FINCEPT TERMINAL · MASTER REFERENCE

EXCLUSIVE · INSTITUTIONAL

THE COMPLETE TERMINAL GUIDE

Every market. Every desk.
One window.

A professional research and trading workstation — streaming markets and execution, institutional-grade analytics, and an agent layer that drives any of it on command.

46	635	864	7
DOCUMENTS	SECTIONS	PAGES	PARTS

MARKETS & EXECUTION

Streaming books and order entry — equities, F&O, 14 crypto venues.

RESEARCH DESKS

Valuation, portfolios, derivatives, M&A, backtesting, a quant lab.

AGENTS & DATA

53 agents, 114 workflow nodes, 48 macro data sources.

FINCEPT RESEARCH DESK · EDITION 5.0 · AUGUST 2026

· Contents

46 DOCUMENTS · 7 PARTS

PART 0 The Prospectus

What the terminal is, where it stands, and what it costs.

01 · The Terminal 8 sections · 10 pp 21
02 · Where It Stands 7 sections · 9 pp 31
03 · The Platform 8 sections · 10 pp 40
04 · The Data 16 sections · 18 pp 50
05 · Plans & Pricing 8 sections · 14 pp 68

PART I Getting Started

The terminal, end to end — what it is and how it is laid out.

The Terminal Shell 21 sections · 26 pp 82
Dashboard 10 sections · 12 pp 108
Dashboard Widgets 18 sections · 23 pp 120
Docs 8 sections · 12 pp 143
About 10 sections · 12 pp 155
Credits & Billing 20 sections · 23 pp 167

PART II Markets & Trading

Live prices, order entry and automated execution.

Markets 8 sections · 12 pp 190
Equity Trading 24 sections · 39 pp 202
Crypto Trading 11 sections · 15 pp 241
Algo Trading 21 sections · 23 pp 256
Watchlist 7 sections · 9 pp 279

PART III Research & Analytics

The analyst desks — valuation, risk, portfolios and strategy.

Equity Research 12 sections · 25 pp 288
Portfolio 18 sections · 25 pp 313
Backtest 11 sections · 21 pp 338
Derivatives 14 sections · 24 pp 359
Alternative Investments 15 sections · 26 pp 383
M&A Analytics 13 sections · 18 pp 409
Relationship Map 17 sections · 19 pp 427
Surface Analytics 9 sections · 18 pp 446

AI Quant Lab 9 sections · 16 pp 464
Alpha Arena 8 sections · 12 pp 480

PART IV Data & Intelligence

Where the numbers come from — macro, sovereign, market and news feeds.

Global Data 18 sections · 20 pp 492
Global Economy 23 sections · 27 pp 512
Data Sources 10 sections · 14 pp 539
Data Mapping 12 sections · 15 pp 553
News 11 sections · 13 pp 568
RSS Feeds 9 sections · 13 pp 581
Geopolitics 18 sections · 21 pp 594
Maritime 21 sections · 24 pp 615
Prediction Markets 23 sections · 26 pp 639

PART V Automation & AI

Agents, workflows and the tooling that drives the terminal for you.

AI Chat 8 sections · 10 pp 665
Agent Studio 15 sections · 27 pp 675
Node Editor 23 sections · 28 pp 702
MCP Servers 22 sections · 24 pp 730
Code 7 sections · 10 pp 754
Report Builder 12 sections · 17 pp 764

PART VI Workspace & System

The everyday tools, your account and how the terminal is configured.

Excel 12 sections · 14 pp 781
Notes 12 sections · 14 pp 795
File Manager 11 sections · 13 pp 809
Settings 23 sections · 26 pp 822
Profile 14 sections · 16 pp 848

About this volume. Each guide below is reproduced complete, with its own contents page and section numbering intact. Every row on this page and in the detailed contents that follows is a live link — click it to jump straight to the page. The running tag at the top right of any page returns you to that guide's own contents.

· Detailed contents

EVERY SECTION

PART 0 · THE PROSPECTUS

01 · The Terminal 21

01 The problem with terminals What USD 32,000 a seat actually buys 23
02 What Fincept is One window, thirty-seven destinations 24
03 Seven reasons it exists The pain points, and the answer to each 25
04 The principles Six rules the code is held to 26
05 How it is built Native C++, one gateway, your keys 27
06 Who it is for Five desks and what changes for them 28
07 What it is not The honest limits, in writing 29
08 Where it goes next What is designed, and what is shipped 30

02 · Where It Stands 31

01 Four tiers, one gap How the category actually segments 33
02 What each one costs Reported 2026 pricing, side by side 34
03 Against Bloomberg A scorecard with the losses left in 35
04 Against the mid-market Koyfin, YCharts, TradingView 36
05 Four things only we do The genuinely unmatched claims 37
06 Where they beat us Unhedged, and worth knowing 38
07 How to choose Who should buy which 39

03 · The Platform 40

01 One window, six groups How thirty-seven screens are organised 42
02 Markets & data Dashboard, Markets, Watchlist, News, Global Data 43
03 Trading Equities, crypto, derivatives, algos, prediction 44
04 Portfolio & risk Holdings, analytics, backtesting, surfaces 45
05 Research & intelligence Equity, M&A, alternatives, geopolitics, maritime 46
06 AI & quant The assistant, the agents, the lab, the nodes 47
07 Tools & workspace Notebook, spreadsheet, reports, files, notes 48
08 What makes it a terminal Why the pieces are worth more together 49

04 · The Data 50

01 Seven kinds of data How to read everything that follows 52
02 Live prices Where a quote on your screen comes from 53
03 The economic catalogue Sixty-seven million labelled series 54
04 Who publishes it Fifty-eight institutions, named and counted 55
05 What the numbers describe Subjects, places, frequencies, history 56
06 Why it is bitemporal The correctness property nobody advertises 57
07 How the catalogue grows A provider is a file, not a release 58
08 Instruments and issuers Nine million securities, and who issued them 59
09 Filings and company facts The full EDGAR corpus, pulled once for everyone 60

10 Ships and ports 776,479 vessels, three years of positions 61
11 Conflict and political events 2.4 million dated, placed events 62
12 What the world trades Thirty years of bilateral flows 63
13 Physical infrastructure The plants, pipes, cables and crossings 64
14 The wire and the odds News, monitors and prediction markets 65
15 Bring your own data 118 connectors and a mapping wizard 66
16 Where coverage stops The limits, stated plainly 67

05 · Plans & Pricing 68

01 The shape of the offer Why no plan hides a screen 70
02 The three plans Price, positioning, allowance 71
03 What actually differs The full limit matrix 72
04 Capabilities by tier The specific actions each plan unlocks 74
05 How credits work What a credit is, and what stays free 76
06 The price list What every metered action costs 77
07 Billing, seats and top-ups The rules, including the awkward ones 80
08 Which plan, and what it replaces Total cost against the alternatives 81

PART I · GETTING STARTED

The Terminal Shell 82

01 What this guide covers 84
02 Anatomy of the window 85
03 The header, left to right 86
04 File — windows, layouts and data 87
05 Navigate — every screen, and the pinned slots 88
06 View — display, panels and workspaces 90
07 Help — about, legal and sign-out 91
08 The command bar 92
09 The main tab bar 94
10 Panels, tabs and the dock 95
11 The tab menu 96
12 Floating, tearing off and second windows 97
13 Workspaces — the ten desks 98
14 Layouts — saving, opening and moving them 99
15 The Component Browser 100
16 Symbol groups — one symbol, many panels 101
17 The pin strip 102
18 The status bar 103
19 Every keyboard shortcut 104
20 Where things are not 105
21 Quick reference and honest notes 106

Dashboard 108

01 What the Dashboard is 110
02 Anatomy of the screen 111
03 The top bar and navigation 112
04 Board controls — the toolbar 113
05 The ticker tape and the status bar 114
06 Working with tiles 115
07 The widget catalogue 116
08 Templates, and what RESET does 117
09 The Market Pulse rail 118
10 Quick reference — how do I… 119

Dashboard Widgets 120

01 The board 122
02 The toolbar 123
03 The ticker, the rail and the status line 124
04 Adding a tile 125
05 Every tile, the same moves 126
06 Refresh — three layers 127
07 Where your board lives 128
08 Templates, and what RESET does 129
09 Markets — twelve tiles 130
10 Research — seven tiles 131
11 Portfolio — five tiles 132
12 Trading — five tiles 133
13 Tools — seven tiles 134
14 Geopolitics — six tiles 136
15 What a board costs 138
16 All 42 widgets, one table 139
17 Honest notes 141
18 Quick reference — how do I… 142

Docs 143

01 What the Docs tab is 145
02 Anatomy of the tab 146
03 All thirty-six topics 147
04 What a topic page contains 148
05 How pages load and re-style 150
06 What is covered — and what is not 151
07 Using it well 152
08 Quick reference and honest notes 153

About 155

01 What the About tab is 157
02 Anatomy of the tab 158
03 Version information 159
04 Check for Updates 160
05 How updates are delivered 161
06 Release notes, in the app 162
07 Licence 163
08 Diagnostics and crash dumps 164
09 Resources and contacts 165
10 Quick reference and honest notes 166

Credits & Billing 167

01 What a credit is 169
02 The balance in the header 170
03 The price list 171
04 What the tab in front of you costs 172
05 The five ways a price is charged 173
06 What is free, and always will be 175
07 Finding a price 176
08 AI and agents 177
09 Trading and execution 178
10 Portfolio, quant and deal work 179
11 Markets, research and data 180
12 Alternative data 181
13 The charges that run without you 182
14 The three plans 183
15 Monthly and quarterly 184
16 Buying credit packs 185
17 Running out 186
18 Every priced action, one table 187
19 Honest notes 188
20 Quick reference — how do I… 189

PART II · MARKETS & TRADING

Markets 190

01 What the Markets tab is 192
02 Anatomy of the screen — the five regions 193
03 The header bar 194
04 Reading a market panel 195
05 The nine default panels 196
06 Building your own board (+ the panel editor) 197
07 Refresh, states & good to know 199

08 Quick reference — “How do I…” 200

Equity Trading 202

01 What the Equity tab is 204
02 Anatomy of the workstation 205
03 The command bar 206
04 Broker accounts 207
05 Paper, live, and the approval gate 209
06 The watchlist 210
07 Charts, compare and chart trading 211
08 Market depth 213
09 The order ticket: the core form 214
10 Advanced, GTT and option strategies 215
11 Sending one order to several accounts 217
12 Blotter: Positions and Holdings 218
13 Blotter: Orders, Funds and Stats 220
14 US brokers, feeds and the unified monitor 222
15 The F&O desk 223
16 The option chain 224
17 Trading from the chain 226
18 The strategy builder 227
19 The risk model, and trading a structure 230
20 Open interest and straddles 232
21 Screening, flows, positions and cost 235
22 Data, persistence and agent tools 238
23 Quick reference 239
24 Honest notes on this build 240

Crypto Trading 241

01 What the Crypto Trading tab is 243
02 Anatomy — the six regions 244
03 Command bar, exchanges & modes 245
04 The watchlist 246
05 The chart 247
06 Order book, depth & tape 249
07 The order ticket 250
08 The blotter 252
09 Paper, live & connecting an exchange 253
10 AI tools & the data engine 254
11 Quick reference & honest notes 255

Algo Trading 256

01 What the ALGO tab is 258
02 Anatomy: the frame and the seven tabs 259

03 Builder — the four-step flow 260
04 Builder — writing rules in sentences 261
05 Builder — the safety net and F&O legs 262
06 Builder — the AI strategy assistant 263
07 Strategies — your library 264
08 Backtest Lab — one symbol, one report 265
09 Backtest Lab — baskets, and what it does not model 266
10 Scan — the one-time check 267
11 Scan — the live universe sweep 268
12 Alerts — background watches 269
13 The deploy dialog, field by field 270
14 Baskets and F&O options baskets 271
15 Dashboard — the fleet 272
16 Dashboard — positions, fills and controls 273
17 History — the permanent ledger 274
18 How a deployment actually runs 275
19 The safety rails, in numbers 276
20 Driving the desk from AI Chat 277
21 Quick reference and honest notes 278

Watchlist 279

01 What the Watchlist tab is 281
02 Anatomy — the four regions 282
03 Your watchlists — the sidebar 283
04 Adding symbols & the two views 284
05 Live prices, CSV & backtest 285
06 AI tools & where lists appear 286
07 Quick reference & honest notes 287

PART III · RESEARCH & ANALYTICS

Equity Research 288

· Overview — what the Equity Research workstation is 290
1 Anatomy of the workstation (title bar, quote bar, rail, dock) 291
2 OVERVIEW — the one-screen snapshot 293
3 FUNDAMENTALS — Financials · Valuation · Analysis · Quality 295
4 MARKET — Technicals · Peers · Ownership · Cross-Asset 299
5 CONTEXT & RESEARCH — Macro · Quant Lab · Scenario · Filings 303
6 FLOW — News · Sentiment 306
7 The research dock — Notes · Copilot · Alerts 308
8 Toolbar actions — Buy/Sell · Backtest · CSV export 308
9 Where the numbers come from (data & provenance) 310
10 AI agent tools (45 MCP tools) 310
11 Quick reference & important notes 312

Portfolio 313

01 What the Portfolio tab is 315
02 Workspace anatomy & the command bar 316
03 The live dashboard — holdings, chart, sectors, positions 318
04 Sectors & concentration 320
05 Performance & risk 321
06 Portfolio optimization 322
07 QuantStats report 325
08 Reports & attribution 326
09 Custom indices 327
10 Risk management 328
11 Financial planning 329
12 Economics & macro factors 330
13 Trading — order entry, buy / sell / dividend 331
14 Managing portfolios — create, import, export 332
15 AI & Agent insights 333
16 Analytics engine, live data & metrics 334
17 Agent automation tools 335
18 Quick reference & honest notes 336

Backtest 338

· Overview — what the Backtest tab is 340
1 Anatomy of the workbench (top bar, three panels, status bar) 341
2 Providers — the 6 backtest engines 343
3 Commands — the 11 operations & how the UI morphs 344
4 Configuring a run — market data, execution, advanced, benchmark, strategy 345
5 The five result windows (Summary · Equity · Metrics · Details · Raw JSON) 346
6 Core commands — Backtest, Optimize, Walk-Forward 350
7 Analysis commands — Indicators, Signals, ML Labels, CV Splits, Returns, Sweep 352
8 Strategy catalog (per engine) + the Custom Combo builder 356
9 Engine, data & AI agent tools 357
10 Quick reference — “How do I…?” & important notes 358

Derivatives 359

· Overview — what the Derivatives tab is 361
1 Anatomy of the workbench (shell, rail + canvas, charts) 362
2 CHAIN — the option chain 364
3 VOL SURFACE — SVI / SSVI calibration 366
4 VOL ANALYSIS — realized vs implied, VRP, VIX strip 368
5 STRATEGY LAB — 23 templates, payoff & greeks 370
6 RISK — greeks, scenarios & VaR 372
7 POSITIONING — dealer gamma (GEX) 374
8 EVENTS — earnings / event vol & density 376

9 PRICERS — 16 multi-asset calculators 378
10 Data sources & the seeded demo 380
11 Analytics engine — capability catalog 380
12 AI agent tools (35 MCP tools) 381
13 Quick reference — “How do I…?” & important notes 382

Alternative Investments 383

· Overview — what the Alt Investments tab is 385
1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 386
2 PROGRAM — set up & edit the LP program 388
3 PERFORMANCE — IRR / TVPI / PME & J-curves 390
4 PACING — Takahashi-Alexander projection & commitment solver 392
5 LIQUIDITY — calls vs distributions, coverage & redemption ladder 394
6 EXPOSURE — concentration, treemap, matrix & de-smoothing 396
7 SCENARIOS — historical & custom stress, denominator effect 398
8 DEAL LAB — waterfalls, fees, secondaries, RE/infra underwriting 400
9 INSTRUMENT LAB — 24 instrument calculators 402
10 MARKET — live proxies & embedded benchmarks 404
11 The seeded demo program (reference numbers) 406
12 Analytics engine — capability catalog 407
13 AI agent tools (30 MCP tools) 407
14 Quick reference — “How do I…?” & important notes 408

M&A Analytics 409

· Overview — what the M&A Analytics tab is 411
1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 412
2 DEAL DESK — set up the deal 414
3 VALUATION — DCF, comps, precedents, LBO & the football field 416
4 MERGER MODEL — accretion/dilution, sources & uses, synergies 418
5 STRUCTURE — payment mix, exchange ratio, collar, earnout, CVR 420
6 MARKET INTEL — deal database, live EDGAR scan, cross-deal analytics 421
7 SPECIALIZED — startup valuation & industry lens 423
8 RISK & FAIRNESS — Monte Carlo, fairness opinion, premium, process 424
9 The seeded sample deal & deal database (reference) 425
10 Analytics engine — capability catalog 425
11 AI agent tools (47 MCP tools) 425
12 Quick reference — “How do I…?” & important notes 426

Relationship Map 427

01 What the Relationship Map is 429
02 Anatomy of the screen 430
03 The header bar 431
04 Finding a company 432
05 How the graph is arranged 433

06 The company card 434
07 Board and officers 435
08 Insiders — SEC Form 4 436
09 Metrics, governance and short interest 437
10 Corporate events — SEC 8-K 438
11 Peers — and what they really are 439
12 Holders and funds 440
13 Analysts 441
14 Filters 442
15 Moving around the canvas 443
16 Where the data comes from, and what it costs 444
17 Quick reference and honest notes 445

Surface Analytics 446

· Overview — what the Surface Analytics tab is 448
1 Anatomy of the workbench 449
2 The 7 categories & 35 surfaces — the catalog 450
3 The Control Panel — inputs & parameters 457
4 The three views — 3D, Table, Line 458
5 The Data Inspector — raw data, lineage, status 460
6 Data & connectivity — the four tiers, Databento, CSV 460
7 AI agent tools (15 MCP tools) 462
8 Quick reference & important notes 462

AI Quant Lab 464

· Overview — what the AI Quant Lab is 466
1 Anatomy — the 24-module workbench 467
2 CORE — factors, models, backtests, signals 469
3 AI / ML — agents, RL, online & meta learning 471
4 ADVANCED — HFT, optimization, feature engineering 473
5 ANALYTICS — GS Quant, statsmodels, CFA, forecasting 475
6 The engine — Python, credits & dependencies 477
7 Driving Quant Lab by AI — the MCP tool suite 478
8 Quick reference & honest caveats 479

Alpha Arena 480

01 What Alpha Arena is 482
02 Anatomy — the live dashboard 483
03 Setting up — the competition wizard 484
04 Running & controlling a competition 485
05 Reading the arena — the six panels 486
06 Under the hood — the trading engine 488
07 Live mode & its beta status 489
08 Cost, AI tools & honest notes 491

Global Data 492

01 What the Global Data tab is 494
02 Anatomy of the screen 495
03 The command strip 496
04 The twenty-eight sources 497
05 Browsing by topic 498
06 Reading the results list 499
07 Ask AI — questions instead of keywords 500
08 Loading a series 501
09 The series rail 502
10 Window, style and transform 503
11 Comparing — two charts, one screen 504
12 Reading the chart 505
13 The data table 506
14 AI analysis of what you charted 507
15 Exporting to CSV 508
16 What it costs 509
17 What carries over between sessions 510
18 Quick reference and honest notes 511

Global Economy 512

01 What the Global Economy tab is 514
02 Anatomy of the screen 515
03 Getting here, and what the tab remembers 516
04 Asking a question 517
05 What it costs 518
06 Chord — one country against its partners 519
07 Treemap — the same answer, sized 520
08 The ranking table 521
09 Products — what the country actually trades 522
10 The map, and how to read it 523
11 The indicator picker 524
12 The year timeline and playback 526
13 Zoom, pan and the country readout 527
14 The Layers menu 528
15 The seventeen layers 529
16 Reading a layer 530
17 Infrastructure over an indicator 532
18 The four dashboard tiles 533
19 Every control on the tab 534
20 Where the numbers come from 535
21 Every status message 536

22 Honest notes 537
23 Quick reference — how do I… 538

Data Sources 539

· Overview — what the Data Sources tab is 541
1 Anatomy of the screen (header, stats, tabs, two views) 542
2 BROWSE — search & filter the catalog 543
3 The connector Inspector 544
4 The Connection Config window (add / edit / clone) 545
5 Testing connections & live status 547
6 CONNECTIONS — manage saved connections 548
7 Import / Export / Template 548
8 The connector catalog — 118 connectors by category 550
9 Security & data handling · Quick reference · Notes 552

Data Mapping 553

· Overview — what the Data Mapping tab is 555
1 Anatomy of the screen (chrome, panels, status) 556
2 View · MAPPINGS — the saved-mappings list 557
3 View · TEMPLATES — 14 ready-made APIs 558
4 Wizard Step 1 · API Config 560
5 Wizard Step 2 · Schema 561
6 Wizard Step 3 · Field Mapping 562
7 Wizard Step 4 · Cache & Security 563
8 Wizard Step 5 · Test & Save 564
9 Reference — the 7 schemas, parsers, transforms, templates 565
10 Under the hood — network, storage, security 566
11 Quick reference & notes 567

News 568

01 What the News tab is 570
02 Anatomy of the screen 571
03 Filtering & searching the wire 572
04 Counts, sentiment & board controls 573
05 Reading the wire 574
06 Clusters, the banner & the ticker 575
07 The article detail panel 576
08 The Intelligence drawer 577
09 Managing news sources 578
10 Live feed, refresh & states 579
11 Quick reference — “How do I…” 580

RSS Feeds 581

01 What the RSS Feeds tab is 583

02 Anatomy — the board & toolbar 584
03 Feed boxes — cards & tables 586
04 Adding a feed — the Add Feed dialog 587
05 Field mapping & one-click Discover 588
06 Managing feeds — edit, refresh, remove 590
07 Under the hood — the feed engine 591
08 AI feed tools & cross-app reuse 592
09 Quick reference & honest notes 593

Geopolitics 594

01 What the Geopolitics tab is 596
02 Anatomy of the screen 597
03 Filters and the legend 598
04 The four workspaces 599
05 The conflict map 600
06 The events table 601
07 Overview, categories and hotspots 602
08 Mapped against loaded 603
09 What HDX is, and its four views 604
10 Datasets — the view that works 605
11 Explorer — searching HDX yourself 606
12 Relations — a fixed reference chart 607
13 Trade — four analyses, none of which run 608
14 Credits — what costs what 609
15 Driving the tab from the AI assistant 610
16 Where the data comes from 611
17 Status and message reference 612
18 Quick reference and honest notes 613

Maritime 615

01 What the Maritime tab is 617
02 Anatomy of the screen 618
03 The filter bar 619
04 The seven workspaces 620
05 Load actions and the draw tools 621
06 Fleet intelligence and trade corridors 622
07 Market pulse and chokepoint transits 623
08 The map, its toolbar and its layers 624
09 Vessel search, place search and ports 625
10 Tracking a ship and replaying a voyage 626
11 FLOWS — corridors and top lanes 627
12 PORTS — league table and congestion 628
13 COUNTRIES — seaborne arrivals 629

14 FLEET — tonnage and flag states 630
15 SIGNALS — four fleet indicators 631
16 WATCH — standing screens over the fleet 632
17 Credits — what costs what 633
18 Plans — what actually needs Exclusive Pro 634
19 Driving the tab from the AI assistant 635
20 Where the data comes from 636
21 Quick reference and honest notes 637

Prediction Markets 639

01 What the Prediction Markets tab is 641
02 Anatomy of the screen 642
03 The command bar 643
04 Choosing an exchange 644
05 The views, and what each one lists 645
06 Categories, chips and series 646
07 Search 647
08 Sorting, and how to read a market card 648
09 The detail panel 649
10 The price chart 650
11 The order book 651
12 Outcomes, status and open interest 652
13 Holders, comments and related markets 653
14 The trades feed 654
15 The leaderboard 655
16 Connecting an exchange account 656
17 The order ticket 657
18 The open-orders blotter 658
19 Amending, cancelling and cancel-all 659
20 Polymarket vs Kalshi — what actually changes 660
21 Driving the tab from the AI assistant 661
22 Where the data comes from, and what it costs 662
23 Quick reference and honest notes 663

PART V · AUTOMATION & AI

AI Chat 665

01 What the AI Chat tab is 667
02 Anatomy of the screen 668
03 Getting started 669
04 Reading a conversation 670
05 The agentic engine 671
06 Conversations & history 672
07 The Quick-Chat bubble 673

08 Quick reference — “How do I…” 674

Agent Studio 675

· Overview — what the Agent Studio tab is 677
1 Anatomy — the Agent Studio shell 678
2 AGENTS — browse & configure the persona catalog 680
3 CREATE — author a custom agent 682
4 TEAMS — multi-agent teams & the six modes 684
5 WORKFLOWS — one-click analyses 686
6 PLANNER — plan, edit, then execute 688
7 DESKS — standing, scheduled agent desks 690
8 TOOLS — the tool catalog & assignment console 692
9 CHAT — talk to an agent 694
10 SYSTEM — the capabilities dashboard 696
11 The agent engine — how a run executes 697
12 Driving Agent Studio by AI — the MCP tool suite 698
13 Advanced — Agentic Mode (developer) 699
14 Quick reference & honest caveats 700

Node Editor 702

01 What the Node Editor is 704
02 Anatomy of the workbench 705
03 The toolbar, button by button 707
04 The node palette 708
05 The canvas: move, zoom, connect 710
06 A node up close 711
07 The properties panel 712
08 How a workflow actually runs 713
09 The execution results panel 714
10 Triggers, Core, Control Flow 716
11 Data Transform and Data Format 717
12 Market Data and Analytics 718
13 Utilities, Files and Data Sources 719
14 Trading, Safety, Agents and MCP 720
15 What costs money, what touches the network 721
16 Trading safely: paper, approval, audit 722
17 Expressions and data flow 723
18 The fifteen templates 724
19 Saving, loading, import and export 725
20 Deploy — and what it does not do 726
21 Driving workflows from AI Chat 727
22 A worked example: Economic Dashboard 728
23 Quick reference and honest notes 729

MCP Servers 730

01 What the MCP Servers tab is 732
02 Anatomy — three views and a status bar 733
03 MARKETPLACE — the curated catalogue 734
04 INSTALLED — and adding your own 735
05 How an external server connects 736
06 TOOLS — the master switchboard 737
07 What the internal tools are 738
08 The catalogue in one page 739
09 Trading & execution — 106 tools 740
10 Research & analytics — 238 tools 741
11 Data & markets — 104 tools 742
12 Workspace & documents — 165 tools 743
13 Agents & orchestration — 114 tools 744
14 Portfolio & account — 29 tools 745
15 Finding a tool among 760 746
16 The task spine and playbooks 747
17 Destructive tools and the approval gate 748
18 Plan gating 749
19 Timeouts, async and the schema budget 750
20 Persistence and security 751
21 Driving the tab from the agent 752
22 Quick reference and honest notes 753

Code 754

01 What the Code tab is 756
02 Anatomy & the two views 757
03 The Notebook Library 758
04 The Editor 759
05 Cells & running code 760
06 Notebooks, saving & files 762
07 Quick reference — “How do I…” 763

Report Builder 764

· Overview — what the Report Builder tab is 766
1 The workspace at a glance 767
2 The Home view — three ways to start 769
3 The template library 770
4 Editing on the canvas 771
5 The block catalogue 773
6 Data blocks & charts 774
7 Document setup — metadata, branding & themes 776
8 Export & the PDF 777

9 The AI & MCP agent tools 778
10 Storage, autosave & cloud sync 779
11 Quick reference & honest caveats 780

PART VI · WORKSPACE & SYSTEM

Excel 781

01 What the Excel tab is 783
02 Anatomy of the tab 784
03 The toolbar 785
04 Entering data: the grid and the fx bar 786
05 Formulas — functions and arithmetic 787
06 Formula limits you should know 788
07 Sheets — add, rename, delete 789
08 The right-click menu 790
09 Import and export 791
10 What is saved, and what is not 792
11 Driving the sheet from AI Chat 793
12 Quick reference and honest notes 794

Notes 795

01 What the Notes tab is 797
02 Anatomy of the tab 798
03 The category sidebar 799
04 The notes list 800
05 Reading a note 801
06 Writing a note 802
07 The metadata model 803
08 Exporting a note as markdown 804
09 Favourite, archive and delete 805
10 Where notes are stored 806
11 Driving notes from AI Chat 807
12 Quick reference and honest notes 808

File Manager 809

01 What the File Manager is 811
02 Anatomy of the tab 812
03 The header and the storage bar 813
04 Finding a file: search, sort, source chips 814
05 The file card and its four buttons 815
06 The preview panel 816
07 Selecting and deleting 817
08 The eleven producing screens 818
09 Storage, quota and what is on disk 819

10 Driving it from AI Chat 820
11 Quick reference and honest notes 821

Settings 822

01 What the Settings screen is 824
02 Anatomy — the rail and the two scopes 825
03 General 826
04 Appearance 827
05 Notifications — the fifteen providers 828
06 Notifications — alert triggers 829
07 Keybindings 830
08 Logging — levels and the log file 831
09 Logging — shipping logs to the backend 832
10 Usage Analytics — what is recorded 833
11 Usage Analytics — control, status and erasure 834
12 Developer 835
13 Profiles 836
14 Credentials 837
15 Security 838
16 Data Sources 839
17 Python Env 840
18 Storage — disk usage and categories 841
19 Storage — files, state and the SQL console 842
20 Cloud Sync — the three tiers 843
21 Cloud Sync — per-domain control and first sync 844
22 Where each setting is stored 845
23 Quick reference and honest notes 846

Profile 848

01 What the Profile tab is 850
02 Anatomy of the tab 851
03 The header bar 852
04 The ACCOUNT panel 853
05 Credits and usage 854
06 The two usage breakdowns 855
07 The credit ledger 856
08 Edit profile 857
09 Changing your password 858
10 Subscription and billing 859
11 Devices and seats 860
12 Logout, refresh and what is stored 861
13 Driving it from AI Chat 862
14 Quick reference and honest notes 863

THE PLATFORM

A terminal you can afford.

Institutional-grade market intelligence, research and execution in one native desktop application — built for the analysts, funds and desks the incumbents priced out.

Contents

8 SECTIONS

THE ARGUMENT

01 **The problem with terminals** What USD 32,000 a seat actually buys 3

02 **What Fincept is** One window, thirty-seven destinations 4

03 **Seven reasons it exists** The pain points, and the answer to each 5

HOW WE BUILD

04 **The principles** Six rules the code is held to 6

05 **How it is built** Native C++, one gateway, your keys 7

FIT

06 **Who it is for** Five desks and what changes for them 8

07 **What it is not** The honest limits, in writing 9

08 **Where it goes next** What is designed, and what is shipped 10

About this document. Every figure, count and price in this set was read out of the shipping source tree rather than from a brochure — the tier tables from `pkg/authz`, the price list from `pkg/credits`, the connector and venue counts from the registries that build them. Where something is a demonstration rather than a live feed, it is labelled as one. Section 7 is a list of what the product does not do yet.

THE FIVE DOCUMENTS IN THIS SET

01	**The Terminal**	What it is, why it exists, and what it is not
02	Where It Stands	The landscape, and an honest comparison
03	The Platform	Every capability, screen by screen
04	The Data	Prices, the indexed store, providers, coverage
05	Plans & Pricing	Three tiers, the credit meter, what changes

01 The problem with terminals

A Bloomberg Terminal costs about **USD 31,980 per seat per year** on a single-terminal contract, roughly **USD 28,320** on a multi-terminal one, billed quarterly in advance against a two-year minimum. There are around **325,000** of those seats in the world. There are several million people who do this work.

That gap is the whole business. It is not a gap in ambition or in skill — the analyst at a four-person family office in Mumbai runs the same screens as the one at a bulge bracket in London. It is a gap in procurement. One of them works somewhere that can sign a two-year, six-figure data contract; the other does not, and so does the job across a browser, a spreadsheet, a broker's web platform and a Jupyter notebook, joining them by hand.

Price is only the visible half of it. The rest is structural, and it is what people who *do* have a terminal complain about.

The complaint	What it looks like in practice
Locked data	Export and download limits are enforced but not published. There is no programmatic way to ask how much of your monthly allowance you have used, and pulling data into your own Python environment is restricted rather than supported.
Feature sprawl	Over thirty thousand functions accumulated over four decades. Power is not the problem; discovery is. Most users work a few dozen screens and never find the rest.
A seat, not a licence	The entitlement follows the terminal, not the person. Leave the firm and you leave the tooling, the layouts and the saved work behind.
Coverage that stops	Depth is superb where the vendor's customers are concentrated and thin where they are not. If your universe is Indian mid-caps, West African sovereigns or spot LNG, you are paying institutional money for retail coverage.
AI arrived last, and bolted on	Agentic assistants reached the major terminals in 2026, years after the workflow was set, and each is confined to its own vendor's data.

None of that is incompetence. It is the predictable shape of a product that has been compounding since 1981 and whose economics reward depth for the largest accounts over reach for everyone else. It does leave a very large, very specific hole.

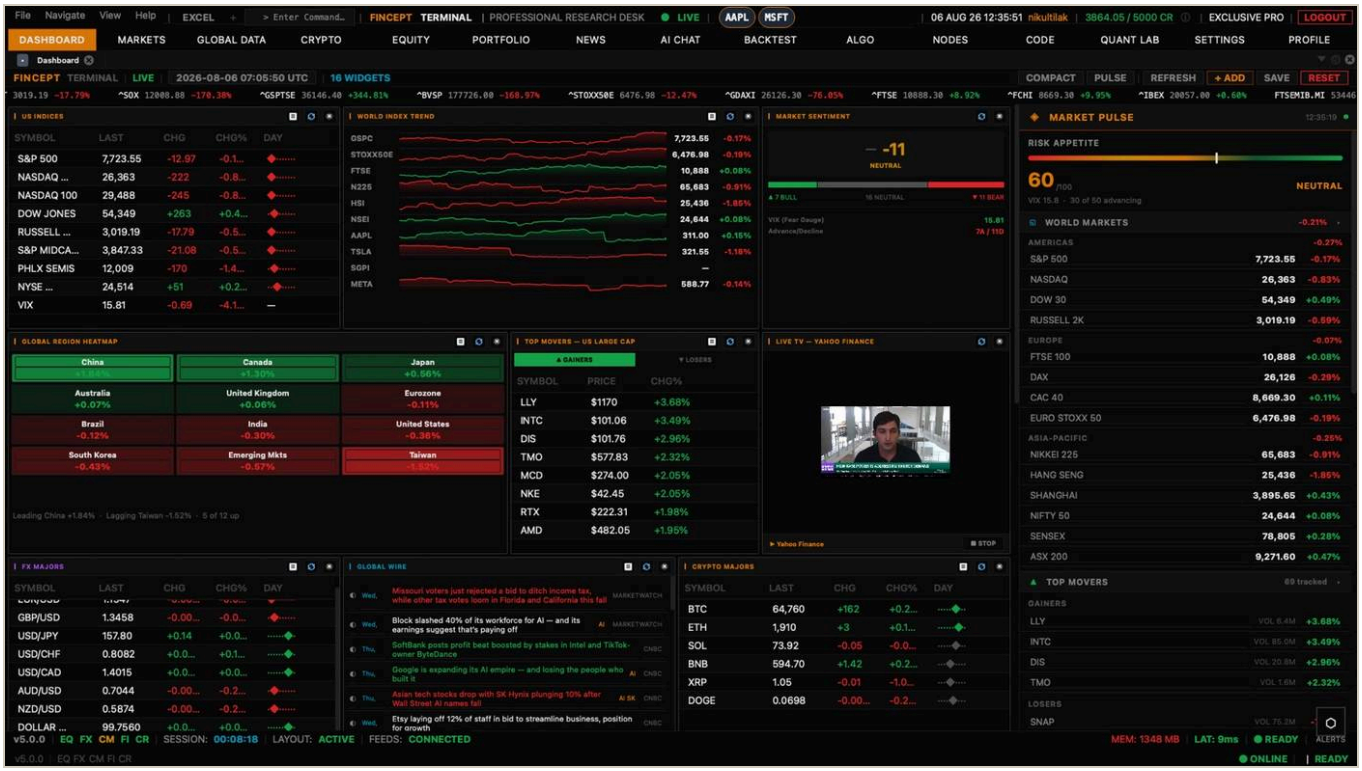
We are not trying to beat the incumbent at its own game. We are trying to serve the people it was never priced for — and to be honest about the difference.

02 What Fincept is

Fincept Terminal is a native desktop financial-intelligence workstation. Not a web app in a wrapper — a C++20 / Qt6 binary with embedded Python, running on Windows, macOS and Linux, talking to one gateway.

37	22	14	790
DESTINATIONS IN THE NAVIGATE MENU	BROKERS, NATIVELY INTEGRATED	CRYPTO EXCHANGES, DIRECT	TOOLS THE AI CAN CALL

One window covers the arc most people currently run across five applications: watch the tape, research a name, model the trade, place it, monitor the book, write it up. The screens are dockable, so the layout is yours; the command bar jumps anywhere by name; and an AI assistant sits over the whole thing on F9, wired to the same tools you are clicking.



The Dashboard — a grid the user assembles from thirty-eight live widgets, with the Market Pulse rail at right and a streaming ticker below. Every screen shares this frame: toolbar and command bar above, status bar beneath.

What it is *not* is a collection of separate products behind one login. The portfolio you build in one tab is the universe the backtester reads, the book the risk views price, and the context the AI answers from. That is the point of a terminal, and it is the part that a folder full of subscriptions cannot reproduce at any price.

03 Seven reasons it exists

THE ARGUMENT · 03

Each of these is a complaint we heard often enough to build against. The answer under each one is shipped, not planned.

Cost puts the tooling out of reach

USD 32,000 a year, two-year minimum, quarterly in advance — a procurement process before a product decision.

FINCEPT USD 99 to USD 299 a month, billed monthly, no minimum term — between four and eleven per cent of an incumbent seat.

The work is scattered across five applications

A terminal for the tape, a spreadsheet for the model, a notebook for the research, a broker portal to execute, a document tool to write it up. Every boundary is a copy-paste and a chance to be wrong.

FINCEPT All five live in one process and share one data layer — including a real spreadsheet, a Jupyter-style notebook, and a report compiler whose preview is the exported PDF.

Your data is hard to get out

Undisclosed download caps, no way to query your own usage, restricted programmatic access.

FINCEPT Bulk CSV and Parquet export of the indexed store, a documented HTTP surface, and one hundred and eighteen connectors that read your own databases and warehouses.

Coverage stops at the borders that matter to you

Deep US and European equity data; thin everywhere the vendor's largest accounts are not.

FINCEPT Seventeen Indian brokers alongside US and European ones, fourteen crypto venues natively, and an indexed macro store built from the world's statistical agencies rather than from one vendor's licence.

The AI cannot reach your actual work

A chat window trained on the vendor's own corpus, unable to see your positions, your notes or your models.

FINCEPT The assistant calls the same seven hundred and ninety tools the interface does. It reads your portfolio, runs your screens, pulls the filing and writes the note.

You cannot tell what anything costs until the invoice

Usage-based charges that surface at the end of the month.

FINCEPT Every metered action states its price on the control before you press it, and every recurring meter states its rate before it is switched on.

A cheaper plan means a broken product

Entry tiers that grey out half the interface, so the buyer evaluates a demo rather than the product.

FINCEPT Every screen is open on every paid plan and renders real data. Plans differ in depth and in specific actions, never in whether a screen works.

04 The principles

Six rules the codebase is actually held to. They are worth stating because each one costs us something, and each is the reason a specific class of complaint does not apply here.

1 — A LOCKED FEATURE MUST NEVER LOOK BROKEN

A gated control keeps its label, gains a lock, and explains itself when clicked. It is never simply disabled, because a disabled button cannot tell you why. Every quota is rendered as a meter beside the thing it limits — *3 of 3 watchlists* — so the ceiling is visible before it is hit rather than discovered at it.

2 — THE PRICE IS STATED BEFORE THE CLICK

Anything that spends credits carries its cost on the control. The four charges that accrue without a click — algo runtime, watchlist alerts, scheduled refreshes, news monitors — state their rate before they are armed, and the scheduling dialog quotes the monthly total rather than the per-run unit, because the unit reads as trivial and the total is what is being agreed to.

3 — HISTORY IS BITEMPORAL, SO BACKTESTS CANNOT CHEAT

Economic data is revised. Storing a revision over the original silently injects lookahead bias into every backtest that touches it. Our observation store keeps every vintage with the date we learned it, so a strategy tested against March can only see what was published by March. Split and dividend adjustment is likewise computed at read time against the corporate actions known as of your chosen date — never stored, because an adjusted price is a fact about a day *and* the day you are standing on.

4 — YOUR BROKER KEYS NEVER REACH OUR SERVERS

Broker and exchange credentials are written to the operating system keychain and stay on the workstation. Only account metadata goes to the platform. A live order is signed locally and routed by the machine in front of you.

5 — A NEW DATA PROVIDER IS A CONFIGURATION FILE

Providers are declared as JSON manifests describing endpoints, paging, authentication and field mapping — not as code. Adding a national statistics office is a file and a crawl, with no release and no downtime. That is why the catalogue can keep growing between versions.

6 — RUNNING OUT OF CREDITS NEVER DESTROYS A POSITION

An algorithm that cannot pay its daily runtime has new entries blocked and keeps everything else: exits, stop-losses, trailing stops and bracket unwinds all continue, and open positions are untouched. Flattening a live book over a billing state is indefensible, so the code cannot do it.

05 How it is built

The engineering choices below are the ones a buyer can feel — in latency, in what happens when the network drops, and in who can read your credentials.

Layer	What it is	Why it matters to you
Desktop client	C++20 / Qt6.8.3 native binary with an embedded Python 3.11 runtime for analytics. Windows, macOS, Linux.	Charts and blotters redraw at desk speed rather than browser speed, and the analytics run in-process without a round trip.
Gateway	One public edge on which every request is authenticated, entitled and rate-limited before it reaches a service.	One place to reason about access. The shipping build compiles the gateway address in, so no setting, environment variable or registry key can point the client somewhere else.
Services	Around twenty Go services behind that edge — market data, paper and live trading, analytics, news, the data plane, files, releases, voice.	One slow provider cannot stall the desk; ingestion runs in a separate process from everything you read.
Licensing	An Ed25519-signed, device-bound entitlement receipt with a short expiry, verified against a key compiled into the binary.	Your plan cannot be forged, and it also cannot be silently downgraded by a bad response — an unfetched entitlement reads as allowed, never as locked.
Credentials	OS keychain on the workstation for broker and exchange keys; server-held keys only for services we pay for on your behalf.	We cannot leak what we do not hold.
Analytics	In-process C++ model libraries for derivatives, M&A and alternatives; Python for the research surface; server-side only where the server genuinely does it better.	The local models are free and unmetered. You are charged for our compute, not for arithmetic your own machine did.

> **Offline behaviour is designed, not accidental.** Screens whose data lives in your account need the gateway; screens that do not, keep working. The client watches connectivity, tells you which state it is in, and reconnects on its own rather than requiring a restart.

06 Who it is for

FIT · 06

Five desks, and the specific thing that changes for each. If none of these is recognisably you, the product is probably not a fit — and we would rather say so here than after the invoice.

Desk	Today	With Fincept
The emerging-markets analyst	Institutional pricing for coverage that thins out exactly where the mandate is. Local data pulled from a broker portal by hand.	Seventeen Indian brokers wired natively, an indexed macro store built from national agencies rather than one vendor's licence, and a research desk that reads both.
The small fund or family office	Two or three people cannot justify a six-figure data line, so the stack is a patchwork of consumer subscriptions.	One seat at USD 99–299 a month covers the tape, the research, the book, the backtest and the client report.
The quantitative researcher	Jupyter plus a vendor API plus a backtester, with reproducibility maintained by discipline rather than by architecture.	Point-in-time data by construction, six backtest engines, an append-only trial ledger that deflates a Sharpe by the number of experiments actually run, and a notebook in the same window.
The systematic or active trader	A broker platform for execution, a charting subscription for analysis, a spreadsheet to reconcile them.	Multi-broker order entry across every linked account at once, fourteen crypto venues, a strategy builder that writes rules in plain English, and paper trading on the same engine as live.
The independent analyst or student	Priced out entirely; learns on tools they will not use professionally.	The same interface, the same data model and the same AI tooling as the paying desk above them, at a price a person can carry personally.

> **Seats and devices.** A plan is a person, not a workstation: the base and middle tiers include one device, Pro includes two, and additional seats can be added to an account at the same price at any time.

07 What it is not

FIT · 07

Anything a buyer would find out in the first month, they should find out here. These are the real limits of the current build.

DATA

- **We do not resell a licensed premium vendor.** Live prices come from the broker or exchange connection you already hold; reference, fundamental and macro data comes from public and government sources we index ourselves. A licensed vendor tier is on the roadmap and is not in the product today.
- **Fifteen of the thirty-five analytic surfaces are synthetic demonstrations** — the fixed-income, FX, credit and macro families. They are labelled DEMO in the interface. The equity and crypto volatility surfaces run on live data.
- **Depth of history varies by source.** Where a provider publishes no revision history, an as-of query says so rather than guessing.

COVERAGE AND WORKFLOW

- **No fixed-income execution and no OTC dealer connectivity.** Bonds are covered analytically, not transactionally.
- **No compliance, order-audit or regulatory-reporting module.** If you need supervised communications or a books-and-records archive, that is not this product.
- **Algorithmic execution runs locally.** Strategies execute on your machine with the server mirroring state; the terminal is not a colocated execution venue and is not built for latency arbitrage.
- **Prediction markets are read-mostly.** Browsing, odds, book and tape are live; treat the trading path as beta.

COMPANY

- **We are small and new.** There is no thirty-year data archive behind us and no global support desk. Support is email and in-app, with priority handling on the upper tiers.
- **Some of what an incumbent does, we do not do at all** — and where the comparison goes against us, Document 02 says so directly rather than omitting the row.

If a feature is a demonstration, it says DEMO on the screen. If a price would charge you, it is printed on the button. We would rather lose a sale on page seven than lose a customer in month two.

08 Where it goes next

Two lists. The first is shipped and running in the build this document describes. The second is designed and not built — it is here so that nobody mistakes a roadmap item for a feature.

SHIPPED IN V5.0

Area	State
Trading — paper engine, brackets, multi-broker routing, reconciliation	**Live**
Algorithmic strategies — builder, backtest, scanner, deployment, runtime meter	**Live**
Indexed data plane — manifest connectors, bitemporal store, catalogue search, bulk export	**Live**
News — RSS corpus, batch sentiment, server-side keyword monitors that fire while the terminal is shut	**Live**
AI — multi-provider gateway, agentic tool loop, teams, deep research, voice in and out	**Live**
Research — equity, derivatives, M&A, alternatives, surfaces, quant lab	**Live**
Alternative data — maritime vessel positions and geopolitical events on live feed tables	**Live**
Prediction markets	**Partial — browse live, trading beta**

DESIGNED, NOT BUILT

Area	State
A licensed premium data vendor for the exclusive core	**Not procured**
Event bus and streaming fan-out tier for very high connection counts	**Not built**
Column-store analytics tier for tick-scale history	**Not built**
Multi-region deployment	**Not built**

> **Why publish the second table.** The architecture document these rows come from carries an instruction to its own engineers: call these *planned*, not *current*, when reasoning about the system. The same rule should apply when we are selling it.

THE LANDSCAPE

Where it stands.

An honest comparison against Bloomberg, LSEG Workspace, FactSet and the mid-market research platforms — including the rows where we lose.

Contents

7 SECTIONS

THE MARKET

01 **Four tiers, one gap** How the category actually segments 3

02 **What each one costs** Reported 2026 pricing, side by side 4

HEAD TO HEAD

03 **Against Bloomberg** A scorecard with the losses left in 5

04 **Against the mid-market** Koyfin, YCharts, TradingView 6

05 **Four things only we do** The genuinely unmatched claims 7

THE VERDICT

06 **Where they beat us** Unhedged, and worth knowing 8

07 **How to choose** Who should buy which 9

About the numbers. Competitor pricing is negotiated and rarely published; the figures here are the 2026 market-reported ranges from procurement and benchmarking sources, not quotations. They are directional and will differ from any particular contract. Fincept's own prices are exact — they are read from the billing service's plan catalogue and are the same numbers the checkout charges.

THE FIVE DOCUMENTS IN THIS SET

01	The Terminal	What it is, why it exists, and what it is not
02	**Where It Stands**	The landscape, and an honest comparison
03	The Platform	Every capability, screen by screen
04	The Data	Prices, the indexed store, providers, coverage
05	Plans & Pricing	Three tiers, the credit meter, what changes

01 Four tiers, one gap

Global spend on financial market data reached a record **USD 49.2 billion** in 2025, growing 6.5 per cent on the year. Almost none of that growth is reaching the people this document is written for, and the reason is structural rather than commercial.

The category sorts into four tiers, and they are not competing with each other so much as serving four different procurement processes.

Tier	Who it serves	What it costs	The catch
Institutional terminals **Bloomberg, LSEG Workspace, FactSet, S&P Capital IQ**	Banks, large asset managers, anyone with a market-data budget line and a procurement function.	USD 4,000–50,000 per seat per year, multi-year.	The contract is the product decision. Below a certain firm size the answer is simply no.
Mid-market research **Koyfin, YCharts, AlphaSense**	Advisers, wealth managers, boutique research shops.	USD 400–6,000 per user per year.	Read-only. Excellent charts and screens; no execution, no backtest engine, no notebook, no automation.
Retail charting **TradingView and peers**	Active traders.	USD 15–200 per month.	Built around the chart. Research, portfolio analytics and macro data are thin or absent.
Do it yourself **Python, a vendor API, a broker portal**	Quants, engineers, anyone technical enough to assemble it.	Cheap in cash, expensive in weeks.	You own the integration forever, and reproducibility is maintained by discipline rather than by architecture.

The gap is the diagonal: a firm that needs the *capability* of tier one and can only sign the *contract* of tier three. That is a two-person credit fund, a family office, an emerging-markets research boutique, a systematic trader running their own book, a doctoral researcher. Today they buy tier two and rebuild the missing half in Python.

Fincept is tier-one capability at tier-three procurement. That is the entire positioning, and every trade-off in the product follows from it.

02 What each one costs

Prices as reported by procurement-benchmarking sources for 2026. Vendor pricing is negotiated, so treat every row but the last as a range rather than a quotation.

Platform	Reported price	Term	Annual, one seat
Bloomberg Terminal	USD 31,980 / seat / yr single-terminal; USD 28,320 multi-terminal	Two-year minimum, billed quarterly in advance	**~USD 32,000**
LSEG Workspace	~USD 22,000 / yr typical; reduced configurations reported from ~USD 3,600	Annual, role-based tiers	**~USD 22,000**
FactSet Workstation	From ~USD 4,000 / yr base to ~USD 50,000 loaded; reported median ~USD 27,500	Annual, modular add-ons	**~USD 27,500**
Koyfin	USD 39 / user / mo Plus; USD 79 / user / mo Pro	Monthly or annual	**~USD 950**
YCharts	Not published; third-party estimates ~USD 3,600–6,000 / yr	Annual	**~USD 4,800**
TradingView	Roughly USD 15–200 / mo across tiers; repriced April 2026	Monthly or annual	**~USD 600**
Fincept Terminal	**USD 99 / 199 / 299 per month.** Quarterly billing takes 15 per cent off	**No minimum term**	**USD 1,188 – 3,588**

3.7%
OF A BLOOMBERG SEAT, AT OUR ENTRY TIER

11%
OF A BLOOMBERG SEAT, AT OUR TOP TIER

0
MONTHS MINIMUM COMMITMENT

15%
OFF IF BILLED QUARTERLY

What the comparison does not capture. An institutional seat includes licensed real-time exchange feeds, a global sales and support organisation, and decades of curated fundamental history. We include none of those things, and Section 06 says so plainly. The point of this table is not that we are the same product at a discount — it is that for a large population, the alternative to us is not a cheaper terminal, it is *no terminal*.

03 Against Bloomberg

HEAD TO HEAD · 03

The comparison everyone asks for. Every row we lose is left in, because a scorecard with no losses is a brochure and gets read as one.

	Bloomberg Terminal	Fincept Terminal
Price, one seat, one year	~USD 32,000	**USD 1,188–3,588**
Minimum term	Two years	**None**
Fixed income & OTC depth	**Definitive**	**Analytical only**
Licensed real-time consolidated feeds	**Yes, global**	**Via your own broker / exchange link**
Curated fundamental history	**Decades, restated**	**Public filings & open sources**
The messaging network	**The moat**	**None**
Compliance archiving & supervision	**Yes**	**No**
Sell-side research distribution	**Yes**	**No**
Bitemporal macro history	**Separate product**	**How the store is built**
Multi-broker retail execution	**Institutional routing**	**22 brokers, 17 Indian**
Native crypto venue coverage	**Limited**	**14 exchanges, direct**
Backtesting in the product	**Separate tooling**	**6 engines, 110+ strategies**
Notebook & spreadsheet in-window	**External**	**Both, in-process**
Read your own databases	**No**	**118 connectors**
Bulk export of what you queried	**Capped, undisclosed**	**CSV & Parquet**
Agentic AI over the product itself	**Vendor data, 2026 beta**	**790 tools, whole terminal**
Support organisation	**24/7 global**	**Email & in-app**

Read honestly, the table says one thing. Bloomberg wins every row that depends on being Bloomberg — licences, network, curation, forty years of accumulated correctness. We win every row that depends on *not* being locked.

04 Against the mid-market

HEAD TO HEAD · 04

This is the tier most of our buyers are actually on today, and the comparison is much less balanced. The mid-market platforms are excellent at what they do and structurally cannot do the rest of it: they are browser products with no execution path, no compute layer and no local runtime.

Capability	Koyfin / YCharts / AlphaSense	TradingView	Fincept
Screening & charting	**Strong**	**Best in class**	**Yes**
Fundamentals & filings	**Strong**	**Thin**	**Live EDGAR pipeline**
Place a real order	**No**	**Via partner brokers**	**22 brokers, 14 venues**
Paper trading on the live engine	**No**	**Simulated**	**Same engine as live**
Backtest an actual strategy	**No**	**Pine only**	**6 engines**
Deploy an automated strategy	**No**	**Alerts only**	**Paper and live**
Portfolio & risk analytics	**Basic**	**No**	**Nine analytic views**
Options & volatility surfaces	**No**	**Chain only**	**Native C++ workbench**
Python notebook	**No**	**No**	**In-window**
Spreadsheet	**Excel add-in**	**No**	**In-window**
Macro / economic data	**Curated set**	**Curated set**	**Millions of indexed series**
Connect your own database	**No**	**No**	**118 connectors**
Agentic AI across the product	**Search & summarise**	**No**	**790 tools, 53 agents**
Runs natively, works part-offline	**Browser**	**Browser**	**Native binary**

The honest summary: if the job is to look things up and chart them, the mid-market is cheaper than us and perfectly good. The moment the job includes acting — testing a hypothesis, sizing a position, placing the order, automating it and writing the note — the browser tier stops and the price gap closes to nothing.

05 Four things only we do

Not “we do it better” — claims we can point at in the code, where no competitor at any price does the same thing in the same product.

1 — EVERY SCREEN IS OPEN ON EVERY PAID PLAN

The entire industry ladder is built on withholding screens. Ours is built on withholding *depth*: a base-tier subscriber opens all thirty-seven destinations and gets real data in each one, and the plan changes how many watchlists, how far back, how many concurrent streams, which specific actions. The rule is written into the access layer — prefer a quota to a capability, and a capability to a feature — because a screen that renders blank teaches the buyer the product is broken, while a screen that renders full and says *3 of 3 · Exclusive+ raises this to 25* sells the next tier every time it is opened.

2 — THE AI DRIVES THE TERMINAL, NOT A COPY OF IT

The assistant reaches seven hundred and ninety built-in tools over the Model Context Protocol — the same operations the buttons perform. It can pull the quote, run the screen, price the option, read your portfolio, fetch the filing and write the note, and you can switch any individual tool off. Fifty-three specialist agents and six team modes run on top of that surface. You can also add your own MCP servers, so the agent reaches your systems, not only ours.

3 — THE HISTORY CANNOT BE QUIETLY REWRITTEN

Every observation in the indexed store is bitemporal: the value, the date it describes, and the date we learned it. A revision adds a vintage and never overwrites one, so no plan — including the cheapest — can be served a restated number dressed as an original. Price adjustment for splits and dividends is likewise computed at read time against the corporate actions known as of the date you asked about, never stored as a column. Querying the store *as of* a past date is a Pro capability; storing it correctly for everybody is not a tier, because a store that loses its own history cannot be fixed by upgrading.

4 — A NEW DATA PROVIDER SHIPS WITHOUT A RELEASE

Providers are declarative manifests, not code. Adding a national statistics office, a central bank or a government open-data portal is a configuration file and a crawl — no new endpoint, no migration, no client update. That is why the catalogue grows between versions instead of at them.

06 Where they beat us

If any of the following is load-bearing for your work, buy the incumbent. We would rather you read this page than discover it in month two.

BUY BLOOMBERG OR LSEG INSTEAD IF…

- **You trade fixed income, credit or OTC derivatives professionally.** Evaluated pricing, dealer runs and the liquidity picture are their business and are not reproducible from public sources.
- **You need the messaging network.** Three hundred and twenty-five thousand professionals on one system is not a feature we can build; it is the reason the price holds.
- **You are regulated and need supervised communications or a books-and-records archive.** We have no compliance module and no plans to pretend otherwise.
- **You depend on licensed consolidated real-time feeds across many venues.** Ours come through your own broker and exchange connections, which is excellent for the venues you trade and nothing at all for the ones you do not.
- **You need decades of restated, analyst-curated fundamentals.** We read filings; we do not maintain a forty-year normalised financial database.

BUY FACTSET OR S&P INSTEAD IF…

- **Your workflow is institutional portfolio analytics and attribution against a committee-approved benchmark set**, with the vendor's own risk model.
- **You need cross-vendor entity concordance as a contractual product** — a symbology master maintained and warranted by someone else.

BUY KOYFIN, YCHARTS OR TRADINGVIEW INSTEAD IF…

- **You only need to look and chart.** If nothing downstream of the screen matters — no execution, no backtest, no automation, no code — they cost less and will serve you well.
- **You want zero installation.** They run in a browser. We are a native application you install, which is the source of the speed and also the cost of it.

> **What none of them will do.** Give you a single window in which the research, the model, the order, the automation and the written output share one data layer — for less than four per cent of a terminal seat. That is the trade we are asking you to make.

07 How to choose

Five questions. Answer them honestly and the decision usually makes itself.

Ask yourself	If yes	If no
I already control a terminal seat my firm pays for.	Keep it. Add us for what it will not do — automation, notebooks, crypto venues, your own databases.	Read on.
My work stops at looking things up.	The mid-market tier is cheaper and sufficient.	Read on.
I trade, test, automate or write — not just read.	This is the product, from USD 99 a month.	Reconsider whether you need a terminal at all.
My universe is emerging-market, crypto or macro-driven.	The coverage argument is strongly ours — the one place we are deeper than a seat costing thirty times more.	We are comparable; the argument reverts to price.
I am regulated and need supervised archiving.	Buy the incumbent. We do not do this.	Nothing here disqualifies us.

THE PRAGMATIC ANSWER

A very common outcome is that both are correct. A desk keeps one shared institutional seat for the licensed feeds and the network, and gives every analyst a Fincept licence for the daily work — the research, the modelling, the backtesting, the automation and the writing. One Bloomberg seat pays for twenty-six Fincept Exclusive licences, or nearly nine Pro ones. That is not a competitive comparison so much as an arithmetic one.

The question is not whether we are better than a Bloomberg Terminal. It is whether the work you do actually requires one — and for a very large number of capable people, the honest answer has been no for years, and nobody has been selling to them.

Sources for the market figures in this document: Burton-Taylor International Consulting (TP ICAP) for industry spend; published 2026 procurement and benchmarking data for per-seat pricing; vendor announcements for the 2026 agentic-AI releases. Fincept prices are exact and taken from the billing service's own plan catalogue.

THE CAPABILITIES

Thirty-seven destinations.

An upper-level tour of everything the terminal does — markets, trading, portfolio, research, AI and the workspace — and how the pieces connect to each other.

Contents

8 SECTIONS

THE SHAPE OF IT

01 **One window, six groups** How thirty-seven screens are organised 3

THE DESKS

02 **Markets & data** Dashboard, Markets, Watchlist, News, Global Data 4

03 **Trading** Equities, crypto, derivatives, algos, prediction 5

04 **Portfolio & risk** Holdings, analytics, backtesting, surfaces 6

05 **Research & intelligence** Equity, M&A, alternatives, geopolitics, maritime 7

06 **AI & quant** The assistant, the agents, the lab, the nodes 8

07 **Tools & workspace** Notebook, spreadsheet, reports, files, notes 9

THE JOIN

08 **What makes it a terminal** Why the pieces are worth more together 10

About this document. A survey, not a manual. Each screen gets a paragraph and the numbers that matter; the full screen-by-screen walkthroughs are separate product guides, one per tab, and ship with the terminal as in-app documentation. Counts here come from the registries that build the interface — the navigation catalogue, the broker and exchange registries, the connector registry and the tier tables.

THE FIVE DOCUMENTS IN THIS SET

01	The Terminal	What it is, why it exists, and what it is not
02	Where It Stands	The landscape, and an honest comparison
03	**The Platform**	Every capability, screen by screen
04	The Data	Prices, the indexed store, providers, coverage
05	Plans & Pricing	Three tiers, the credit meter, what changes

01 One window, six groups

Thirty-seven destinations, grouped by the job they belong to rather than by the technology behind them. Fourteen sit on the tab bar; the rest are one keystroke away in the command bar or the Navigate menu.

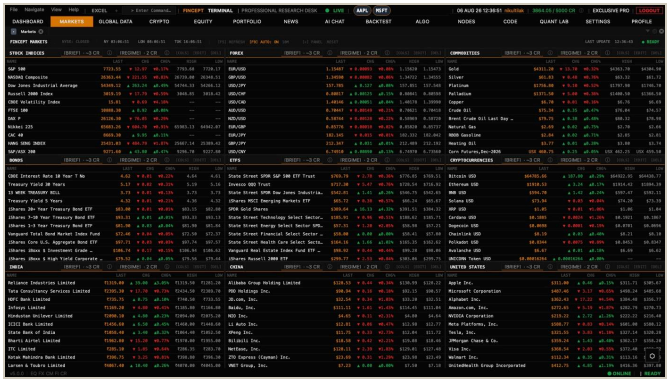
Group	Screens	What the group is for
Markets & Data	6	Dashboard · Markets · Watchlist · News · RSS Feeds · Global Data
Trading	6	Equity Trading · Crypto Trading · Algo Trading · Derivatives · Prediction Markets · Alpha Arena
Portfolio & Risk	3	Portfolio · Backtesting · Surface Analytics
Research & Intelligence	6	Equity Research · M&A Analytics · Alt. Investments · Relationship Map · Geopolitics · Maritime
AI & Quant	5	AI Chat · AI Quant Lab · Agent Config · MCP Servers · Node Editor
Tools & Workspace	8	Data Sources · Data Mapping · Code Editor · Excel · Report Builder · Trade Viz · File Manager · Notes
Account	3	Docs · Profile · Settings

Every screen is a dockable panel. Tabs can be torn off into their own windows, pinned to the toolbar, or arranged into saved layouts — and the layout follows the account, so a second machine opens where the first one left off.

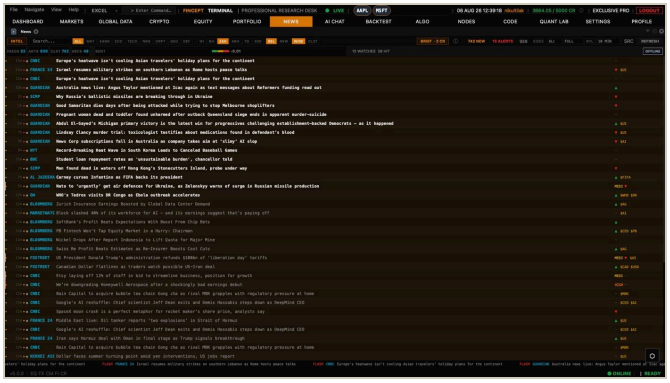
> **All of them open on every paid plan.** The tier changes depth — how many watchlists, how far back the history goes, how many concurrent streams, which specific actions are permitted — and never whether a screen works. See Document 05.

02 Markets & data

Where the day starts. Six screens covering the tape, the wire and the world's economic record.



Markets — a grid of streaming quote panels the user arranges, across equities, FX, commodities, bonds and crypto. Every quote, column and layout is free permanently.



News — roughly sixty sources in one dense wire with priority tiers, sentiment, clustering and keyword monitors that keep running while the terminal is shut.

Screen	What it does
Dashboard	A home board assembled from thirty-eight live widgets across six categories, with a Market Pulse rail and a streaming ticker. Multiple named layouts.
Markets	A quote-monitoring board of streaming panels. Two priced actions sit beside it and neither is the data: an AI written brief over the snapshot, and a cross-sectional regime scan that stores yesterday's reading so trend is answerable.
Watchlist	Many named lists in a live table or card grid, priced from your own broker stream. Server-side price alerts fire while the application is closed.
News	A live wire with source tiering, sentiment scoring, story clustering and saved keyword monitors. Monitors are evaluated server-side so a move is not missed overnight.
RSS Feeds	A board of resizable panels that turns any RSS, Atom, XML, JSON or CSV feed into a self-updating tile.
Global Data	One explorer over the platform's indexed economic store — search, a subject tree, a two-chart stage, transforms, a data table and CSV export. Ask for a series in plain English and it finds it.

03 Trading

Six screens covering equities, futures and options, crypto, automated strategies and prediction markets — all on the same order and position model, in paper or live mode.

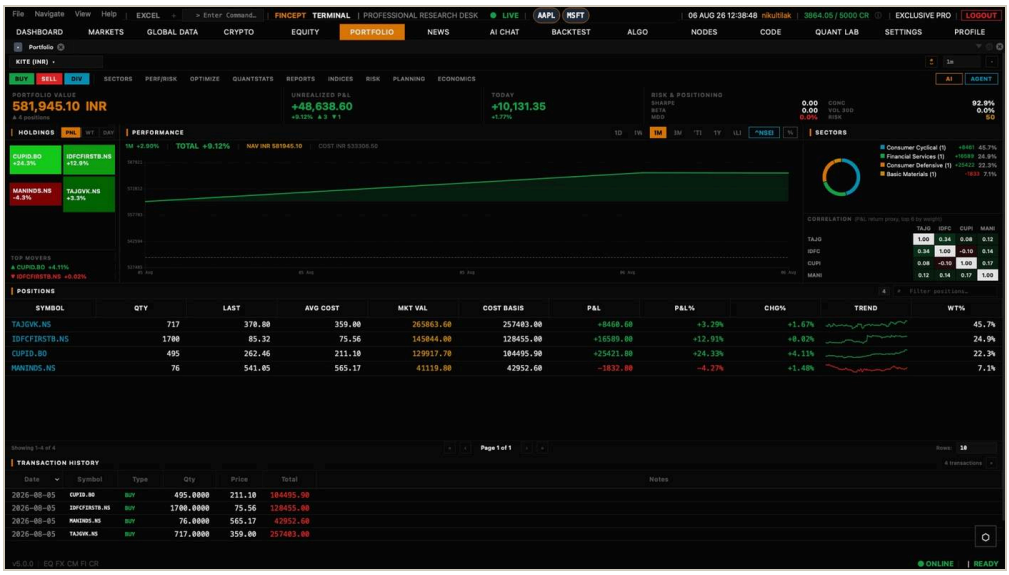
22	14	17	1
BROKERS, NATIVELY INTEGRATED	CRYPTO EXCHANGES, DIRECT	OF THOSE BROKERS INDIAN	ENGINE BEHIND PAPER AND LIVE

Screen	What it does
Equity Trading	A multi-broker desk for stocks and F&O: watchlist, chart, order book, ticket and blotter, working across every linked account at once. Brackets and stop-loss / take-profit are held server-side so they survive the client closing.
Crypto Trading	A full workstation over fourteen exchanges connected natively rather than through a bridge — chart, order book, depth, tape, ticket and blotter, spot and perpetuals, in paper or live mode.
Algo Trading	Build strategies as plain-English sentence rules or by conversation with the AI, backtest them, scan the market for candidates, then deploy to paper or live. Circuit breakers, cooldowns and a stale-feed watchdog are built in, and a deployment that cannot pay its runtime is blocked from new entries while every exit stays armed.
Derivatives	A native C++ options and volatility workbench — eight modules around one underlying's live chain, with forty-one models and no Python round trip.
Prediction Markets	Browse odds, order book, chart and tape across prediction venues. Free to browse.
Alpha Arena	A live competition in which language models trade crypto perpetuals against each other on real market data, round by round, with a full audit trail.

> **Paper trading runs on the live engine.** Same matcher, same brackets, same settlement sweep — virtual money against real prices, with resting limit orders filled by a worker while the terminal is shut. A simulator that behaves differently from production teaches the wrong lesson.

04 Portfolio & risk

Three screens: what you hold, how it would have behaved, and what the cross-sectional picture looks like.



Portfolio — a live position monitor with nine institutional analytics views and a built-in AI analyst that reads pre-computed analytics rather than guessing from a position list.

Screen	What it does
Portfolio	Multiple portfolios, live marking from the broker feed, nine analytic views (allocation, performance, risk, factor, attribution and more), broker position import, end-of-day attribution, and a unified all-accounts monitor.
Backtesting	Six engines, over a hundred and ten built-in strategies, eleven commands and five result windows. Costs and slippage are modelled rather than assumed, and an optimiser sweeps parameters on the upper tiers.
Surface Analytics	Thirty-five quantitative surfaces across seven asset classes, rendered in 3D. The equity and crypto volatility families run on live data; the fixed-income, FX, credit and macro families are labelled DEMO in the interface and are illustrative.

> **Backtests cannot see the future, by construction.** The indexed store keeps every revision as its own vintage with the date it was published, and split and dividend adjustment is computed at read time against the corporate actions known as of the test date. A strategy tested to March sees March's numbers, not today's restatement of them.

05 Research & intelligence

Six screens that answer the question behind the trade rather than the trade itself — and the two that most buyers do not expect to find at this price.

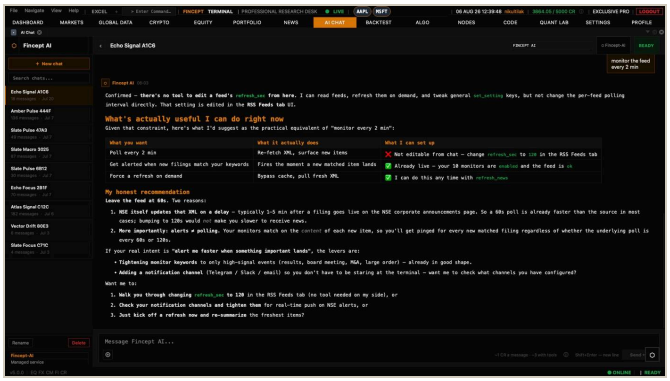
Screen	What it does
Equity Research	A single-symbol analyst workstation: fifteen sections spanning fundamentals, valuation, quality, technicals, peers, ownership and cross-asset, on live market data and a shared SEC EDGAR filing pipeline. Forty-five of the AI's tools point here.
M&A Analytics	A native C++ deal workbench — seven modules over one live merger with forty-two models, a live EDGAR filing scan, and a screened comparable set assembled server-side out of filings.
Alt. Investments	An LP-program workbench: commitments, performance and PME, pacing, liquidity, exposure, stress, a deal lab, an instrument lab and a proxy market. All in-process.
Relationship Map	Entity graphs built from real SEC Form 4 and 8-K filings — insider edges, ownership links, and expandable hops out from any node.
Geopolitics	A live world map of geopolitical events on the platform's own crawler-populated event tables, with humanitarian data, a relations reference and a trade-policy workbench.
Maritime	Live vessel tracking on a world map by sea area or single hull, with voyage replay, port lookup and seven analytic workspaces over corridors, ports and flag states.

Geopolitics and Maritime are not decoration. When the question is why a commodity moved, the answer is frequently a hull in a strait or a decision in a capital, and no mid-market platform will show you either.

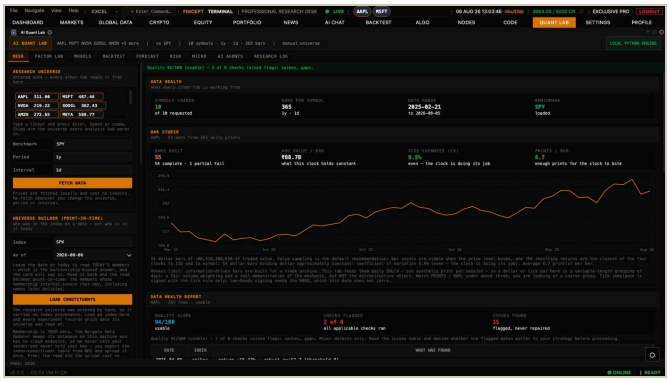
06 AI & quant

The part of the product that is genuinely new rather than cheaper. The assistant is not a chat window beside the terminal — it operates the terminal.

790	53	6	110
BUILT-IN TOOLS OVER MCP	SPECIALIST AGENTS	MULTI-AGENT TEAM MODES	NODE TYPES IN THE WORKFLOW EDITOR



AI Chat — the assistant answers with sources and shows which tools it called. Summon it over any screen with F9.

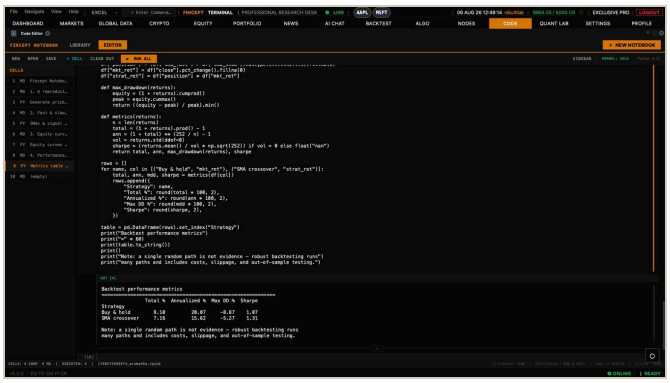


AI Quant Lab — eight workflow tabs over one shared universe: set the tickers once on the desk, then factor-test, train, backtest, forecast, risk-check and automate.

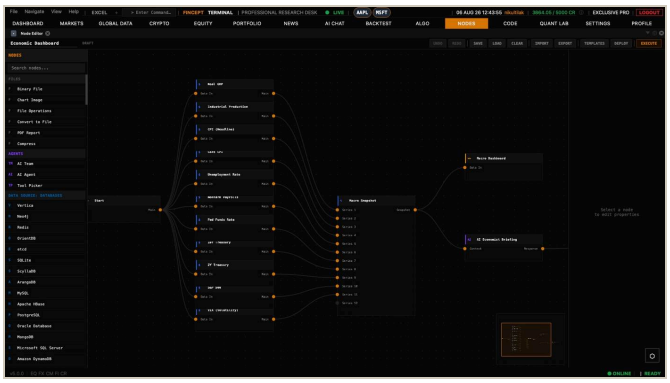
Screen	What it does
AI Chat	An agentic research assistant with a tool-calling loop, cross-turn memory, streamed answers, voice in and out, and a citation trail showing every tool it used.
AI Quant Lab	A research workbench in eight workflow tabs, plus an append-only trial ledger: the number of experiments you have run deflates the Sharpe of the one you are about to publish, and deleting research cannot reset the count.
Agent Config	Build, configure and orchestrate agents — fifty-three personas across nine sub-tabs, six team modes, and desks that bind an agent to a screen.
MCP Servers	Add external Model Context Protocol servers so the agent reaches your own systems, and switch any of the seven hundred and ninety built-in tools on or off individually.
Node Editor	A visual workflow builder — one hundred and ten node types across fourteen categories, wired by typed ports, from market-data fetchers through analytics and safety checks to order placement.

07 Tools & workspace

Eight screens that exist because the alternative was leaving the terminal — which is where the copy-paste errors come from.



Code — a Jupyter-style Python notebook with a curated library of ready-to-run finance and quant notebooks.



Node Editor — typed ports, one hundred and ten node types, run interactively or deploy as a standing pipeline.

Screen	What it does
Data Sources	One hundred and eighteen built-in connectors across eleven categories — relational and NoSQL databases, warehouses, time-series stores, object storage, streaming buses, market-data vendors, open banking and file formats.
Data Mapping	A five-step wizard that turns any JSON API into one of seven standard schemas, with fourteen verified templates and a live test harness. Mapped feeds can be refreshed on a schedule.
Code Editor	A Python notebook and editor with a curated starter library.
Excel	A real multi-sheet spreadsheet with a formula bar and .xlsx / .csv import and export — in-process, not an add-in.
Report Builder	A block canvas with live data blocks and a single compiler, so the preview is exactly the exported PDF. Twelve templates; your own branding on the upper tiers.
Trade Viz	Visual reconstruction of trade sequences against price.
File Manager	A cloud file library with a durable upload queue and an evictable local cache, or a purely local library if you prefer nothing leaves the workstation.
Notes	A three-panel research and trade journal, categorised, prioritised and sentiment-tagged — and readable by the AI agent.

08 What makes it a terminal

THE JOIN · 08

A list of screens is not a terminal. What makes it one is that they share a data layer, so work done in one is context in the others. Five joins do most of that work.

THE PORTFOLIO IS THE UNIVERSE

Positions built in the Portfolio screen are what the backtester tests against, what the risk views price, what end-of-day attribution decomposes, and what the AI reads when asked how the book is doing. Nothing is re-entered.

ONE ORDER MODEL, FOUR VENUES

Equity, F&O, crypto and algorithmic orders share one order and position model, one blotter and one reconciliation path. A paper fill and a live fill differ in where they were routed, not in how they are recorded.

ONE INDEXED STORE, EVERY SCREEN

The same bitemporal observation store answers the Global Data explorer, the macro context in Equity Research, the scheduled refresh that runs overnight, the notebook's data calls and the agent's tool calls. One provider added is one provider everywhere.

ONE TOOL CATALOGUE

The seven hundred and ninety built-in tools are the same surface used by AI Chat, by every agent and team, and by the node editor. A capability added for one is available to all three, and any of them can be switched off in one place.

ONE ACCOUNT, ANY MACHINE

Layouts, watchlists, notes, notebooks, reports, dashboards, monitors and feeds live in the account. A new workstation signs in and finds its desk already set up.

Any one of these screens exists somewhere else, usually done well. What does not exist somewhere else is all of them behind one login, sharing one book, one catalogue and one tool surface — for a hundred dollars a month.

THE DATA

What we actually hold.

Live prices, the indexed economic store, instruments and issuers, filings, ships, conflict, trade, infrastructure and your own systems — what is in each, how much of it there is, and where the coverage stops.

· Contents

16 SECTIONS

THE SHAPE OF THE DATA

01 **Seven kinds of data** How to read everything that follows 3

02 **Live prices** Where a quote on your screen comes from 4

THE INDEXED STORE

03 **The economic catalogue** Sixty-seven million labelled series 5

04 **Who publishes it** Fifty-eight institutions, named and counted 6

05 **What the numbers describe** Subjects, places, frequencies, history 7

06 **Why it is bitemporal** The correctness property nobody advertises 8

07 **How the catalogue grows** A provider is a file, not a release 9

INSTRUMENTS, COMPANIES AND VENUES

08 **Instruments and issuers** Nine million securities, and who issued them 10

09 **Filings and company facts** The full EDGAR corpus, pulled once for everyone 11

THE WORLD BEHIND THE PRICE

10 **Ships and ports** 776,479 vessels, three years of positions 12

11 **Conflict and political events** 2.4 million dated, placed events 13

12 **What the world trades** Thirty years of bilateral flows 14

13 **Physical infrastructure** The plants, pipes, cables and crossings 15

14 **The wire and the odds** News, monitors and prediction markets 16

YOUR OWN DATA

15 **Bring your own data** 118 connectors and a mapping wizard 17

16 **Where coverage stops** The limits, stated plainly 18

> **About this document.** Data is the claim buyers check hardest and the one vendors are loosest about, so this document separates seven things that usually get merged: what streams live, what we index ourselves, what we hold as reference, what we proxy, what we crawl, what you bring, and what is a demonstration. Every count in it is read from the running system, not estimated.

THE FIVE DOCUMENTS IN THIS SET

01 The Terminal — What it is, why it exists, and what it is not

02 Where It Stands — The landscape, and an honest comparison

03 The Platform — Every capability, screen by screen

04 **The Data** — Prices, the indexed store, providers, coverage

05 Plans & Pricing — Three tiers, the credit meter, what changes

01 Seven kinds of data

Almost every argument about market-data coverage is really an argument about categories being merged. A vendor says “millions of data points” and means five different things at once. Here are the seven, kept apart, with what we do in each.

Kind	What it means	Where ours comes from
Live prices	A quote, a book, a tape, right now.	Your own broker and exchange connections — 29 brokers, a MetaTrader bridge and 14 crypto venues, streamed straight to the workstation.
Indexed time series	The economic and statistical record — rates, prices, trade, energy, labour, fiscal.	Our own bitemporal store: 67.5 million labelled series crawled from 58 national agencies, central banks and supranational bodies.
Reference & entities	What an instrument is, who issued it, where it trades, who owns whom.	9.1 million securities mapped to 98,569 issuers, 485,000 corporate and fund relationships, and 2,875 registered trading venues.
Filings & fundamentals	What a company reported, in its own words and its own numbers.	The whole public US filing corpus through one shared, cached pipeline, plus public market-data sources.
News & text	The wire, and what it means.	An RSS/Atom corpus over 53 sources in six categories, scored for sentiment, clustered by story, with server-side keyword monitors.
Alternative	Signals that are not prices — ships, conflict, trade, infrastructure, odds.	Crawlers we operate: a 776,479-vessel register, 2.4 million political events, 270 million trade flows, 318,000 infrastructure assets, three prediction venues.
Yours	The data you already pay for or already own.	118 connectors into your databases, warehouses, object stores, streams and vendor APIs.

> **The distinction that matters most.** We do not resell a licensed premium vendor feed. Where a competitor's value is a redistribution licence, ours is the indexing, the correctness and the joins — and the fact that your own already-paid-for broker feed is a first-class citizen rather than an afterthought.

02 Live prices

A price on your screen arrives one of three ways, and it is worth knowing which, because they have different latency, different entitlement and different cost to you.

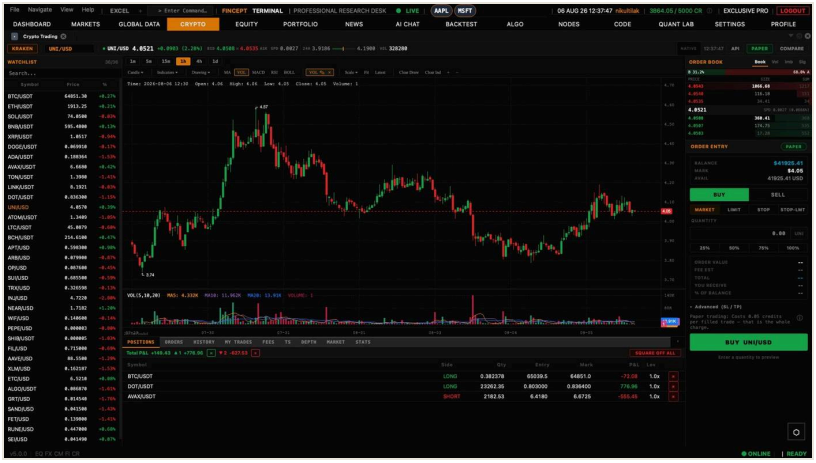
Route	Covers	Notes
Your broker's stream	Equities, futures, options, currencies and CFDs on the exchanges your broker serves.	Real time, at the entitlement you already hold with them. Twenty-nine brokers connected directly — seventeen Indian, seven US, five European and global — plus a MetaTrader 4/5 bridge.
Exchange websockets	Crypto spot and perpetuals across fourteen venues.	Connected natively, not through a third-party bridge library. Order book, depth, trades and funding all stream directly.
Public market data	Everything you have not linked an account for — equities and funds on the major exchanges worldwide, indices, FX, commodities, cross-asset context, historical candles.	Public sources, cached and normalised through the platform. Adequate for research and context; not a substitute for a licensed consolidated feed.

29
BROKERS, PLUS A METATRADER BRIDGE

14
CRYPTO VENUES, NATIVELY CONNECTED

500
CONCURRENT STREAMING SYMBOLS, TOP TIER

0
CREDENTIALS STORED ON OUR SERVERS



Crypto Trading — chart, order book, depth ladder, tape, ticket and blotter over one of fourteen natively-connected venues.

> **Options and volatility.** Chains and surfaces are driven by live options feeds on the upper tiers. The models that consume them run in-process, in C++.

03 The economic catalogue

The part of the product we built rather than bought. A single indexed store of labelled, dated statistical series, crawled from the institutions that publish them and served from our own infrastructure — so a read never waits on a third party.

67.5M	19,111	58	10,689	1,371
SERIES IN THE CATALOGUE	DATASETS INDEXED	PUBLISHING INSTITUTIONS	DISTINCT GEOGRAPHIES	DISTINCT UNITS OF MEASURE

Ten institutions account for four fifths of it. The rest of the list — another forty-eight, from national statistics offices to central banks to energy regulators — is on the facing page, because a catalogue's credibility is the long tail, not the headline.

Largest ten publishers	What they cover	Series
ILOSTAT	Labour: employment, hours, wages, informality, safety, by country and sex	15,895,123
Eurostat	The whole European statistical programme — 1,311 dataflows	12,226,524
IMF Data	Balance of payments, trade, government finance, financial soundness, portfolio positions	8,921,267
Australian Bureau of Statistics	The full Australian national programme	8,916,945
US Bureau of Labor Statistics	Employment, prices, productivity — 65 survey families	8,287,846
World Bank	Development, demographic and economic indicators by country	5,103,564
Bank for International Settlements	Credit, debt securities, property prices, effective exchange rates, derivatives	1,158,476
European Central Bank	Rates, money and banking, payments, supervision, financial vehicles	937,782
FRED / ALFRED	US and international macro, with full revision history on every series	845,503
Climate TRACE	Asset-level greenhouse-gas emissions by facility and sector	732,713

> **Nothing is invented.** Every series carries its source, its unit, its frequency, its geography where the provider publishes one, and the taxonomy the provider itself files it under. If a provider does not state a unit, the field is empty rather than guessed.

04 Who publishes it

Fifty-eight institutions, each crawled directly and served from our own store. Grouped by where they publish from, with what we currently hold from each.

Group	Institutions	Series
Global & supranational 14 bodies	ILOSTAT · IMF Data · IMF DataMapper · IMF PortWatch · World Bank · BIS · OECD Data Explorer · OECD Main Economic Indicators · WTO · UN SDG Database · WHO Global Health Observatory · Climate TRACE · Ember · JODI	32,547,539
Europe 18 bodies	Eurostat · European Central Bank · European Banking Authority · Deutsche Bundesbank · Bank of England · Norges Bank · Statistics Norway · Statistics Sweden · Statistics Finland · Statistics Denmark · Statistics Netherlands · Statistics Iceland · Swiss Federal Statistical Office · STATEC Luxembourg · UK Office for National Statistics · France DVF property register · ENTSOG · Energy-Charts	15,079,044
United States 12 bodies	Bureau of Labor Statistics · FRED / ALFRED · Treasury Fiscal Data · TreasuryDirect · Bureau of Economic Analysis · Census economic indicators · Energy Information Administration · FDIC BankFind · CFTC Commitments of Traders · Federal Housing Finance Agency · Federal Reserve Bank of New York · USDA Agricultural Marketing Service	10,441,644
Asia-Pacific 6 bodies	Australian Bureau of Statistics · Taiwan Stock Exchange · Taiwan Futures Exchange · Taipei Exchange · Singapore Department of Statistics · Bank of Japan	9,432,022
Americas beyond the US 8 bodies	Bank of Canada · Statistics Canada · Open Government Canada · Ontario Open Data · Ontario Fuel Prices · Banco Central do Brasil · Banco Central do Brasil Focus survey · Banco Central de la República Argentina	85,815

WHAT THAT MIX BUYS YOU

The heavyweights give breadth: labour, European statistics and the IMF programme are three quarters of the catalogue on their own. The small entries are why the store answers questions the big ones cannot — a Luxembourg tax series, a Taiwanese exchange statistic, an Icelandic price index, the Brazilian central bank's own forecast survey, a French property register at transaction level, gas flows through European transmission points. Fifty-two of the fifty-eight are open under attribution or public-domain terms; the remaining six carry share-alike or non-commercial conditions and are labelled as such in the product.

> **The list is not the ceiling.** Adding an institution is writing one file and running a crawl — no schema change, no client release, no downtime. Section 07 explains why that is structural rather than a promise.

05 What the numbers describe

A count of series says nothing about whether the one you need is in there. These are the four axes that answer that: subject, place, frequency and depth.

SUBJECT

Every series is filed under the taxonomy its own publisher uses, and mapped into twelve subject families with fifty-five sub-topics beneath them, so one search crosses providers without you knowing which one holds the answer.

Family	What sits under it
Labour & demographics	Employment, unemployment, participation, wages, productivity, population
External sector & trade	Trade in goods and services, balance of payments, foreign investment, reserves
Government & public finance	Debt, revenue, expenditure, deficits, issuance and auctions
National accounts	GDP and output, income, consumption, investment, flow of funds
Money, banking & credit	Money supply, central-bank operations, condition of banks, credit and lending
Prices & inflation	Consumer prices, producer and import prices, deflators
Financial markets	Interest rates and yields, equities, foreign exchange, bonds, spreads
Energy & commodities	Petroleum, gas, electricity, coal, renewables and nuclear, commodity prices
Housing & construction	House prices, starts, permits and sales, inventory, mortgages
Business & production	Manufacturing, services, retail and wholesale, industrial output, surveys
Environment & climate	Emissions, natural resources, agriculture and land
Health, education & social	Health, education, poverty and inequality, social protection, gender

PLACE, FREQUENCY AND DEPTH

48.5M
SERIES CARRYING A GEOGRAPHY

1,077
COUNTRY CODES IN USE

52.6M
ANNUAL SERIES; 7.4M MONTHLY

5.3M
QUARTERLY SERIES; 505K DAILY

845k
SERIES WITH FULL REVISION HISTORY

Sub-national geography is carried wherever the publisher files it — 10,689 place codes in all, from world aggregates down to European regions, US states and Norwegian municipalities. The earliest observation is dated 1086; the furthest-out projection runs to 2036.

> **Catalogue first, numbers on demand.** Every series is indexed and searchable from the moment its crawl lands; the observations behind one are fetched the first time somebody asks and kept from then on. The catalogue is complete while the value history fills in behind it.

06 Why it is bitemporal

This is the single most valuable property of the store and the hardest to retrofit, which is why it was built in on day one rather than added later.

THE PROBLEM

Economic data is revised. US GDP for a quarter is published, then revised, then revised again, sometimes years later. A store that keeps only the latest value has quietly destroyed the historical record: a backtest run against it sees numbers that did not exist on the day the strategy would have traded. The result is a strategy that looks profitable and is not, and nothing downstream can detect it.

WHAT WE DO INSTEAD

Every observation is stored with three dates — the date it describes, the date we learned it, and the date it was superseded. A revision adds a vintage; it never overwrites one. Ask for a series as of any past date and you get exactly what a subscriber would have seen on that date, with the answer stamped so you can prove it. Where a provider publishes no revision history, the store says the answer is approximate rather than implying precision it does not have.

The as-of query is an Exclusive Pro capability. The storage is not a tier and never will be: a store that overwrites its own history has destroyed something no upgrade can restore, so every plan is served out of the same uncorrupted record.

THE SAME ARGUMENT, APPLIED TO PRICES

An adjusted price is not a fact about a day. It is a fact about a day *and* the date you are standing on, because every future split rewrites the history behind it. So adjustment is computed at read time from the corporate actions known as of your chosen date, never stored as a column. An adjusted answer computed from no action history is refused rather than returned unadjusted under an adjusted label.

> The major vendors sell the vintage archive as a separate product, which is fair — it is expensive to keep. What should never be optional is keeping it at all, and that is the part we made structural.

> **Symbology.** A cross-provider join only works if the same company resolves to the same entity everywhere, including through ticker recycling and delistings. Identifier leases are dated, so a query about 2009 resolves the ticker as it stood in 2009 — the same discipline the majors sell as a concordance product.

07 How the catalogue grows

The reason to expect this catalogue to keep expanding is architectural rather than aspirational. Adding a provider does not touch the product.

A PROVIDER IS A MANIFEST

Each one is a declarative file describing its endpoints, its authentication by environment-variable *name* (never a secret), its rate limit, its paging, its crawl strategy and its field mapping. There is no code path per provider. Adding a national statistics office means writing that file and running a crawl — no new endpoint, no schema migration, no client release, no downtime. The read surface stays the same whether we hold fifty-eight providers or five hundred.

It works: the catalogue went from nine institutions and 10.5 million series to fifty-eight and 67.5 million without a single change to the terminal that reads it.

THREE RULES THE FRAMEWORK ENFORCES

Rule	Why it exists
Paging stops on an empty page, never on a reported count.	Providers over-report. One major source reports roughly four times its true series count; trusting it truncates the crawl and nothing looks wrong.
Observations are bitemporal; a revision adds a vintage.	The alternative silently injects lookahead bias into every backtest downstream.
A row that fails to map is dropped and counted, never invented.	Only counts and values are evidence that a crawl worked. “It ran without errors” is not.

THE TARGET

The stated goal for the catalogue is coverage of the financial and economic record of every country, plus everything that moves the price of any asset — so that a scenario analyst reaching for the driver of a shock finds a labelled, dated series already indexed, whether that driver is a rate decision, a drought, a port closure or an election. That is a long programme, and it is the one the architecture above exists to serve.

> **What this buys the buyer.** Coverage that grows without waiting for us to ship a version — and a catalogue whose breadth is bounded by what the world publishes rather than by what one vendor chose to licence.

08 Instruments and issuers

A price is only useful if you know what it is a price of. The reference plane answers that: what an instrument is, who issued it, who owns the issuer, and where the thing trades — all of it dated, so a question about 2009 gets 2009's answer.

9.14M
SECURITIES CARRYING AN ISIN

98,569
DISTINCT ISSUING LEGAL ENTITIES

484,925
OWNERSHIP AND FUND RELATIONSHIPS

2,875
REGISTERED TRADING VENUES, 149 COUNTRIES

WHAT THE NINE MILLION COVERS

Bonds, notes, equities, funds and structured issues — every security a registered issuer has brought to market, worldwide. It is an issuance universe rather than a listing one, and that is the join that makes a bond screen, a credit exposure map or an issuer-level filing search possible at all.

Relationship	What it tells you	Edges
Fund managed by	Which manager runs a fund — the backbone of any fund-flow or manager-exposure view	146,090
Ultimately consolidated by	The top of a group: who ultimately owns this entity	132,522
Directly consolidated by	The immediate parent, one step up	126,361
Sub-fund of	Umbrella structures broken out into their compartments	73,098
Operated by	Which entity operates a trading venue	2,246
International branch of	Cross-border branches tied back to their head office	1,938
Feeder to · Segment of	Feeder-master fund links, and venue market segments	2,670

VENUES

2,875 venue codes across 149 countries — 2,293 currently active, the rest expired and deliberately kept, because an instrument's historical trading venue still has to resolve. They break down as 1,309 exchange-level codes, 379 multilateral trading facilities, 349 systematic internalisers, 302 regulated markets, 142 alternative trading systems, 128 organised trading facilities, 30 swap execution facilities and 29 crypto-asset service providers.

> **Where this is honest.** Issuers self-report their instruments, so one that never reported leaves its bonds absent rather than unissued; and both ends of an ownership edge must be registered entities, so a parent without a registration reads as no parent. Both limits are stated in the product next to the answer, not buried here.

09 Filings and company facts

Every public US filing, and the structured financial facts inside them, reached through one shared pipeline rather than five tabs each scraping the same source from your own address.

What you get	Contents
The full filing corpus	Annual and quarterly reports, current reports, insider transactions, proxy statements, registration statements, prospectuses — every form type, back to the start of electronic filing
Company facts	The tagged financial statement line items an issuer has ever reported, as a time series per concept — revenue, margin, debt, cash flow — without you parsing a document
The filing documents	The original exhibits and attachments, fetched on demand and cached
The insider graph	Insider transaction reports assembled into a who-bought-what-when graph per company, rather than a stack of forms
The ticker map	Ticker to filer identity, so a symbol on your watchlist resolves to the entity that files

WHY IT IS ONE SHARED PIPELINE

Four surfaces read filings — Equity Research, the M&A desk, the Relationship Map and the AI agent — and before this they each fetched independently, from your machine, with their own ad-hoc caching. Centralising means one cache for the whole platform: the second person to open a filing this month gets it instantly and it costs them nothing, because the allowance pays for a fetch we actually make. Every plan includes a monthly pull quota; a cache hit never touches it.

> **And it is a proxy with a fixed menu, not a pass-through.** There is no generic “fetch this URL” route. Five explicit endpoints cover everything the product needs, over an allow-listed source — which is the difference between a proxy and an open redirect with our address attached.

> **Beyond the US.** Company fundamentals outside the United States come from public market-data sources rather than a filings pipeline, and are labelled that way. Building the equivalent regulator-level pipeline for other jurisdictions is on the roadmap and is not shipped.

10 Ships and ports

Physical trade is the part of the economy that moves slowly enough to watch and fast enough to matter. We run the crawlers, hold the positions and derive the analytics ourselves — so a question about a corridor is a query, not an API call to somebody else.

776,479	2.31bn	16.4M	2.72M	34,031
VESSEL IDENTITIES ON FILE	DEADWEIGHT TONNES ON THE FLEET	POSITION FIXES SINCE MAY 2023	VOYAGES RECONSTRUCTED	PORTS, IN 187 COUNTRIES

THE REGISTER, LAYER BY LAYER

Layer	What each one carries	Vessels
Identities on file	Name and transponder identity for every ship seen	776,479
Carrying an IMO number	The ocean-going merchant fleet	85,510
Full particulars	Type, flag, tonnage, dimensions, year built	52,648
Tracked fleet	Position history, voyages, port calls and analytics	49,446

Tracked fleet, by segment	Ships	Tracked fleet, by segment	Ships
Dry bulk	12,039	**Ro-ro and car carriers**	1,318
Container	6,291	**Passenger and cruise**	949
Product tankers	4,656	**Reefer**	443
General cargo	3,064	**Offshore and support**	246
Chemical tankers	2,879	**Gas carriers**	24
Crude tankers	1,754	**Type not stated at source**	15,783

WHAT IS DERIVED FROM IT

Positions on their own are dots on a map. What makes them answerable is the derivation that runs on top every fifteen minutes: the latest fix per ship, voyages with origin, destination, laden state and tonnage, daily traffic density, port call histories, country and flag rollups, and transit counts through eight chokepoints — Hormuz, Suez, Bab el-Mandeb, Malacca, Panama, the Turkish and Danish Straits and the Cape of Good Hope, 219,815 transits so far. 170 flag states and 91 source vessel types are represented, mapped into twelve segments; hulls date from 1960 to 2025.

> **Why we run our own crawlers here.** This was originally proxied to a third-party API, and a proxy means your coverage ends when somebody else's business decision does. Reading our own tables also means an unset connection answers with an explicit error rather than an empty success — the failure that actually costs people money, because it looks like “nothing happened today”.

11 Conflict and political events

A dated, placed, categorised record of what is happening politically — the same shape of dataset that specialist conflict-data vendors sell on its own, held here beside the prices and the economic series so the join is a click rather than a project.

2.42M	15	6,470	143
CLASSIFIED EVENTS ON FILE	EVENT CATEGORIES	CITIES LOCATED BY NAME	PUBLISHER SOURCES READ CONTINUOUSLY

Category	Events	Category	Events
Armed conflict	940,560	**Political violence**	45,317
Protests	351,497	**Diplomacy**	7,072
Crisis	297,935	**Elections**	2,089
Violence against civilians	296,326	**Maritime incidents**	1,715
Explosions	145,275	**Sanctions**	965
Strategic developments	119,967	**Cyber**	588
Riots	117,707	**Nuclear**	528
Terrorism	94,553		

WHAT ONE EVENT CARRIES

A date, a latitude and longitude with its precision stated, a city and country, a category, the publisher it came from, a confidence score, and a one-line summary of what actually happened — which reads far better in a table than a headline does. The record runs from November 2022 to today and grows continuously. Country, city, category and source pickers are free to browse: charging to discover what you may ask for is charging for the menu.

> **What this is and is not.** It is a classified record of what published reporting says happened, geolocated and dated. It is not field-verified casualty data and does not claim to be. Where the coordinate is a country centroid rather than a located town, the record says so rather than implying a precision it does not have.

12 What the world trades

Bilateral merchandise trade at product level — who sold what to whom, every year, for thirty years. It is the largest single table we hold, and the one that turns a tariff headline or a chokepoint closure into a number.

270M	30	231	5,018
BILATERAL TRADE RECORDS	YEARS, 1995 TO 2024	COUNTRIES AND TERRITORIES	PRODUCT CODES, SIX-DIGIT

Each row is one exporter, one importer, one product, one year, with value and quantity. Products roll up from six-digit lines through four- and two-digit headings to twenty-one sections, so the same corpus answers “India's pharmaceutical exports to the EU” and “who supplies the world's neodymium magnets” without a second dataset.

Also held alongside it	What it adds	Rows
Tariff schedules and projections	Applied and projected duty lines by country pair and product	20,942,220
World development indicators	The full cross-country development series, aligned to the same country codes	7,588,544
Services trade	Cross-border services flows by category, where goods data stops	2,097,574
Long-run trade classification	The older standard classification, for series that predate the current one	886,159
Economic complexity	Product complexity and revealed comparative advantage across five classification revisions	—
Institutional holdings	Registered investment managers and their reported positions	300,907

WHY IT IS OURS RATHER THAN PROXIED

Aggregating 270 million rows at request time is only possible if the rows are on our own infrastructure with our own indexes. A partner-ranking query that would be a rate-limited paginated crawl against somebody's public API is a single grouped read here — which is why the Trade view can redraw a country's whole partner history while you drag a slider.

> **One honest detail.** The country list contains regional aggregates as well as countries, and summing across them would double-count. Every aggregation in the product is scoped to real countries for exactly that reason — a mistake worth naming, because a chart that silently double-counts looks completely normal.

13 Physical infrastructure

The assets that carry the economy: where power is generated, where gas and oil flow, where data moves, where goods cross. Held as mapped geometry with capacity and operator, so a shock can be located rather than described.

318,056
ASSETS MAPPED

16
INFRASTRUCTURE LAYERS

230
COUNTRIES REPRESENTED

3
DOMAINS: ENERGY, TRANSPORT, TELECOM

Domain	Layer	Assets
Energy	Power plants, with fuel type and capacity	142,163
	Electricity transmission lines	52,244
	Gas pipeline routes, international	46,908
	Gas pipelines, US network	32,851
	Oil and refined-product pipeline routes	7,386
	Coal mines	3,522
	Hydrogen pipeline routes	209
Transport	Border crossings	8,159
	Airports	5,272
	Ports	2,926
	Maritime chokepoints	8
Telecom	Terrestrial fibre backbone routes	5,789
	Interconnection facilities	5,257
	Data centres	4,470
	Submarine cables	849
	Cable landing stations	43

Why this sits in a financial terminal. The chokepoint polygons that count ship transits are the same objects as these layers; the power plants carry the capacity that a generation series is measured against; the cables and landing stations are where a connectivity outage becomes a regional risk. Infrastructure is the join between a map and a number, and it is the layer most terminals leave out entirely.

14 The wire and the odds

Two feeds that are not prices and not statistics, and that carry a disproportionate amount of the signal: what is being said, and what is being wagered.

THE WIRE

Tier	Who is in it	Sources
Regulators and central banks	The sources a monitor exists to catch — securities regulators, the Federal Reserve, the ECB, and the like, publishing their own decisions	4
Major financial media	The global and national financial press, markets and world desks	18
Specialist and regional	Technology, crypto, energy and regional desks across the US, Europe, India and Asia	31

Fifty-three sources in six categories — markets, geopolitics, technology, economics, regulatory and crypto — polled into one shared corpus, scored for sentiment and clustered by story so twelve write-ups of one event read as one event. You can add your own feeds to the same pipeline, and they get the same scoring and clustering.

KEYWORD MONITORS

A monitor is a phrase you care about — a company, a commodity, a policy — evaluated by a server-side worker rather than by your machine. That is the whole reason it exists: a monitor that only fires while the terminal is open is a monitor that misses the thing you built it for. It fires while the terminal is shut, and the hit is waiting when you open it.

PREDICTION MARKETS

Odds, order books and tape from three venues, including the two largest regulated and decentralised prediction exchanges, connected natively with signed authentication and full order-book snapshot and delta streaming. Browsing is free and unmetered; live odds streaming is an upper-tier capability. Prediction symbols are first-class elsewhere in the terminal — they can be priced, charted and traded inside a competition alongside equities and crypto.

> **The pattern behind sections 10 to 14.** Every one of these feeds reads a table we populate and own. Nothing here is a live pass-through to a third-party API that can change its terms, its pricing or its existence — which is what makes the coverage a property of the product rather than of somebody else's roadmap.

15 Bring your own data

The data that matters most to a professional is usually already theirs. One hundred and eighteen built-in connectors across eleven categories exist so it can sit beside everything else rather than in another window.

Category	Count	Examples
Market data	25	Yahoo Finance, Alpha Vantage, Finnhub, IEX, Twelve Data, Nasdaq Data Link, Polygon, Tiingo, Intrinio, EOD Historical, FactSet, LSEG, PitchBook, OpenFIGI, Binance, Coinbase, Kraken, CoinGecko
Databases	23	PostgreSQL, MySQL, MariaDB, SQLite, Oracle, SQL Server, CockroachDB, Vertica, MongoDB, Redis, Cassandra, DynamoDB, Neo4j, Cosmos DB, Firestore, HBase, ScyllaDB
File formats	14	CSV, Excel, JSON, Parquet, XML, Avro, ORC, Feather, Arrow, Iceberg, Delta Lake, GeoParquet, FTP, SFTP
Alternative data	10	Sentiment, satellite, footfall, transaction-panel and workforce vendors
Object storage	10	S3, Google Cloud Storage, Azure Blob, R2, MinIO, Wasabi, Backblaze, DigitalOcean, Oracle, IBM
Time-series databases	8	InfluxDB, QuestDB, TimescaleDB, Prometheus, VictoriaMetrics, OpenTSDB, kdb+, ClickHouse
Streaming	8	WebSocket, MQTT, Kafka, RabbitMQ, NATS, Redpanda, Flink, FIX
APIs	5	REST, GraphQL, gRPC, SOAP, OData
Search & analytics	5	Elasticsearch, OpenSearch, Solr, Algolia, MeiliSearch
Data warehouses	5	BigQuery, Redshift, Databricks, Synapse, Snowflake
Open banking	5	Plaid, MX, Finicity, TrueLayer, FDX

Anything not on that list is still reachable: the Data Mapping wizard turns an arbitrary JSON API into one of seven standard schemas in five steps, with fourteen verified templates and a live test harness. A mapped feed can then be refreshed automatically on a schedule, so a source you defined once keeps arriving without you.

> **And it goes out as well as in.** Series and query results export as CSV, and the indexed store exports in bulk as Parquet on the upper tiers. Data you can only look at is not data you own.

16 Where coverage stops

The page a competitor would leave out. Every item below is something a buyer would discover within the first few weeks.

NOT IN THE PRODUCT

- **No licensed premium vendor feed.** We do not redistribute a commercial market-data licence. On the roadmap; not shipped.
- **No evaluated pricing for illiquid instruments.** No dealer runs, no consensus marks on OTC credit or structured products.
- **No curated multi-decade fundamentals database.** We read filings and public sources; we do not maintain a forty-year normalised, restated financial history.
- **No tick-level historical archive.** Intraday depth is what your venue provides live; we store no consolidated tick history.

LABELLED AS DEMONSTRATION INSIDE THE PRODUCT

- **Fifteen of the thirty-five analytic surfaces are synthetic** — the fixed-income, FX, credit and macro families, each carrying a DEMO marker on screen. Equity, commodity and crypto run live.

KNOWN COVERAGE CAVEATS

- **Depth of history varies by plan.** One year on the entry tier, five years in the middle, full history on Pro.
- **Point-in-time is exact only where the provider publishes revision history.** 845,503 series carry full vintages; for the rest the response says the as-of answer is approximate.
- **The catalogue leads the value history.** Observations are pulled on first read, then kept.
- **The instrument universe is self-reported by issuers.** One that has never reported leaves its securities absent rather than unissued, and ownership edges join registered entities only.
- **The vessel register is layered.** 776,479 identities are on file, 52,648 carry full particulars and 49,446 are tracked with voyage history — and 15,783 of those state no type at source.
- **Trade data ends at 2024** — where the reporting cycle for complete bilateral figures currently stands.
- **Political events are classified from published reporting**, not field-verified, with coordinate precision stated per event.

A data sheet that lists only what a product has is a sales document. A data sheet that also lists what it does not have is a specification — and it is the only kind worth signing a contract against.

PLANS & PRICING

Three plans.
No blank tabs.

Every screen opens on every paid plan. What changes is how much of each you get, and which specific actions are permitted — set out here in full, with the price of every metered action.

Contents

8 SECTIONS

THE OFFER

01 **The shape of the offer** Why no plan hides a screen 3

02 **The three plans** Price, positioning, allowance 4

03 **What actually differs** The full limit matrix 5

04 **Capabilities by tier** The specific actions each plan unlocks 7

THE METER

05 **How credits work** What a credit is, and what stays free 9

06 **The price list** What every metered action costs 10

07 **Billing, seats and top-ups** The rules, including the awkward ones 13

THE DECISION

08 **Which plan, and what it replaces** Total cost against the alternatives 14

> **About these numbers.** Every figure in this document is read from the servers that enforce it — the tier tables in `pkg/authz` and the price list in `pkg/credits`. There is no second copy: the pricing screen in the product, the meters beside each control, and the checkout all read the same tables, so what is printed here cannot drift from what you are charged.

THE FIVE DOCUMENTS IN THIS SET

01	The Terminal	What it is, why it exists, and what it is not
02	Where It Stands	The landscape, and an honest comparison
03	The Platform	Every capability, screen by screen
04	The Data	Prices, the indexed store, providers, coverage
05	**Plans & Pricing**	Three tiers, the credit meter, what changes

01 The shape of the offer

Almost every subscription product in this category ladders by withholding screens. Ours does not, and the reason is worth a page because it changes what you are evaluating when you buy the cheapest plan.

EVERY TAB IS OPEN ON EVERY PAID TIER

A subscriber on the entry plan opens all thirty-seven destinations and gets real data in each one. What differs is depth — how many watchlists, how far back the history goes, how many symbols stream at once — and a short list of specific actions. Not whether a screen works.

This is a deliberate design rule and it is written into the access layer, which enforces in three ordered ways and is instructed to prefer the gentlest one that will do:

Layer	The question it answers	What a refusal looks like
Quota	How many, how deep, how far back?	A visible meter: *3 of 3 watchlists · Exclusive+ raises this to 25*. You see the ceiling before you hit it.
Capability	Is this specific action permitted?	The control stays visible and clickable, gains a lock, and opens an upgrade sheet naming the exact plan that permits it.
Feature	Is this whole plane reachable at all?	Reserved for one thing only: the broker plane, which holds real credentials and real positions, and has no useful partial version.

WHY WE BOTHERED

A tab that renders empty because the plan is off gives you nothing to interpret. It reads as a broken product, and it means the cheapest plan is a demonstration rather than the product. A tab that renders fully, with its limit stated on the screen, sells the next tier every single time it is opened — and, more importantly, it means the thing you bought works.

You are never evaluating a crippled version. You are evaluating the product, with smaller numbers.

02 The three plans

THE OFFER · 02

Three tiers, priced monthly with no minimum term. Quarterly billing takes fifteen per cent off and grants the whole quarter's credits on activation.

	Exclusive	Exclusive+	Exclusive Pro
Per month	**USD 99**	**USD 199**	**USD 299**
Per quarter	USD 252.45	USD 507.45	USD 762.45
Effective monthly, quarterly	USD 84.15	USD 169.15	USD 254.15
Positioning	Monitor & track	Research & analyse	Execute & automate
Credits per month	400	2,000	5,000
Cost per credit	USD 0.248	USD 0.100	USD 0.060
Devices included	1	1	2
Screens available	All 37	All 37	All 37
Live broker routing	—	—	**Yes**
Live algo deployment	—	—	**Yes**
Broker accounts	—	1	**Unlimited**
AI tool-calling & agents	—	**Yes**	**Yes, priority**

99
USD / MONTH, ENTRY

400
CREDITS INCLUDED WITH IT

15%
OFF ON QUARTERLY BILLING

37
SCREENS, ON ALL THREE PLANS

READING THE LADDER

Exclusive is a complete terminal for someone who watches, researches and simulates. One portfolio, three watchlists, a year of history, paper trading, every screen open. It is not a trial.

Exclusive+ is where the terminal becomes an analyst's tool: the AI tool-calling loop and agent teams, a linked broker account for position import, five years of history, the backtest optimiser, scheduled refreshes, watermark-free exports.

Exclusive Pro is where money moves: live broker routing, live algo deployment, crypto perpetuals, level-two depth, full history, as-of vintage queries, bulk export, your own branding on reports, and two devices.

03 What actually differs

The limits that shape the buying decision, exactly as the servers hold them. “Unlimited” means no cap; an em dash means the resource is not included on that plan.

MARKETS, PORTFOLIO AND TRADING

Resource	Exclusive	Exclusive+	Pro
Portfolios	1	10	Unlimited
Portfolio analytic views	3	9	9
Accounts on the unified monitor	1	5	Unlimited
Watchlists × symbols	3 × 25	25 × 100	Unlimited
Streaming symbols at once	25	100	500
Price history	1 year	5 years	Full
Paper accounts	1	5	Unlimited
Broker accounts	—	1	Unlimited
Crypto exchanges connected	3	8	14
Algo strategies	3	25	Unlimited
Algo deployments — paper / live	1 / —	5 / —	Unlimited
Backtest engines / strategies	2 / 20	6 / 111	6 / 111
Backtest history	2 years	10 years	Unlimited

RESEARCH AND ANALYTICS

Resource	Exclusive	Exclusive+	Pro
Equity research sections	5	15	15
Derivatives modules / models	2 / 10	8 / 41	8 / 41
M&A modules / models	3 / 15	7 / 42	7 / 42
Alternative-investment modules	4	9	9
Volatility surfaces	5	20	35
Quant Lab modules	6	18	24
Relationship-map hops	1	3	Unlimited
Geopolitics panels / history	2 / 30 days	4 / 365 days	4 / Unlimited

03 What actually differs

CONTINUED

DATA

Resource	Exclusive	Exclusive+	Pro
SEC filing pulls / month	10	Unlimited	Unlimited
Distinct data series / month	300	1,500	Unlimited
Economic / government providers	12 / 4	48 / 12	48 / 12
Data connections / mappings	3 / 3	20 / 20	Unlimited
Scheduled refresh jobs	—	10	Unlimited

NEWS

Resource	Exclusive	Exclusive+	Pro
News sources	10	Unlimited	Unlimited
News history	7 days	90 days	Unlimited
Keyword monitors / RSS feeds	2 / 10	10 / 50	Unlimited

AI, AUTOMATION AND WORKSPACE

Resource	Exclusive	Exclusive+	Pro
Agents / team modes	5 / —	25 / 3	53 / 6
External MCP servers	—	5	Unlimited
Node-graph nodes	10	50	Unlimited
Alpha Arena rounds / month	—	2	Unlimited
Dashboard widgets / layouts	6 / 1	20 / 5	Unlimited
Notes / notebooks / Excel sheets	50 / 5 / 5	∞ / 50 / 50	Unlimited
Report templates	3	12	12
Cloud storage	1 GB	20 GB	100 GB
Devices included	1	1	2

> **Limits are enforced when you create, never when you list.** Editing something you already own never consumes a slot — being at the limit is exactly when you need to fix what is already there. And a refusal always names the cheapest plan that lifts it, so you are never left guessing which upgrade to buy.

04 Capabilities by tier

Twenty-four named actions, each granted at one tier and cumulative upward. This is the complete list — there is nothing gated that is not on it.

EXCLUSIVE — THE BASE PLAN

Every screen, real data in all of them, paper trading on the live engine, the whole indexed data catalogue, AI chat, the local analytics models, notebooks, spreadsheet, reports, notes, the node editor and cloud sync of your workspace.

EXCLUSIVE+ ADDS FOURTEEN

Capability	What it lets you do
AI tool-calling	The agentic loop — the assistant calls terminal tools rather than only answering
Agent teams	Multi-agent team runs across six coordination modes
Deep research	Long-form, multi-step sourced research reports
Backtest optimiser	Parameter sweeps, not single runs
Broker import	Pull live positions from a linked brokerage account
Advanced orders	Brackets, OCO and good-till-triggered beyond market and limit
Scheduled refresh	Standing jobs that re-pull your data while the terminal is shut
News sentiment	Scored sentiment across the news corpus
SEC insider edges	Form 4 insider relationships on the relationship map
Realtime Asia quotes	Real-time rather than delayed Asian market data
Live prediction odds	Streaming odds rather than snapshot reads
Live Deribit feed	The live crypto options feed behind volatility surfaces
Clean export	Exports and reports without the Fincept watermark
Priority support	Ahead of the queue

EXCLUSIVE PRO ADDS TEN

Capability	What it lets you do
Live broker routing	Send a real order to a real broker
Live algo deployment	Run a deployed strategy against live routing
Crypto perpetuals	Perpetual futures on live crypto venues
Level-two depth	Full order-book depth rather than top of book
Live Databento feed	The professional feed behind equity volatility surfaces
Live vessel AIS	Real-time vessel positions on the maritime map
Point-in-time queries	Ask the indexed store for a series *as of* a past date
Bulk data export	Download the store in bulk as Parquet
Report branding	Your own name and marks on generated reports
Priority inference	A priority queue for AI work

> **Two capabilities move real money** — live broker routing and live algo deployment — and both are Pro-only. Anything ambiguous, such as a deployment with no mode set, is treated as live and refused rather than guessed as paper.

Twenty-four gated actions across three plans. That is the complete list. If something is not on it, no plan is withholding it from you.

05 How credits work

Credits meter consumption — the things that cost us money when you use them, which in practice means AI inference and a small number of expensive computations. They do not gate access to anything.

WHAT A CREDIT IS

Roughly a thousand tokens of language-model work. Every action costs between one and twenty-five credits, except the one that is deliberately smaller: a filled paper trade costs 0.05, twenty fills to the credit, because a simulator people hesitate to use is a broken simulator.

WHAT IS FREE, PERMANENTLY

NO CREDIT IS EVER CHARGED FOR ANY OF THESE

- **Every quote, chart, order book, tape and layout.** Reading the tape is the product.
- **Placing, modifying and cancelling orders** — paper or live. Live orders route through your own broker keys, so they cost us nothing and are free forever.
- **Loading and charting any data series**, within a generous monthly allowance of distinct series, and re-loading anything you already opened this month.
- **Every local analytic model** — derivatives, M&A, alternatives, technicals. They run on your machine in C++, so there is nothing to bill.
- **Notebooks, spreadsheet, notes, reports, dashboards, watchlists, the node editor** and every screen in the terminal.
- **Typing.** Voice is metered per minute because a vendor bills us per minute; the keyboard is not.

THE FOUR RULES THE METER FOLLOWS

Rule	What it means for you
The price is on the control	Every metered button states its cost before you press it. Nothing is discovered on the invoice.
Charged before the work, refunded on failure	If our provider fails, our sidecar times out or you cancel, the reservation is released. Our outage is not your bill.
Only the server can charge	There is no endpoint a client can call to declare a spend. The ledger is not under the application's control, or yours.
Running out never breaks a position	An algorithm that cannot pay its runtime stops opening and keeps every exit, stop and bracket armed. Alerts stay armed too.

06 The price list

The complete meter. Flat prices cost exactly what is listed; measured prices are the typical total for a normal run, and you are charged the actual usage.

AI AND AGENTS

Action	Credits	Notes
AI chat message	1	Measured from tokens used
AI chat with tool use	3	Fee plus measured
Agent run	5	Fee plus measured
AI research note	6	Fee plus measured
Deep research report	15	Fee plus measured
Agent team run	20	One fee per run, however many agents it coordinates
Quant Lab deep agent	15	Fee plus measured
Alpha Arena round	10	Fee plus measured
Portfolio AI analysis	5	Flat
Report AI generation	8	Flat
M&A deal memo	8	Fee plus measured
AI market brief	3	The writing only — every quote stays free
AI chart analysis	3	The written read; the statistics and the chart are free
AI trade review	3	Fee plus measured
Ask AI for data	1	The interpretation only; search and browsing are free

COMPUTATION WE RUN FOR YOU

Action	Credits	Notes
Backtest run	1	Only when run server-side; a local run is free and unlabelled
Screener run	1	Flat
Backtest optimiser sweep	5	Flat
Market regime scan	2	The cross-section, and it is kept as history
Surface calibration	2	The stored fit; the local fit is free
M&A deal model	3	Flat — a cache hit is free
Alt program simulation	2	The distribution; the local projection is free
Option strategy model	2	On save only

COMPUTATION WE RUN FOR YOU, CONTINUED

Action	Credits	Notes
End-of-day attribution	2	Flat
Broker reconciliation	1	Flat
News sentiment batch	2	Per corpus, not per article

DATA PAST THE INCLUDED ALLOWANCE

Action	Credits	Notes
SEC filing pull	1	Only past your monthly quota. A cache hit is free and consumes no quota either — the credit pays for a fetch we actually make against the SEC's shared rate limit, and charging for a read that never left the building would be charging for the cache
Data series load	0.10	Only past your monthly allowance of DISTINCT series. Re-loading one you already opened this month is free, because the quota measures breadth of exploration rather than clicks

TRADING

Action	Credits	Notes
Paper trade fill	**0.05**	Per FILLED trade — twenty fills to a credit, and a base-tier grant covers about eight thousand of them. Placing, modifying and cancelling are free, and so is every quote, chart and position read
Live broker order	**Free**	Routed through your own keys. We carry no cost and frequently never see the fill, so there is nothing to charge for

> **Why the simulator is priced in hundredths.** The point of a simulator is that people use it heavily, so a price anyone would hesitate over would break the feature in order to bill it. Charging a twentieth of a credit is what let us meter it without changing how anyone uses it.

THE FOUR RECURRING METERS

These accrue without a click, so each states its rate before it is armed — and the scheduling dialog quotes the whole monthly cost rather than the per-run unit, because the unit reads as trivial and the total is what you are agreeing to.

Meter	Rate	Notes
Algo runtime	2 / deployment / day	Stated in the deploy dialog before you confirm, including for paper deployments — same engine, same infrastructure
Watchlist alerts	1 / day	Per account, not per alert, so nobody deletes alerts to save money
News monitor	1 / monitor / day	Per monitor, because that is the unit you create and reason about
Scheduled refresh	1 / job / run	Cadence stops at daily on purpose: at hourly, one job would cost 720 credits a month

VOICE — THE ONLY METER MEASURED IN TIME

Meter	Rate	Notes
Dictation	1 / minute	Rounded up, one-minute minimum. Typing is free forever, and so is the clap-to-open trigger, which listens on your own machine
Spoken replies	1 / minute	Rounded up. Reading the reply on screen is free, and this is the half you can switch off

Voice is metered because a vendor bills us per minute of audio with no ceiling of its own — a microphone left open bills until it is closed. The control states the rate before it opens and never refuses a click, because it is the off switch as well as the on switch.

> **Three rules that hold across the whole list.** A price is shown only where it is actually charged — a quoted price that debits nothing teaches users the meter is decorative. A refund returns credits to the bucket they came from. And every charging route is idempotent, so a retry after a dropped connection cannot bill you twice.

07 Billing, seats and top-ups

The mechanics, including the parts most companies leave for the support ticket.

TWO KINDS OF CREDIT, AND THEY BEHAVE DIFFERENTLY

	Included with your plan	Credit packs you buy
Where it comes from	The plan's periodic grant	A top-up purchase
Ceiling	Twice the period's grant — you carry at most one unused period	None
Expires	Yes, by being capped at the next grant	**Never**
Spent	**First**	Only after the allowance is exhausted

The allowance is spent first precisely because it is the part that expires; burning something you paid cash for while a grant sits unused would destroy value for no reason. A refund returns credits to the bucket they came from, so a failed job cannot quietly convert a permanent credit into an expiring one, or the reverse.

TOP-UP PACKS

Pack	Credits	Price	Per credit
Small	250	USD 29	USD 0.116
Medium	750	USD 69	USD 0.092
Large	2,000	USD 149	USD 0.075

SEATS, DEVICES AND TERMS

THE RULES

- **A plan is a person.** Exclusive and Exclusive+ include one device; Pro includes two. Additional seats are added to the same account at the same price, billed separately.
- **Monthly means monthly.** No minimum term, no notice period, no quarterly prepayment unless you choose the quarterly plan for its fifteen per cent discount.
- **Quarterly grants the whole quarter at once** on activation, and the carry-over ceiling is calculated on the period grant — so buying a quarter does not void most of it on day one.
- **Holding several subscriptions gives you the best of them**, never the newest or the cheapest.
- **Upgrades take effect immediately**, and the refusal you hit names the exact plan that lifts it, so you are never guessing which one to buy.

08 Which plan, and what it replaces

THE DECISION · 08

The honest arithmetic. Most buyers arrive already paying for three or four things this replaces, which changes the comparison considerably.

If you…	Buy	Because
Watch markets, research names and simulate; do not route live orders	**Exclusive** USD 99 / mo	Every screen, real data, paper trading on the production engine, the whole data catalogue. Nothing is a demo.
Research for a living — models, filings, backtests, written output; want the AI to do real work	**Exclusive+** USD 199 / mo	The agentic tool loop, agent teams, deep research, the optimiser, five years of history, scheduled automation, clean exports.
Route live orders, run live strategies, trade perpetuals, or need full history and bulk export	**Exclusive Pro** USD 299 / mo	Live routing and live deployment, level-two depth, as-of queries, Parquet export, your branding, two devices.

WHAT ONE PRO SEAT COSTS AGAINST THE ALTERNATIVES

	Per year	Relative
Bloomberg Terminal, single seat	~USD 32,000	1.0×
FactSet Workstation, reported median	~USD 27,500	0.86×
LSEG Workspace, typical	~USD 22,000	0.69×
A stack of mid-market subscriptions	~USD 5,000–9,000	0.2×
Fincept Exclusive Pro	**USD 3,588**	**0.11×**
Fincept Exclusive Pro, quarterly	**USD 3,050**	**0.10×**
Fincept Exclusive	**USD 1,188**	**0.04×**

One Bloomberg seat pays for twenty-six Exclusive licences, or nearly nine Pro ones. At that ratio the question stops being which product is better, and starts being how many people on your desk currently have nothing at all.

Where to start. Take Exclusive for a month. It is the whole product rather than a trial, and if the depth is not enough the upgrade takes effect the moment you press it, with everything you already built still in the account.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Terminal Shell

A complete walkthrough of the frame around every tab — the menus, the command bar, the docking system, symbol linking, and every key the terminal listens to.

`>_ 4 menus · 37 screens · 10 desks · 10 symbol groups`

Covers · **File, Navigate, View, Help** · **the command bar** · **tabs, panels & windows** · **layouts**
Symbol groups · the pin strip · the status bar · every keyboard shortcut

Demonstrated live on the Fincept Terminal, v5.0.1.

Contents

21 SECTIONS

GETTING ORIENTED

01 What this guide covers 3
02 Anatomy of the window 4

THE TOP BAR

03 The header, left to right 5
04 File — windows, layouts and data 6
05 Navigate — every screen, and the pinned slots 7
06 View — display, panels and workspaces 9
07 Help — about, legal and sign-out 10
08 The command bar 11

THE WORKSPACE

09 The main tab bar 13
10 Panels, tabs and the dock 14
11 The tab menu 15
12 Floating, tearing off and second windows 16
13 Workspaces — the ten desks 17
14 Layouts — saving, opening and moving them 18
15 The Component Browser 19

LINKING SYMBOLS

16 Symbol groups — one symbol, many panels 20
17 The pin strip 21

REFERENCE

18 The status bar 22
19 Every keyboard shortcut 23
20 Where things are not 24
21 Quick reference and honest notes 25

About this guide. Every screenshot is a live capture of the shipping v5.0.1 build, and every control described here was read from the product's source rather than inferred from the screen. This is the one guide that is not about a tab — it is about everything around them.

01 What this guide covers

Every other guide in this library describes what happens *inside* a tab. This one describes everything else: the bar across the top, the menus that hang off it, the box you type commands into, the strip of pinned symbols, the way panels dock and float and tear off into new windows, the layout system that remembers all of it, and the keys that drive it without the mouse.

It is the part of the terminal you use constantly and are never taught. Most of what follows is not discoverable by clicking around — the compound command grammar, symbol groups and half the keyboard map have no visible affordance at all.

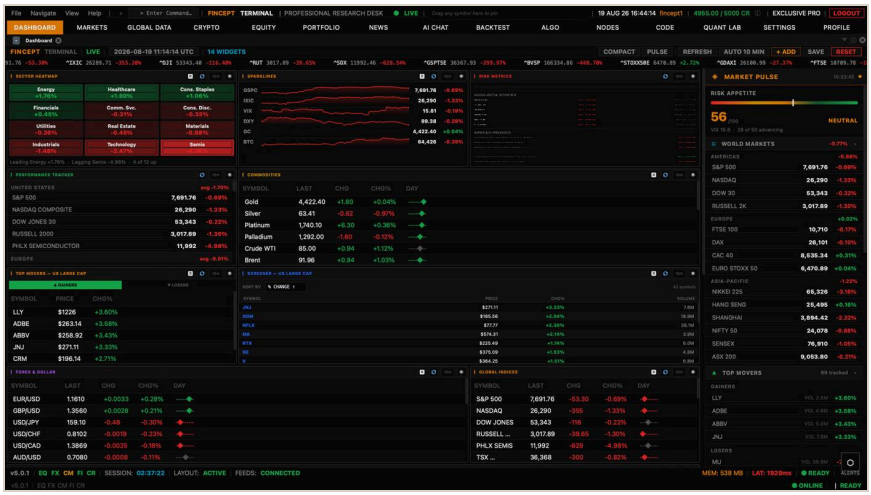


Fig 1 — The shell around a tab: header and menus at the top, the main tab bar below it, the panel area filling the middle, and the status bar at the foot.

By the end of this guide you will be able to…

- Reach any of 37 screens three different ways
- Open two panels side by side in one command
- Pin your three most-used screens to the menu bar
- Tear a panel off onto a second monitor
- Save a layout and reopen it tomorrow
- Load a ready-made desk for your job
- Drive one symbol into six panels at once
- Keep a symbol pinned across sessions
- Read every light in the status bar
- Work the whole terminal from the keyboard

> **Everything here was read from the source.** The shell in the working tree matches the shipping v5.0.1 binary, so unlike some tab guides this one is grounded end to end rather than inferred from the screen.

02 Anatomy of the window

Fourteen regions, in four horizontal bands.

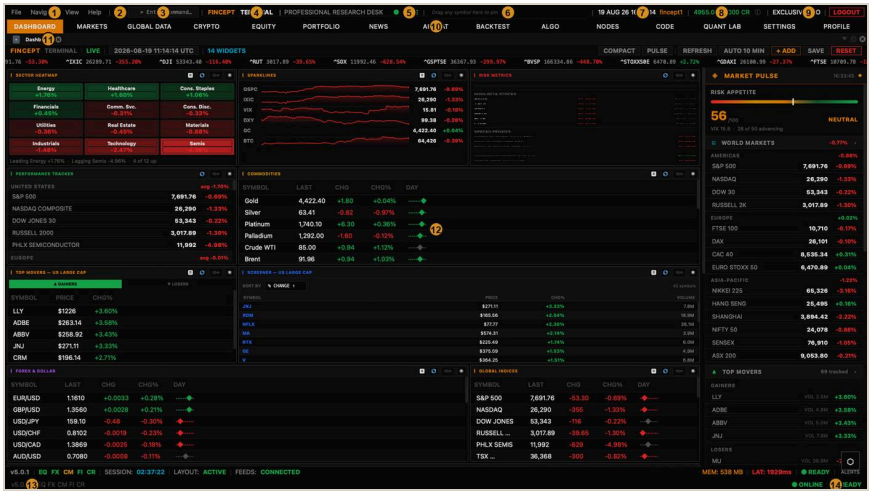


Fig 2 — The fourteen regions, numbered.

#	Region	What it is
1	Menu bar	File · Navigate · View · Help (§04–§07)
2	Pinned slots	Up to three screens of your choosing, plus the + that adds them (§05)
3	Command bar	Type a screen name, a symbol, or a compound command (§08)
4	Brand	**FINCEPT TERMINAL** · PROFESSIONAL RESEARCH DESK
5	● **LIVE**	Market-data connection state
6	Pin strip	Your pinned symbols — *Drag any symbol here to pin* (§17)
7	Clock and user	UTC date-time, then the signed-in account
8	Credits	Balance against your monthly allowance, with an info button
9	Plan and sign-out	Your tier, then **LOGOUT**
10	Main tab bar	The fifteen top-level screens (§09)
11	Panel tab strip	One tab per open panel, with its close button (§10)
12	Panel area	Where screens are drawn, docked, split and floated (§10)
13	Version and feeds	Build number, then the five feed codes (§18)
14	Connection and readiness	● **ONLINE** and **READY** (§18)

> **Two tab bars, doing different jobs.** Region 10 is a fixed row of the fifteen most-used screens and behaves like a channel selector — clicking one *replaces* your workspace. Region 11 is the dock's own tab strip and lists the panels actually open. They look similar and are not the same thing; §09 and §10 take them in turn.

03 The header, left to right

Fig 3 — The left half: menus, the pinned-slot button, the command bar, the brand and the live lamp.

Fig 4 — The right half: the pin strip, clock, account, credits, plan and sign-out.

Element	What it does	Detail
`File / Navigate / View / Help`	The four menus	§04–§07
`+`	Adds a pinned screen slot	Opens the same dialog as Customize Menu Bar (§05)
`> Enter Command…`	The command bar	Click it or just start typing (§08)
`FINCEPT TERMINAL`	Branding	Not a control
`● LIVE`	Market-data state	Green when quotes are streaming
`Drag any symbol here to pin`	The empty pin strip	Drop a symbol on it to pin (§17)
`19 AUG 26 16:44:14`	UTC clock	Always UTC, never local time
`fincept1`	Signed-in account	The username, not the email
`4955.00 / 5000 CR`	Credits	Remaining against your monthly allowance
ⓘ	Credit info	Explains what credits are and what spends them
`EXCLUSIVE PRO`	Your plan	The tier the servers enforce
`LOGOUT`	Sign out	Ends the session

> **The clock is UTC by design.** A trading desk spanning exchanges needs one unambiguous reference, so the header never shows local time. Panels that show market hours convert to the venue's own zone separately.

> **Credits are a monthly allowance, not a wallet balance.** The figure left of the slash is what remains this period. Individual tabs print their own per-action prices — the Maritime tab, for example, marks an 8-credit area search directly on the button. Nothing in the shell itself spends credits.

04 File — windows, layouts and data

Four groups: windows, layouts, import/export, and a shortcut to the file library.

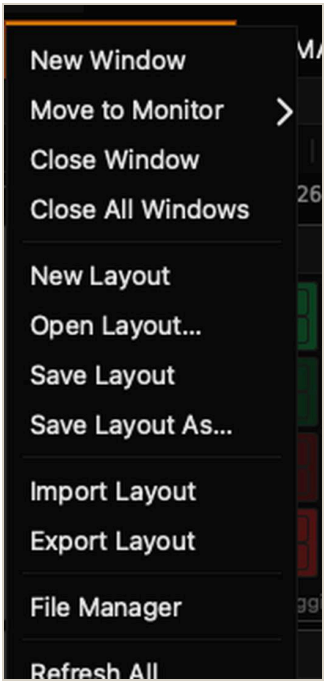


Fig 5 — The File menu.

Entry	What it does
`New Window`	Opens a second terminal window with its own panels and its own layout
`Move to Monitor ▸`	Lists your displays by name and resolution; moves this window to the chosen one. Rebuilt each time it opens, so plugging a monitor in mid-session is picked up
`Close Window`	Closes this window only
`Close All Windows`	Closes every terminal window
`New Layout`	Clears the workspace to an empty layout
`Open Layout…`	Loads a saved layout (§14)
`Save Layout`	Saves over the current layout
`Save Layout As…`	Saves under a new name
`Import Layout`	Reads a layout from a file
`Export Layout`	Writes the current layout to a file — how you share a desk with a colleague
`File Manager`	Jumps to the File Manager screen
`Refresh All`	Refreshes every open panel

> **Move to Monitor names displays, it does not number them.** The menu emits the monitor's name rather than its index, because indices shuffle when a display is unplugged and names do not. With one display attached the submenu shows a disabled *(single monitor)* entry rather than being empty.

05 Navigate — every screen, and the pinned slots

NAVIGATE MENU · 05

The complete list of screens, in six job-shaped groups, with three account destinations below them.

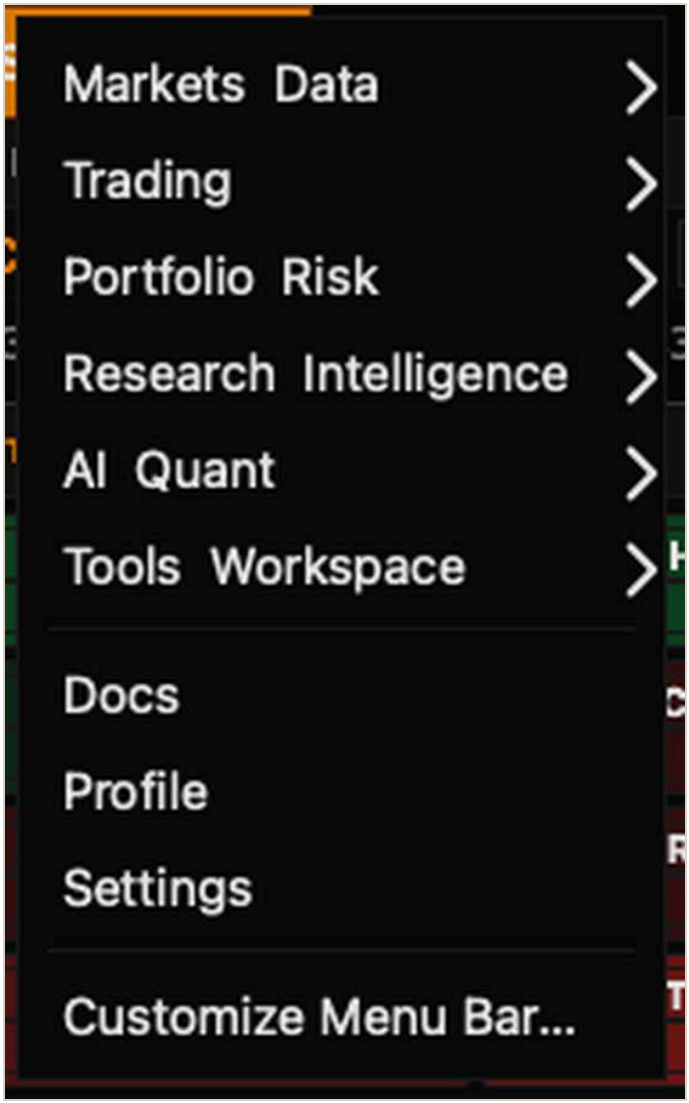


Fig 6 — Six groups, then Docs / Profile / Settings.

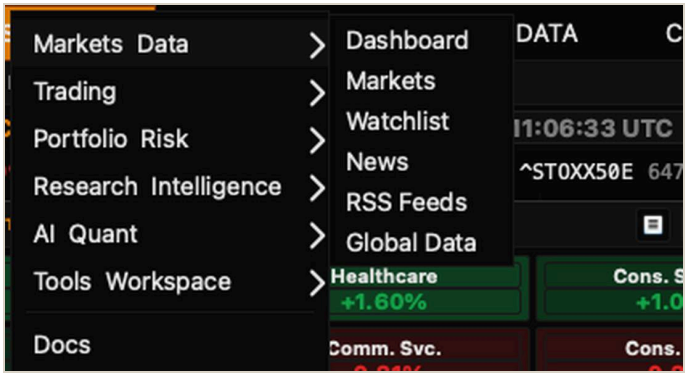


Fig 7 — Markets & Data expanded.

Group	Holds
`Markets & Data`	Dashboard · Markets · Watchlist · News · RSS Feeds · Global Data
`Trading`	Equity · Crypto · Algo · Derivatives · Prediction Markets · Alpha Arena
`Portfolio & Risk`	Portfolio · Backtesting · Surface Analytics
`Research & Intelligence`	Equity Research · M&A · Alt. Investments · Relationship Map · Geopolitics · Maritime
`AI & Quant`	AI Chat · AI Quant Lab · Agent Config · MCP Servers · Node Editor
`Tools & Workspace`	Data Sources · Data Mapping · Code Editor · Excel · Report Builder · Trade Viz · File Manager · Notes
`(loose)`	Docs · Profile · Settings

Pinning your three most-used screens

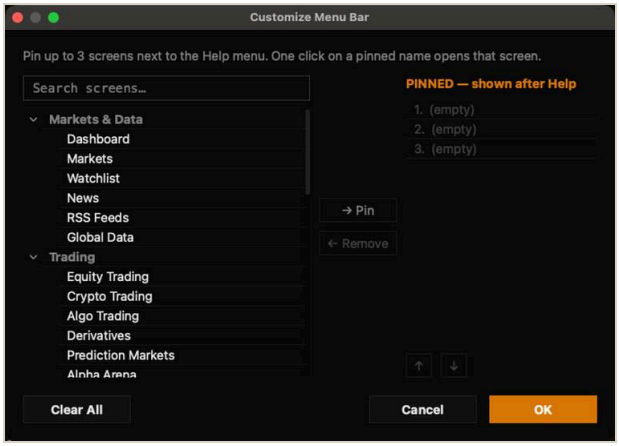


Fig 8 — Customize Menu Bar: search every screen, pin up to three.

Customize Menu Bar… at the foot of the menu — or the + in the header — opens a searchable tree of every screen with three pinned slots beside it. Pinned names appear immediately after the Help menu, and one click opens that screen. **Pin** and **Remove** move entries between the panes, the arrows reorder them, and **Clear All** empties the slots.

> The entry point sits at the bottom of the very menu the pins are a shortcut past — which is exactly where somebody who has just walked the full list for the third time is looking.

06 View — display, panels and workspaces

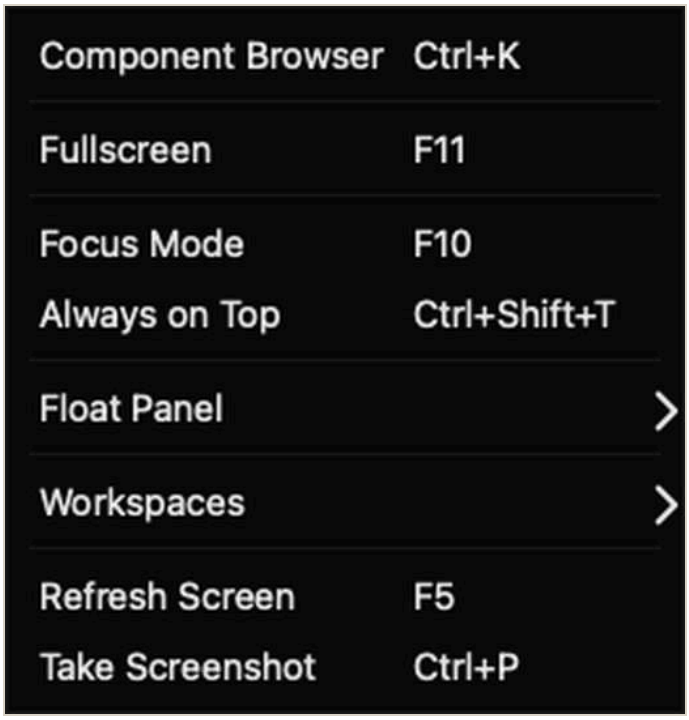


Fig 9 — The View menu, with its shortcuts.

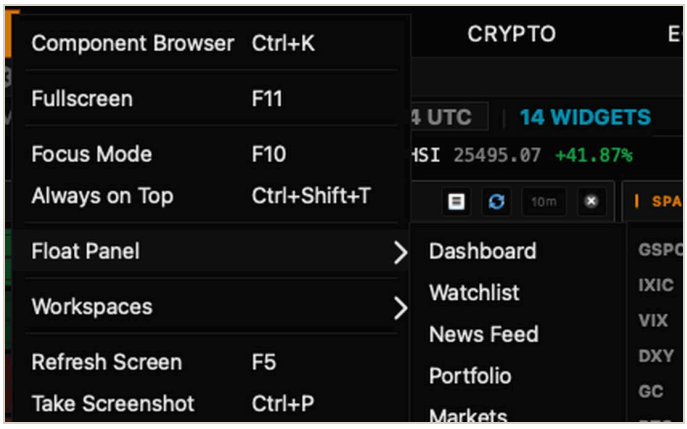


Fig 10 — Float Panel, opening a screen as its own window.

Entry	Key	What it does
Component Browser	`Ctrl+K`	A searchable card grid of every screen (§15)
Fullscreen	`F11`	Fills the display, hiding the window chrome
Focus Mode	`F10`	Strips the shell back to the panel you are working in
Always on Top	`Ctrl+Shift+T`	Keeps this window above other applications
Float Panel ▸	—	Opens one of twelve screens directly as a floating window: Dashboard, Watchlist, News Feed, Portfolio, Markets · Crypto, Equity, Algo Trading · Equity Research, Global Data, Geopolitics, AI Chat
Workspaces ▸	—	The ten ready-made desks, plus Save Current Workspace (§13)
Refresh Screen	`F5`	Refreshes the active panel
Take Screenshot	`Ctrl+P`	Captures the terminal window to an image

> **Always on Top is not a tick box here.** The action is deliberately not checkable — the real state lives on the window, and a checkable menu item drifts out of sync as focus moves between windows. The tab menu's own Always on Top *is* checkable, because there it describes one specific floating panel (§11).

> **Float Panel is a shortcut, not the only route.** It opens twelve chosen screens as floating windows in one click. Any panel at all can be floated from its tab menu (§11).

07 Help — about, legal and sign-out

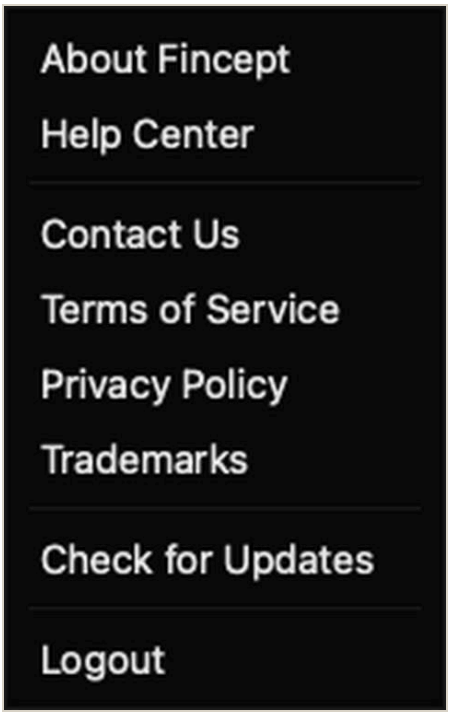


Fig 11 — The Help menu.

Entry	What it opens
`About Fincept`	Build and licence information for this installation
`Help Center`	The in-app help screen
`Contact Us`	How to reach the desk
`Terms of Service`	The terms you are operating under
`Privacy Policy`	What the terminal collects and why
`Trademarks`	Trademark attributions
`Check for Updates`	Asks the release service whether a newer build exists
`Logout`	Ends the session and returns to sign-in

The four legal pages and the help centre are ordinary screens — they open in the panel area like any other tab, and they are deliberately kept out of the Navigate menu so that list stays a list of places you work.

> **Updates install in place.** Check for Updates downloads and applies the new build over the installation you are running, so there is never a second copy and your session survives the update. You are not asked where to install.

> **Logout ends the session everywhere in this window.** Panels that hold unsaved local state — an unsaved note, an unrun notebook cell — are not prompted for by the shell. Save before signing out.

08 The command bar

The fastest route to anything. It accepts three different kinds of input, and the second and third are invisible unless you know they exist.

1 · A screen name

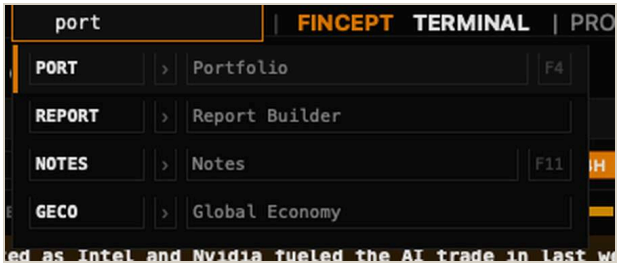


Fig 12 — Typing a few letters ranks matching screens, with their key.

Type any part of a screen's name or one of its aliases and press `Return`. Aliases are generous — `poly`, `kalshi` and `prediction` all reach Prediction Markets; `marine` and `shipping` both reach Maritime.

2 · A symbol, by asset class

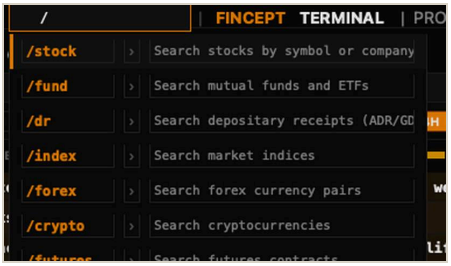


Fig 13 — Typing `/` lists the nine asset classes.

Prefix	Searches
`/stock`	Stocks by symbol or company name
`/fund`	Mutual funds and ETFs
`/dr`	Depositary receipts (ADR/GDR)
`/index`	Market indices
`/forex`	Currency pairs
`/crypto`	Cryptocurrencies
`/futures`	Futures contracts
`/bond`	Bonds and fixed income
`/economic`	Economic indicators

The command bar, continued

CONTINUED · 08

3 · A compound command — two panels at once

Type this	And you get
`markets add settings`	Settings opened **alongside** Markets, split
`markets replace news`	News in place of Markets
`markets remove`	Markets closed

> **This is the only way to open two panels from the keyboard**, and nothing on screen advertises it. Every other navigation route — the tab bar, the Navigate menu, a plain command-bar entry — opens the target *exclusively* and closes what you had (§09). If you have ever wondered why you cannot get two screens side by side, this is the answer.

How the grammar works

A compound command is always `<screen> <verb> [<screen>]`. The first name is the panel you are acting on; the verb says what to do; the second name, where the verb takes one, is the panel to act with. Both names accept the same aliases as a plain lookup, so `poly add news` works exactly as well as the full titles.

Verb	Takes a second screen?	Effect
`add`	Yes	Opens the second screen alongside the first, splitting the area
`replace`	Yes	Swaps the first screen out for the second in the same place
`remove`	No	Closes the named screen

The bar guides you through it: type a screen name followed by a space and it offers the three verbs; pick one and it offers the screens. You can also type the whole thing straight through and press `Return`.

> **Worked example — a research desk in three commands.** Type `equity_research`, then `research add news`, then `research add portfolio`. You now have three panels split across the area. Link all three to Group A (§16) and they will follow one symbol together — then **File ▸ Save Layout** to keep it.

09 The main tab bar

The row under the menus holds fifteen top-level screens: **DASHBOARD · MARKETS · GLOBAL DATA · CRYPTO · EQUITY · PORTFOLIO · NEWS · AI CHAT · BACKTEST · ALGO · NODES · CODE · QUANT LAB · SETTINGS · PROFILE**.

> **Clicking one closes everything else.** The tab bar navigates *exclusively*: the selected screen replaces whatever was open and fills the whole panel area. This is deliberate — each tab is meant to be a fresh, independent screen rather than something nested inside your previous arrangement — but it means the tab bar will discard a two-panel split you just built. Save the layout first (§14), or use a compound command instead (§08).

Which routes replace, and which add

Route	Behaviour
`Main tab bar`	**Replaces** — exclusive
`Navigate menu`	**Replaces** — exclusive
`A pinned slot`	**Replaces** — exclusive
`Command bar, plain screen name`	**Replaces** — exclusive
`Command bar, X add Y`	**Adds** — splits alongside
`Tab menu ▸ Duplicate Panel`	**Adds** — a second copy as a new tab
`Tab menu ▸ Float / Tear off`	**Adds** — as a floating window (§12)
`View ▸ Float Panel`	**Adds** — as a floating window
`A workspace template`	**Replaces** — with a whole multi-panel desk (§13)
`The AI assistant's navigation tool`	**Adds** by default — auto-tiled alongside

> **The assistant is the exception.** Its navigation tool defaults to non-exclusive, so asking the AI to open a screen tiles it next to what you have rather than replacing it. Callers that want a clean switch have to ask for one explicitly.

10 Panels, tabs and the dock

Everything below the main tab bar is a dock. A **panel** is one screen; panels carry tabs, sit side by side in split areas, or float free as their own windows.

Fig 14 — The panel tab strip: one tab per open panel, each with a close button.

Gesture	Result
`Click a tab`	Brings that panel to the front of its area
`Drag a tab within the strip`	Reorders panels
`Drag a tab onto another area`	Moves the panel there — drop zones highlight as you go
`Drag a tab to an edge`	Splits the area and docks the panel against that edge
`Drag a tab out of the window`	Floats it as its own window (§12)
`Click the × on a tab`	Closes that panel
`Right-click a tab`	The tab menu (§11)
`Drag a splitter`	Re-proportions two neighbouring areas

Moving between panels without the mouse

Key	Does
`Ctrl+Tab`	Focus the next panel
`Ctrl+Shift+Tab`	Focus the previous panel
`Ctrl+1` … `Ctrl+9`	Focus window 1–9
`Ctrl+Alt+Tab`	Next window
`Ctrl+Alt+Shift+Tab`	Previous window

> **Why `Ctrl+Alt+Tab` and not `Alt+Tab`.** Window cycling would naturally be `Ctrl+Alt+Tab`, but that is a window-manager shortcut on several platforms. The terminal wraps the same gesture with an extra modifier so it still feels related without being stolen by the OS. Every binding here is rebindable in Settings.

11 The tab menu

Right-click any panel tab. The menu is **contextual** — three of its entries appear only when they can do something.

Fig 15 — On an ordinary panel.

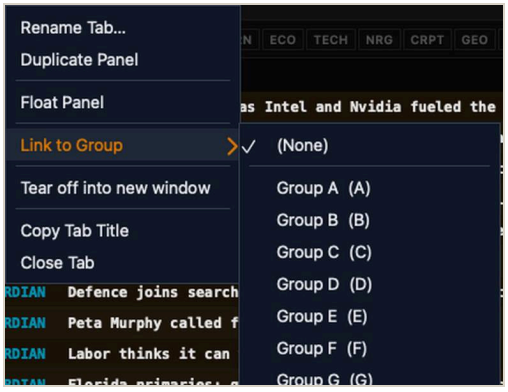


Fig 16 — On a linkable one, with Link to Group open.

Entry	What it does	Shown when
`Rename Tab…`	Gives the tab your own label	Always
`Duplicate Panel`	A second independent copy of the same screen	Always
`Float Panel / Re-dock Panel`	Detaches the panel as its own window, or puts it back	Always
`Always on Top`	Keeps that floating panel above other windows — a real tick box here	Only while floating
`Link to Group ▸`	Joins the panel to a symbol group (§16)	Only on the ten linkable screens
`Tear off into new window`	Moves the panel into a brand-new terminal window	Always
`Move to window ▸`	Sends the panel to another open window, listed by number	Only with a second window open
`Copy Tab Title`	Copies the tab's label to the clipboard	Always
`Close Tab`	Closes the panel	Always

> **Duplicate Panel is genuinely a second instance**, not a second view of the same one. Two copies of Equity Research can therefore sit on two different companies — and if you link them to different symbol groups (§16), each will follow its own.

> **Rename is how you keep two copies straight.** Duplicated panels arrive with identical titles; renaming one *AAPL* and the other *MSFT* is the difference between a usable desk and a row of indistinguishable tabs.

12 Floating, tearing off and second windows

Three different things that all sound similar, and are not.

Operation	What you get	Best for
`Float Panel`	The panel becomes a small free-floating window, still owned by this terminal window	Watching one panel while you work in another
`Tear off into new window`	The panel moves into a **new full terminal window** — its own menus, tab bar and status bar	Giving a screen a whole monitor
`File ▸ New Window`	An empty second terminal window you then fill yourself	Building a second desk from scratch

Working across monitors

Want to…	Do this
`Put this window on another display`	**File ▸ Move to Monitor**, or `Ctrl+Shift+1…9`
`Open a new window on the next display`	`Ctrl+Shift+N`
`Give one screen its own monitor`	Tear it off, then move that window
`Move a panel between existing windows`	Tab menu ▸ **Move to window**
`Jump between windows`	`Ctrl+1` … `Ctrl+9`, or `Ctrl+Alt+Tab`

> **Every window is a full terminal.** A torn-off window is not a satellite — it has its own menu bar, its own command bar, its own tab bar, its own layout and its own status bar. That is why **Close All Windows** exists as a separate action from **Close Window**.

> **Move to window is hidden when there is nothing to move to.** With a single window open the entry is not greyed out — it is absent entirely, which can read as a missing feature. Open a second window and it appears.

13 Workspaces — the ten desks

View ▸ Workspaces holds ten ready-made multi-panel desks, each built for a job. Picking one replaces your workspace with a proportioned arrangement of panels — not an auto-tiled grid.

Desk	Panels
`Equity Trading Desk`	Trading · Watchlist · Algo · Positions · News · Research
`Options & F&O Desk`	F&O Chain · Pricing · Vol Surface · Execution · Positions
`Crypto Trading Desk`	Crypto · Watchlist · Cross-Asset · Portfolio · Arena · News
`Portfolio Management`	Portfolio · Live P&L · Risk · Rebalance · Research · Reporting
`Equity Research`	Research · Coverage · M&A · Ownership · News · Notes
`Macro Research`	Global Data · Markets · News · Geopolitics · Notes
`Quant Research`	Quant Lab · Notebook · Backtest · Live Algos · Trade Analysis
`Global Market Monitor`	Dashboard · Markets · Global Data · News · Feeds · Watchlist
`Geopolitical Intelligence`	Geopolitics · Maritime · Relationships · News · Markets · Notes
`AI Agent Desk`	AI Chat · Agent Studio · Workflows · MCP Tools · Dashboard

Each carries a one-line description as a tooltip — *"Execution-first desk with watchlist, algos and live positions"*, *"Read-only world view for the morning check or a wall display"* — so you can tell them apart without loading each in turn.

Making your own

Save Current Workspace... at the foot of the submenu turns whatever you have arranged into a desk of your own, alongside the ten built-ins.

> **These are the same desks the Launchpad picker offers**, drawn from one list. The menu used to carry a separate set of "perspectives" that had already drifted apart from the picker's — the menu knew about desks the picker did not and vice versa — and the menu's version could only ever produce an auto-tiled grid with no proportions and no tabbed panels.

14 Layouts — saving, opening and moving them

SAVED LAYOUTS · 14

A workspace template is a starting point. A **layout** is your arrangement, saved by name.

Action	What it does
`File ▸ New Layout`	Clears to an empty workspace
`File ▸ Open Layout…`	Loads one of your saved layouts
`File ▸ Save Layout`	Saves over the layout you are in
`File ▸ Save Layout As…`	Saves the current arrangement under a new name
`File ▸ Export Layout`	Writes it to a file you can send to someone
`File ▸ Import Layout`	Loads a layout file from someone else
`View ▸ Workspaces ▸ Save Current Workspace…`	Promotes the arrangement to a named desk in the Workspaces menu

What a layout remembers

- Which panels are open, and in which dock areas
- How the splitters are proportioned
- Which panels are tabbed together, and which tab is in front
- Which panels are floating, and where
- Any tab titles you renamed

> **Export is how a desk travels.** A layout file is the practical way to hand a colleague the arrangement you have tuned — or to move your setup to another machine. Import it on the far side and the panels come back in the same places.

> **A layout is an arrangement, not a session.** It restores which panels are open and where — it does not restore what each panel was *looking at*. Screens persist their own state separately and to their own rules; the Maritime tab, for instance, keeps its IMO field and its bounding box but resets its basemap and drawn shapes.

15 The Component Browser

`Ctrl+K`, or **View ▸ Component Browser**. A card grid of every screen in the terminal, searchable and categorised — the discovery surface for a product with 37 of them.

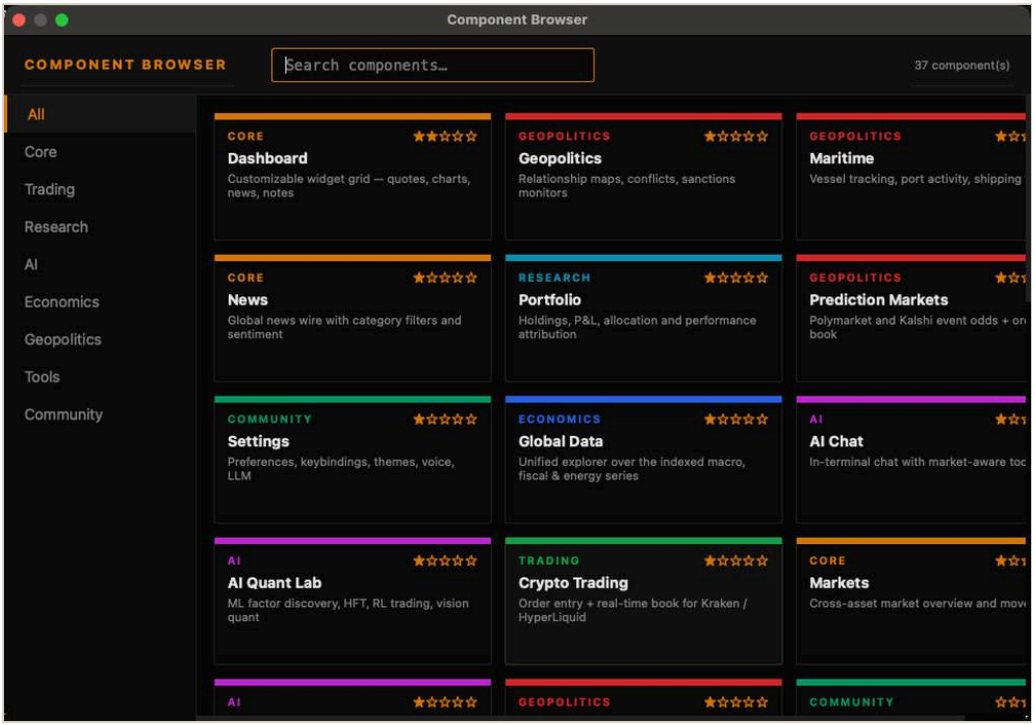


Fig 17 — The Component Browser: 37 components, categorised, each card carrying a one-line description and a popularity rating.

Element	What it is
Search components…	Filters by name and description as you type
Category rail	All · Core · Trading · Research · AI · Economics · Geopolitics · Tools · Community
37 component(s)	The count for the current filter
Card colour bar	The component's category, colour-coded
Card stars	How often you use it — the browser surfaces frequently-used screens first
Card text	A one-line description of what the screen is for

> **The stars are your own usage, not an editorial rating.** Every navigation bumps a popularity counter for that screen, and the browser sorts on it — so the grid reshapes itself around how you actually work rather than staying in a fixed order.

> **It is the best answer to "what else is in here?"** The Navigate menu is a list of names; this is a list of names with descriptions, which is what you want when you do not yet know what you are looking for.

16 Symbol groups — one symbol, many panels

This is the feature that turns a set of panels into a desk. Link several panels to the same **group**, and setting a symbol in any one of them drives all the others.

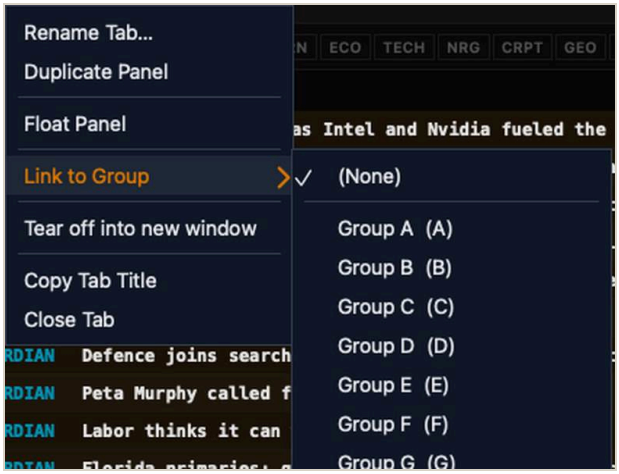


Fig 18 — Link to Group on the News panel: (None) is ticked, with Group A through G available.

How to use it

1. Open the panels you want to move together.
2. Right-click each one's tab ▸ **Link to Group** ▸ **Group A**.
3. Set a symbol in any of them. The rest follow.

The ten linkable screens

AI Chat	**Equity Trading**
Backtesting	**F&O**
Crypto Trading	**News**
Equity Research	**Portfolio**
Surface Analytics	**Watchlist**

Group	Meaning
`(None)`	Unlinked — the panel ignores other panels and is ignored by them. The default
`Group A … Group G`	Seven independent channels, enabled out of the box
`Groups H, I, J`	Three further slots, off by default

> **Two groups let you run two ideas at once.** Put Research, News and Portfolio on Group A and a second Research panel plus Equity Trading on Group B, and you can compare two names side by side without either set disturbing the other.

> **Only those ten screens link.** Every other panel ignores groups entirely and shows no **Link to Group** entry at all. A missing entry means the screen is not linkable — nothing is broken.

17 The pin strip

The strip in the header holding symbol chips. Empty, it reads *Drag any symbol here to pin* — which is the whole instruction.

Gesture	Result
`Drag a symbol onto the strip`	Pins it
`Left-click` `a chip`	Broadcasts that symbol to the default group — every panel linked to it follows
`Drag` `a chip onto a panel`	Sets that one panel to the symbol, without touching the others
`Right-click` `a chip`	Remove · Broadcast to Group A/B/… · Copy ticker
`Drag a chip within the strip`	Reorders your pins
`Scroll over the strip`	Scrolls horizontally when you have more pins than fit

> **Pins follow you.** They are stored against your account rather than against a layout or a window, so they survive closing the terminal, and they are the same in every window and every workspace.

> **Click to broadcast, drag to target.** That distinction is the whole point of the strip: a click moves your linked desk to a name, while a drag pushes it into exactly one panel. The right-click menu adds a third option — broadcast to a *specific* group rather than the default one, which is how you drive Group B without disturbing Group A.

> **Pinning is idempotent, and unpinning is immediate.** Pinning a symbol twice does nothing the second time, and **Remove** takes effect without confirmation. There is no undo — but re-pinning is one drag away.

18 The status bar

Fig 19 — Left: build number and the five feed codes.

Fig 20 — Right: connection and readiness.

Element	Reads	Meaning
`Version`	`v5.0.1`	The build you are running
`Feeds`	`EQ FX CM FI CR`	The five data planes — equities, foreign exchange, commodities, fixed income, crypto
`Connection`	● **ONLINE**	The backend is reachable
	● **RECONNECTING**	The link dropped and is being re-established
	● **OFFLINE**	No backend. Local-only screens keep working
`Link`	varies	What the active panel is currently linked to
`Readiness`	**READY** / **BUSY**	Whether the terminal is idle or working

> **OFFLINE is not a failure state.** Screens that run on local Python or cached data keep working; the ones that need the backend say so in their own panel. The banner and this lamp are how you tell which kind of silence you are looking at.

> **BUSY means the terminal is working, not hung.** A long analytics run or a large area search will hold it — give it time before clicking again, since a second click usually queues a second request rather than cancelling the first.

19 Every keyboard shortcut

All of these are rebindable in **Settings ▸ Keybindings**. The defaults:

Global

Key	Action
`F5`	Refresh the active screen
`F9`	Toggle the AI chat bubble
`F10`	Focus Mode
`F11`	Fullscreen
`Ctrl+P`	Take a screenshot
`Ctrl+K`	Component Browser
`Ctrl+Shift+T`	Always on Top
`Ctrl+L`	Lock the terminal now

Windows and panels

Key	Action
`Ctrl+1` … `Ctrl+9`	Focus window 1–9
`Ctrl+Shift+1` … `Ctrl+Shift+9`	Move this window to monitor 1–9
`Ctrl+Shift+N`	New window on the next monitor
`Ctrl+Tab` / `Ctrl+Shift+Tab`	Next / previous panel
`Ctrl+Alt+Tab` / `Ctrl+Alt+Shift+Tab`	Next / previous window

The command bar

`↓` and `↑` move through the suggestions, `Tab` accepts the highlighted one, `Return` runs what you have typed, and `Esc` dismisses the dropdown.

In particular screens

Key	Action	Where
`J` / `K`	Next / previous article	News
`Return` / `Esc`	Open / close an article	News
`Ctrl+Return`	Run the current cell	Code editor
`Shift+Return`	Run the cell and move to the next	Code editor
`F2`	Rename the cell	Code editor

> **`Ctrl+L` locks immediately**, without waiting for the inactivity timer — the fastest way to leave a shared desk safely.

20 Where things are not

A short list of things people look for in the wrong place, with where they actually are.

Looking for…	It is…
`Two screens side by side`	Not on the tab bar — use `X add Y` in the command bar (§08), or Duplicate / Float from a tab menu (§11)
`A second window`	**File ▸ New Window**, or tear a panel off (§12)
`The screen list`	**Navigate**, or the Component Browser for descriptions (§15)
`Prediction Markets, Maritime, Relationship Map`	Not on the main tab bar — they are Navigate-only screens
`The F&O screen`	Not in Navigate either — it opens from the Equity Trading tab's F&O button, because it needs that tab's symbol and leg context
`Symbol linking`	The tab menu, not Settings (§16)
`Your pinned symbols after a restart`	Still there — pins are stored against your account (§17)
`Your panel arrangement after a restart`	Save it first — **File ▸ Save Layout** (§14)
`Local time`	Nowhere in the shell. The header clock is always UTC (§03)
`Move to window`	Absent until a second window exists (§12)
`Link to Group on most screens`	Only ten screens are linkable (§16)

> **The Navigate menu can show a lock and a tier name beside a screen**, and the machinery to do it is live — but no screen is configured as plan-gated today, so in practice the marks never appear and every screen in the menu opens on any paid plan. Gating that exists is enforced per-action inside a tab instead, which is why a tab can open and still refuse one of its buttons.

21 Quick reference and honest notes

How do I…

Goal	Do this
Open any screen fast	Click the command bar, type a few letters, `Return`
Open two screens side by side	`markets add news` in the command bar
Look up a symbol	Type `/` and pick an asset class
Keep three screens one click away	**Navigate ▸ Customize Menu Bar…**, pin up to three
Give a screen its own monitor	Tab menu ▸ Tear off, then **File ▸ Move to Monitor**
Compare two companies at once	Duplicate a panel, rename both, link them to Groups A and B
Drive six panels from one symbol	Link them all to Group A, then click a pinned chip
Load a ready-made desk	**View ▸ Workspaces**
Keep the arrangement you built	**File ▸ Save Layout**
Send your desk to a colleague	**File ▸ Export Layout**
See everything the terminal has	`Ctrl+K`
Lock the terminal	`Ctrl+L`

Honest notes

- **The main tab bar destroys your arrangement.** Clicking a top-level tab closes every other panel — deliberate, but it will discard a split you just built (§09).
- **The compound command grammar is unadvertised.** `X add Y` is the only keyboard route to a two-panel split and nothing on screen mentions it (§08).
- **The AI assistant navigates differently from you.** Its tool defaults to non-exclusive, so it tiles where your click would replace (§09).
- **Only ten screens support symbol linking**; on everything else the menu entry is absent rather than disabled (§16).
- **Move to window and Always on Top are hidden, not greyed**, when they do not apply — which reads as a missing feature rather than an inapplicable one (§11, §12).
- **Plan-lock marks in the Navigate menu never appear** — the machinery is live but no screen is configured as gated (§20).
- **Groups H, I and J exist but are off by default**, so the menu shows seven channels, not ten (§16).
- **A layout restores arrangement, not content.** Each screen persists its own state to its own rules (§14).
- **Nothing in the shell prompts you to save.** Logout and Close All Windows act immediately; unsaved panel state is the panel's business (§07).
- **The header clock is UTC only.** There is no setting for local time (§03).
- **Float Panel in the View menu covers twelve screens**, not all of them; any panel can be floated from its own tab menu instead (§06).
- **Duplicated panels arrive identically titled.** Rename them or you will not be able to tell them apart (§11).
- **Window cycling is `Ctrl+Alt+Tab`**, deliberately not the OS-standard gesture, to avoid being intercepted (§10).
- **Nothing in the shell spends credits.** Every price is charged inside a tab (§03).

> Everything in this guide was read from the shipping source and then confirmed on screen on the v5.0.1 build. The shell is the one part of the terminal with no in-app documentation at all, which is why several of the notes above are simply *where a thing is* rather than a defect — undiscoverable is its own kind of broken.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Dashboard

The terminal's home screen — a grid you build yourself, a fixed whole-market rail beside it, and the controls that manage the board as a whole.

`>_ 42 widgets · 9 templates · a market rail · a board that remembers itself`

Covers · **the anatomy of the screen** · **the toolbar** · **working with tiles**
templates & RESET · the Market Pulse rail · the ticker and the status bar

Photographed on a live board, v5.0.1, without altering it.

Contents

10 SECTIONS

ORIENTATION

01 What the Dashboard is 3

02 Anatomy of the screen 4

THE FURNITURE

03 The top bar and navigation 5

04 Board controls — the toolbar 6

05 The ticker tape and the status bar 7

BUILDING THE BOARD

06 Working with tiles 8

07 The widget catalogue 9

08 Templates, and what RESET does 10

THE FIXED RAIL

09 The Market Pulse rail 11

REFERENCE

10 Quick reference — how do I… 12

About this guide. This is the guide to the **screen**. The catalogue of what you can put on it — all 42 widgets, category by category — is a companion volume, *Dashboard Widgets*, so that list lives in one place and stays current. Every figure here is a live capture of a working board.

01 What the Dashboard is

The Dashboard is the terminal's home screen, and the only one you build yourself. It is a grid of **tiles** — quotes, news, a chart, your portfolio, a live TV stream — each updating on its own. Beside the grid sits a fixed **Market Pulse** rail summarising the whole market, a scrolling ticker above it and a status bar reporting feed health along the bottom.

Nothing about the arrangement is fixed. Add, remove, drag, resize and configure tiles; the board saves itself and comes back the way you left it, on any machine you sign in from.

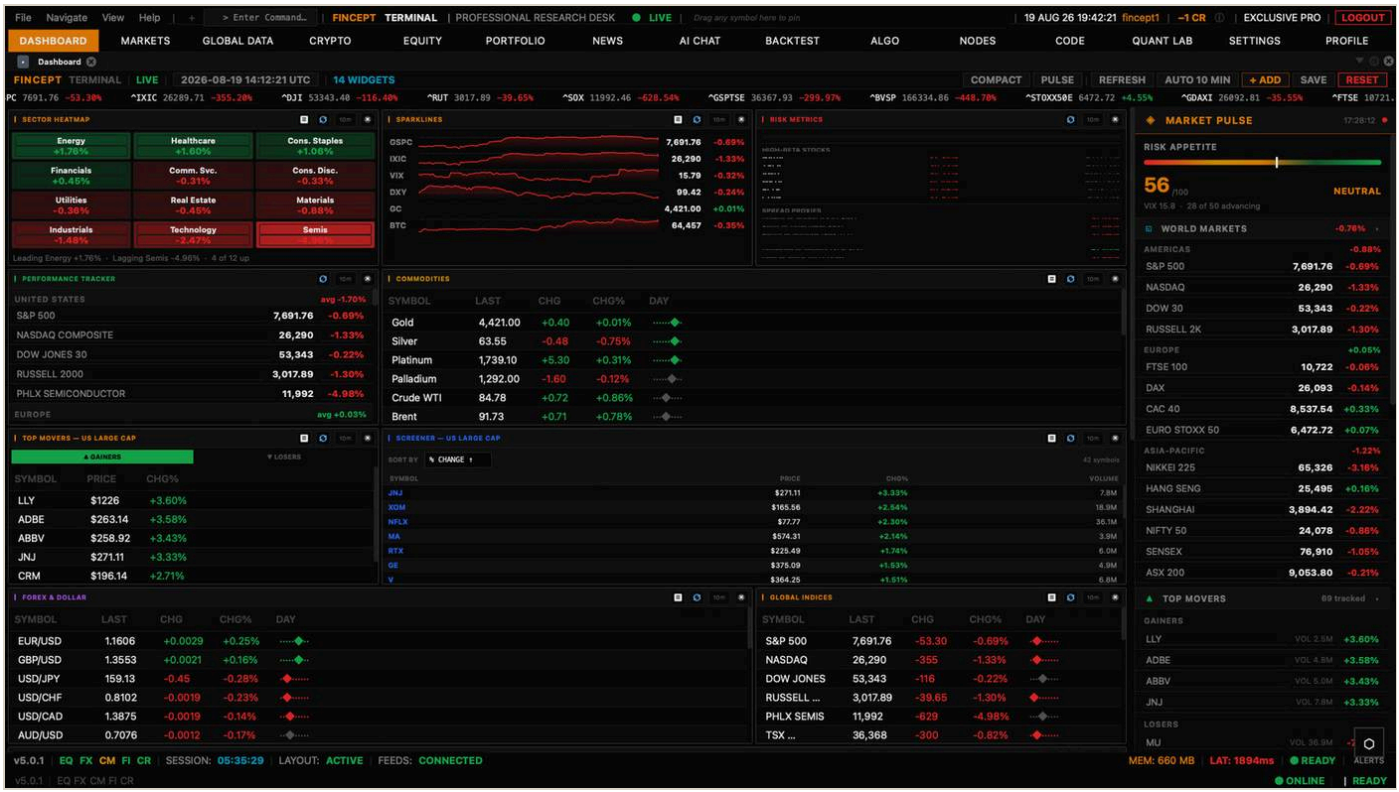


Fig 1 — A working board. Toolbar across the top, ticker under it, your tiles on a twelve-column grid, the Market Pulse rail down the right, and the status line along the foot.

At a glance, on the Dashboard you can…

- Build a layout from **42 widget types** and drag them into place
- Resize any tile on the grid, and set its own refresh interval
- Start from one of **9 ready-made boards** instead of an empty grid
- Track live markets — indices, watchlists, sparklines, heatmaps, movers
- Follow news and the economic calendar beside your positions
- Watch financial TV, or paste any stream you like
- Read the whole market at a glance in the Pulse rail
- Keep two accounts, four symbols or six feeds side by side at once
- Have it all reappear, arranged, the next time you sign in

> **This guide is the screen. Its companion is the parts.** What each of the 42 widgets shows, what it needs before it shows anything and what it costs is the *Dashboard Widgets* guide — see §07.

02 Anatomy of the screen

Seven regions, three of which are yours to change and four of which are furniture.

Region	What it is for	Yours?
1 Top bar	App menus, the command bar, pinned symbols, your account, credit balance, plan and sign-out. Present on every screen, not just this one. §03	—
2 Tab bar	The 15 top-level workspaces; another 37 screens open from **Navigate**. §03	—
3 Dashboard toolbar	Controls for the board itself — density, the rail, refresh, auto-refresh, add, save, reset. §04	—
4 Ticker tape	A scrolling marquee of your symbols with live price and change. §05	Symbols
5 Widget grid	Your tiles — the heart of the screen, and the only part you arrange. §06	**Yes**
6 Market Pulse rail	A fixed whole-market overview down the right. §09	Show / hide
7 Status bar	Feed lights, session timer, layout state, memory, latency, alerts and connectivity. §05	—

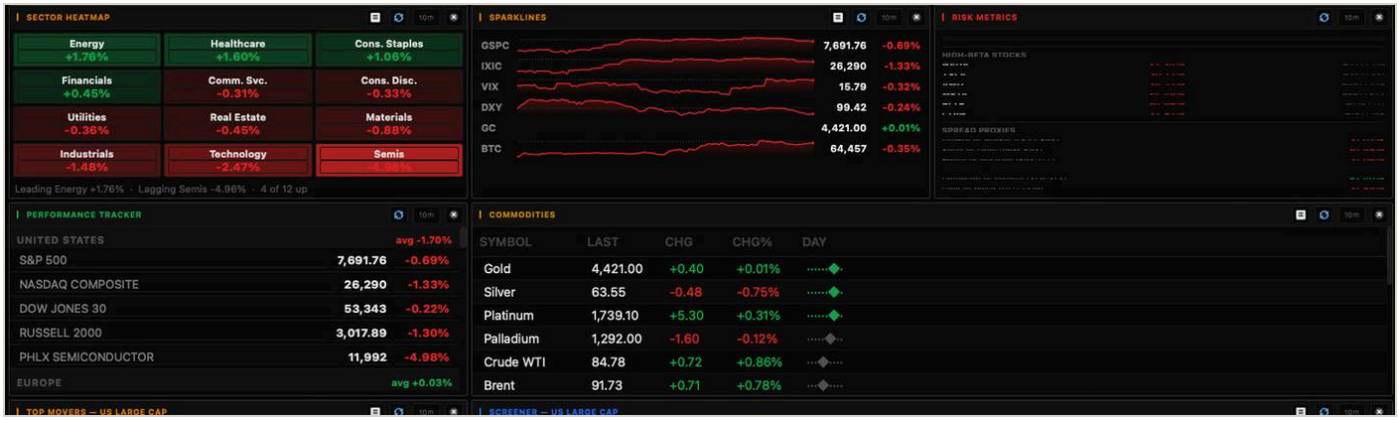


Fig 2 — Region 5, the grid. Every tile is independent: its own data, its own settings, its own refresh cadence.

> **The rail is not a widget.** Market Pulse cannot be added, removed or configured from the picker — it is part of the screen, and the toolbar's **PULSE** button is the only control it has. Some of what it shows is also available as tiles you *can* configure (Market Indices, Top Movers, Market Sentiment), which is the point: the rail is the fixed view, the tiles are the ones you aim.

03 The top bar and navigation

The two rows across the top of every screen — identity, navigation and your account.

Fig 3 — Row 1: menus, the command bar, the pin strip, then the clock, your account, the credit balance, your plan and sign-out.

Element	What it does
File / Navigate / View / Help	The app menus. **Navigate** jumps to any of the terminal's 37 non-tab screens; **View** carries fullscreen, focus mode, floating panels and screenshot
Command bar	Type a screen name and pick the row — `markets`, `portfolio`, `global economy`
Pinned symbols	Drag any ticker here to pin it. Click a chip to broadcast that symbol to linked panels, drag it onto a panel, or right-click to copy or remove it
Clock	The session clock, in the terminal's own time base
Account & credits	Your signed-in name and the metered credit balance (`CR`) — click it for the full price list
Plan badge	Your subscription tier. Click to open Plans & Pricing
LOGOUT	Signs out of the terminal

Fig 4 — Row 2: the fifteen top-level workspaces.

The tab bar carries **Dashboard · Markets · Global Data · Crypto · Equity · Portfolio · News · AI Chat · Backtest · Algo · Nodes · Code · Quant Lab · Settings · Profile**. Everything else — research desks, the data explorers, the workspace tools — opens from **Navigate** or by typing into the command bar.

> **If you are holding the July edition of this guide, its tab list is wrong.** It named a *QuantLib* tab, which no longer exists, and omitted **Global Data**, which replaced the five separate data-provider screens in 5.0. The list above is the shipping one.

04 Board controls — the toolbar

The row directly under the tab strip acts on the **whole board**. Its left half is a live status line; its right half is seven controls.

Fig 5 — Left: feed state (LIVE or OFFLINE), a clock you can click to swap UTC for local time, and a running count of the tiles on the board.

Fig 6 — Right: the seven board controls.

Control	What it does
COMPACT	Toggles a denser grid — shorter tile rows, so more fits on one screen. Press again to go back
PULSE	Shows or hides the Market Pulse rail, giving the grid the full width
REFRESH	Force-refreshes every live source at once — ticker, rail and every tile, whatever their own intervals say
AUTO *n*	The board-wide auto-refresh interval. Nine choices from **10 minutes to 24 hours**; tiles without an interval of their own follow it
+ ADD	Opens the widget picker — the only way to put a new tile on the board (§07)
SAVE	Writes the layout now. The board also saves itself shortly after any change, so this is belt and braces
RESET	Opens the template picker (§08). Nothing is cleared until you press APPLY on a card

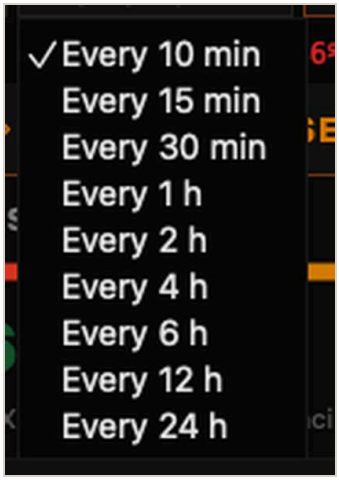


Fig 7 — The AUTO menu. Ten minutes is both the default and the floor — there is no faster option anywhere on the board.

> **COMPACT and PULSE are session toggles, not saved preferences.** They change what you are looking at now; the tiles, their sizes and their settings are what persist. Reopen the Dashboard and the rail is back.

05 The ticker tape and the status bar

The ticker

Fig 8 — A continuous marquee of your symbols with live price and change, green up and red down, immediately under the toolbar.

By default it carries the major indices and the day's movers. **Right-click the tape** and choose *Edit Symbols…* to set your own comma-separated list; it is saved for next time. Like the rail, the tape is part of the screen rather than a tile.

The status bar

Fig 9 — The foot of the screen: build, feed lights, session timer, layout state, feed state, memory, latency, alerts and connectivity.

Segment	What it means
`v5.0.1`	The build you are running
`EQ FX CM FI CR`	Feed lights for Equities, FX, Commodities, Fixed Income and Crypto
SESSION	How long the terminal has been open
LAYOUT	**ACTIVE** when the board has tiles on it, **EMPTY** when it does not
FEEDS	**CONNECTED** or **DISCONNECTED** — the market-data state
MEM · LAT	Memory in use, and the round-trip latency to the servers
ALERTS & bell	A pending-approvals count, and a bell that opens your notifications
ONLINE	Your connection to Fincept — **ONLINE**, **RECONNECTING** or **OFFLINE**
`G · symbol`	The symbol currently shared across linked panels, set by clicking a pinned chip

> **FEEDS and ONLINE are different questions.** ONLINE is whether the terminal can reach Fincept at all; FEEDS is whether market data is flowing. You can be online with a disconnected feed — that is the state where tiles hold their last value rather than going blank.

06 Working with tiles

Learn one tile and you have learned all of them. Each sits in a frame with its name on the left of the title bar and up to four controls on the right.

Fig 10 — A tile's title bar. The name is always upper-case; the controls sit at the right-hand end.

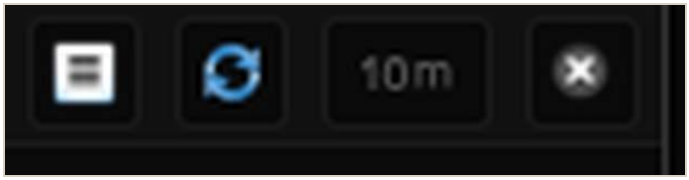


Fig 11 — The four controls, in order: configure, refresh now, refresh interval, close.

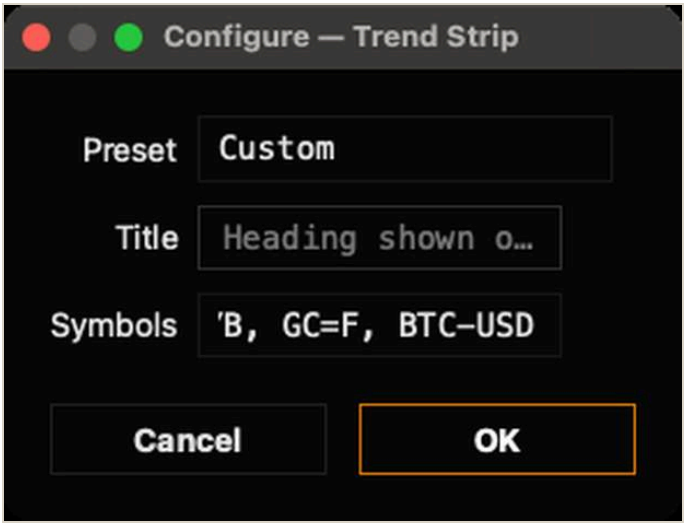


Fig 12 — A gear dialog. Every configurable tile opens one of these, titled *Configure — <tile>*, with OK and Cancel.

Action	How
Move	Drag the title bar. Other tiles slide out of the way and the grid re-flows behind you
Resize	Drag the bottom-right corner. It snaps to the twelve-column grid, and each widget has a minimum it will not go below
Refresh now	The ↻ icon reloads just that tile
Set an interval	The pill showing the current interval — `10m` — opens a menu with *Follow dashboard* at the top and the same nine choices as the toolbar
Configure	The list icon opens that tile's settings. Only tiles that have settings show it
Remove	The ✕ at the far right takes the tile off the board

> **Two tiles of the same kind never share settings.** Configuration belongs to the tile, not the widget type — which is what makes four charts on four symbols, or two positions tiles on two broker accounts, work at all.

> **✕ removes without asking.** No confirmation, no undo, and the tile's own settings go with it. Re-adding gives you a fresh, empty one.

07 The widget catalogue

+ ADD opens the picker: a search box, a row of category filters, and the catalogue as two columns of cards. There are **42 widgets in six categories**.

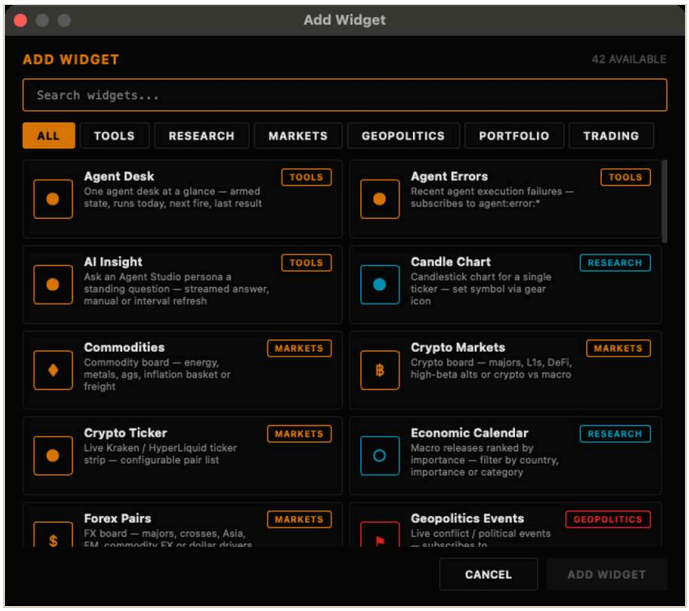


Fig 13 — The picker. The header counts what your plan can use; clicking a card only selects it, and nothing reaches your board until **ADD WIDGET**.

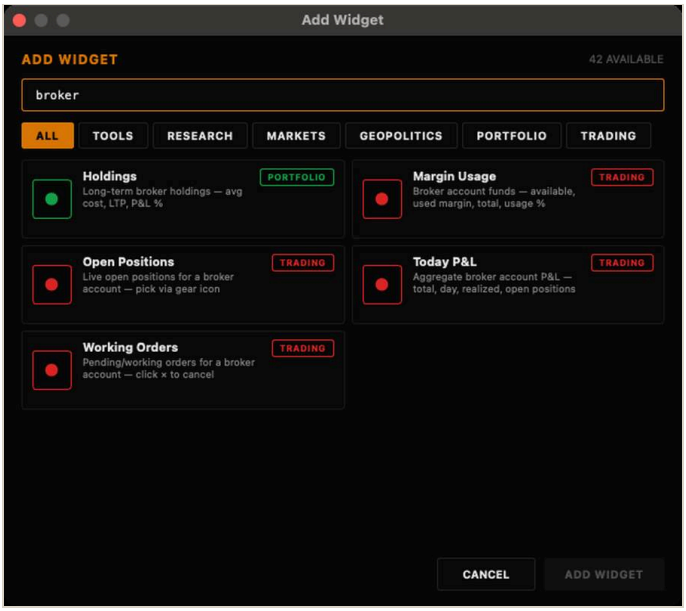


Fig 14 — Search matches name, description and category — typing *broker* finds the five tiles that need a broker account.

Category	Tiles	What lives there
Markets	12	Prices across every asset class — index, FX, crypto and commodity boards, heatmap, movers, sentiment, tickers, sparklines, the tape
Research	7	News, the economic calendar, single-name quote and chart, the screener, RSS
Portfolio	5	Watchlist, performance, portfolio summary, risk metrics, broker holdings
Trading	5	Order entry and live broker state — all five need a connected account
Tools	7	Agent output, AI insight, media, files, the scraper, notes
Geopolitics	6	Events, shipping, and the four Global Economy tiles — trade flows, treemap, world map, infrastructure

> **The full catalogue is a guide of its own.** *Dashboard Widgets* documents all forty-two, one row each: what it shows, what it needs before it shows anything, and what it costs. It is kept in one place on purpose — a catalogue copied into two documents is a catalogue that disagrees with itself within a release.

> **Only one tile on the whole board ever spends credits. AI Insight** runs an Agent Studio persona, and can be armed to do so on a timer; the two Global Economy trade tiles pass through the trade-corpus meter. The other thirty-nine are free however many you add and however often they refresh.

08 Templates, and what RESET does

RESET opens *Choose Dashboard Template* — nine ready-made boards, each shown as a miniature of its layout with a description and a tile count. Pressing **APPLY** on one replaces your board with it; pressing **CANCEL** does nothing at all.

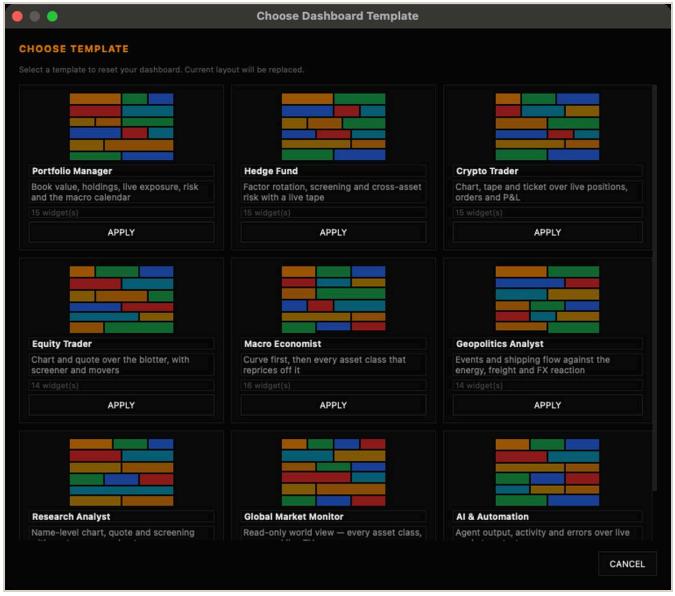


Fig 15 — The template picker. Every card carries a layout preview, a one-line description, the number of tiles it places and its own APPLY button.

Template	Its idea	Tiles
Portfolio Manager	Book value, holdings, live exposure, risk and the macro calendar	15
Hedge Fund	Factor rotation, screening and cross-asset risk with a live tape	15
Crypto Trader	Chart, tape and ticket over live positions, orders and P&L	15
Equity Trader	Chart and quote over the blotter, with screener and movers	14
Macro Economist	Curve first, then every asset class that reprices off it	16
Geopolitics Analyst	Events and shipping flow against the energy, freight and FX reaction	14
Research Analyst	Name-level chart, quote and screening with sector news and notes	13
Global Market Monitor	Read-only world view — every asset class, news and live TV	16
AI & Automation	Agent output, activity and errors over live market context	15

> **A template is a starting point, not a mode.** Applying one gives you an ordinary board you then edit tile by tile; nothing about it stays special afterwards.

> **APPLY replaces the whole board and the saved copy with it.** The dialog says so: *Current layout will be replaced*. There is no undo. If your board took work to build, look at the templates and press CANCEL.

09 The Market Pulse rail

The fixed column down the right-hand side: the same whole-market read for everybody, always current, with no settings. The toolbar's **PULSE** button hides it when you want the width.

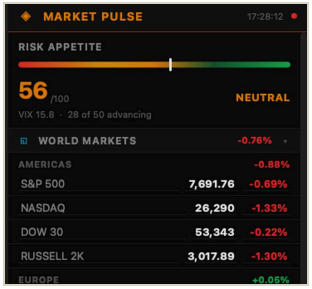


Fig 16 — Risk appetite: a 0–100 score with a label, the VIX and an advancing count beneath it.

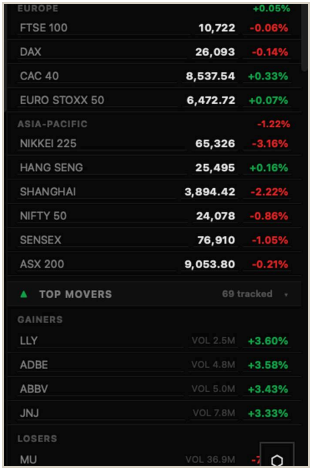


Fig 17 — World markets by region, then the day's movers with volume.

Section	What it shows
Risk appetite	A 0–100 sentiment score with a worded label, the VIX level, and how many of the tracked names are advancing
World markets	Index levels and moves grouped **Americas**, **Europe** and **Asia-Pacific**, with a headline move per region
Top movers	The day's strongest and weakest names, with volume

> **Some of the rail is computed here, not quoted.** The risk-appetite score and the advancing count are derived inside the terminal from live quotes — quick indicators, not official exchange statistics. The index levels and the movers are quoted data.

> **The rail and its tile equivalents are not duplicates.** Market Indices, Top Movers and Market Sentiment cover the same ground as tiles you can point at a region, a sector or a basket of your own. The rail is the view you do not have to configure.

10 Quick reference — how do I…

I want to…	Do this
Add a tile	**+ ADD** → click a card → **ADD WIDGET**
Find a tile without scrolling	Type in the picker's search box — it matches names, descriptions and categories
Know what a tile does before adding it	Its card carries a one-line description; the *Dashboard Widgets* guide carries the full row
Move a tile	Drag its title bar; the others make room
Make a tile bigger	Drag its bottom-right corner; it snaps to the grid
Change a tile's symbols or account	Its gear icon → **OK**
Remove a tile	The ✕ in its title bar — no confirmation, no undo
Reload one tile	Its ↻ icon
Reload everything	**REFRESH** on the toolbar
Change how often the board updates	**AUTO** — nine choices, 10 min to 24 h
Give one tile its own cadence	Its interval pill (`10m`) → pick an interval instead of *Follow dashboard*
Fit more on screen	**COMPACT**, and **PULSE** to reclaim the rail
Change the ticker's symbols	Right-click the tape → *Edit Symbols…*
Start from a professional layout	**RESET** → a template card's **APPLY**
Look at the templates without risk	**RESET** → read → **CANCEL**
Save right now	**SAVE** — though the board saves itself after any change
Get the same board on another machine	Nothing — dashboard layouts follow your account
Check whether data is actually flowing	The status bar: **FEEDS CONNECTED**, and the five feed lights
Avoid spending credits from the board	Leave **AI Insight** on *Manual only*; everything else here is free
Read about a specific widget	The *Dashboard Widgets* guide — all 42, one table

> **The short version.** Press **+ ADD**, pick a card, press **ADD WIDGET**. Drag to move, corner-drag to resize, gear to configure, ✕ to remove. Everything saves itself and follows your account, and only one tile in the catalogue ever costs anything.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Dashboard Widgets

Every tile you can put on the board — what it shows, what it needs, what it costs, and the handful of mechanics that are the same on all of them.

`>_ 42 widgets · 6 categories · 9 templates · 1 that spends credits`

Covers · **the picker** · **tile chrome and resizing** · **the three layers of refresh**
All six categories, tile by tile · templates, persistence and the one metered tile

Photographed on the running terminal, v5.0.1, without altering a live board.

Contents

18 SECTIONS

GETTING ORIENTED

01 The board 3
02 The toolbar 4
03 The ticker, the rail and the status line 5

BUILDING A BOARD

04 Adding a tile 6
05 Every tile, the same moves 7
06 Refresh — three layers 8
07 Where your board lives 9
08 Templates, and what RESET does 10

THE CATALOGUE

09 Markets — twelve tiles 11
10 Research — seven tiles 12
11 Portfolio — five tiles 13
12 Trading — five tiles 14
13 Tools — seven tiles 15
14 Geopolitics — six tiles 17

REFERENCE

15 What a board costs 19
16 All 42 widgets, one table 20
17 Honest notes 22
18 Quick reference — how do I… 23

About this guide. Every widget name, category and one-line description here is the terminal's own text, read out of the live **+ ADD** picker rather than rewritten. Where a tile needs setting up before it shows anything, that is said in its row — those are the tiles that look broken if you skip the gear.

01 The board

The Dashboard is the only screen in the terminal you build yourself. Everything on it is a **tile** you chose, sized and placed — and the catalogue you choose from holds **42 widgets across six categories**. This guide is that catalogue, plus the small set of mechanics that work the same way on every tile.

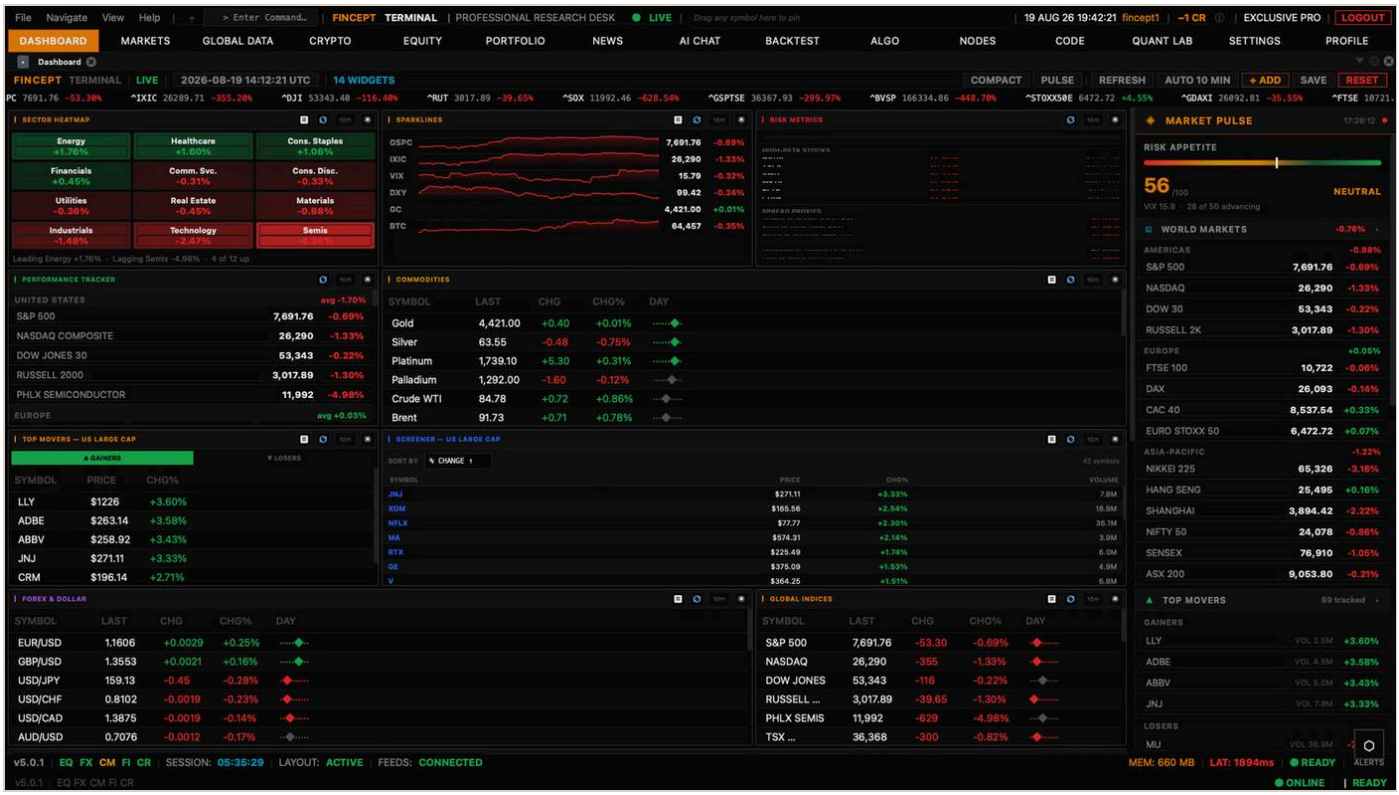


Fig 1 — A working board. Toolbar across the top, a scrolling ticker under it, your tiles on a twelve-column grid, the fixed Market Pulse rail down the right, and a status line along the bottom. Only the grid is yours to arrange.

The six categories

- **Markets** — 12 tiles: prices across every asset class
- **Research** — 7 tiles: news, the economy, single names
- **Portfolio** — 5 tiles: holdings, performance, risk
- **Trading** — 5 tiles: order entry and live broker state
- **Tools** — 7 tiles: agents, media, files, scratch
- **Geopolitics** — 6 tiles: events, shipping, trade, infrastructure

> **Nothing here is plan-locked.** The picker's header counts the widgets your plan can use, and on the account this guide was written from it counted **42 of 42**. Tiles that need a broker, a portfolio or a configured feed will say so in their own body rather than being hidden from you.

02 The toolbar

The row directly under the tab strip acts on the **whole board**. Its left half is a status line; its right half is seven controls.

Fig 2 — Left: feed state (LIVE or OFFLINE), a clock you can click to swap UTC for local time, and a running count of the tiles on the board.

Fig 3 — Right: the seven board controls.

Control	What it does
COMPACT	Halves the grid row height so more fits on one screen. A toggle — press it again to go back.
PULSE	Shows or hides the Market Pulse rail on the right, giving the grid the full width.
REFRESH	Force-refreshes every live source at once — ticker, rail and every tile, regardless of their own intervals.
AUTO *n*	The board-wide auto-refresh interval. Opens a menu of nine choices from 10 minutes to 24 hours (see §06).
+ ADD	Opens the widget picker (§04). The only way to put a new tile on the board.
SAVE	Writes the layout now. The board also saves itself shortly after any change, so this is a belt-and-braces button rather than a requirement.
RESET	Opens the template picker (§08). Nothing is cleared until you press APPLY on a template card.

> **RESET is safe to open.** It does not clear anything by itself — it opens a chooser, and the board is only replaced when you press **APPLY** inside one of the template cards. **CANCEL** leaves your layout exactly as it was.

> **COMPACT and PULSE are session toggles, not saved preferences.** They change what you are looking at now; the tiles, their sizes and their settings are what persist. Reopen the Dashboard and the rail is back.

03 The ticker, the rail and the status line

Three parts of the Dashboard are **not** tiles and cannot be added, removed or configured from the picker. They are worth knowing so you do not go looking for them in the catalogue.

Fig 4 — The ticker: a continuous marquee of index and symbol quotes with their change, immediately under the toolbar. It is part of the screen, not a widget.

The Market Pulse rail

The fixed column down the right-hand side — a risk-appetite gauge with a 0–100 reading, world markets by region, and a movers list. It is the same for everybody and always current; the **PULSE** button hides it when you want the width back.

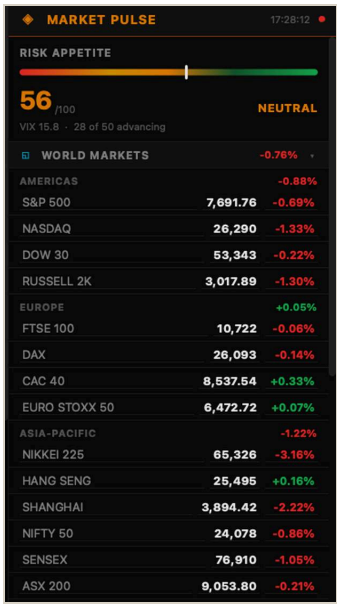


Fig 5 — Market Pulse: risk appetite, then world markets grouped Americas / Europe / Asia-Pacific, then top movers.

Fig 6 — The status line: build number, feed lights, session timer, layout state, memory, latency and readiness.

> **Some rail content also exists as tiles, deliberately.** Market Indices, Top Movers and Market Sentiment cover the same ground as parts of the rail. The rail is a fixed whole-market view; the tiles are the versions you can point at a region, a sector or a basket of your own.

04 Adding a tile

+ ADD opens the widget picker: a search box, a row of category filters, and the catalogue as two columns of cards. Each card carries the widget's icon in its category colour, its name, its one-line description and a category badge.

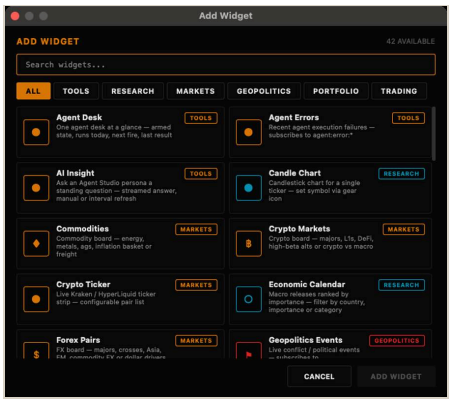


Fig 7 — The picker on ALL. The header counts what your plan can use; the cards are ordered by internal id, which is why Agent Desk, Agent Errors and AI Insight lead.

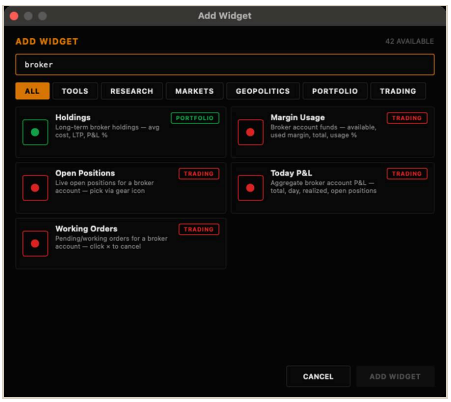


Fig 8 — Search matches name, description and category. Typing *broker* finds the five tiles that need a broker account.

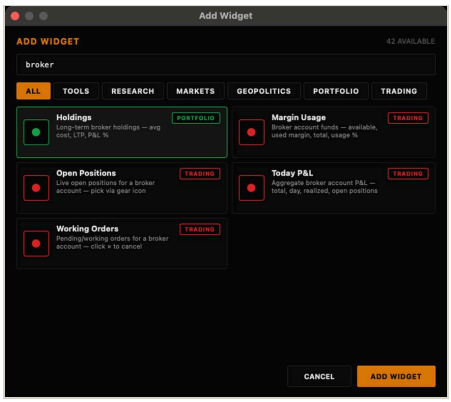


Fig 9 — Selected, not added. Holdings is highlighted in Portfolio green and ADD WIDGET has gone live. CANCEL still changes nothing.

Choose, then add

Clicking a card **selects** it — the card takes its category colour as a border and the **ADD WIDGET** button turns from grey to amber. Nothing reaches your board until you press that button, so you can browse and click freely.

Control	Behaviour
Search box	Filters as you type, against the widget name, its description and its category — so *crypto*, *broker* or *news* all work. Combines with the category filter.
Category buttons	ALL plus the six categories. Single-select — picking one replaces the previous filter.
A card	One click selects. There is no double-click shortcut and no drag-to-board.
ADD WIDGET	Disabled until something is selected. Adds one tile at the first free spot on the grid and closes the dialog.
CANCEL	Closes with nothing added. **Esc** does the same.

> **Duplicates are allowed and are the point.** Add Candle Chart four times and set a different symbol on each; add Open Positions twice for two broker accounts. Each tile keeps its own settings, so the catalogue is a list of *kinds* of tile, not a list of slots.

05 Every tile, the same moves

Learn one tile and you have learned all forty-two. Each sits in a frame with its name on the left of the title bar and up to four controls on the right.

Fig 10 — A tile's title bar. The name is always upper-case; the controls sit at the right-hand end.

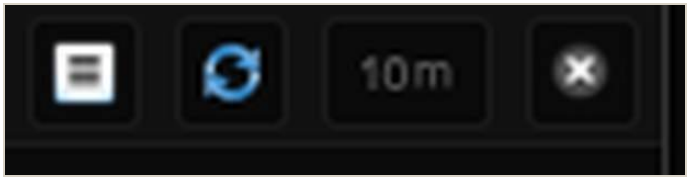


Fig 11 — The four controls, in order: configure, refresh now, refresh interval, close.

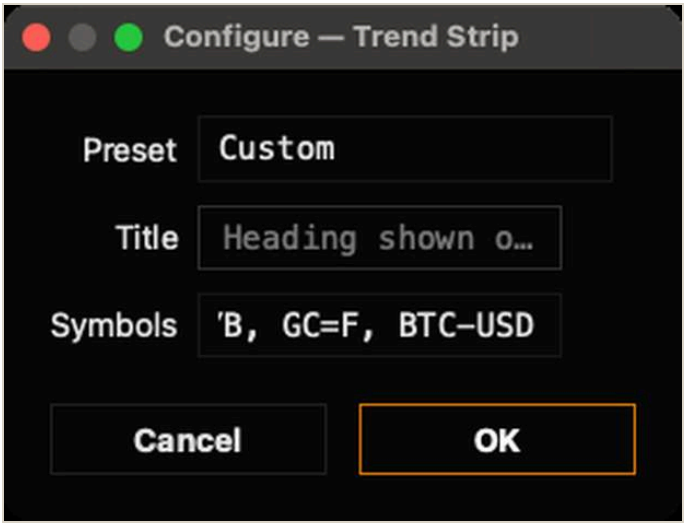


Fig 12 — A gear dialog. Every configurable tile opens one of these, titled *Configure — <tile>*, with OK and Cancel.

Action	How
Move	Drag the title bar. Other tiles slide out of the way and the grid re-flows behind you.
Resize	Drag the bottom-right corner. It snaps to the twelve-column grid, and each widget has a minimum below which it will not go.
Refresh now	The ↻ icon reloads just that tile.
Set an interval	The pill showing the current interval — `10m` — opens a menu (§06).
Configure	The list icon opens that tile's settings. Only tiles that have settings show it.
Remove	The ✕ at the far right takes the tile off the board.

> **A tile that is loading says so.** Tiles show `LOADING…` while they fetch, and one that comes back with nothing shows *No data yet — click refresh to retry* rather than an empty rectangle. An empty-looking tile is usually one that has not been configured yet, not one that is broken.

> **✕ removes without asking.** There is no confirmation and no undo on a tile close. The tile's own settings go with it, so a Watchlist you had typed a symbol list into is gone rather than parked. Re-adding gives you a fresh, empty one.

06 Refresh — three layers

Data arrives three different ways, and knowing which is which explains almost every "why is this number old" question.

1 · Streams, which do not need refreshing at all

Several tiles subscribe to a live feed rather than polling — Trade Tape, Crypto Ticker, Sparklines, Agent Errors, Geopolitics Events, Maritime Vessels. They update when the feed does. The interval control on those tiles governs their fallback fetch, not the stream.

2 · The board default

The toolbar's **AUTO** button sets one interval for the whole board: *widgets without their own interval re-fetch every n*. Nine choices, from 10 minutes to 24 hours — 10 minutes is both the default and the floor.

3 · Per tile

Any tile can opt out of the board default and keep its own cadence. Its interval pill opens the same list with **Follow dashboard** at the top, which is where every tile starts.

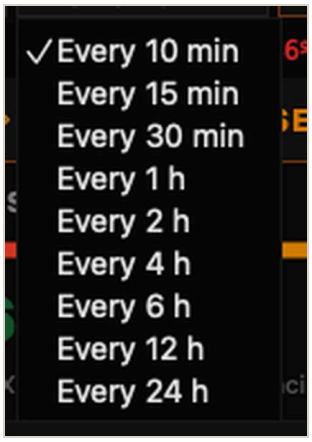


Fig 13 — The board-wide interval menu, from the AUTO button. Ten minutes is the shortest allowed.

Fig 14 — The same list on one tile, with *Follow dashboard* at the top. The pill on the title bar always shows what is in force.

Where	Label	Range
Toolbar	`AUTO 10 MIN`	10 min · 15 min · 30 min · 1 h · 2 h · 4 h · 6 h · 12 h · 24 h
Tile	`10m`	Follow dashboard, then the same nine choices
Toolbar	`REFRESH`	Once, now, everything — ignores every interval
Tile	↻	Once, now, this tile only

> **Ten minutes is a floor, not a suggestion.** There is no one-minute option anywhere in either menu. Prices you need to watch tick by tick belong on a streaming tile — Crypto Ticker, Trade Tape — or on the Markets and trading screens, not on a polled dashboard tile.

07 Where your board lives

PERSISTENCE · 07

A board is three things saved together: **which tiles**, **where and how big** each one is, and **each tile's own settings**. All three are written whenever you change something, and again when you press SAVE.

Question	Answer
When is it saved?	Shortly after any change — add, remove, move, resize or a settings change. **SAVE** forces it immediately.
What is saved per tile?	Its type, its grid position and size, and its configuration object — the symbol list, the chosen broker account, the persona and prompt, and so on.
Does it follow me to another machine?	Dashboard layouts are one of the domains mirrored to your account, so a second machine picks up the board rather than starting blank.
What is *not* saved?	The COMPACT and PULSE toggles, and scroll positions inside tiles. Those are per-session.
What happens on a fresh install?	An empty grid with the template picker one click away — see §08.

Two tiles of the same kind never share settings. Configuration is stored against the tile instance, not the widget type, which is what makes four Candle Charts on four symbols — or two Open Positions tiles on two broker accounts — work at all.

The board is per-user, not per-window. Opening the Dashboard in a second dock panel shows the same board; it is not a second workspace. If you want two different arrangements, that is what the terminal's saved layouts are for, one level up from this screen.

08 Templates, and what RESET does

RESET opens *Choose Dashboard Template* — nine ready-made boards, each shown as a miniature of its layout with a description and a tile count. Pressing **APPLY** on one replaces your board with it. Pressing **CANCEL** does nothing at all.

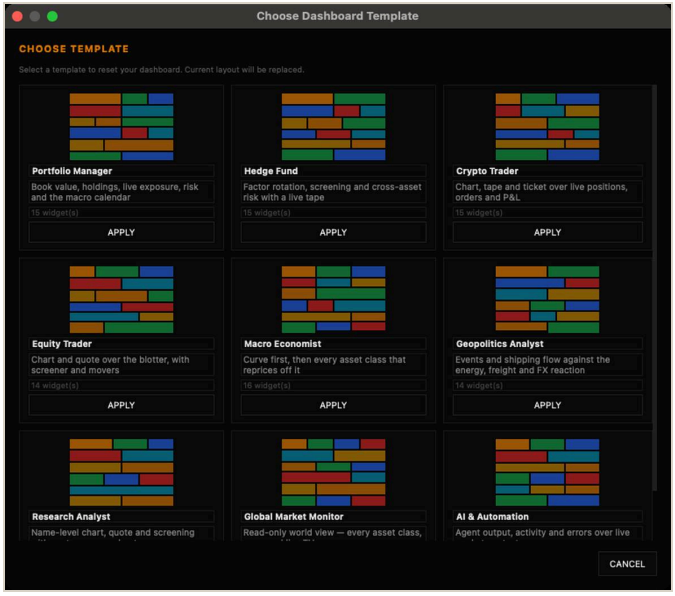


Fig 15 — The template picker. Every card carries a layout preview, a one-line description, the number of tiles it places and its own APPLY button.

Template	Its idea	Tiles
Portfolio Manager	Book value, holdings, live exposure, risk and the macro calendar	15
Hedge Fund	Factor rotation, screening and cross-asset risk with a live tape	15
Crypto Trader	Chart, tape and ticket over live positions, orders and P&L	15
Equity Trader	Chart and quote over the blotter, with screener and movers	14
Macro Economist	Curve first, then every asset class that reprices off it	16
Geopolitics Analyst	Events and shipping flow against the energy, freight and FX reaction	14
Research Analyst	Name-level chart, quote and screening with sector news and notes	13
Global Market Monitor	Read-only world view — every asset class, news and live TV	16
AI & Automation	Agent output, activity and errors over live market context	15

> **A template is a starting point, not a mode.** Applying one gives you an ordinary board you can then edit tile by tile — nothing about it stays special afterwards, and no tile behaves differently for having arrived that way.

> **APPLY replaces the whole board and the saved copy with it.** The dialog says so: *Current layout will be replaced.* There is no undo. If your board took work to build, look at the templates and press CANCEL.

09 Markets — twelve tiles

The largest category: live prices across every asset class. Four of them — Indices, Forex, Crypto and Commodities — are **boards**, meaning one tile that can be pointed at any of several preset baskets from its gear.

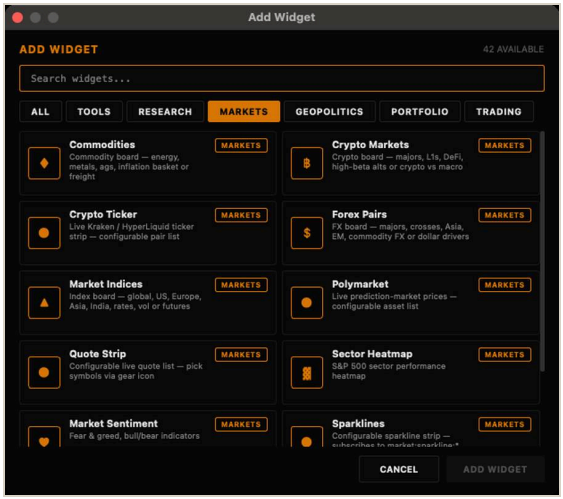


Fig 16 — The Markets category. Amber is the Markets accent; every card in the picker is coloured by category.

Widget	What it shows	Set up
Market Indices	Index board — global, US, Europe, Asia, India, rates, vol or futures	Pick the board from the gear
Forex Pairs	FX board — majors, crosses, Asia, EM, commodity FX or dollar drivers	Pick the board from the gear
Crypto Markets	Crypto board — majors, L1s, DeFi, high-beta alts or crypto vs macro	Pick the board from the gear
Commodities	Commodity board — energy, metals, ags, inflation basket or freight	Pick the board from the gear
Sector Heatmap	S&P 500 sector performance heatmap — tiles shaded by the size of the move	None
Top Movers	Biggest gainers and losers — rank US equity, a sector, crypto, FX or commodities	Optional: choose the universe
Market Sentiment	Fear & greed, bull/bear indicators, advancers against decliners and the VIX	None
Quote Strip	Configurable live quote list — pick symbols via gear icon	Yes — it is empty until you name symbols
Crypto Ticker	Live Kraken / HyperLiquid ticker strip — configurable pair list	Yes — set the pair list
Polymarket	Live prediction-market prices — configurable asset list	Yes — add the markets to track
Sparklines	Configurable sparkline strip — a mini trend line and last price per symbol	Yes — choose the symbols
Trade Tape	Live trade prints for a crypto pair, as they happen	Yes — set exchange and pair

> **Four of these arrive empty on purpose.** Quote Strip, Crypto Ticker, Polymarket and Sparklines have no sensible default symbol list, so a freshly added tile shows its setup prompt rather than someone else's watchlist. Open the gear and name what you want.

10 Research — seven tiles

News, the economy, and single-name work. Two of these — Stock Quote and Candle Chart — are the per-symbol tiles that make duplication worthwhile: add one per name you follow.

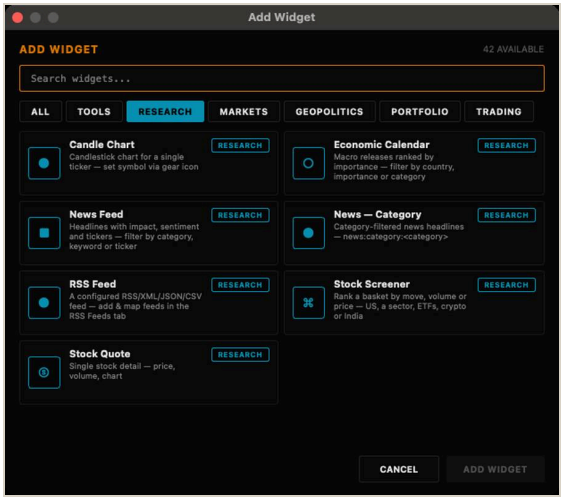


Fig 17 — The Research category, in cyan.

Widget	What it shows	Set up
News Feed	Headlines with impact, sentiment and tickers — filter by category, keyword or ticker	Optional filters
Stock Quote	Single stock detail — price, change, open, previous close, high, low and volume	Set the symbol (defaults to AAPL)
Candle Chart	Candlestick chart for a single ticker — set symbol via gear icon	Set the symbol
Stock Screener	Rank a basket by move, volume or price — US, a sector, ETFs, crypto or India	Optional: choose the basket and sort
Economic Calendar	Macro releases ranked by importance — filter by country, importance or category	Optional filters
News — Category	Category-filtered news headlines, narrowed to one topic	Yes — pick the category
RSS Feed	A configured RSS/XML/JSON/CSV feed — add & map feeds in the RSS Feeds tab	Yes — the feed must exist first

RSS Feed is the one tile that depends on another tab. It shows a feed you have already added and mapped in the **RSS Feeds** screen; its gear picks from that list rather than taking a URL. If the list is empty, add the feed there first. **Web Scraper** (§13) is the tile that does take a bare URL.

News Feed and News — Category are not the same tile twice. News Feed is the whole wire with filters you can change; News — Category is bound to one topic and stays there, which is what you want when three of them sit side by side.

11 Portfolio — five tiles

Your holdings, your performance and your risk. Watchlist is the odd one out — it is the only tile in the catalogue you edit *in place*, by typing symbols into the tile itself.

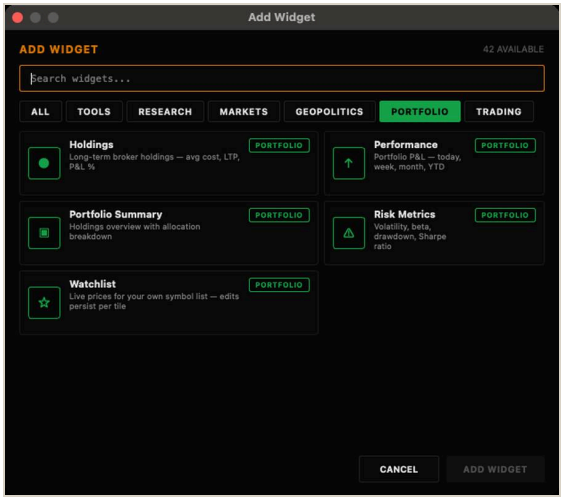


Fig 18 — The Portfolio category, in green.

Widget	What it shows	Set up
Watchlist	Live prices for your own symbol list — edits persist per tile	Type comma-separated symbols into the tile and press GO
Performance	Portfolio P&L — today, week, month, YTD	A portfolio must exist
Portfolio Summary	Holdings overview with allocation breakdown — total value, day P&L, total P&L, holdings count and a positions table	A portfolio must exist
Risk Metrics	Volatility, beta, drawdown, Sharpe ratio	A portfolio must exist
Holdings	Long-term broker holdings — avg cost, LTP, P&L %	A connected broker account

> **Holdings is a broker tile filed under Portfolio.** It reads your *broker's* long-term holdings, not the portfolio you keep in the Portfolio screen — so it stays empty until a broker is connected under Settings → Credentials, however full your portfolios are. The four tiles above it read the portfolios.

> **Watchlist tiles and the Watchlist screen are separate lists.** A symbol list typed into a tile lives with that tile. Use several tiles for several themes; use the Watchlist screen when you want one list with alerts on it.

12 Trading — five tiles

Order entry and live account state. **All five need a connected broker** — they are the five tiles the search in Fig 8 turned up, and they are the ones most likely to look broken if you add them before connecting an account.

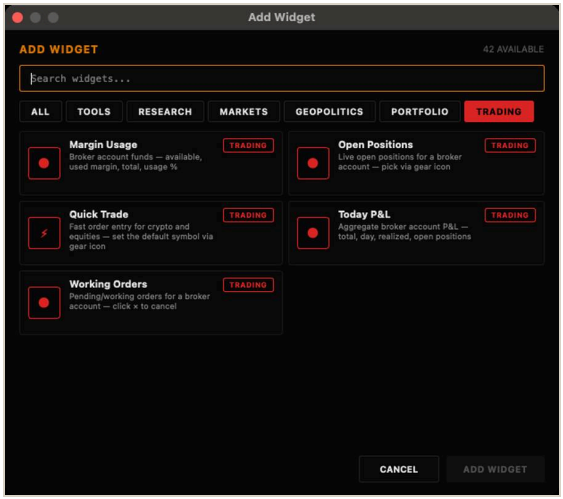


Fig 19 — The Trading category, in red.

Widget	What it shows	Set up
Quick Trade	Fast order entry for crypto and equities — set the default symbol via gear icon	Broker; optionally a default symbol
Open Positions	Live open positions for a broker account — pick via gear icon	Broker; pick the account
Working Orders	Pending/working orders for a broker account — click × to cancel	Broker; pick the account
Margin Usage	Broker account funds — available, used margin, total, usage %	Broker; pick the account
Today P&L	Aggregate broker account P&L — total, day, realized, open positions	Broker; pick the account

Working Orders cancels live orders from the board. The × on a row is an order cancel, not a tile close — and the tile's own close button is the × up in the title bar. They are in different places and do very different things; check which one you are reaching for.

Quick Trade sends real orders. It is an order ticket on a dashboard tile, with the same reach as the trading screens. Treat a board carrying one the way you would treat an open ticket.

Two accounts, two tiles. Each of these binds to one broker account through its gear, so the way to watch two accounts is two tiles side by side rather than a switcher inside one.

13 Tools — seven tiles

Everything that is not a price: agent output, media, files and scratch. Two of these — **Agent Desk** and **AI Insight** — are new since the previous edition of this guide and are worth reading properly, because one of them is the only tile on the board that spends credits.

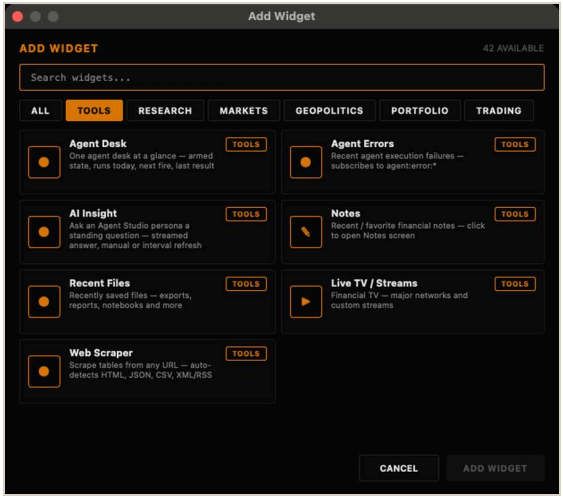


Fig 20 — The Tools category, in amber.

Widget	What it shows	Set up
Agent Desk	One agent desk at a glance — armed state, runs today, next fire, last result	Pick the desk from the gear
AI Insight	Ask an Agent Studio persona a standing question — streamed answer, manual or interval refresh	Pick a persona and write the question
Agent Errors	Recent agent execution failures — an empty list means everything is healthy	None
Notes	Recent / favorite financial notes — click to open Notes screen	None
Recent Files	Recently saved files — exports, reports, notebooks and more	None
Live TV / Streams	Financial TV — major networks and custom streams, plus any YouTube, HLS or MP4 URL	None; paste a URL for a custom stream
Web Scraper	Scrape tables from any URL — auto-detects HTML, JSON, CSV, XML/RSS	Yes — give it a URL

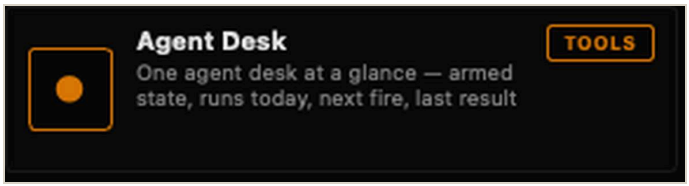


Fig 21 — Agent Desk.

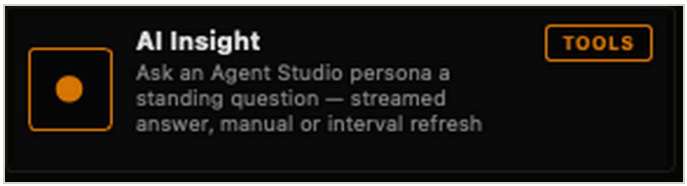


Fig 22 — AI Insight.

Tools — Agent Desk and AI Insight

Agent Desk

A read-only status card for **one** desk from Agent Studio. Bind it with the gear — the dialog is titled *Configure — Agent Desk* and offers a single **Desk** list — and the tile then shows that desk's name and domain, whether it is ⏻ **ARMED** or **DISARMED**, a *RUNNING* flag while a run is in flight, *Runs today* with the *Next fire* time, and a snippet of the last result. Before it is bound it reads *No desk selected — pick a desk via the gear icon*; if the desk is later deleted it says so rather than going blank.

It **observes**. Nothing on this tile arms, disarms or triggers a desk, and it costs nothing — it polls status while it is visible and stops when it is not.

AI Insight

A **standing question** pinned to the board. Choose an Agent Studio persona — an agent or a team — type a question such as *"What moved crypto today?"*, and the tile shows that persona's answer, streamed in as it is written, with an *Updated* stamp. Its gear dialog, *Configure — AI Insight*, has three fields: **Persona**, **Prompt** and **Refresh every** *n* min, where zero means **Manual only**.

Field	What it does
Persona	Any agent or team from Agent Studio. A **team** costs about four times an agent, so this field changes the tile's price.
Prompt	The standing question. It is asked verbatim, every time.
Refresh every	0 = manual only, the default. Otherwise a timer. **Minimum 15 minutes for an agent, 60 for a team.**
↻ on the tile	Asks now. It carries its own price on the control and refuses cleanly if the balance will not cover it.

> **This tile can charge you with no click at all.** Set an interval and the timer spends credits on its own schedule. The configuration dialog states the rate before you arm it — per ask, per day and the projected monthly total — because a per-run figure reads as nothing and a monthly one does not. The 15- and 60-minute floors exist for the same reason.

> **Leave it on Manual only unless you want a subscription.** Manual costs exactly what you press, and the answer stays on the tile to re-read for free. Re-reading is never charged; only asking again is.

14 Geopolitics — six tiles

The category that has changed most. It was two tiles; it is now six, because the four views of the **Global Economy** screen — the chord, the treemap, the world choropleth and the infrastructure map — each ship as a tile you can pin to a board.

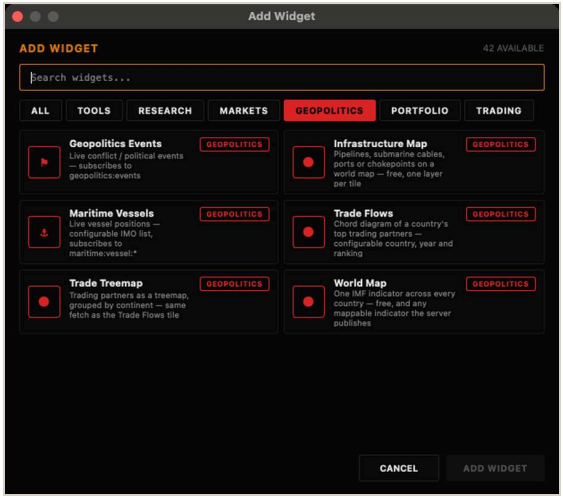


Fig 23 — All six Geopolitics tiles on one screen. Four of them are new.

Widget	What it shows	Set up
Geopolitics Events	Live conflict / political events with category, region and date	None
Maritime Vessels	Live vessel positions — configurable IMO list	Optional: name the ships to follow
Trade Flows	Chord diagram of a country's top trading partners — configurable country, year and ranking	Country, year, ranking, partner count
Trade Treemap	Trading partners as a treemap, grouped by continent — same fetch as the Trade Flows tile	Same four settings
World Map	One IMF indicator across every country — free, and any mappable indicator the server publishes	Pick the indicator
Infrastructure Map	Pipelines, submarine cables, ports or chokepoints on a world map — free, one layer per tile	Pick the layer

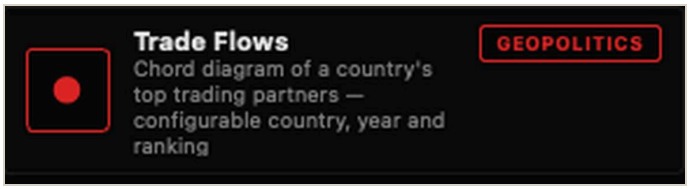


Fig 24 — Trade Flows.

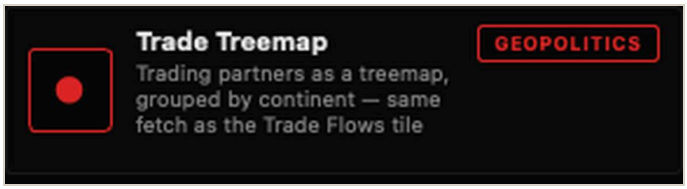


Fig 25 — Trade Treemap.

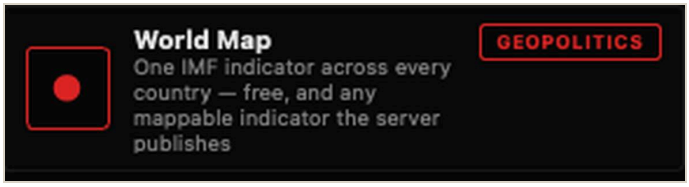


Fig 26 — World Map.

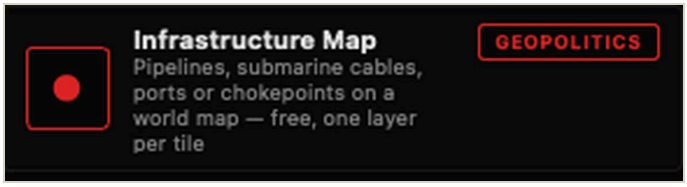


Fig 27 — Infrastructure Map.

The four Global Economy tiles

These are not screenshots of the Global Economy screen dropped on a board — they are separate tiles with their own settings, their own status line and their own fetch. Two of them share one.

Trade Flows and Trade Treemap

The same query, drawn two ways. Both take four settings from *Configure — Trade Flows*: **Country**, **Year** (with *Latest available* as the default), **Rank by** and **Partners drawn**. Rank by offers **Total Trade**, **Imports**, **Exports** or **Trade Balance**. The chord draws each partner as a ribbon whose width is the flow; the treemap draws the same partners as rectangles grouped by continent.

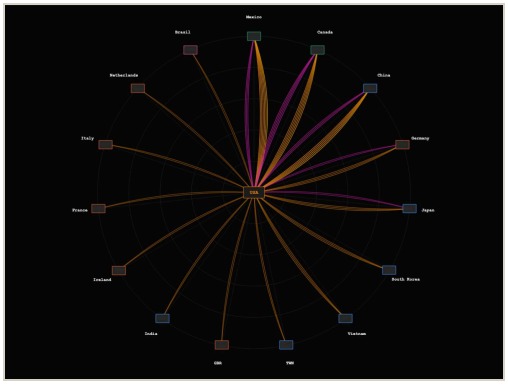


Fig 28 — What the Trade Flows tile draws, shown full-size on the Global Economy screen: partners around the ring, ribbons scaled to the flow.

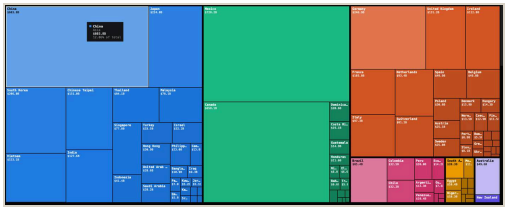


Fig 29 — The treemap of the same query — the same partners, area-scaled and grouped by continent.

World Map

Every country shaded by **one** economic indicator, chosen in *Configure — World Map* along with a **Colour** mode: **Continent hue**, which tints each continent its own colour, or **Single ramp**, one scale across the world. It opens on IMF nominal GDP and offers any mappable indicator the server publishes. Unconfigured it reads *Pick an indicator in this tile's settings*.

Infrastructure Map

One layer per tile — pipelines, submarine cables, ports or chokepoints — with two further filters in *Configure — Infrastructure*: a **Status** (*Every status*, operational, *Under construction*, *Proposed*, *Cancelled or shelved*) and a **Viewport**, a bounding box entered as `minLon,minLat,maxLon,maxLat` and left empty for the whole world.

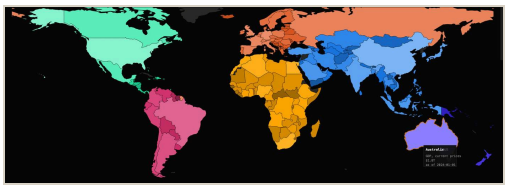


Fig 30 — The choropleth the World Map tile draws.

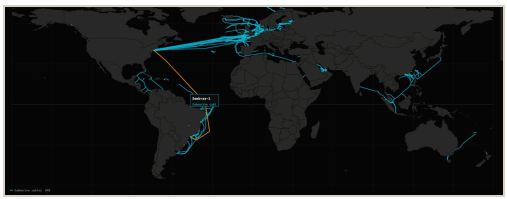


Fig 31 — An infrastructure layer — submarine cables — as the Infrastructure Map tile draws it.

> **The two map tiles are free; the two trade tiles are metered.** The picker says so in the descriptions themselves — *free, and any mappable indicator*, *free, one layer per tile*. Trade Flows and Trade Treemap run the same corpus query the Global Economy screen's **Load** button runs, and are priced the same way. See §15.

15 What a board costs

Almost nothing. Of the forty-two tiles, **one** spends credits by design and **two** pass through a metered query; the other thirty-nine are free however many you add and however often they refresh.

Tile	Charge	When
AI Insight	One agent run per ask — about four times that for a team	Every ↻, and every tick of its interval if you arm one
Trade Flows	The same trade-corpus query the Global Economy screen's **Load** charges for	On the tile's fetch, and on each refresh
Trade Treemap	The same query again — it is the same fetch, drawn differently	As above
Everything else	Free	—

The rules that make this safe

- **Market data is never metered.** Every price, index, quote, chart, heatmap, screener and ticker on the board is free, at any refresh interval you like.
- **Broker data is never metered.** Positions, orders, margin, holdings and P&L come from your own account.
- **Re-reading is never metered.** An answer already on a tile can be read as often as you like; only asking again costs.
- **Anything that runs on your machine is free** — notes, files, the scraper, local computation.

> **The one thing to watch is an armed AI Insight tile.** It is the only control on the Dashboard that can spend money while you are not looking at it. If you want the answer once a day, set 24 h — not the floor. If you want it when you ask, leave the interval at **Manual only**, which is where it starts.

> **A trade tile on a short interval is a repeated query.** Trade Flows and Trade Treemap re-run their corpus query on every refresh, so two of them on the same country and year are two queries, not one shared one. Give them a long interval — the data is annual.

16 All 42 widgets, one table

FULL CATALOGUE · 16

The whole catalogue, in the order the picker lists it — by internal id, which is why the names do not read alphabetically. **Needs** is what the tile wants before it shows anything useful.

id	Widget	Category	What it shows	Needs	Cost
`agent_desk`	Agent Desk	Tools	One agent desk at a glance — armed state, runs today, next fire, last result	Desk	Free
`agent_errors`	Agent Errors	Tools	Recent agent execution failures	—	Free
`agent_insight`	AI Insight	Tools	Ask an Agent Studio persona a standing question — streamed answer	Persona, prompt, interval	**Metered**
`candle_chart`	Candle Chart	Research	Candlestick chart for a single ticker	Symbol	Free
`commodities`	Commodities	Markets	Commodity board — energy, metals, ags, inflation basket or freight	Board	Free
`crypto`	Crypto Markets	Markets	Crypto board — majors, L1s, DeFi, high-beta alts or crypto vs macro	Board	Free
`crypto_ticker`	Crypto Ticker	Markets	Live Kraken / HyperLiquid ticker strip	Pair list	Free
`econ_calendar`	Economic Calendar	Research	Macro releases ranked by importance	Filters	Free
`forex`	Forex Pairs	Markets	FX board — majors, crosses, Asia, EM, commodity FX or dollar drivers	Board	Free
`geopolitics_events`	Geopolitics Events	Geopolitics	Live conflict / political events	—	Free
`holdings`	Holdings	Portfolio	Long-term broker holdings — avg cost, LTP, P&L %	Broker account	Free
`indices`	Market Indices	Markets	Index board — global, US, Europe, Asia, India, rates, vol or futures	Board	Free
`infra_map`	Infrastructure Map	Geopolitics	Pipelines, cables, ports or chokepoints on a world map	Layer, status, viewport	Free
`margin_usage`	Margin Usage	Trading	Broker account funds — available, used, total, usage %	Broker account	Free
`maritime_vessels`	Maritime Vessels	Geopolitics	Live vessel positions	IMO list	Free
`news`	News Feed	Research	Headlines with impact, sentiment and tickers	Filters	Free
`news_category`	News — Category	Research	Category-filtered news headlines	Category	Free
`notes`	Notes	Tools	Recent / favorite financial notes	—	Free
`open_positions`	Open Positions	Trading	Live open positions for a broker account	Broker account	Free
`performance`	Performance	Portfolio	Portfolio P&L — today, week, month, YTD	—	Free
`polymarket_prices`	Polymarket	Markets	Live prediction-market prices	Asset list	Free
`portfolio_summary`	Portfolio Summary	Portfolio	Holdings overview with allocation breakdown	—	Free
`quick_trade`	Quick Trade	Trading	Fast order entry for crypto and equities	Broker, default symbol	Free
`quote_strip`	Quote Strip	Markets	Configurable live quote list	Symbols	Free

All 42 widgets, one table (continued)

id	Widget	Category	What it shows	Needs	Cost
`recent_files`	Recent Files	Tools	Recently saved files — exports, reports, notebooks	—	Free
`risk_metrics`	Risk Metrics	Portfolio	Volatility, beta, drawdown, Sharpe ratio	—	Free
`rss_feed`	RSS Feed	Research	A configured RSS/XML/JSON/CSV feed	Feed (from the RSS Feeds tab)	Free
`screener`	Stock Screener	Research	Rank a basket by move, volume or price	Basket, sort	Free
`sector_heatmap`	Sector Heatmap	Markets	S&P 500 sector performance heatmap	—	Free
`sentiment`	Market Sentiment	Markets	Fear & greed, bull/bear indicators	—	Free
`sparklines`	Sparklines	Markets	Configurable sparkline strip	Symbols	Free
`stock_quote`	Stock Quote	Research	Single stock detail — price, volume, chart	Symbol	Free
`today_pnl`	Today P&L	Trading	Aggregate broker account P&L	Broker account	Free
`top_movers`	Top Movers	Markets	Biggest gainers and losers	Universe	Free
`trade_chord`	Trade Flows	Geopolitics	Chord diagram of a country's top trading partners	Country, year, rank, partners	**Metered**
`trade_tape`	Trade Tape	Markets	Live trade prints for a crypto pair	Exchange, pair	Free
`trade_treemap`	Trade Treemap	Geopolitics	Trading partners as a treemap, grouped by continent	Country, year, rank, partners	**Metered**
`video_player`	Live TV / Streams	Tools	Financial TV — major networks and custom streams	—	Free
`watchlist`	Watchlist	Portfolio	Live prices for your own symbol list	Symbols, in the tile	Free
`web_scraper`	Web Scraper	Tools	Scrape tables from any URL	URL	Free
`working_orders`	Working Orders	Trading	Pending/working orders — click ✕ to cancel	Broker account	Free
`world_choropleth`	World Map	Geopolitics	One IMF indicator across every country	Indicator, colour mode	Free

Counts, for reconciliation

- **Markets** 12 · **Research** 7 · **Portfolio** 5
- **Trading** 5 · **Tools** 7 · **Geopolitics** 6
- **42** widgets in the registry, and 42 in the picker's header
- **3** of them ever charge; **39** never do
- **5** require a connected broker
- **9** ready-made templates, placing 13 to 16 tiles each

17 Honest notes

Things that are true of the shipped build and worth knowing before they surprise you.

What you see	What is going on
A tile is called **SPARKLINES** but its settings dialog is titled *Configure — Trend Strip*	Two names for one widget: the registry name and the class's own. Nothing is wrong; the settings do belong to that tile.
✕ on a tile removes it with no confirmation	There is no undo and no bin. Its configuration goes with it.
✕ inside Working Orders cancels a live order	Two different ✕ on the same tile, doing very different things — one on the row, one in the title bar.
A freshly added tile is blank	Most likely it needs setting up — see the **Needs** column in §16 — or it is a broker tile with no account connected.
Ten minutes is the shortest auto-refresh anywhere	Deliberate. Streaming tiles are the answer for anything faster.
The in-app **Docs** page for the Dashboard does not mention the **AUTO** button	The bundled help text predates the control. The button is real; the help is behind.
COMPACT and **PULSE** forget themselves between sessions	They are view toggles, not saved settings. Tiles, sizes and tile settings do persist.

This edition documents 42 widgets; the July edition documented 36. Six tiles had shipped with no entry anywhere — Agent Desk, AI Insight, Trade Flows, Trade Treemap, World Map and Infrastructure Map. If you are holding a printed copy that says *36 widgets* on its cover, it is the older one.

Where the four Global Economy tiles came from. They are the Global Economy screen's four views — chord, treemap, choropleth and infrastructure map — packaged as tiles. Their full behaviour, including what the trade corpus contains and how the indicator list is built, is in the *Global Economy* guide; this guide covers only what changes when they are pinned to a board.

18 Quick reference — how do I…

I want to…	Do this
Add a tile	**+ ADD** → click a card → **ADD WIDGET**
Find a tile without scrolling	Type in the picker's search box — it matches names, descriptions and categories
See only one category	The category buttons under the search box; press **ALL** to go back
Watch two brokers at once	Add the tile twice and bind each copy to one account with its gear
Chart four symbols	Four Candle Chart tiles, one symbol each — settings are per tile
Move a tile	Drag its title bar; the others make room
Make a tile bigger	Drag its bottom-right corner; it snaps to the grid
Fit more on screen	**COMPACT**, and **PULSE** to reclaim the right-hand rail
Reload one tile	Its ↻ icon
Reload everything	**REFRESH** on the toolbar
Change how often the board updates	**AUTO** on the toolbar — nine choices, 10 min to 24 h
Give one tile its own cadence	Its interval pill (`10m`) → pick an interval instead of *Follow dashboard*
Change a tile's symbols, account or feed	Its gear icon; **OK** to apply
Remove a tile	The ✕ in its title bar — no confirmation, no undo
Start over from a professional layout	**RESET** → a template card's **APPLY**
Look at the templates without risk	**RESET** → read → **CANCEL**
Save right now	**SAVE** — though the board also saves itself after any change
Get the same board on another machine	Nothing — dashboard layouts follow your account
Add a news feed of my own	Add and map it in the **RSS Feeds** tab, then add the RSS Feed tile
Pull a table off a web page	The **Web Scraper** tile — give it the URL
Watch one agent desk	The **Agent Desk** tile — read-only, free
Keep a standing question answered	**AI Insight** — and read §15 before arming its timer
Avoid spending credits from the board	Leave AI Insight on **Manual only** and give the two trade tiles long intervals
Know what a tile will cost before adding it	§16 — the **Cost** column. Three tiles say *Metered*; the other thirty-nine say *Free*

> **The short version.** Press **+ ADD**, pick a card, press **ADD WIDGET**. Drag to move, corner-drag to resize, gear to configure, ✕ to remove. Everything saves itself and follows your account. Only **AI Insight** — and the two trade tiles — ever cost anything.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The In-App Documentation

A complete walkthrough of the Docs tab — the terminal's own manual: thirty-six illustrated topics across nine categories, always the same version as the build you are running.

`>_ 36 topics · 9 categories · ~200 screenshots · offline`

Covers · **the sidebar tree** · **all 36 topics** · **what a topic page contains**
how pages load · what is and is not covered · using it alongside these PDF guides

Demonstrated live on the Fincept Terminal Docs tab.

Contents

8 SECTIONS

GETTING ORIENTED

01 What the Docs tab is 3

02 Anatomy of the tab 4

THE MANUAL ITSELF

03 All thirty-six topics 5

04 What a topic page contains 6

05 How pages load and re-style 8

COVERAGE AND WORKFLOW

06 What is covered — and what is not 9

07 Using it well 10

REFERENCE

08 Quick reference and honest notes 11

About this guide. Every screenshot is a live capture of the Docs tab in the shipping build. Topic and category counts in this guide are the ones the tab itself reports, because it counts its own pages at runtime.

01 What the Docs tab is

ORIENTATION · 01

Docs is **the terminal's own manual, inside the terminal**. A sidebar of nine categories opens thirty-six illustrated topics — one per major screen, plus three orientation pages — each written as a walkthrough with real screenshots of the software you are running and reference tables for its controls.

Because it ships inside the build, it cannot drift out of step with a release, and it needs no network: the whole manual, including roughly two hundred screenshots, is on disk beside the application. Open it on a plane, behind a firewall, or on a locked-down desk and it reads exactly the same.

Fig 1 — The command bar names the tab, states the build version, and counts its own contents: **36 TOPICS I 9 CATEGORIES**. Both numbers are counted at runtime, so they are always true of the build in front of you.

Where to find it

Route	How
Navigate menu	**Navigate ‣ Docs** — below the four category submenus, next to **About**
Command bar	**Not available.** Docs has no command-bar alias — use the Navigate menu

At a glance, in the Docs tab you can…

- Browse thirty-six topics grouped into nine categories
- Read a screen-by-screen walkthrough of any major tab
- See real screenshots of the build you are running
- Look up controls in reference tables
- Learn the global keyboard shortcuts
- Follow a first-steps orientation as a new user
- Read skill-level guidance from beginner to pro
- Do all of it offline, with no key and no credits

> **Docs is free and offline.** Reading the manual makes no network request and spends no Fincept credits — unlike asking the AI assistant, which is metered. When the answer is likely to be in the manual, this is the cheaper place to look.

02 Anatomy of the tab

Three regions: a command bar, the topic tree, and the page.

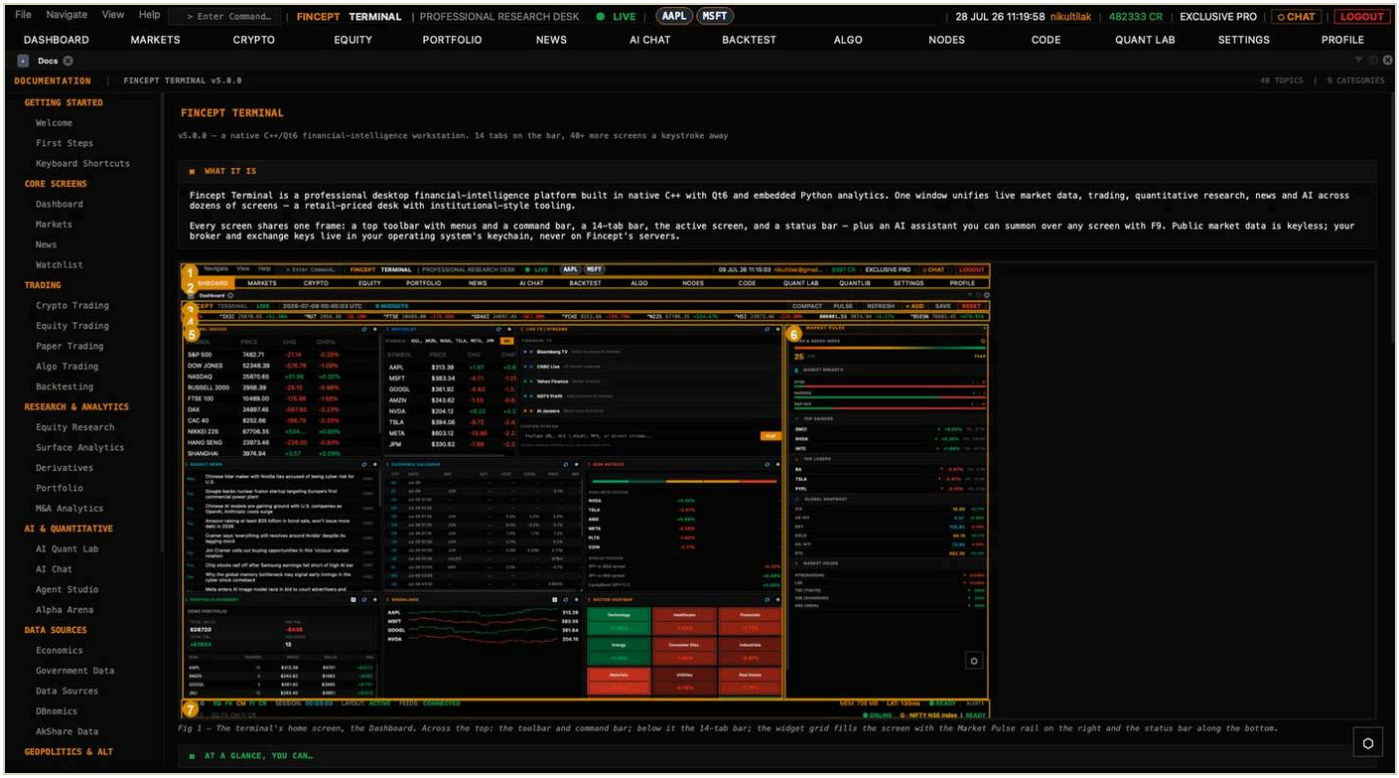


Fig 2 — The Docs tab on open, showing the Welcome topic. The tree is on the left; the page fills the rest of the width.

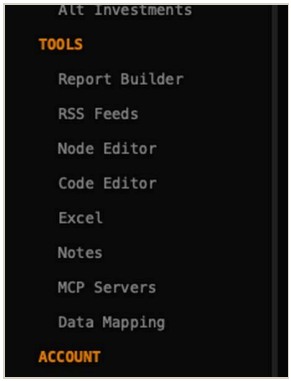


Fig 3 — Scrolling the tree reveals the lower categories: **TOOLS** carries eight topics, including Excel and Notes.

#	Region	What it is for
1	Command bar	**DOCUMENTATION**, the build version, and a live count of topics and categories.
2	Topic tree	Nine amber category headings, each expanded, with their topics beneath. Category rows are labels — click a topic to read it.
3	Page area	The topic: title, subtitle, then panels, figures, tables and tip boxes. Scrolls independently of the tree.

The divider between tree and page can be dragged — widen the tree while hunting for a topic, then give the width back for reading. The tab always opens on **Welcome**; it does not restore the topic you were last reading.

03 All thirty-six topics

The whole manual on one page, in tree order.

Category	Topics
GETTING STARTED `(3)`	Welcome · First Steps · Keyboard Shortcuts
CORE SCREENS `(4)`	Dashboard · Markets · News · Watchlist
TRADING `(5)`	Crypto Trading · Equity Trading · Paper Trading · Algo Trading · Backtesting
RESEARCH & ANALYTICS `(5)`	Equity Research · Surface Analytics · Derivatives · Portfolio · M&A Analytics
AI & QUANTITATIVE `(4)`	AI Quant Lab · AI Chat · Agent Studio · Alpha Arena
DATA SOURCES `(1)`	Data Sources
GEOPOLITICS & ALT `(4)`	Geopolitics · Maritime · Prediction Markets · Alt Investments
TOOLS `(8)`	Report Builder · RSS Feeds · Node Editor · Code Editor · Excel · Notes · MCP Servers · Data Mapping
ACCOUNT `(2)`	Settings · Profile

The three orientation topics

Topic	What it gives you
`Welcome`	What the terminal is, an annotated picture of the Dashboard, and an *at a glance* summary of what the product does
`First Steps`	The path a new user should take through the software on day one
`Keyboard Shortcuts`	The real global keys, the command bar, the Navigate menu, and what each top-level menu holds

> **Start with Keyboard Shortcuts even if you are experienced.** It is the page that distinguishes the terminal's five genuine global keys from the bracketed mnemonics screens print on their own buttons — a distinction that saves a lot of pressing keys that do nothing.

04 What a topic page contains

Every screen topic follows the same shape, so once you have read one you know how to read the other thirty-one.

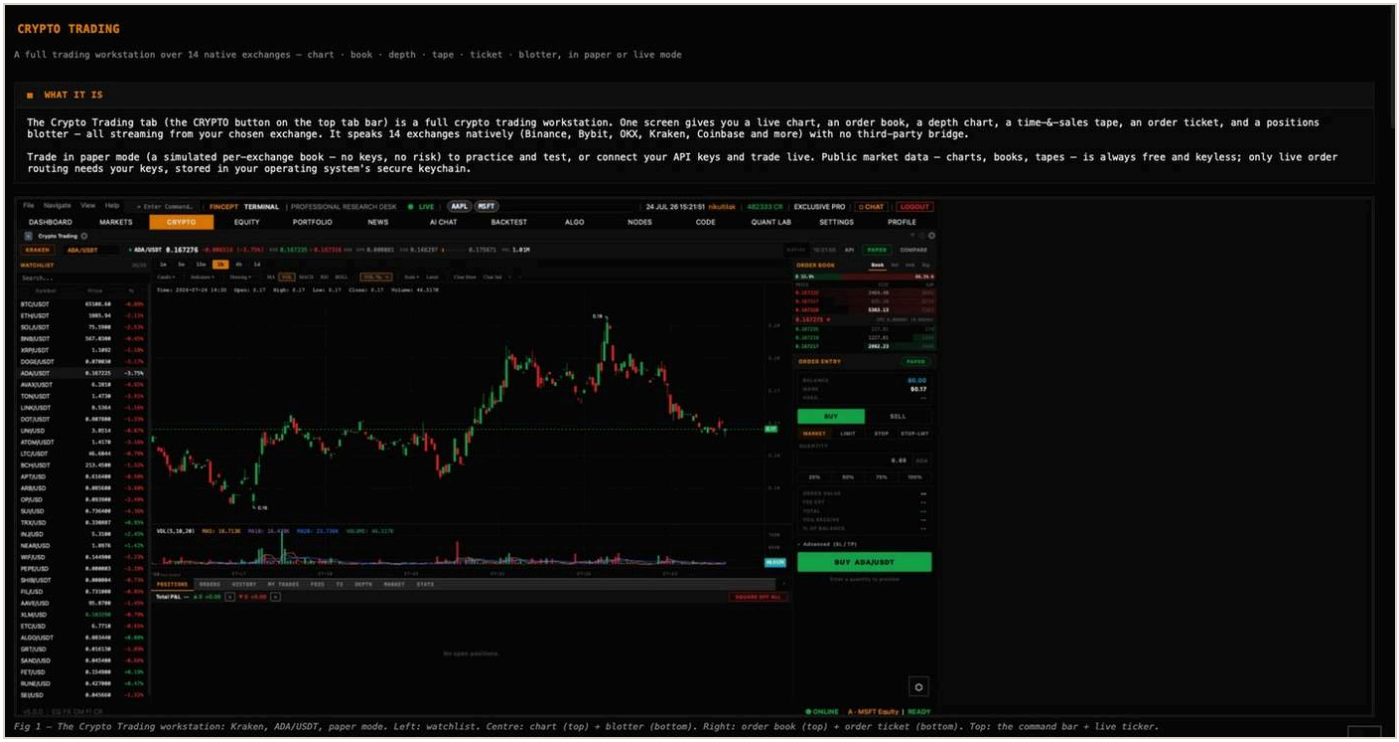


Fig 4 — The Crypto Trading topic: amber title, one-line subtitle, a **WHAT IT IS** panel, then a full-width annotated screenshot with a numbered caption.

Element	What it is
`Title + subtitle`	The screen's name, and a one-line statement of what it does
`Panels`	Bordered blocks with a ■ header — **WHAT IT IS**, **AT A GLANCE, YOU CAN…**, and topic-specific sections
`Figures`	Real screenshots with a *Fig N* — caption describing what to look at. They rescale to the page width as you resize the window or drag the divider
`Reference tables`	Amber-headed tables listing controls, options or values with what each one does
`Tip boxes`	Short coloured call-outs for the thing that catches people out
`Skill levels`	On some topics, a four-row panel — **BEGINNER**, **INTERMEDIATE**, **ADVANCED**, **PRO** — suggesting how far to take the screen

GLOBAL SHORTCUTS

Key	Action
F11	Toggle fullscreen — use the whole monitor
F10	Toggle focus mode — hide the tab bar and status bar for maximum content area
F9	Toggle AI chat mode — the agentic assistant over the current screen
F5	Refresh the current screen's data
Ctrl+P	Take a screenshot — a save dialog lets you choose the file

THE COMMAND BAR

■ TYPE TO JUMP

The command bar sits at the top-left of the toolbar. Type a screen name — markets, economics, watchlist, crypto — and press Enter to jump straight there, without hunting through menus. It matches on names and common aliases, so partial words usually find the right screen.

THE NAVIGATE MENU

Category	Example screens
Markets & Data	Economics, GOVT Data, DBnomics, Screener and more
Trading & Portfolio	Equity Trading, Derivatives, Watchlist
Research & Intelligence	Equity Research, M&A Analytics, Alt. Investments, Geopolitics, Maritime, Surface Analytics
Tools	Agent Config, MCP Servers, Data Mapping, Data Sources, Report Builder, Excel, Trade Viz, File Manager, Notes

Fig 5 — Reference tables from the Keyboard Shortcuts topic: the five global keys, then what each Navigate category holds.

> **Figures never upscale.** A screenshot is drawn at its natural size or smaller, never stretched, so a narrow window shows a smaller sharp image rather than a large blurred one. Widen the page area to read fine detail in a figure.

05 How pages load and re-style

Pages are built when you first open them

The manual holds around two hundred screenshots. Rather than decode all of them the moment the tab opens, each topic is **built on its first visit** and then kept for the rest of the session.

Behaviour	What you see
`First visit to a topic`	A brief pause while the page and its screenshots are prepared — longest on the screenshot-heavy topics
`Every later visit`	Instant. The built page is cached for the session and simply re-shown
`Opening the tab`	Instant, and always on **Welcome** — no other page is built until you ask for it
`Closing and reopening the tab`	Pages are built again on first visit, because the cache lives with the open screen

> **Why it is designed this way.** Building all thirty-six pages up front would decode every screenshot in one go and freeze the window each time you opened the manual. Loading each topic on demand keeps the tab instant to open and spreads the cost over the pages you actually read.

It follows your appearance settings

The manual is not a static document — it is built from the same theme tokens as the rest of the terminal. Change the font family, size or density in **Settings ‣ Appearance** and the tree and the page are rebuilt immediately in the new typeface, keeping you on the topic you were reading. The same happens if you switch the interface language.

> **A practical consequence.** If the manual feels cramped on a large monitor, raise the interface font in Settings ‣ Appearance — the documentation reflows with everything else, and figures rescale to match.

> **Missing figures fail quietly.** If a screenshot cannot be found on disk — an incomplete installation, for instance — the page simply omits that figure and renders the rest of the text. A topic with no pictures where you expected some is a sign the installation's resources folder is incomplete, not that the topic is unwritten.

06 What is covered — and what is not

COVERAGE · 06

Thirty-six topics cover thirty-two screens, plus the three orientation pages and one on paper trading as a discipline rather than a screen. Seven screens have no topic in this build:

Screen not in the manual	Where to read about it instead
`File Manager`	**Fincept-File-Manager-Guide.pdf** — this guide's companion volume
`Global Economy`	**Fincept-Global-Economy-Guide.pdf**
`Global Data`	**Fincept-Global-Data-Guide.pdf**
`Docs`	This guide
`About`	**Fincept-About-Guide.pdf**
`Relationship Map`	**Fincept-Relationship-Map-Guide.pdf**
`F&O`	No guide yet; it opens from the Equity Trading tab's F&O button

Other limits worth knowing

- **There is no search.** The tree is the only way in, so scan the nine category headings rather than looking for a search field. With thirty-six topics in nine groups, the heading you want is usually one glance away.
- **Topics cannot be bookmarked** and the tab does not remember where you were — it opens on Welcome every time.
- **Pages are read-only** and cannot be printed or exported from the tab. For a document you can send to a colleague, use the PDF guide set.
- **Topic names are not always tab names.** Agent Studio is the Agent Config tab, Prediction Markets is the Polymarket screen, and Paper Trading is a mode of the trading tabs rather than a screen of its own.

> **How the two sets fit together.** The in-app manual is the quick answer while you work: what a control does, where a panel is, what a screen is for. The PDF guides go deeper on one tab at a time — every control, the data behind it, the agent tools, and the honest caveats — and can be shared with someone who does not have the terminal in front of them.

07 Using it well

WORKFLOW · 07

Three habits that pay off

- **Read the topic before you drive the screen.** Most Fincept tabs have one non-obvious convention — a symbol suffix, a run button that must be pressed, a paper-versus-live toggle. The topic's **WHAT IT IS** panel and its first figure usually surface it in under a minute.
- **Keep Docs open in a second window.** Because the manual is offline and cheap, running it in its own terminal window on a second monitor costs nothing and saves flicking away from your work. **File ▸ New Window**, then Navigate ▸ Docs.
- **Use it as the free first answer.** The AI assistant is metered; the manual is not. Look here first for "what does this button do", and save the assistant for questions about *your* data.

A reading order for a new desk

Step	Topic	Why
1	Welcome	What the product is, and the shape of the Dashboard
2	Keyboard Shortcuts	The five real global keys and how the command bar works
3	First Steps	A sensible day-one path through the software
4	Dashboard, Markets, Watchlist	The screens you will keep open all day
5	Your desk's screen	Equity Research, Portfolio, Crypto Trading, Derivatives — whichever you actually run
6	AI Chat and Agent Studio	Where the metered features are, and what they cost

> **For the tabs the manual does not cover** — File Manager, Global Economy, Global Data, Docs and About — the companion PDF guides do, and they are written to the same structure: what it is, anatomy, region by region, then a quick reference and an honest list of what does not work as advertised.

08 Quick reference and honest notes

How do I…?

Question	Answer
`…open the manual?`	**Navigate ▸ Docs**. There is no command-bar alias
`…find a topic?`	Scan the nine amber category headings in the tree — there is no search
`…read about a tab I am using?`	Look under the category it belongs to: TRADING, RESEARCH & ANALYTICS, TOOLS, and so on
`…learn the keyboard shortcuts?`	GETTING STARTED ▸ **Keyboard Shortcuts**
`…make a figure bigger?`	Drag the divider right, or widen the window — figures rescale to the page
`…make the text bigger?`	Settings ▸ Appearance — the manual rebuilds in the new font at once
`…get back to where I was?`	Click the topic again; the tab always opens on Welcome
`…read about File Manager, Global Economy, Global Data, Docs or About?`	The companion PDF guides — they are not in the in-app manual
`…print a topic?`	Not from the tab. Use the PDF guide set, or the terminal screenshot key (**Ctrl+P**)

> The counts in this guide — thirty-six topics, nine categories — are the ones the tab computes and displays for itself, so they stay correct as the manual grows. Everything else here was read from the shipping source and confirmed on screen.

Honest notes on this build

- **No search**, no bookmarks, and no history — the tree is the only navigation.
- **The tab always opens on Welcome**; it does not restore the last topic you read.
- **Seven screens have no topic:** File Manager, Global Economy, Global Data, Docs, About, Relationship Map and F&O.
- **The manual shrank from forty topics to thirty-six in v5.0.1.** The DATA SOURCES category lost four pages — Economics, Government Data, DBnomics and AkShare Data — when those five screens were retired into one Global Data explorer. The replacement screen has no topic of its own, so a whole area of the product went from four pages to none.
- **The Welcome page still describes v5.0.0** and says the bar carries fourteen tabs; the shipping build is v5.0.1 with fifteen. The header beside it reads the real version at runtime, so the tab disagrees with itself on its own first page.
- **The first visit to a screenshot-heavy topic pauses briefly** while the page is built; later visits are instant.

- **Category rows are not clickable** — they are headings, and only the topics under them open a page.
- **Some topic names differ from the tab names** (Agent Studio, Prediction Markets, Government Data), which can make a topic hard to find by tab name alone.
- **Missing screenshots are omitted silently** rather than reported.
- **No export or print** from the tab, and no way to copy a whole topic as text.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

About & Diagnostics

A complete walkthrough of the About tab — the build you are running, the licence it is running under, where its crash reports go, and how to reach the people who maintain it.

`>_ version · updates · release notes · licence · crash dumps`

Covers · **version information** · **Check for Updates** · **the in-app release notes** · **the licence** crash dumps · resources and contacts · the sibling legal pages

Demonstrated live on the Fincept Terminal About tab.

Contents

10 SECTIONS

GETTING ORIENTED

01 What the About tab is 3

02 Anatomy of the tab 4

THE FIVE PANELS

03 Version information 5

04 Check for Updates 6

05 How updates are delivered 7

06 Release notes, in the app 8

07 Licence 9

08 Diagnostics and crash dumps 10

09 Resources and contacts 11

REFERENCE

10 Quick reference and honest notes 12

> **About this guide.** Every screenshot is a live capture of the About tab in the shipping build, including the real result of pressing **Check for Updates** on an up-to-date installation.

01 What the About tab is

ORIENTATION · 01

About is the terminal's **identity card and support desk**. It states exactly which build you are running, offers a one-press update check, sets out the licence the software is used under, tells you where crash reports are written, and lists the addresses to write to.

It is a small page, and it earns its place on the days that matter: raising a support ticket, answering a compliance question about licensing, or confirming that the version on your desk is the version a colleague is describing.

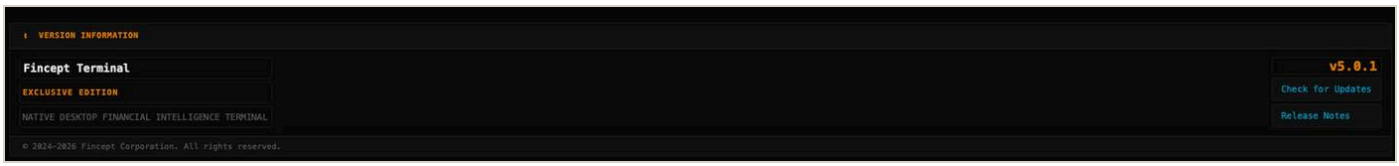


Fig 1 — The version panel: product name, the **EXCLUSIVE EDITION** badge, the build version in amber, and the update check.

Where to find it

Route	How
Help menu	**Help ▸ About Fincept** — the conventional route
Navigate menu	**Navigate ▸ About**, at the foot of the menu next to **Docs**
Command bar	Type `about`, `info` or `version`

At a glance, on the About tab you can…

- Read the exact build version you are running
- Confirm this is the Exclusive (subscriber) edition
- Check for a newer release in one press
- Read the licence terms in four lines
- Find the crash-dump folder path
- Open that folder to attach a report
- Reach the official website
- See the support, commercial, security and legal addresses

> **Why the edition badge matters.** Fincept ships two terminals — a public open-source build and this private subscriber edition. The **EXCLUSIVE EDITION** badge, and the proprietary licence in section 07, are how you confirm at a glance which one is installed.

02 Anatomy of the tab

Five stacked panels, each with an amber header. The page scrolls if the window is short; at a normal desk resolution all five are visible at once.

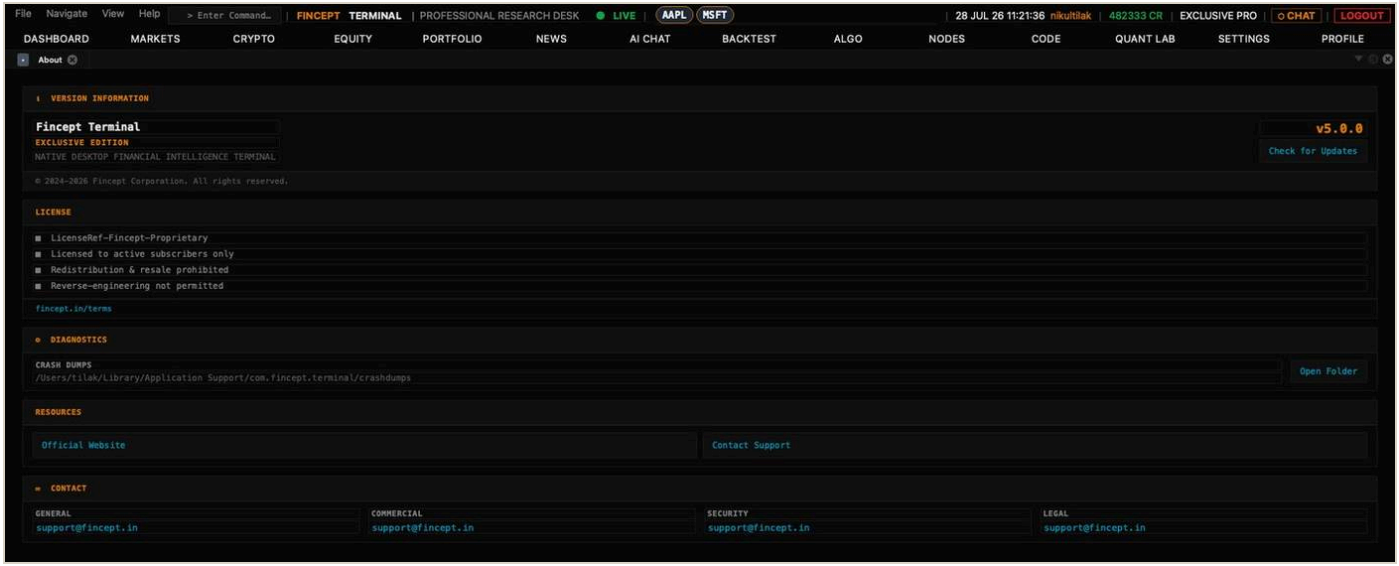


Fig 2 — The whole About tab: version, licence, diagnostics, resources and contact.

#	Panel	What it holds
1	VERSION INFORMATION	Product name, edition badge, one-line description, the build version, **Check for Updates**, **Release Notes**, and the copyright line. Sections 03–06.
2	LICENCE	The licence identifier and its three practical terms, plus the terms-of-service address. Section 07.
3	DIAGNOSTICS	The crash-dump folder path and an **Open Folder** button. Section 08.
4	RESOURCES	**Official Website** and **Contact Support**. Section 09.
5	CONTACT	Four addressed routes — general, commercial, security and legal. Section 09.

> There are only **three interactive controls** on the whole tab: **Check for Updates**, **Open Folder**, and the two buttons in RESOURCES. Everything else is text — including the email addresses and the terms address, which are shown for reading and copying rather than clicking.

03 Version information

VERSION · 03

Everything needed to identify the installation, in one panel.

Element	What it tells you
`Fincept Terminal`	The product name
`EXCLUSIVE EDITION`	This is the private, subscriber-only build — not the public open-source terminal
`NATIVE DESKTOP FINANCIAL INTELLIGENCE TERMINAL`	The one-line description of what the product is
`v5.0.1`	The build version, in amber — whatever the running application reports. This is the number to quote in a support ticket
`Check for Updates`	A user-initiated check against Fincept's release service. Sections 04–05
`Release Notes`	Opens the changelog that ships inside the application (§06)
`© 2024–2026 Fincept Corporation. All rights reserved.`	The copyright line, in the panel footer

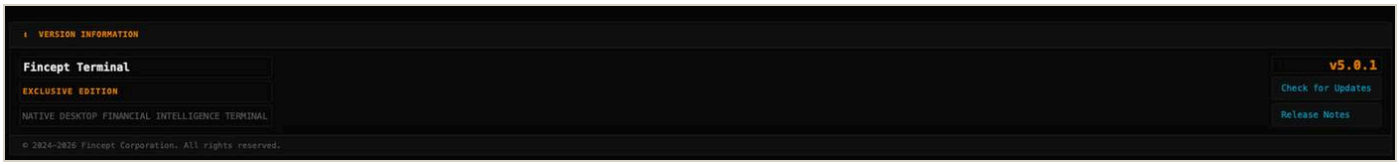


Fig 3 — The version panel on a current installation. Two buttons stack at the right-hand end: **Check for Updates** above, **Release Notes** below.

> **The version is read from the running application**, not typed into the page, so it cannot drift. The same number appears in the terminal's status bar at the bottom-left of the window and in the Docs tab's command bar — if all three agree, you are looking at one installation.

> **What to quote when raising a ticket.** The version from this panel, your operating system and processor architecture, and — if the terminal has crashed — the newest file from the crash-dump folder in section 08. That is enough for support to identify the exact build and its platform package.

04 Check for Updates

Press the button and the terminal asks Fincept's release service what this build should upgrade to, then compares that version with the one you are running. The button reads *Checking…* and is disabled while the request is in flight. Section 05 covers how the release itself is delivered.

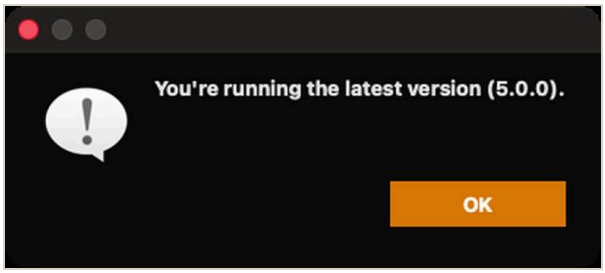


Fig 4 — The result on an up-to-date installation: *You're running the latest version (5.0.0).*

Outcome	What you see
`Already current`	*You're running the latest version (5.0.0).* — one **OK** button
`A newer release exists`	An **Update Available** dialog with the version, size and changelog, offering **Download & Install**, **View Release Notes** (when the release provides a link), **Remind Me Later** and **Skip This Version**
`A required release exists`	**Update Required** — the same dialog minus the ways to stay behind: **Download & Install** or **Quit**
`Development build`	*This is a development build … Auto-update is disabled.* — a non-release version string cannot be compared
`No network, or the service is unreachable`	An error message. Nothing is downloaded and nothing changes

05 How updates are delivered

DELIVERY · 05

Where the build comes from, and what stops the terminal from running something it should not.

Aspect	Detail
Who is asked	Fincept's own release service, through the terminal's authenticated edge — it carries your session, so this is a private lookup rather than a public file
Subscription needed?	**No.** Updates are an account-level service: a lapsed subscriber still receives builds, including security fixes
How your build is matched	By platform and processor architecture (for example `macos-arm64`, `windows-x64`) and by release **channel** — `stable` by default, with `beta` available
How versions compare	Numerically by major, minor and patch, so 5.10.0 beats 5.9.0
Where the installer comes from	Fincept's own download host. The URL must be HTTPS and on an allow-list before anything is fetched, and the server applies the same list
Integrity	Every installer carries a **sha256** that is verified as the file streams to disk. A missing or mismatched checksum aborts the update and deletes the download
Also available from	**Help ▸ Check for Updates** — the same action, from the menu
Automatic checks	One quiet check per run of the terminal, after login. It stays silent unless a release is found, respects a version you have skipped, and can be switched off entirely for managed installs — the button here keeps working either way

Download & Install fetches a verified installer, then asks to quit. Nothing is replaced behind your back: the download shows progress and can be cancelled, the checksum is checked before the installer is allowed to run, and you choose **Quit & Install** or **Later**. Close unsaved work first — remember the Excel scratch pad is session-only. To read what changed without updating, use **View Release Notes**.

A required update is deliberately hard to dismiss. When a release is marked mandatory — or your build falls below the minimum the service still supports — the dialog offers only **Download & Install** or **Quit**, and **Skip This Version** is withheld. That is the mechanism behind a forced security rollout.

06 Release notes, in the app

Release Notes, under **Check for Updates** in the version panel, opens a screen rather than a web page. The changelog is compiled into the application as a resource, so the build you are running always carries its own history — **no network call, and it works offline**.

Fig 5 — The Release Notes screen. Versions down the left, the selected release on the right, grouped and colour-coded by the kind of change.

Element	What it is
`← Back to About`	Returns to the tab. This screen replaces the About panel rather than opening beside it
`RELEASE NOTES`	Header, with *Fincept Terminal · N version(s)* underneath
`VERSIONS · N`	The rail: every released version, newest first, with its date
`● NOW`	Marks the version you are running
`▲ NEW`	Marks a release newer than yours — you have not updated yet
`LATEST`	Badge on the newest release the build knows about
`INSTALLED`	Badge on the one you are running. Both appear together when you are current
`Group headings`	**ADDED**, **CHANGED**, **FIXED**, **REMOVED**, **SECURITY** — coloured by the leading word, and often suffixed with the area (*ADDED — SPREADSHEETS*)
`Entries`	One line per change, written for customers rather than engineers

> **This is not the same as *View Release Notes* in the update dialog.** That button opens the *upstream* release's notes for a version you have not installed, and only when the release provides a link. This screen shows the history that is already inside your build. One is about the future, the other about the past.

> **Unreleased work is deliberately not shown.** The changelog file accumulates entries under an *Unreleased* heading until a version is tagged, and this screen skips that section entirely — so a feature you can see in the product may legitimately have no entry here yet. The count in the header (*N version(s)*) is releases, not changes.

07 Licence

The licence panel is short by design: one identifier and three plain-language terms.

Fig 6 — The licence panel, with the terms-of-service address in the footer.

Line	What it means in practice
`LicenseRef-Fincept-Proprietary`	The licence identifier. This is proprietary, closed software — the public Fincept terminal is a separate, differently licensed product
`Licensed to active subscribers only`	The right to run this build follows an active subscription. It is per-subscriber, not per-machine-forever
`Redistribution & resale prohibited`	You may not pass the installer, the application or a copy of it to anyone outside your licence
`Reverse-engineering not permitted`	Decompiling, disassembling or extracting internals is outside the licence
`fincept.in/terms`	Where the full terms of service live. The panel shows the address for you to type or copy — it is not a clickable link

> **For a compliance question, this panel plus the version panel is usually the whole answer:** which product, which edition, which build, under which licence. Screenshot both — they sit adjacent on the page — and the record is unambiguous.

> **The full terms are online rather than in the application**, so they can be updated without a release. The Help menu also carries **Terms of Service**, **Privacy Policy** and **Trademarks** as their own screens inside the terminal — see section 07.

08 Diagnostics and crash dumps

When the terminal crashes it writes a minidump — a small snapshot of what the application was doing at the moment it stopped. This panel exists so you never have to go hunting for those files.

Fig 7 — The diagnostics panel: the crash-dump folder path, and the button that opens it.

Element	Behaviour
`CRASH DUMPS path`	The full path to the folder, inside your terminal profile directory. The text is selectable, so you can copy it into a ticket
`Open Folder`	Opens the folder in your operating system's file browser. If it does not exist yet — because the terminal has never crashed — it is created first, so the button never fails

What to do after a crash

1. Reopen the terminal and come to this tab.
2. Press **Open Folder** and take the **newest** file.
3. Send it to support with the version from section 03 and a note of what you were doing.

> **An empty folder is good news.** If **Open Folder** shows nothing, this installation has not written a crash report — the folder was just created for you. A crash that produced no dump is worth reporting anyway, with the version and the steps that led to it.

> **Where the folder sits.** Crash dumps live beside the rest of your profile data — the same profile directory that holds the File Manager's `files` folder. One profile, one set of diagnostics, and a second profile on the same machine keeps its own.

> **Minidumps can contain fragments of in-memory data.** Treat a crash dump like a diagnostic artefact from your desk: send it through your normal support channel rather than posting it somewhere public.

09 Resources and contacts

Two buttons, four addresses, and a set of sibling pages in the Help menu.

Fig 8 — The resources panel: two buttons that open in your browser or mail client.

Button	What it opens
`Official Website`	`fincept.in` in your default browser
`Contact Support`	A new message to `support@fincept.in` in your default mail client

Fig 9 — The contact panel: four routes, each labelled by the kind of enquiry.

Route	Use it for
`GENERAL`	Product questions, usage help, anything that does not fit below
`COMMERCIAL`	Licensing, subscriptions, invoicing, adding seats
`SECURITY`	Vulnerability reports and security questions
`LEGAL`	Contractual, compliance and legal correspondence

> **All four routes currently reach the same mailbox** — `support@fincept.in`. The labels are still worth using: put the category in your subject line and the message lands with the right person faster. Note that these addresses are shown as text, not links; use **Contact Support** in RESOURCES to open a message directly.

The sibling pages in the Help menu

Help menu entry	What it opens
`About Fincept`	This tab
`Help Center`	The in-terminal help screen
`Contact Us`	A dedicated contact screen
`Terms of Service`	The terms, inside the terminal
`Privacy Policy`	The privacy policy screen
`Trademarks`	Trademark notices
`Check for Updates`	The same update check as section 04

10 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	**Help ‣ About Fincept**, Navigate ‣ About, or type `about` in the command bar
…find my exact version?	The amber number in the version panel — also in the status bar and the Docs command bar
…confirm this is the subscriber build?	The **EXCLUSIVE EDITION** badge, and the proprietary licence in section 06
…check for a new release?	**Check for Updates** here, or **Help ‣ Check for Updates**
…read the release notes without updating?	**View Release Notes** in the update dialog — it installs nothing
…postpone an update?	**Remind Me Later**, or **Skip This Version** to stop being asked — unless the release is required
…find my crash reports?	**Open Folder** in the diagnostics panel; take the newest file
…email support?	**Contact Support** in RESOURCES opens a message to `support@fincept.in`
…read the full terms?	`fincept.in/terms`, or **Help ‣ Terms of Service**
…report a security issue?	Use the SECURITY route with a clear subject line

Honest notes on this build

- **The email addresses and the terms address are plain text**, not clickable links. Use **Contact Support**, or copy the address.
- **All four contact routes go to the same mailbox** — the categories are a convention, not separate inboxes.
- **The update check is user-initiated for the visible dialog.** One quiet check also runs per run of the terminal and stays silent unless a release is found; it honours a skipped version and can be disabled for managed installs.
- **Releases come from Fincept's own service, not a public file** — authenticated, channel-aware (`stable` / `beta`), host-allow-listed and sha256-verified.
- **Development builds cannot self-update**, and say so plainly when checked.
- **Nothing on this tab is configurable** — it is a read-only page with three actions, and it keeps no state between visits.
- **No agent tools** — the assistant cannot read your version or crash folder from here — but the page does follow **Settings ‣ Appearance** and rebuilds in a new font immediately.

> Read from the shipping source and confirmed on screen — including the update check, run live against the release service, which reported this installation as current.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Credits and Billing

What every action in the terminal costs, what is free and always will be, how the three plans differ, and what happens when the balance runs out.

`>_ 43 priced actions · 5 price shapes · 3 plans · 6 things never metered`

Covers · **the price list** · **every metered action, tab by tab** · **plans & packs**

The standing per-day charges, the allowance meters, and what an empty balance stops

Read live from the Fincept billing API on an Exclusive Pro account, v5.0.1.

Contents

20 SECTIONS

GETTING ORIENTED

01 What a credit is 3
02 The balance in the header 4
03 The price list 5

HOW PRICING WORKS

04 What the tab in front of you costs 6
05 The five ways a price is charged 7
06 What is free, and always will be 9
07 Finding a price 10

THE CATALOGUE

08 AI and agents 11
09 Trading and execution 12
10 Portfolio, quant and deal work 13
11 Markets, research and data 14
12 Alternative data 15
13 The charges that run without you 16

PLANS AND TOP-UPS

14 The three plans 17
15 Monthly and quarterly 18
16 Buying credit packs 19
17 Running out 20

REFERENCE

18 Every priced action, one table 21
19 Honest notes 22
20 Quick reference — how do I… 23

About this guide. Every price here was read from the terminal's own price list, which the dialog states is served by the API rather than held in the client. It is therefore what the server charged on 2026-08-19; treat the in-app dialog as the live authority and this guide as the map of it.

01 What a credit is

A credit is the unit the terminal charges in when an action costs **somebody else** money — a model's tokens, a vendor's data call, a machine on the server doing work your laptop is not doing. Nothing else is metered. The overwhelming majority of the product — every price, every chart, every order, every local model — is free to use as often as you like.

Your plan includes an allowance of credits each month. It resets on a fixed date; unused allowance does not roll over. If you need more before the reset you can buy packs, and purchased credits never expire.

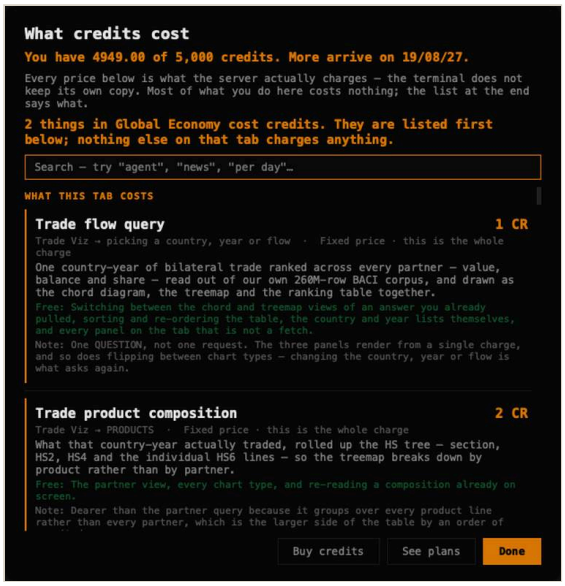


Fig 1 — The price list, opened from the credits chip in the header. It is served by the API, not compiled into the terminal, which is why it is the authority on every number in this guide.

By the end of this guide you will know…

- What every metered action in the terminal costs
- Which six categories are never metered
- How to check a price before you press the button
- The difference between a fixed price and an estimate
- Which charges keep running after you close the terminal
- What your plan includes before anything is billed
- What changes if you pay quarterly
- What a credit pack costs and when it is spent
- Exactly what stops when the balance reaches zero

> **The terminal never guesses a price.** Every priced button carries its cost on its face — `Load · 1 CR`, `TRACK · 1 CR`, `REFRESH · 8 CR` — and an ⓘ beside it that explains the charge in a sentence and opens the full list. You should never have to press something to find out what it costs.

02 The balance in the header

The credit balance sits in the top-right of every screen, between your username and your plan.

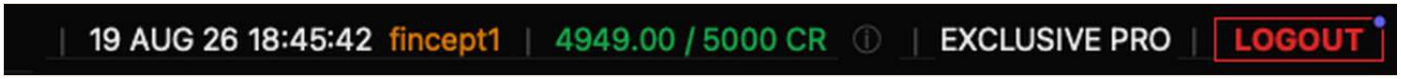


Fig 2 — The header, right-hand end: clock, account, balance, the info mark, plan and sign-out.

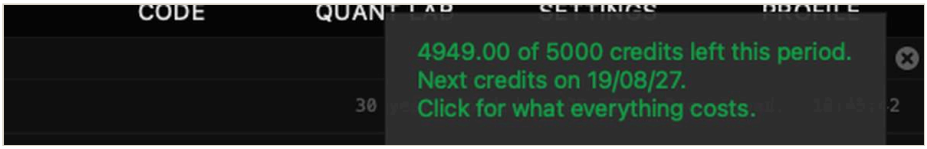


Fig 3 — Hovering the balance tells you the three things that matter: what is left, when more arrives, and that clicking opens the price list.

Element	Reads	What it means
`Balance`	`4949.00 / 5000 CR`	Credits left this period, against your plan's monthly allowance
`Tooltip line 1`	*4949.00 of 5000 credits left this period.*	The same figure in words
`Tooltip line 2`	*Next credits on 19/08/27.*	The reset date — allowance does not roll over
`Tooltip line 3`	*Click for what everything costs.*	The balance itself is a button
ⓘ	—	Opens the same price list (§03)
`Plan`	`EXCLUSIVE PRO`	Your tier (§14)

Two decimal places, and why

Balances are carried to hundredths, not whole numbers, because not every meter is a whole credit — a paper-trading fill is **0.05 CR** and a series load past your allowance is **0.1 CR**. A balance that reads `4949.00` is exact, not rounded.

> **The balance is a period figure, not a wallet.** The number before the slash is what is left of *this period's allowance*. Purchased credits sit behind it and are only spent once the allowance is exhausted (§16), so a top-up does not immediately change the left-hand number.

03 The price list

Click the balance, or the ⓘ beside it, or the ⓘ beside any priced button anywhere in the terminal, and you get the same window.

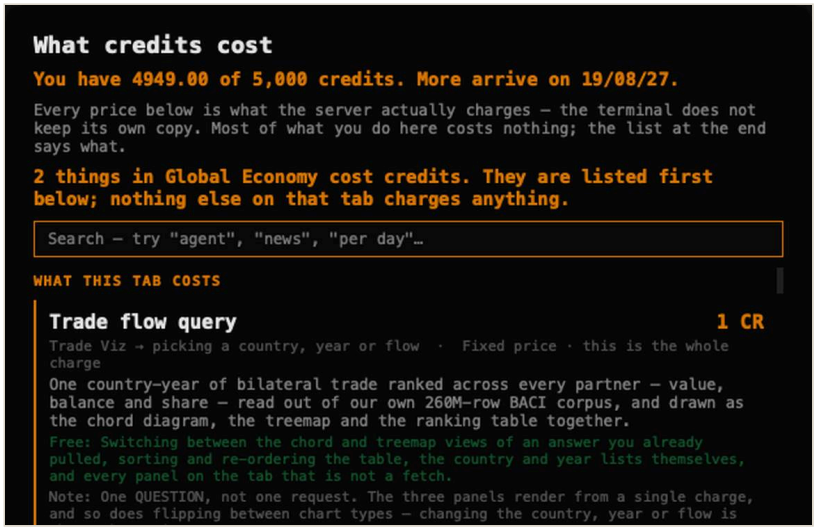


Fig 4 — The head of the list. The balance and reset date, the sentence that makes the server the authority, and a line telling you how much of the tab in front of you is metered.

Region	What it is
`Title`	**What credits cost**
`Balance line`	*You have 4949.00 of 5,000 credits. More arrive on 19/08/27.*
`Provenance`	*Every price below is what the server actually charges — the terminal does not keep its own copy.*
`Tab line`	How many things on the current tab are metered (§04)
`Search`	Filters the whole catalogue (§07)
`WHAT THIS TAB COSTS`	The metered actions of the tab you came from, listed first
`EVERYTHING ELSE`	The rest of the catalogue, grouped by area
`FREE — ALWAYS`	The six things that are not metered (§06)
`Buy credits`	Opens the pack picker (§16)
`See plans`	Opens Plans & Pricing (§14)
`Done`	Closes

This list is fetched, not built in. If the terminal cannot reach the API the dialog says *Prices load with your plan. Reconnect and reopen this to see them.* rather than showing a stale table. That is deliberate — a cached price that has since changed is worse than no price — but it does mean you cannot check a cost while offline.

Every entry carries four things: the **action**, the **price**, the **route** that triggers it (e.g. *Maritime → AREA SEARCH*), and a **Free:** line naming what the same feature does *not* charge for. Several also carry a **Note:** with the billing edge case. Read the Free line — it is usually the more useful half.

04 What the tab in front of you costs

PER TAB · 04

The list is context-aware. Open it from a tab and the first thing it tells you is how much of *that* tab is metered, in one sentence, before any prices.

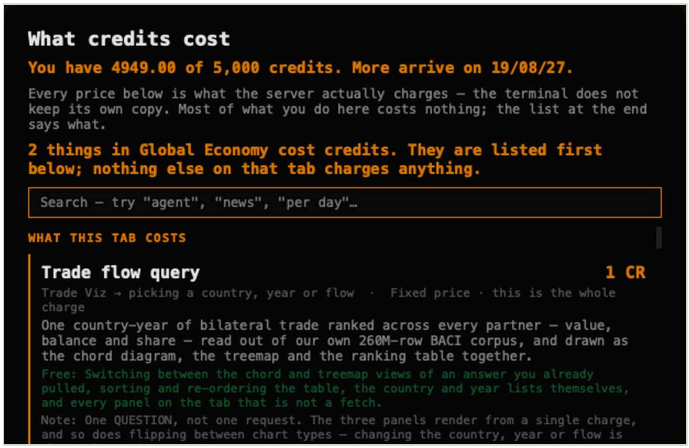


Fig 5 — Opened from Global Economy: two metered actions, listed first.

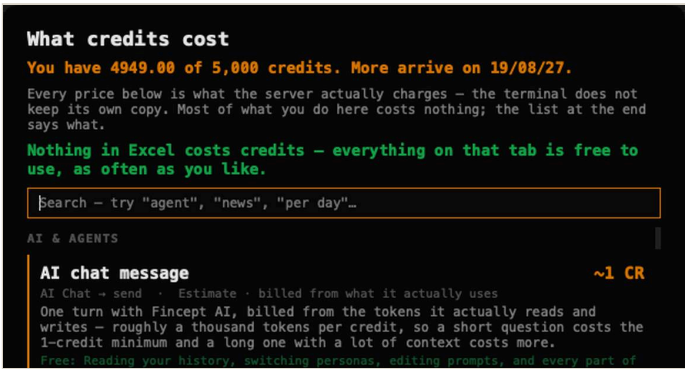


Fig 6 — Opened from Excel: nothing on that tab is metered at all.

The line reads	When
*Nothing in **X** costs credits — everything on that tab is free to use, as often as you like.*	The tab has no metered action. Most tabs
*One thing in **X** costs credits. It is listed first below; nothing else on that tab charges anything.*	Exactly one
***N** things in **X** cost credits. They are listed first below; nothing else on that tab charges anything.*	Two or more

How many tabs are actually metered

Of the 37 screens in the terminal, **most are entirely free**. Excel, Notes, File Manager, Report Builder's editor, the Node Editor, Data Sources, Docs, Profile, Settings, the Dashboard, the Watchlist itself and every local model carry no meter at all. The metered tabs are the ones that reach a model, a vendor or a server engine — and even there, usually one or two buttons out of dozens.

> **This is the fastest way to audit a tab.** Open any screen, click the header ⓘ, and the first sentence tells you whether anything on it can spend. It is quicker and more reliable than reading the buttons, and it is the same answer the server would give.

05 The five ways a price is charged

Every entry names its billing model in grey under the title. There are five, and telling them apart is the difference between a predictable bill and a surprise.

Backtest run — **1 CR**

Algo Trading → Backtest Lab → RUN BACKTEST · Fixed price · this is the whole charge

One backtest executed on the server engine.

Free: The local engines. A backtest that runs on your own machine is free and is not labelled with a price at all.

Note: Charged only when the server engine is switched on.

Fig 7 — **Fixed price**. The number is the whole charge, every time. Most of the catalogue.

AI chat with tools — **~3 CR**

AI Chat → a turn that calls tools · Estimate · 2 CR to start, then what it actually uses

A chat turn that can act — search filings, pull a series, read your portfolio — plus the tokens the tool loop spends. The fee prices the orchestration; the tokens price the volume.

Free: A plain answer with no tool call is billed as an ordinary chat message, not this.

Fig 8 — **Estimate**. A stated amount to start, then metered by what the work actually consumes. The ~ in front of the price is the warning.

Algo runtime — **2 CR per deploy, per day**

Algo Trading → deploy a strategy · Recurring · about 60 CR a month if left on

A day of a deployed strategy running: its order journal, reconciliation, marks and event fills.

Free: Building, testing and backtesting strategies. Paper deploys pay the same rate as live ones — same engine, same infrastructure.

Note: Running out of credits blocks NEW entries only. Exits, stop-losses and open positions are never touched by a billing state.

Fig 9 — **Recurring**. A rate per day, with the monthly cost spelled out. Keeps charging while the thing is left on.

Shape	Reads	What to expect
`Fixed price`	*Fixed price · this is the whole charge*	The printed number, exactly. Nothing further
`Estimate`	*Estimate · N CR to start, then what it actually uses*	At least the starting figure. The ~ price is a typical case, not a cap
`Recurring`	*Recurring · about N CR a month if left on*	A daily rate that keeps going after you close the terminal (§13)
`While open`	*While open · rounded up to the minute, stops when you close it*	Charged by the minute for as long as the feature is held open
`Allowance`	*Only past your plan's included allowance*	Free until you exceed what your plan includes, then priced per unit

Voice dictation — **1 CR per minute of microphone audio, rounded up**

AI Chat → MIC (and the Voice toggle, which keeps it open) · While open · rounded up to the minute, stops when you close it

Speaking to the terminal instead of typing: your microphone transcribed live, tuned for tickers and market vocabulary so "buy a hundred AAPL" arrives as text you can send rather than something about fruit.

Free: Typing, every keyboard shortcut, and clap-to-open-mic — the clap trigger listens on your machine, sends nothing anywhere and is never metered.

Note: The meter runs only while the mic is open, and the minimum is one minute. Push-to-talk closes it after a single message; the Voice toggle holds it open until you switch it off.

Fig 10 — **While open**, with the minimum spelled out: the meter runs only while the microphone is open, and one minute is the floor.

SEC filing pull — **1 CR**

Equity Research → Filings · Only past your plan's included allowance

One fetch from SEC EDGAR past the number your plan already includes.

Free: Every pull inside your monthly allowance, and every cache hit forever — a filing somebody already fetched costs nothing and does not touch the allowance.

Note: Charged only past your plan's included pulls.

Fig 11 — **Allowance**. Note the colour: an allowance meter is drawn in cyan, a metered action in amber.

> **The tilde is the whole story on an estimate.** `~15 CR` for a deep research run means it starts at 10 and finishes wherever the work finishes. Long context, more tool calls and bigger documents all push it up. If you need a guaranteed number, use a fixed-price action.

06 What is free, and always will be

NEVER METERED · 06

The list closes with a section headed **FREE — ALWAYS**, prefaced *These are not metered and are not going to be*. It is a commitment, not a description of the current build.

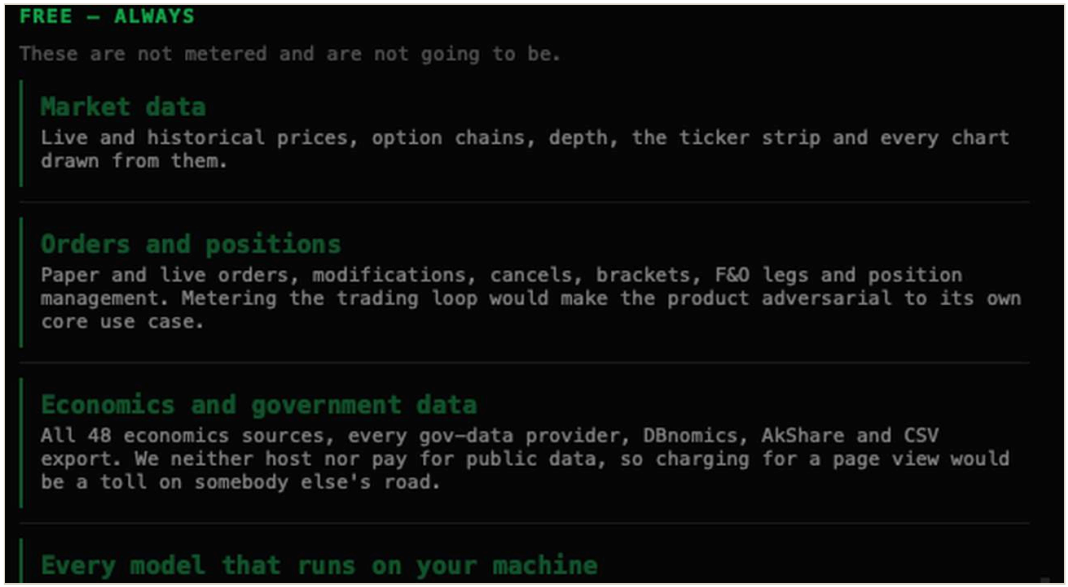


Fig 12 — The head of the free list, with the reasoning attached to each entry.

Never metered	Which is to say
`Market data`	Live and historical prices, option chains, depth, the ticker strip and every chart drawn from them
`Orders and positions`	Paper and live orders, modifications, cancels, brackets, F&O legs and position management
`Economics and government data`	All 48 economics sources, every gov-data provider, DBnomics, AkShare and CSV export
`Every model that runs on your machine`	DCFs, LBOs, accretion/dilution, payoffs, Greeks, local SVI fits, pacing projections, local backtests and all 24 Quant Lab modules
`Reading anything you already have`	The news feed and per-article sentiment, notes, notebooks, dashboards, saved scans, finished runs, past reports and your file library
`Your account and your data`	Cloud sync across devices, file storage, settings, layouts, and the credit statement itself

The reasoning, in the product's own words

- On orders — *"Metering the trading loop would make the product adversarial to its own core use case."*
- On public data — *"We neither host nor pay for public data, so charging for a page view would be a toll on somebody else's road."*
- On local compute — *"If the compute is yours, so is the bill."*

> **Re-reading is always free.** This is the single most useful rule in the whole scheme: once you have paid for an answer, looking at it again — re-opening it, re-sorting it, switching chart types, exporting it — never charges again. The charge is for *asking*, not for *having*.

07 Finding a price

The search box at the top of the list filters the whole catalogue, not just the current tab. It matches the action name, the route and the billing model — which is what makes it useful.

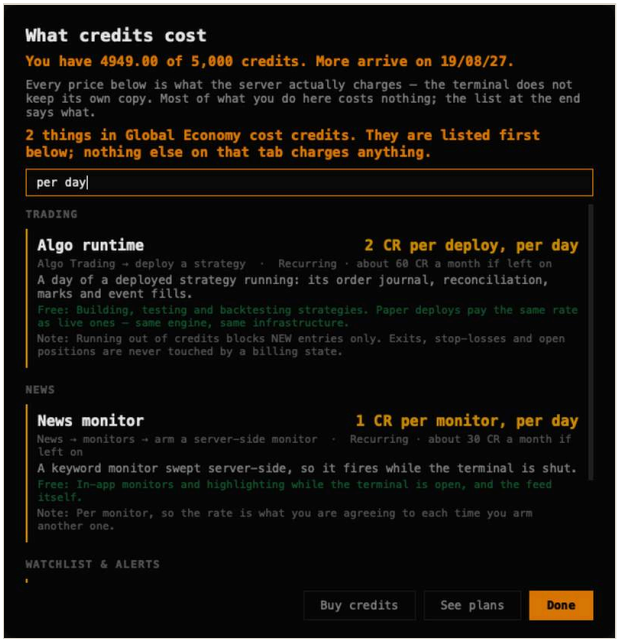


Fig 13 — Searching `per day` returns exactly the meters that keep running without you.

Try	To get
`per day`	Every recurring charge
`estimate`	Everything whose final cost is not fixed
`agent`	The agent and deep-research meters
`news`	Article analysis, corpus sentiment, monitors
`maritime`	The four vessel meters
`allowance`	The meters that only bill past your plan
`free`	— nothing; free items are the tail section

If nothing matches, the dialog says *Nothing matches "X". Everything not listed here is free.* — which is itself an answer.

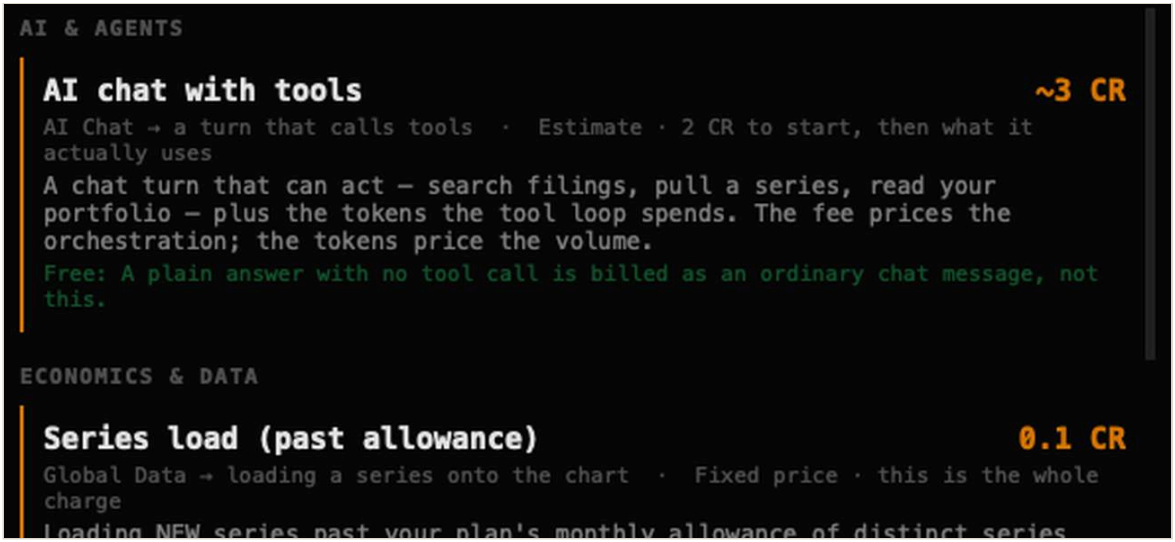


Fig 14 — Searching `series` finds the smallest charge in the terminal: a tenth of a credit, and only past the allowance your plan already includes.

> **The catalogue is one flat list across the whole product.** Searching it is the only way to answer questions like *what else bills me daily?* — the per-tab view cannot, because each tab sees only its own meters.

08 AI and agents

The largest group, and the only one where most prices are estimates rather than fixed. Everything here reaches a hosted model, so the bill follows the tokens.

Action	Where	Shape	Price
AI chat message	AI Chat → send	Estimate, from tokens	**~1 CR**
AI chat with tools	AI Chat → a turn that calls tools	Estimate · 2 CR to start	**~3 CR**
Deep research run	AI Chat → run in the background	Estimate · 10 CR to start	**~15 CR**
Agent run	Agent Studio → RUN	Estimate · 2 CR to start	**~5 CR**
Agent team run	Agent Studio → Teams → RUN	Estimate · 8 CR to start	**~20 CR**
Quant Lab deep agent	AI Quant Lab → Deep Agent	Estimate · 12 CR to start	**~15 CR**
Alpha Arena round	Alpha Arena → start a round	Estimate · 6 CR to start	**~10 CR**
AI research note	Equity Research → Copilot → ANALYZE	Estimate · 4 CR to start	**~6 CR**
M&A deal memo	M&A Analytics → Deal Desk → GENERATE MEMO	Estimate · 4 CR to start	**~8 CR**
AI report generation	Report Builder	Estimate	**~8 CR**
AI market brief	Markets → BRIEF THIS PANEL	Estimate · 2 CR to start	**~3 CR**
AI trade review	Equity Trading → bottom panel → AI REVIEW	Estimate · 2 CR to start	**~3 CR**
AI chart analysis	Global Data → AI ANALYSIS	Fixed	**3 CR**
Ask AI for data	Global Data → ASK AI	Fixed	**1 CR**
Analyse an article	News → open a story → ANALYSE	Fixed	**3 CR**
Sentiment over a corpus	News → agent → SCORE CORPUS	Fixed	**2 CR**
Voice dictation	AI Chat → MIC, or the Voice toggle	While open, per minute	**1 CR/min**
Spoken replies	AI Chat → Voice toggle	While open, per minute	**1 CR/min**

A plain answer is not a tool turn. A chat message with no tool call is billed as an ordinary message (~1 CR), not as a tool turn (~3 CR) — the terminal charges for what the turn actually did, not for which tab you were on. Roughly a thousand tokens per credit, so a short question costs the one-credit minimum and a long one with a lot of context costs more.

The two voice meters are separate and can both run at once. Dictation bills the microphone; spoken replies bill the answer being read aloud. Push-to-talk closes the mic after a single message; the **Voice** toggle holds it open until you switch it off — and the minimum is one minute either way. Reading the same answer on screen is free.

09 Trading and execution

Placing orders is free — deliberately and permanently (§06). What is metered is the *server* doing work: running an engine, sweeping parameters, reconciling against a broker.

Action	Where	Shape	Price
Paper trade	Paper trading → every fill	Fixed, per fill	**0.05 CR**
Backtest run	Algo Trading → Backtest Lab → RUN BACKTEST	Fixed	**1 CR**
Optimiser sweep	Algo Trading → Backtest Lab → OPTIMISE	Fixed	**5 CR**
Server screener run	Algo Trading → Scanner → SERVER SCAN	Fixed	**1 CR**
Broker reconciliation	Equity Trading → bottom panel → RECONCILE	Fixed	**1 CR**
Algo runtime	Algo Trading → deploy a strategy	Recurring · ~60 CR a month	**2 CR/day**

The distinctions worth knowing

- **A local backtest is free.** The 1 CR is for the *server* engine; a backtest that runs on your own machine is not labelled with a price at all. The charge appears only when the server engine is switched on.
- **An optimiser sweep is one charge, not one per run.** Many backtests, ranked in one job — which is why it costs five rather than one each.
- **A paper fill costs the same as nothing much** — five hundredths of a credit — but it is per fill, so a high-frequency paper strategy is the one case where the smallest meter in the product adds up.
- **A paper deploy pays the same daily rate as a live one.** Same engine, same infrastructure.

> **An empty balance never touches an open position.** Running out of credits blocks *new* algo entries only. Exits, stop-losses and open positions are explicitly never affected by a billing state — the product refuses to let a billing problem become a market problem (§17).

10 Portfolio, quant and deal work

All fixed prices, all for a server-side computation. Every equivalent model that runs locally is free, which covers most of what these tabs do.

Action	Where	What it buys	Price
Portfolio AI analysis	Portfolio → QuantStats / FFN / Optimiser → RUN	The tear sheet, the risk decomposition or the optimisation, computed over your whole book	**5 CR**
End-of-day attribution	Portfolio → Performance & Risk → RUN ATTRIBUTION	What contributed, decomposed	**2 CR**
M&A comps set	M&A Analytics → Valuation → PULL COMPS	A screened peer set: candidates matched on industry, fundamentals tidied from XBRL, normalised to one fiscal year and priced against a live quote	**3 CR**
Program pacing simulation	Alt Investments → Program Desk → SIMULATE	Thousands of correlated paths giving a distribution of funding need	**2 CR**
Surface calibration	Derivatives → Surface → SERVER FIT	A calendar-constrained SVI calibration kept as history, so today's fit can be compared with last week's	**2 CR**
Option strategy model	F&O → Builder → SAVE RISK MODEL	A multi-leg risk model computed and saved on the server	**2 CR**

What each of these does not charge for

- **Portfolio** — live marks, position tracking, every card and chart in the tab, and the performance history you already have.
- **M&A** — typing peers in by hand, and all 43 valuation models: the DCF, the LBO, the accretion/dilution, however many times you re-run them.
- **Alt Investments** — the NAV nowcast and every stress scenario.
- **Derivatives** — the local surface fit, and every payoff, Greek and scenario in the tab.
- **F&O** — building the structure. Dragging a strike, refreshing the chain and re-drawing the payoff and Greeks are free however many times you do it; the charge is on **SAVE**.

> **The pattern is consistent across all six.** You pay when the answer is *computed on the server and kept* — a saved model, a stored calibration, a history entry. Exploring is free; committing costs. If you are iterating, iterate locally and save once.

11 Markets, research and data

Two of the five entries here are allowance meters — free until you pass what your plan already includes — which makes this the cheapest group in the catalogue for normal use.

Action	Where	Shape	Price
Market regime scan	Markets → RUN REGIME SCAN	Fixed	**2 CR**
SEC filing pull	Equity Research → Filings	Only past your plan's allowance	**1 CR**
Series load (past allowance)	Global Data → loading a series onto the chart	Fixed, past the allowance	**0.1 CR**
Trade flow query	Global Economy → Load	Fixed	**1 CR**
Trade product composition	Global Economy → PRODUCTS	Fixed	**2 CR**
Scheduled data refresh	Data Mapping → Schedules	Recurring · ~30 CR a month	**1 CR/run**

How the two allowance meters behave

- **SEC filings.** Every pull inside your monthly allowance is free, and **every cache hit is free forever** — a filing somebody has already fetched costs nothing and does not touch your allowance. You are charged only past your plan's included pulls.
- **Global Data series.** The tenth of a credit is charged for loading a *new* series past your monthly allowance of distinct series, and it covers the whole load however many series it brings in. Every **re-load is free forever**: a series counted this month is free to chart, compare, window and export for the rest of it. **Search and browsing never count at all.**

> **Global Data is effectively free to explore.** Searching 1.7 million series, browsing every topic, charting, comparing and CSV export across all 48 sources are all in the never-metered list. Only pulling a brand-new series past your allowance registers, at a tenth of a credit for the batch.

> **A scheduled refresh bills per job, per run.** A cadence set to daily costs about a credit a day *per job*; set it to hourly and the same job costs about a credit an hour. This is the one meter where a single careless setting can outspend everything else on this page combined — check the cadence, not just the job count.

12 Alternative data

Vessel and event data is bought from outside vendors, so it is the most expensive part of the catalogue per action — and the only place where a single button costs eight credits.

Action	Where	What it buys	Price
Vessel position	Maritime → SEARCH	The current position, course, speed and destination of one ship, read out of the AIS collector's own table	**1 CR**
Fleet lookup	Maritime → MULTI-VESSEL	Up to 50 ships positioned in one request — a fleet or a watchlist answered together instead of one at a time	**3 CR**
Vessel track	Maritime → TRACK	One ship's whole recorded voyage — every fix held for it, drawn as a track rather than a point	**5 CR**
Area search	Maritime → AREA SEARCH	Every vessel inside a bounding box over the chosen window — a strait, a port approach or a blockade, swept across the whole position table	**8 CR**
Geopolitics query	Geopolitics → a filtered event search or map fill	One question put to the geopolitical event corpus — 2.4M classified conflict, protest and crisis events from 22 crawled feeds	**3 CR**

Reading the free lines here

- **The map is free.** Panning, zooming and replaying a track you have already pulled costs nothing — and so does re-reading a fix.
- **Drawing the box is free.** You can adjust an area search rectangle as long as you like before you run it; the charge is on the run.
- **Every vessel already on the map is free**, including the ports layer.
- **Geopolitics** — the map, the timeline, opening any event, and the country / city / category / source filter lists are all free. You pay for the question, not for looking at the answer.

> **Area search is the most expensive single button in the terminal.** At 8 CR it is eight times a vessel position and forty times a scheduled refresh. Build the box first, satisfy yourself it is right, and run it once — the guide to Maritime documents an auto-refresh mode that repeats this charge on a timer.

13 The charges that run without you

Three meters are not triggered by a button. They bill for something the server keeps doing on your behalf — including while the terminal is shut — and they are the only way to spend credits without touching the application.

```
Algo runtime                                  2 CR per deploy, per day
Algo Trading → deploy a strategy  ·  Recurring · about 60 CR a month if left on
A day of a deployed strategy running: its order journal, reconciliation,
marks and event fills.
Free: Building, testing and backtesting strategies. Paper deploys pay the same rate
as live ones — same engine, same infrastructure.
Note: Running out of credits blocks NEW entries only. Exits, stop-losses and open
positions are never touched by a billing state.
```

Fig 15 — A recurring entry states both the rate and what a month of it comes to, so you are never working the arithmetic out yourself.

Meter	Rate	A month	What it pays for
`Algo runtime`	2 CR per deploy, per day	~60 CR	A day of a deployed strategy running: its order journal, reconciliation, marks and event fills
`News monitor`	1 CR per monitor, per day	~30 CR	A keyword monitor swept server-side, so it fires while the terminal is shut
`Watchlist alerts`	1 CR per day, whole account	~30 CR	A day of your alerts being evaluated server-side, so they fire while the terminal is shut

Three details that change the arithmetic

- **Algo runtime is per deploy.** Four strategies left running is 8 CR a day, not 2. Paper deploys pay the same rate as live ones.
- **News monitors are per monitor.** The rate is what you agree to each time you arm another one, so ten monitors is ten credits a day.
- **Watchlist alerts are per account, not per alert.** This is the one meter that does *not* multiply: one credit a day covers however many alerts you have armed, and arming more costs no extra. Nothing is charged on a day when you have none.

> **These are the meters worth auditing monthly.** Everything else in the catalogue charges because you pressed something; these charge because you once pressed something. Search the price list for `per day` (§07) to see all three at once, then check each tab for what you have left armed.

> **Free while the terminal is open.** In-app monitors and highlighting, and alerts evaluated locally while the application is running, are not metered at all. The daily charge buys the *server-side* sweep — the part that works when you are not there. If you only need alerts while you are at the desk, you do not need to arm them server-side.

14 The three plans

See plans in the price list, or **View Plans & Pricing** in the header, opens a full-screen comparison. Every tab is on every plan — what changes is how much of each you get.

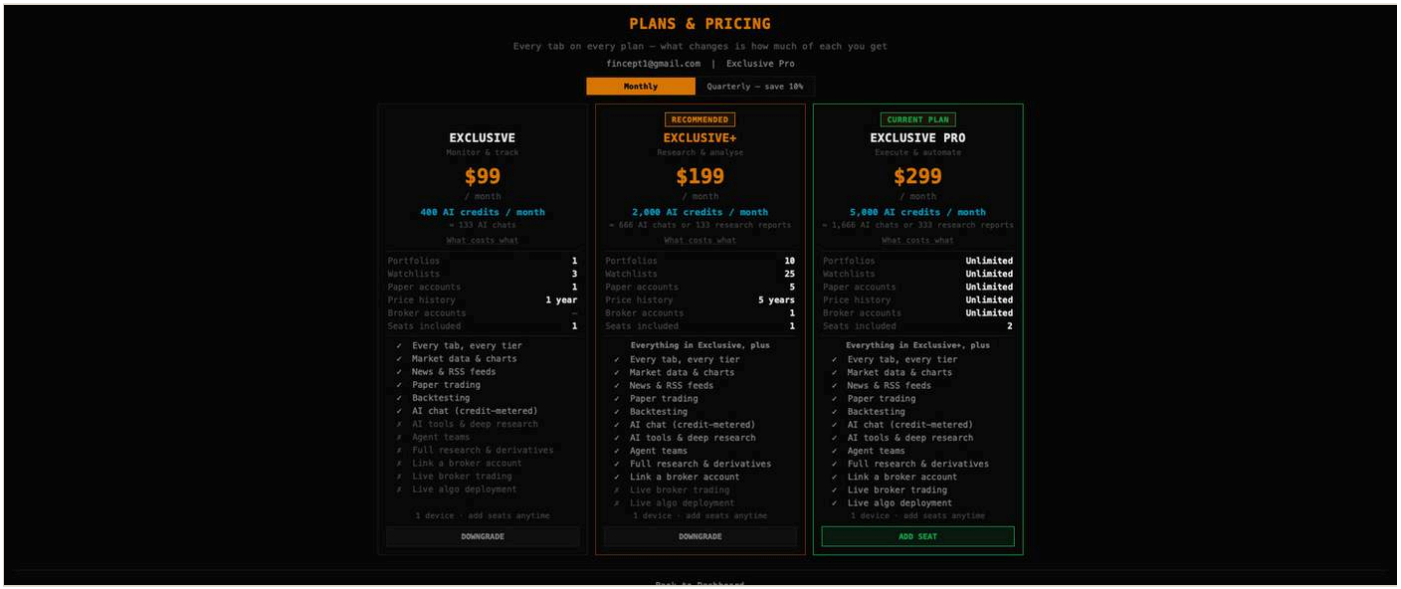


Fig 16 — Plans & Pricing. The screen takes over the whole window, header included; **Back to Dashboard** at the foot is the way out.

	Exclusive	Exclusive+	Exclusive Pro
Positioning	Monitor & track	Research & analyse	Execute & automate
Monthly	**$99**	**$199**	**$299**
AI credits / month	400	2,000	5,000
which is about	133 AI chats	666 chats or 133 research reports	1,666 chats or 333 research reports
Portfolios	1	10	Unlimited
Watchlists	3	25	Unlimited
Paper accounts	1	5	Unlimited
Price history	1 year	5 years	Unlimited
Broker accounts	—	1	Unlimited
Seats included	1	1	2

> **Every tab opens on every plan.** No screen is replaced by an upgrade wall — the tier governs limits and a handful of capabilities, not access. What you cannot do on a lower plan is *link a live broker* and *deploy a live algo*; everything else is a quota.

Capability	Exclusive	Exclusive+	Exclusive Pro
Every tab, every tier	✓	✓	✓
Market data & charts	✓	✓	✓
News & RSS feeds	✓	✓	✓
Paper trading	✓	✓	✓
Backtesting	✓	✓	✓
AI chat (credit-metered)	✓	✓	✓
AI tools & deep research	✗	✓	✓
Agent teams	✗	✓	✓
Full research & derivatives	✗	✓	✓
Link a broker account	✗	✓	✓
Live broker trading	✗	✗	✓
Live algo deployment	✗	✗	✓

15 Monthly and quarterly

The toggle above the plan cards switches the whole comparison between the two cycles. Quarterly is the same product at a **10% discount**, billed once for three months.

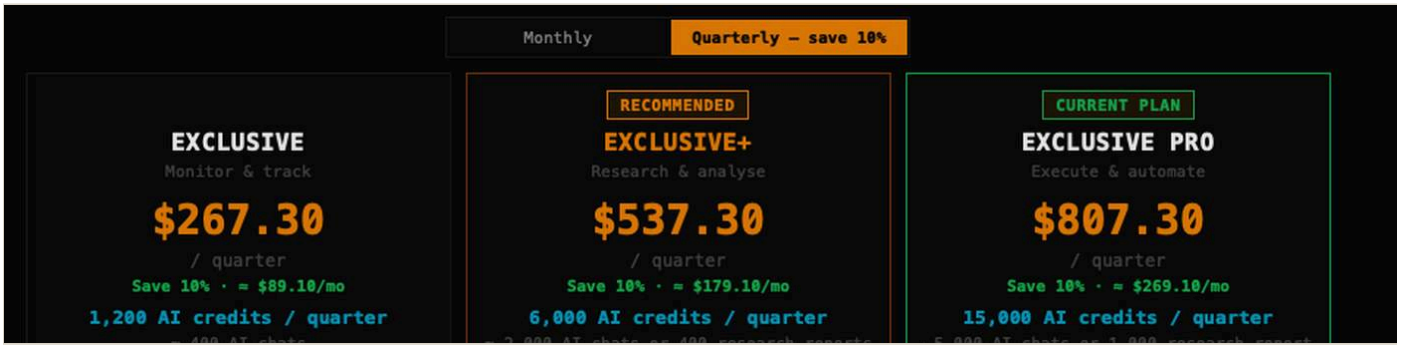


Fig 17 — Quarterly. Each card shows the quarter price, the saving, and the effective monthly rate underneath it.

Plan	Monthly	Quarterly	Effective / month	Credits / quarter
`Exclusive`	$99	**$267.30**	≈ $89.10	1,200
`Exclusive+`	$199	**$537.30**	≈ $179.10	6,000
`Exclusive Pro`	$299	**$807.30**	≈ $269.10	15,000

What the cycle does and does not change

- **The price per credit is identical.** Quarterly buys three months of the same monthly allowance in one payment — 400 × 3 = 1,200 and so on — at 10% off the cash price.
- **The allowance still resets monthly.** The quarterly figure is three monthly grants, not one large pool you can spend in week one. The header still counts against the monthly number and still shows a monthly reset date (§02).
- **Limits are unchanged.** Portfolios, watchlists, seats and history depth are properties of the tier, not the cycle.

> **Where the plan you are on is shown.** Three places agree: the header chip prints the tier name beside the balance, the plan card is outlined in green and badged **CURRENT PLAN**, and the account line above the cards names the signed-in email and tier. The recommended tier is badged separately in amber and is not necessarily yours.

> **Changing plan is not a preview.** The buttons on the cards are live — **DOWNGRADE** on a lower tier, **ADD SEAT** on your own. There is no basket and no summary step described on the screen itself, so read the card you are pressing rather than the one you are looking at.

16 Buying credit packs

Buy credits in the price list opens a small pack picker. Packs are a top-up on the plan, not a replacement for it.

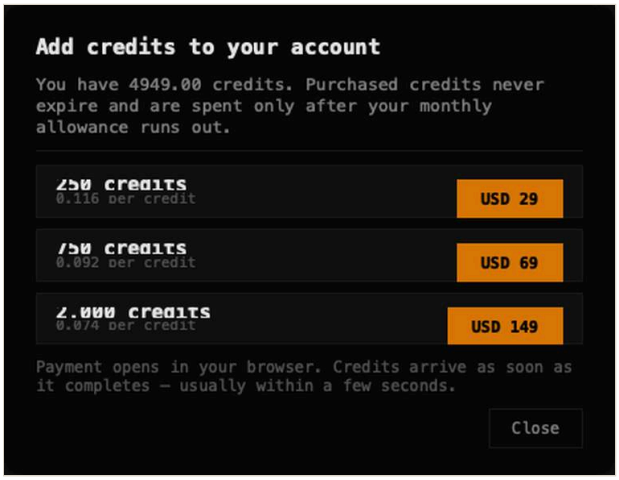


Fig 18 — Three packs, each showing the unit price. Payment completes in your browser.

Pack	Price	Per credit
250 credits	USD 29	0.116
750 credits	USD 69	0.092
2,000 credits	USD 149	0.074

The 2,000-credit pack is **36% cheaper per credit** than the smallest, so the only reason to buy the small one is that you need less than 750.

Rule	Detail
They never expire	Unlike the monthly allowance, purchased credits carry forward indefinitely
Spent last	Your plan's allowance is consumed first; packs are only touched once it runs out
Paid in the browser	*Payment opens in your browser. Credits arrive as soon as it completes — usually within a few seconds.*
Balance updates itself	No restart, no manual refresh

> **The two-stage spend is what makes a pack safe to hold.** Because the allowance is always spent first, buying a pack in a heavy month does not waste it — anything you do not need stays there for the next one. There is no way to spend a pack while allowance remains.

> **Cosmetic, but visible on first open:** the pack titles are clipped along their top edge — *250 credits*, *750 credits* and *2,000 credits* each lose the tops of their letters because the row is shorter than the two lines it holds. The prices and unit costs beside them are unaffected.

17 Running out

An empty balance is a soft stop, not a lockout. The terminal keeps working; the metered actions decline politely and tell you what would fix it.

What you see	When
*This costs **N** credits and you have **M**.*	A fixed-price action you cannot afford
*This usually costs about **N** credits and you have **M**.*	An estimate you cannot afford
*You've used all **N X** on your plan.*	An allowance meter, exhausted
*You've used all **N X** on your plan. **Y** makes them unlimited.*	...where a higher tier removes the limit
*You've used all **N X** on your plan. **Y** raises this to **Z**.*	...where a higher tier raises it
***X** is part of the **Y** plan.*	A capability your tier does not include
This isn't available on your plan.	...with no named upgrade
This isn't available on any current plan.	Not purchasable at any tier

What keeps working regardless

- **Everything in the never-metered list** (§06) — prices, charts, orders, local models, your own data.
- **Exits, stop-losses and open positions.** Stated explicitly on the algo runtime meter: running out blocks *new* entries only and a billing state never touches a position you already hold.
- **Armed alerts stay armed.** The watchlist meter is charged only on days the sweep actually ran; an empty balance defers the charge rather than disarming anything.
- **Reading what you already paid for.** Finished runs, past reports, saved scans and your file library are all in the free list.

> **The refusal always names the price and your balance in the same sentence**, so you can see the gap without opening anything. Where a higher tier would remove the limit rather than raise it, the message says so by name.

18 Every priced action, one table

FULL TABLE · 18

All 43, cheapest first. ~ marks an estimate; everything else is the whole charge.

Price	Action	Tab
0.05 CR	Paper trade, per fill	Paper trading
0.1 CR	Series load, past allowance	Global Data
~1 CR	AI chat message	AI Chat
1 CR	Ask AI for data	Global Data
1 CR	SEC filing pull, past allowance	Equity Research
1 CR	Trade flow query	Global Economy
1 CR	Backtest run, server engine	Algo Trading
1 CR	Server screener run	Algo Trading
1 CR	Broker reconciliation	Equity Trading
1 CR	Vessel position	Maritime
1 CR/min	Voice dictation	AI Chat
1 CR/min	Spoken replies	AI Chat
1 CR/run	Scheduled data refresh	Data Mapping
1 CR/day	News monitor, per monitor	News
1 CR/day	Watchlist alerts, whole account	Watchlist
2 CR	Trade product composition	Global Economy
2 CR	Market regime scan	Markets
2 CR	Sentiment over a corpus	News
2 CR	End-of-day attribution	Portfolio
2 CR	Program pacing simulation	Alt Investments
2 CR	Surface calibration	Derivatives
2 CR	Option strategy model	F&O
2 CR/day	Algo runtime, per deploy	Algo Trading
3 CR	AI chart analysis	Global Data
3 CR	Analyse an article	News
3 CR	M&A comps set	M&A Analytics
3 CR	Fleet lookup	Maritime
3 CR	Geopolitics query	Geopolitics
~3 CR	AI chat with tools	AI Chat
~3 CR	AI market brief	Markets
~3 CR	AI trade review	Equity Trading
5 CR	Optimiser sweep	Algo Trading
5 CR	Portfolio AI analysis	Portfolio
5 CR	Vessel track	Maritime
~5 CR	Agent run	Agent Studio
~6 CR	AI research note	Equity Research
8 CR	Area search	Maritime
~8 CR	M&A deal memo	M&A Analytics
~8 CR	AI report generation	Report Builder
~10 CR	Alpha Arena round	Alpha Arena
~15 CR	Deep research run	AI Chat
~15 CR	Quant Lab deep agent	AI Quant Lab
~20 CR	Agent team run	Agent Studio

Read against a plan. An Exclusive Pro month is 5,000 credits — about 1,666 chat messages, 333 deep research runs, 625 area searches, or one strategy deployed every day of the month with 4,940 credits left over.

19 Honest notes

Things that will surprise you if nobody tells you

- **Three meters bill while the terminal is shut** — algo runtime, news monitors and watchlist alerts. They are the only way to spend without opening the application (§13).
- **Algo runtime and news monitors multiply; watchlist alerts do not.** Four deploys is four times the rate; forty alerts is still one credit a day.
- **A paper deploy costs the same as a live one** — same engine, same infrastructure.
- **A scheduled refresh is priced per run, not per job.** The cadence, not the job count, is what sets the bill.
- **Estimates have no ceiling.** The ~ price is typical, the "to start" figure is the floor, and nothing on screen caps the top.
- **Voice bills by the minute with a one-minute minimum**, and dictation and spoken replies are two separate meters that can run together.
- **Purchased credits are always spent last.** A top-up cannot be consumed while plan allowance remains, which is what makes it safe to hold.

Product defects seen while documenting this

- **Pack titles are clipped along the top edge** in the Buy credits dialog (§16).
- **Some price captions are singularised wrongly** — a one-credit action reads *Costs 1 credits* in its tooltip.
- **Two meters still name a tab that no longer exists.** The Global Economy prices are captioned *Trade Viz → …*, although the summary line above them correctly says *Global Economy*. The tab was renamed in v5.0.1 and the pricing catalogue's captions did not follow.
- **The price list is unavailable offline** by design, so a cost cannot be checked without a connection (§03).

> **Trust the dialog over this guide.** Everything here was read from the live catalogue on 2026-08-19 on an Exclusive Pro account. Because the prices are served rather than compiled in, they can change without a terminal update — and the in-app list will be right on the day you read it while a printed table can only be right on the day it was printed.

20 Quick reference — how do I…

How do I…	Do this
See my balance	Top-right of the header, always visible
See when more credits arrive	Hover the balance — line two is the reset date
Open the full price list	Click the balance, or any ⓘ anywhere in the terminal
Check whether this tab costs anything	Open the list — the first sentence answers it
Check one button before pressing it	Hover the ⓘ beside it for a one-line answer
Tell a fixed price from an estimate	The ~ in front of the number, and the grey line under the title
Find everything that bills daily	Search the list for `per day`
Find what is never metered	Scroll to **FREE — ALWAYS** at the foot of the list
Stop a recurring charge	Undeploy the strategy, disarm the monitor, or delete the alerts
Avoid paying twice for one answer	Re-reading is always free — do not re-run to look again
Iterate a model without paying each time	Work locally; the charge is on the server run or the save
Get the cheapest credits	The 2,000 pack — 0.074 each against 0.116 for the smallest
Keep unused credits	Buy a pack; purchased credits never expire. Plan allowance does not carry over
Pay less per month	Switch to quarterly — 10% off, same allowance each month
See which plan I am on	The header chip, or the green **CURRENT PLAN** card
Compare plans	**See plans** in the price list, or **View Plans & Pricing** in the header
Add a seat	**ADD SEAT** on your own plan card
Know what stops when I run out	Nothing you already hold — new metered actions only
Keep trading with no credits	You already can. Orders, exits and stops are never metered

> **The short version.** Prices, charts, orders and anything computed on your own machine are free forever. You pay when a hosted model thinks, a vendor is queried, or a server keeps working while you are away — and the terminal prints the price on the button before you press it. Three meters bill daily; everything else bills when you ask.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Markets Board

A complete, screen-by-screen walkthrough of the live market-monitoring board — a Bloomberg-style grid of streaming quote panels you arrange yourself.

`>_ indices · fx · commodities · bonds · crypto`

Covers · **the 5-region board** · **reading a panel** · **the 9 default panels** **building your own board** · the panel editor · refresh & states

Demonstrated live on the Fincept Terminal Markets board.

Contents

8 SECTIONS

GETTING ORIENTED

01 What the Markets tab is 3

02 Anatomy of the screen — the five regions 4

03 The header bar 5

04 Reading a market panel 6

BUILDING YOUR BOARD

05 The nine default panels 7

06 Building your own board (+ the panel editor) 8

REFERENCE

07 Refresh, states & good to know 10

08 Quick reference — “How do I…” 11

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Control labels are quoted exactly as they appear on screen; prices shown reflect a single live moment.

01 What the Markets tab is

The Markets tab is a live quote-monitoring **board**. It fills the screen with small **panels**, each a compact table of streaming quotes for a basket of instruments you choose — indices, currencies, commodities, bonds, ETFs, cryptocurrencies, or your own custom watchlists. Panels sit in three resizable columns that you can rearrange, add to, edit and reset. A header strip across the top shows the U.S. market session, world clocks, and refresh controls.

It is a **viewing surface** — there is no order entry here. To trade, use the Equity or Algo tabs; right-clicking a row does offer a one-click jump to Backtesting.

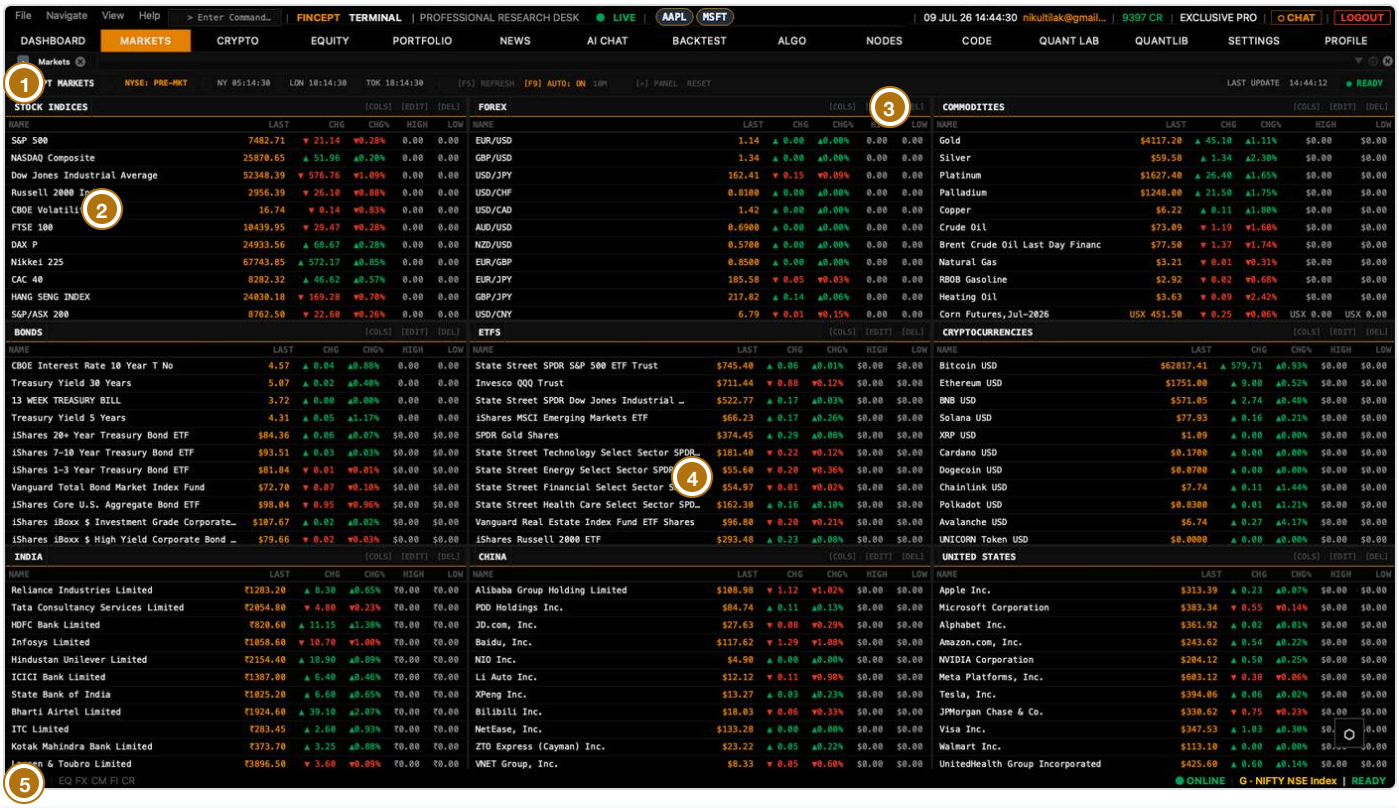
At a glance, you can…

- Track **live prices, change and % change** for dozens of instruments at once, colour-coded green/red.
- Read six markets in one screen: **Stock Indices, Forex, Commodities, Bonds, ETFs and Crypto**, plus regional **India / China / U.S.** baskets.
- **Add, edit, rename and delete panels** — build a board around the markets you actually watch.
- **Search any symbol by name** ("Reliance", "Bitcoin", "Gold") and drop it into a panel.
- **Choose which columns** each panel shows (up to seven) and **resize** panels by dragging.
- Refresh **on demand** (F5) or let it **auto-refresh** on an interval from 5 minutes to 1 day.
- See the **NYSE session state** (pre-market / open / after-hours / closed) and **New York, London & Tokyo clocks** at a glance.
- **Copy a symbol** or send it straight to the **Backtester** from any row.

> **Where the data comes from.** Quotes stream in through the terminal's live data bus. By default they are sourced locally; when the market-data API is enabled they flow from the Fincept edge gateway (`/v1/market/quotes`), authenticated with your session. Either way the board looks and behaves identically. Symbol search uses `/v1/market/search` .

02 Anatomy of the screen

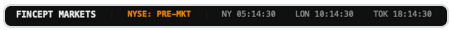
Five regions make up the Markets tab. The numbers below map to the callouts on the screenshot.



#	Region	What it is
1	**Header bar**	Brand label, NYSE session status, world clocks, and all board-wide controls (refresh, auto-refresh, interval, add panel, reset, last-update time, status badge).
2	**Market panel**	One tile = one basket. A titled table of live quotes. The board ships with nine of them.
3	**Panel controls**	Per-panel `[COLS]` `[EDIT]` `[DEL]` buttons in each panel's header.
4	**Three-column board**	Panels are laid out in three columns. Drag the thin dividers between panels and between columns to resize.
5	**Status bar**	App version, connectivity (`ONLINE`), and the global data-feed readiness indicator.

03 The header bar

Everything that acts on the whole board lives here, left to right.

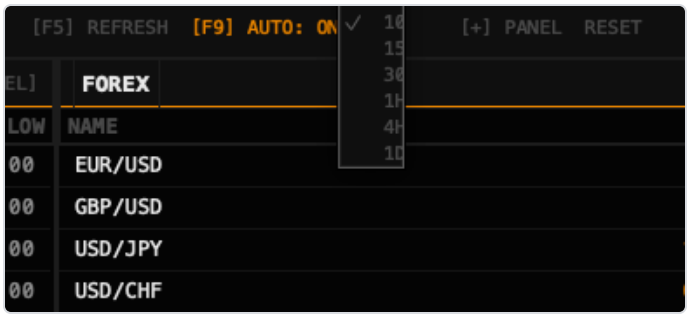


Brand · session · clocks

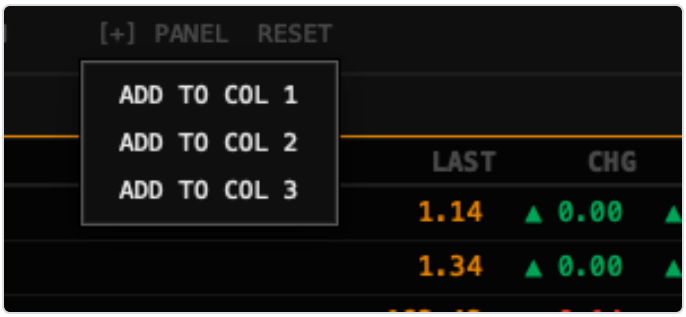
Refresh & layout controls

Last update · status

Control	What it does
NYSE session	Live U.S.-market status computed from New York time: `CLOSED` `PRE-MKT` `OPEN` `AFTER-HRS`. Colour-coded (green = open, amber = pre/after). Updates every minute.
World clocks	`NY` / `LON` / `TOK` wall-clock times, ticking every second.
`[F5] REFRESH`	Force-refresh every panel now. The status badge shows `LOADING` until all panels report back; `LAST UPDATE` then stamps the time.
`[F9] AUTO: ON/OFF`	Toggle automatic periodic refresh. **On by default** (shown amber). Auto-refresh pauses while the tab is hidden.
Interval	How often auto-refresh fires: `5M · 10M · 15M · 30M · 1H · 4H · 1D`. **Default 10M.**
`[+] PANEL`	Add a new panel — pick which column it lands in, then define it in the panel editor.
`RESET`	Restore the nine factory-default panels (asks to confirm first).
`LAST UPDATE`	Time of the most recent successful full refresh (`--:--:--` until the first one completes).
Status badge	`● READY` (green) · `● LOADING` (amber) · `● TIMEOUT` (red, if the feed does not fully answer within 10 seconds).



Interval menu (`5M` – `1D`); the ✓ marks the current choice.



`[+] PANEL` → choose the target column.

> **Tip — the `[F5]` / `[F9]` labels are reminders, not live keys on this screen.** Click the buttons themselves; the bracketed keys are mnemonics.

04 Reading a market panel

Every panel is a title, a set of columns, and one row per instrument. Rows show as many as fit the panel's height — make a panel taller to see more.

STOCK INDICES [COLS] [EDIT] [DEL]

NAME	LAST	CHG	CHG%	HIGH	LOW
S&P 500	7482.71	▼ 21.14	▼0.28%	0.00	0.00
NASDAQ Composite	25870.65	▲ 51.96	▲0.20%	0.00	0.00
Dow Jones Industrial Average	52348.39	▼ 576.76	▼1.09%	0.00	0.00
Russell 2000 Index	2956.39	▼ 26.10	▼0.88%	0.00	0.00
CBOE Volatility Index	16.74	▼ 0.14	▼0.83%	0.00	0.00
FTSE 100	10439.95	▼ 29.47	▼0.28%	0.00	0.00
DAX P	24933.56	▲ 68.67	▲0.28%	0.00	0.00
Nikkei 225	67743.85	▲ 572.17	▲0.85%	0.00	0.00
CAC 40	8282.32	▲ 46.62	▲0.57%	0.00	0.00
HANG SENG INDEX	24030.18	▼ 169.28	▼0.70%	0.00	0.00
S&P/ASX 200	8762.50	▼ 22.60	▼0.26%	0.00	0.00

The **Stock Indices** panel. Green ▲ = up, red ▼ = down; the price is shown in amber.

Columns

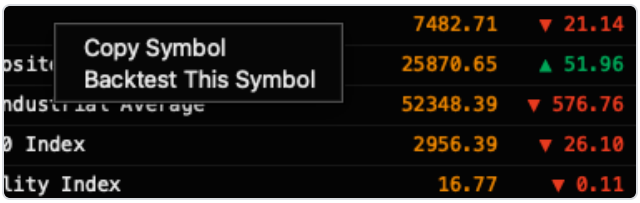
Instruments are listed by **friendly name** (the raw ticker such as `^GSPC` resolves to "S&P 500"; hover a name for its full name and ticker). Prices carry the right currency symbol ($, €, ₹, ¥…).

Column	Meaning
NAME	Friendly instrument name (locked — always shown). Hover for full name + raw ticker.
LAST	Last traded price, currency-prefixed (locked — always shown).
CHG	Absolute change on the day, with a ▲/▼ arrow.
CHG%	Percent change on the day.
HIGH / LOW	Day's range. (Shows `0.00` before regular-session data is available — e.g. pre-market, as in these screenshots.)
VOL	Volume, compact-formatted.
TICKER	The raw symbol (e.g. `^GSPC`, `GC=F`).
BID / ASK / OPEN	Available as columns, but the current feed does not supply them — they display `--`.

Default columns: NAME · LAST · CHG · CHG% · HIGH · LOW.

Right-click a row

Copy Symbol	Copies the raw ticker to the clipboard.
Backtest This Symbol	Loads the symbol into the Backtester and jumps to that tab.



05 The nine default panels

A fresh board (and the `RESET` button) gives you these nine baskets, spread across the three columns.

Panel	Covers
Stock Indices	Global benchmarks — S&P 500, NASDAQ, Dow, Russell 2000, VIX, FTSE 100, DAX, Nikkei 225, CAC 40, Hang Seng, ASX 200, Sensex, Nifty…
Forex	Major currency pairs — EUR/USD, GBP/USD, USD/JPY, USD/CHF, USD/CAD, AUD/USD … through USD/INR.
Commodities	Gold, Silver, Platinum, Palladium, Copper, Crude, Brent, Natural Gas, and more futures.
Bonds	Treasury yields (10Y, 30Y, 5Y, 13-wk) and bond ETFs — TLT, AGG, LQD, HYG …
ETFs	SPY, QQQ, DIA, GLD and the sector SPDRs (XLK, XLE, XLF, XLV…).
Cryptocurrencies	Bitcoin, Ethereum, BNB, Solana, XRP, Cardano, Dogecoin, Chainlink, Polkadot, Avalanche…
India	Reliance, TCS, HDFC Bank, Infosys, HUL, ICICI, SBI, Bharti Airtel, ITC, Kotak, L&T…
China	Alibaba, PDD, JD, Baidu, NIO, Li Auto, XPeng, Bilibili, NetEase, ZTO, VNET…
United States	Apple, Microsoft, Alphabet, Amazon, NVIDIA, Meta, Tesla, JPMorgan, Visa, Walmart, UnitedHealth…

CRYPTOCURRENCIES [COLS] [EDIT] [DEL]

NAME	LAST	CHG	CHG%	HIGH	LOW
Bitcoin USD	$62817.41	▲ 579.71	▲0.93%	$0.00	$0.00
Ethereum USD	$1751.00	▲ 9.00	▲0.52%	$0.00	$0.00
BNB USD	$571.05	▲ 2.74	▲0.48%	$0.00	$0.00
Solana USD	$77.93	▲ 0.16	▲0.21%	$0.00	$0.00
XRP USD	$1.09	▲ 0.00	▲0.00%	$0.00	$0.00
Cardano USD	$0.1700	▲ 0.00	▲0.00%	$0.00	$0.00
Dogecoin USD	$0.0700	▲ 0.00	▲0.00%	$0.00	$0.00
Chainlink USD	$7.74	▲ 0.11	▲1.44%	$0.00	$0.00
Polkadot USD	$0.8300	▲ 0.01	▲1.21%	$0.00	$0.00
Avalanche USD	$6.74	▲ 0.27	▲4.17%	$0.00	$0.00
UNICORN Token USD	$0.0000	▲ 0.00	▲0.00%	$0.00	$0.00

The **Cryptocurrencies** panel — one of the nine defaults.

06 Building your own board

The board is entirely yours to shape. Each panel header carries three buttons.

A panel header: title, then `[COLS]` `[EDIT]` `[DEL]`.

Action	How
Add a panel	Header `[+] PANEL` → pick a column → the panel editor opens for a new, empty panel.
Edit / rename	Panel's `[EDIT]` → change the title and its list of symbols.
Choose columns	Panel's `[COLS]` → tick the columns to show (NAME and LAST are always on; up to **7** total).
Delete	Panel's `[DEL]` → confirm.
Resize / rearrange	Drag the thin dividers between panels and between columns. Your column widths are remembered.
Reset everything	Header `RESET` → restores the nine defaults.

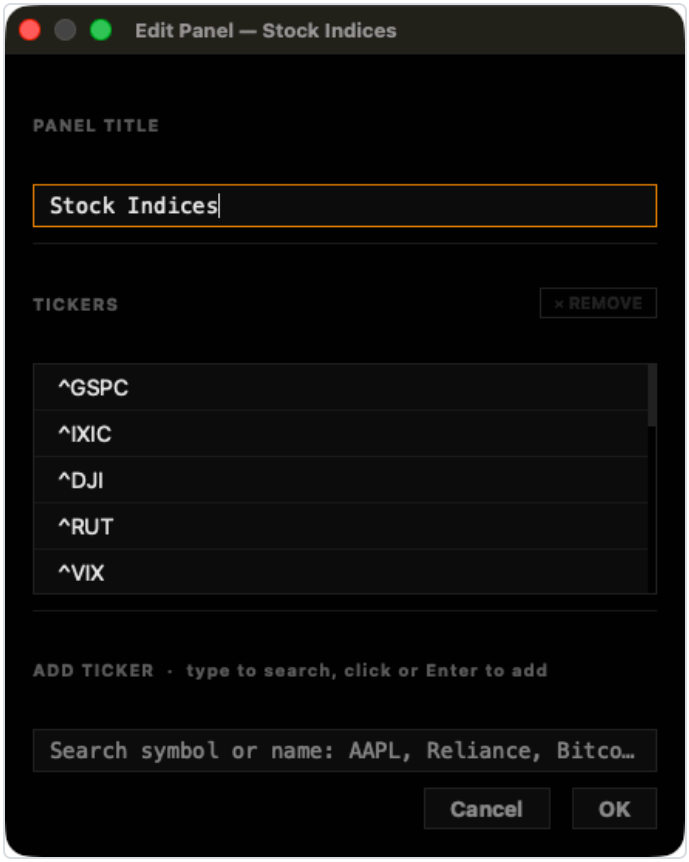
Your layout — panels, symbols, columns and column widths — is saved automatically and restored next time.

Column picker

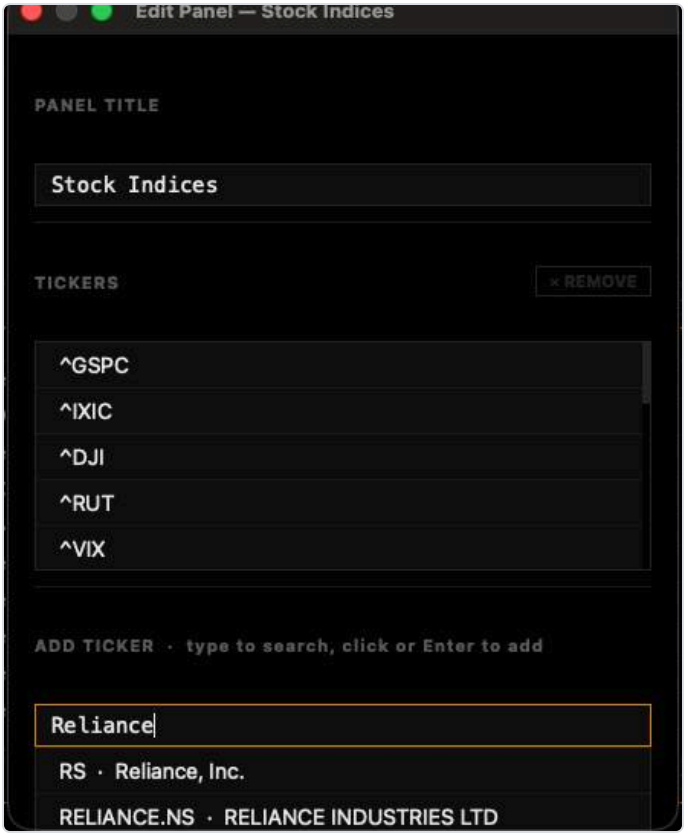
The `[COLS]` menu — in each panel's header — lets you add or remove optional columns for that panel. Ticked columns are shown; NAME and LAST cannot be removed. Once seven columns are active, the remaining ones are disabled until you free a slot.

The panel editor

Adding or editing a panel opens this dialog.



Title, the ticker list with `× REMOVE`, and the search box.



Type a name or symbol — results appear live; click or press Enter to add.

- **Panel title** — name it whatever you like ("My Tech Stocks").
- **Tickers** — the current symbols; select one and click `× REMOVE` (or double-click) to drop it.
- **Add ticker** — start typing a symbol *or* a company/asset name; matches (symbol · name) appear after a moment. Enter or click adds it.
- Click **OK** to keep your changes, or **Cancel** to discard them.

07 Refresh, states & good to know

How refreshing works

Behaviour	Detail
Manual refresh	`[F5] REFRESH` re-pulls every panel immediately.
Auto-refresh	On by default; fires on the chosen interval (default every 10 minutes). Pauses while the tab is not in view.
On returning to the tab	Refreshes only if it has been at least ~5 minutes since the last one — so quick tab-switches don't hammer the feed.
Quote freshness	Each quote is cached briefly and rate-limited behind the scenes, so many symbols update smoothly without overloading the source.

Panel & board states

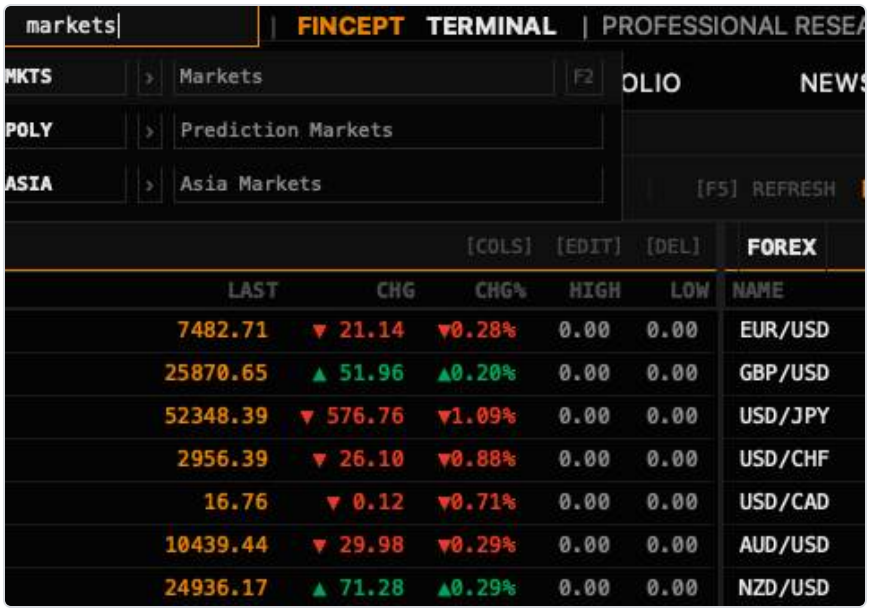
You see…	It means…
`⠿ LOADING` **(in a panel)**	That panel is fetching its first data. It clears as quotes arrive.
`● READY` **(badge)**	The last full refresh completed successfully.
`● LOADING` **(badge)**	A refresh is in progress across the board.
`● TIMEOUT` **(badge)**	Not every panel answered within 10 seconds — usually a slow or rate-limited data source. The next refresh normally recovers it.
`0.00` **in HIGH/LOW**	Intraday range isn't available yet (e.g. pre-market). It fills in during regular trading hours.
`--` **in BID/ASK/OPEN**	Those fields aren't provided by the current feed.

> **Heads up.** `RESET` and a panel's `[DEL]` both change your board. Each asks you to confirm first, so a stray click won't wipe your layout.

08 Quick reference — "How do I..."

I want to...	Do this
Open the Markets tab	Click `MARKETS` in the top tab bar, or open the command bar and type `markets` (hint `F2`).
Refresh prices now	Click `[F5] REFRESH`.
Change how often it updates	Pick a value in the interval dropdown (`5M` – `1D`).
Stop auto-updating	Click `[F9] AUTO` to toggle it off.
Add a market / watchlist	`[+] PANEL` → choose a column → title it and search-add symbols.
Add a symbol to a panel	Panel `[EDIT]` → type its name or ticker in *Add ticker* → Enter.
Remove a symbol	Panel `[EDIT]` → select it → `× REMOVE`.
Show more / fewer columns	Panel `[COLS]` → tick columns (max 7).
Rename a panel	Panel `[EDIT]` → change *Panel title*.
See more rows in a panel	Drag its bottom divider down to make it taller.
Delete a panel	Panel `[DEL]` → confirm.
Start over	`RESET` → confirm (restores the nine defaults).
Copy a ticker	Right-click a row → *Copy Symbol*.
Backtest a symbol	Right-click a row → *Backtest This Symbol*.
Check if the U.S. market is open	Read the `NYSE` status in the header.

Getting to Markets from anywhere



The command bar (top-left) — type `markets` and press Enter.

Fincept Terminal — Markets Guide · screenshots captured from the live application. Prices shown are illustrative of a single pre-market moment.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Equity Trading

A complete, screen-by-screen walkthrough of the Equity tab — the terminal's multi-broker order desk: watchlist, dual charts, market depth, a full order ticket and a five-tab blotter, in paper or live — and the F&O desk it opens.

`>_ 22 brokers · paper + live · depth · GTT · the F&O desk · 39 agent tools`

Covers · **connecting brokers** · **paper vs live and the approval gate** · **the order ticket**
charts & depth · the blotter · multi-account orders · what persists · the agent tools
Futures & Options · the chain · the strategy builder · open interest · institutional flow

Demonstrated live on a working desk, and on a live NIFTY option chain.

Contents

24 SECTIONS

GETTING ORIENTED

01 What the Equity tab is 3
02 Anatomy of the workstation 4
03 The command bar 5

BEFORE YOU TRADE

04 Broker accounts 6
05 Paper, live, and the approval gate 8

READING THE MARKET

06 The watchlist 9
07 Charts, compare and chart trading 10
08 Market depth 12

PLACING ORDERS

09 The order ticket: the core form 13
10 Advanced, GTT and option strategies 14
11 Sending one order to several accounts 16

MANAGING WHAT YOU HOLD

12 Blotter: Positions and Holdings 17
13 Blotter: Orders, Funds and Stats 19
14 US brokers, feeds and the unified monitor 21

FUTURES & OPTIONS

15 The F&O desk 22
16 The option chain 23
17 Trading from the chain 25
18 The strategy builder 26
19 The risk model, and trading a structure 29
20 Open interest and straddles 31
21 Screening, flows, positions and cost 34

REFERENCE

22 Data, persistence and agent tools 37
23 Quick reference 38
24 Honest notes on this build 39

About this guide. Every screenshot is a live capture of the shipping build — the Equity tab on a desk with four configured brokers, and the F&O desk on a connected fyers account with a live NIFTY chain. **No order was placed, modified or cancelled to produce it**: every figure is a read-only view or a dialog that was opened and dismissed. One credential field is masked in section 04, and the one charged control in F&O was run once, deliberately, to photograph its result.

01 What the Equity tab is

ORIENTATION · 01

The Equity tab is the terminal's **order desk**. One screen carries a live watchlist, one or two charts, a market-depth ladder, a full order ticket and a five-tab blotter — wired to as many broker accounts as you have configured, in either paper or live mode.

It is a **multi-account coordinator**, not a single-broker front end. Each configured account keeps its own connection, portfolio and credentials; the tab focuses one of them at a time for the ticket and the chart, can broadcast a single order to several at once, and can show every account's positions together in a side column.

> **This tab sends real orders to real brokers.** In **LIVE** mode the submit button routes to your broker immediately, and — with the default order-approval setting — **without a confirmation dialog**. Section 05 explains exactly what protects you, how to turn on a review step, and how to tell paper from live at a glance. Read it before you use the ticket.

Where to find it

Route	How
Tab bar	**EQUITY** — fourth from the left on the top tab bar
Command bar	Type `equity` and pick the row
From research	**Equity Research** and other screens can open this tab's ticket for a symbol directly

At a glance, in the Equity tab you can…

- Watch a named list of symbols stream live from your broker
- Chart one symbol, or two side by side
- Read a five-level depth ladder with an imbalance read-out
- Place market, limit and both stop order types
- Attach a stop-loss and target to an entry
- Rest a GTT / Forever order until it triggers
- Build a seven-template option strategy as a basket
- Open the full F&O desk — chain, builder, OI and flows — sections 15–21
- Send the same order to several broker accounts at once
- Track positions, holdings, orders, funds and statistics
- Square off one row, the winners, the losers, or everything
- Practise on a paper account with the same ticket
- Drive all of it from AI Chat through 28 agent tools

02 Anatomy of the workstation

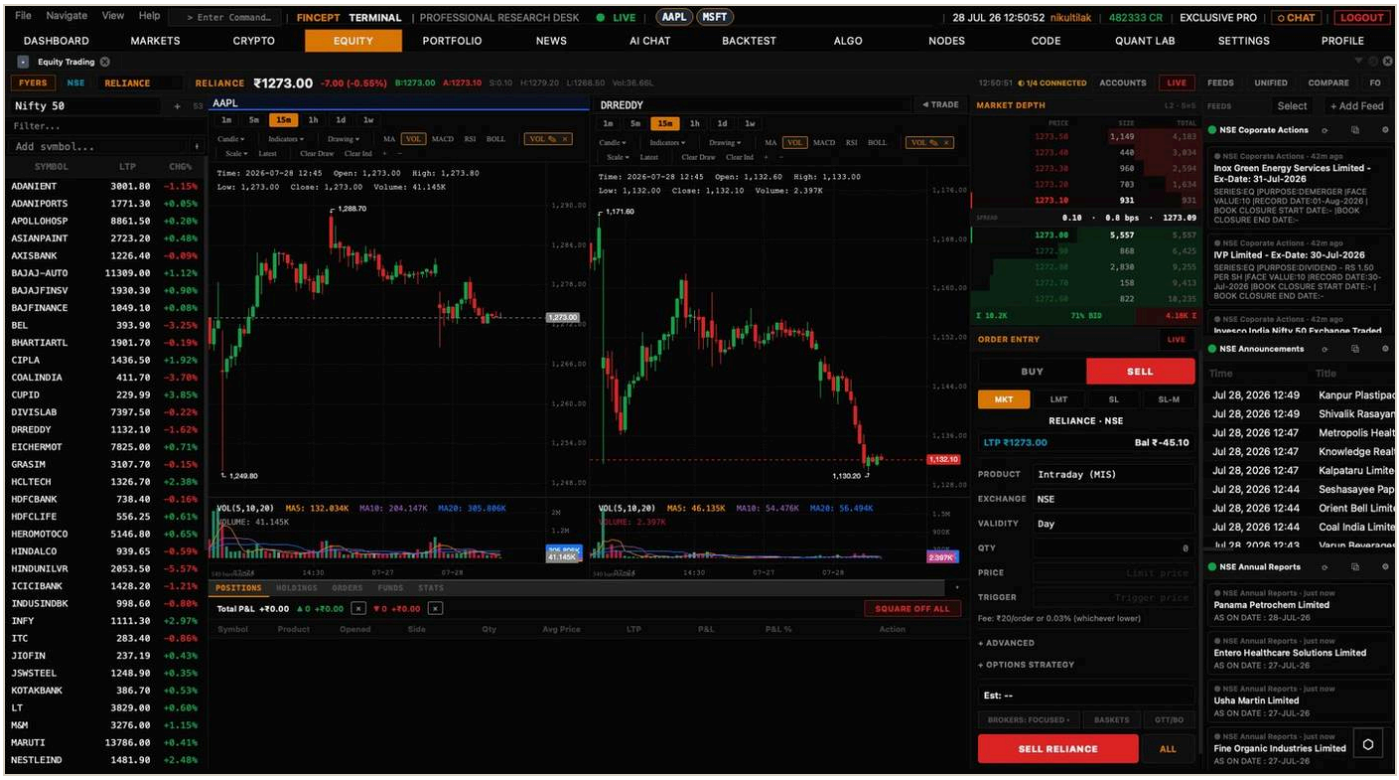


Fig 1 — The Equity tab on a live desk: Fyers focused, RELIANCE selected, compare chart and the feed column both open.

#	Region	What it is for
1	Command bar	Focused account, exchange, symbol box, live ticker ribbon, clock, connection count, and the six toggle buttons. Section 03.
2	Watchlist	The active named list, streaming live, plus the symbols you hold. Section 06.
3	Charts	The primary chart (the traded symbol) and, when **COMPARE** is on, a second live chart beside it. Section 07.
4	Blotter	Positions · Holdings · Orders · Funds · Stats, collapsible, with three more tabs for US brokers. Sections 12–14.
5	Market depth	Five levels each side with cumulative bars, spread and an order-flow imbalance footer. Section 08.
6	Order ticket	The full order form, plus advanced fields, option strategies and the multi-account selector. Sections 09–11.
7	Side columns	Optional feed column (**FEEDS**) and all-accounts portfolio monitor (**UNIFIED**). Section 14.

> Every divider is draggable and the layout is remembered: the width of the feed and unified columns, whether the compare chart is open, which watchlist is active, the symbol, the focused account and even a half-typed order ticket all come back when you return to the tab. Section 22 lists exactly what is stored.

03 The command bar

Fig 2 — The command bar. Left to right: account, exchange, symbol, the live ticker ribbon, clock, connection count, then the toggles.

Control	What it does
`Account button`	Names the focused account (e.g. `FYERS`). Click for the account menu — every configured account, the focused one ticked. Switching re-points the ticket, chart, depth and blotter at that account.
`Exchange`	The exchange of the selected symbol (`NSE`, `SMART`, …). Follows the symbol, not a setting.
`Symbol box`	Type to search the broker's instrument master; the picker shows friendly names (including F&O contracts) and resolves the real symbol on select. For IBKR, suggestions come from a live symbol lookup.
`Ticker ribbon`	The selected symbol's live line: last price, change and %, bid/ask, spread, day high/low and volume.
`Clock`	Local time, ticking while the tab is visible.
`n/m` `CONNECTED`	How many configured accounts currently have a live connection — the fastest check that your broker session is alive.
`ACCOUNTS`	Opens the broker-account manager. Section 04.
`PAPER / LIVE`	The mode of the focused account, and the single most important label on the screen. Section 05.
`FEEDS`	Shows or hides the news-feed column on the far right.
`UNIFIED`	Shows or hides the all-accounts portfolio monitor column.
`COMPARE`	Opens a second chart beside the first, both live.
`F&O`	Jumps to the options-chain tab.

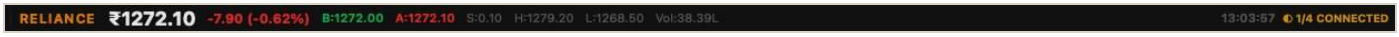


Fig 3 — The ticker ribbon in detail: `RELIANCE ₹1272.10 −7.90 (−0.62%) B:1272.00 A:1272.10 S:0.10 H:1279.20 L:1268.50 Vol:38.39L`.

> **The ribbon is the price of record on this screen.** The ticket's LTP line, the position rows and the ribbon are all fed by the same quote stream, so they agree tick for tick — if a number disagrees with the ribbon, it is a cached or server-side mark, not a fresher price.

04 Broker accounts

Everything about connecting brokers lives behind the **ACCOUNTS** button. The terminal ships **22 broker integrations** plus paper accounts, and a desk can keep several live at once.

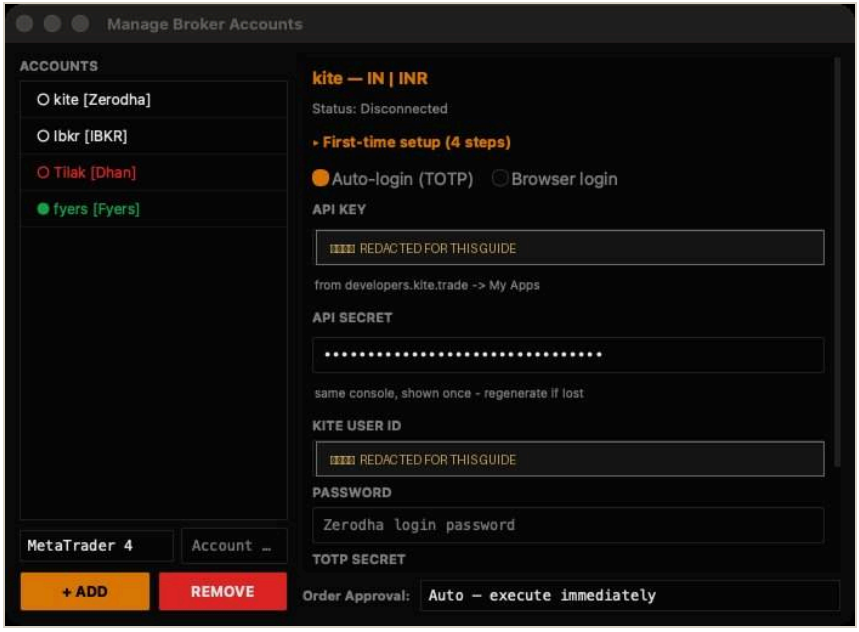


Fig 4 — The account manager. Left: the configured accounts with a status dot each. Right: the selected broker's credential form, its connection status, a first-time-setup walkthrough and the **Order Approval** mode. *Two credential values are masked in this figure.*

Control	Purpose
`Account list`	One row per account, coloured by state: green connected, red needs attention, grey disconnected
`+ ADD / REMOVE`	Add an account for the broker chosen in the picker below the list, or remove the selected one
`RENAME`	Give the account the nickname shown in the command bar
`CONNECT`	Authenticate and start the account's data stream
`Credential fields`	Per-broker: API key and secret, user id, MPIN/password, TOTP secret, vendor code — whatever that broker needs
`Login mode`	Auto-login (TOTP) or Browser login, where the broker supports both
`Order Approval`	**Auto** or **Semi-Auto** — see section 05

Fig 5 — The account menu from the command bar: four accounts, the focused one ticked.

Fig 6 — The blotter tab set for an Indian broker: five tabs.

Aspect	How it works
`Where credentials live`	In the operating system's keychain, never in the terminal's database. Brokers with their own paper environment keep separate live and paper credentials.
`Multiple accounts`	Each account has its own connection, portfolio and instrument master. The tab focuses one; the blotter is cleared on every switch so one account's rows never linger under another.
`Instrument masters`	Downloaded and cached per broker — quotes, charts and depth need them, so the first connection to a broker takes a moment longer.
`Token expiry`	A dead session is surfaced once, not on every request; reconnect from this dialog.

> **While you are filling in the credential form, the API key is shown in clear text** (the secret is masked). Fill it in somewhere private, and treat a screen-share of this dialog the same way you would treat the broker's own portal.

05 Paper, live, and the approval gate

SAFETY · 05

This is the most important page in the guide. Two independent things decide what happens when you press submit: **which mode the focused account is in**, and **that account's order-approval setting**.

1 · Mode — paper or live

Mode	What submit does	How you can tell
`PAPER`	Places into the server-side paper engine. Real prices, simulated fills and margin, no broker involved and no money at risk.	The command-bar button reads **PAPER**; the ticket's top-right chip reads **PAPER**
`LIVE`	Sends the order to the broker.	The command-bar button reads **LIVE**; the ticket chip reads **LIVE**

Mode is a property of the *account*, not a switch you flip per order — the button reflects the focused account. A paper account is the right place to learn this screen: the ticket, the blotter and the square-off buttons all behave identically.

2 · Order approval — Auto or Semi-Auto

Setting	Manual orders from this ticket	Orders from agents, workflows and broadcast
`Auto` `(the default)`	**Sent immediately — no confirmation dialog.**	Sent immediately
`Semi-Auto`	An inline confirmation appears first, listing account, side, symbol, quantity, type, price and estimated value. Nothing is sent until you press **Send Order**.	Queued as pending orders for you to approve or reject from the Action Center

Read that table twice if your account is live. With the default **Auto** setting a live order leaves for the broker the moment you click the coloured submit button — there is no “are you sure”. If you want a review step, set **Order Approval → Semi-Auto** for that account in the ACCOUNTS dialog (section 04). Risk-reducing actions and read-only calls are never queued.

Square-offs and cancels do confirm. The destructive blotter buttons — square off a row, the winners, the losers, all positions, all holdings, cancel all orders — each ask first, in both modes. It is the *entry* path that is immediate in Auto mode.

Two more guards worth knowing. A market order is refused until a quote has arrived (*“Price not available yet — wait for quotes to load”*), so a stale symbol cannot be sent blind; and **GTT / Forever orders are live-only** — a paper account rejects them rather than pretending to rest them.

06 The watchlist

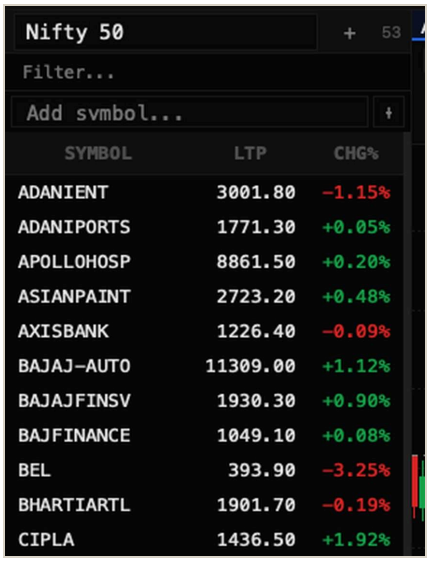


Fig 7 — The watchlist: list switcher, filter box, add box, then **SYMBOL / LTP / CHG%** streaming live.

Control	What it does
`List switcher`	Picks the active named list. Lists are global to the terminal, so the same names appear in the Watchlist tab
`+ / list buttons`	Create, rename and delete lists
`Filter box`	Narrows the visible rows by typing
`Add symbol box`	Adds a symbol to the active list
`Row click`	Loads that symbol into the chart, ticker, depth and ticket
`Right-click`	Remove the symbol from the active list
`LTP / CHG%`	Live from the focused account's stream, coloured by direction, patched per tick

A first run seeds a default list for the broker's region — the NIFTY 50 constituents for an Indian broker, a large-cap US list for a US broker.

> **The list shows more than the list.** The rows you see are the active watchlist *plus every symbol you hold* — open positions and holdings — so anything in your book gets a live price even when it is not on the list. A consequence: **Remove** only does something for real members of the list; on a position-only row there is nothing to remove.

> **Prices come from the focused account.** Switch to a broker that does not serve a symbol's exchange and that row sits at `--`. If a whole list is dashes, check the connection count in the command bar first — an unconnected account streams nothing.

07 Charts, compare and chart trading

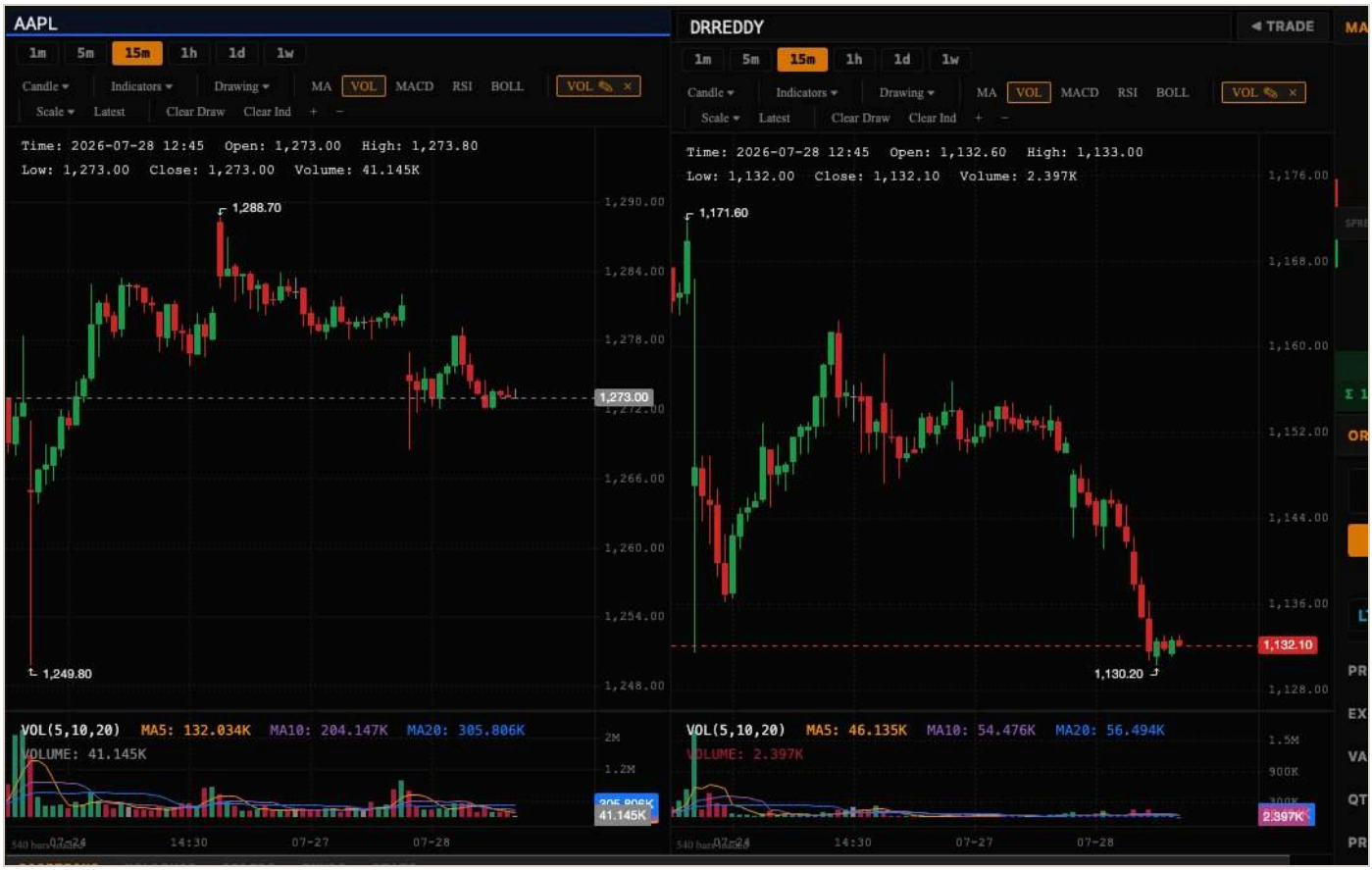


Fig 8 — Two live charts. The left pane is the primary (the traded symbol); the right is the compare pane, with a ◀ **TRADE** button to promote it.

Fig 9 — The chart toolbar: timeframes, then the drawing and indicator controls, then the quick indicator toggles.

Control	What it does
`1m / 5m / 15m / 1h / 1d / 1w`	Timeframe. The chart refetches history for the new slot size
`Candle / Indicators / Drawing`	Chart type, the indicator picker, and the drawing tools
`MA · VOL · MACD · RSI · BOLL`	Quick indicator toggles; volume and moving averages print their legends on the pane
`Scale / Latest / Clear Draw / Clear Ind`	Rescale, jump to the latest candle, and clear drawings or indicators
`Right-click`	**Buy**, **Sell** and **Add to watchlist**. The buy and sell entries open the ticket for the clicked level
`Position card`	When you hold the charted symbol, an overlay card shows the position with its entry line and an **EXIT** button (confirmed before it squares off)

Fig 10 — The chart's right-click menu.

Fig 11 — With a US broker focused the blotter gains three tabs.

The compare pane

- **COMPARE** opens a second chart; both stream live and each keeps its own timeframe.
- Click a pane's header to **focus** it — a watchlist click or a symbol typed in the command bar then loads into the focused pane.
- ◀ **TRADE** promotes the compare symbol to primary, swapping the panes so the ticket, ticker and depth follow it.

> **A stale pane header to know about.** When the tab settles on a different broker account as it opens, the primary pane's header can keep showing the previous symbol while the ticker bar, the ticket and the chart's own candles have already moved to the new one. In Fig 8 the header reads `AAPL` while the traded symbol is `RELIANCE` — the candles are RELIANCE's. **Trust the ticker ribbon and the ticket's symbol line**, and click the symbol again if you want the header to catch up.

08 Market depth

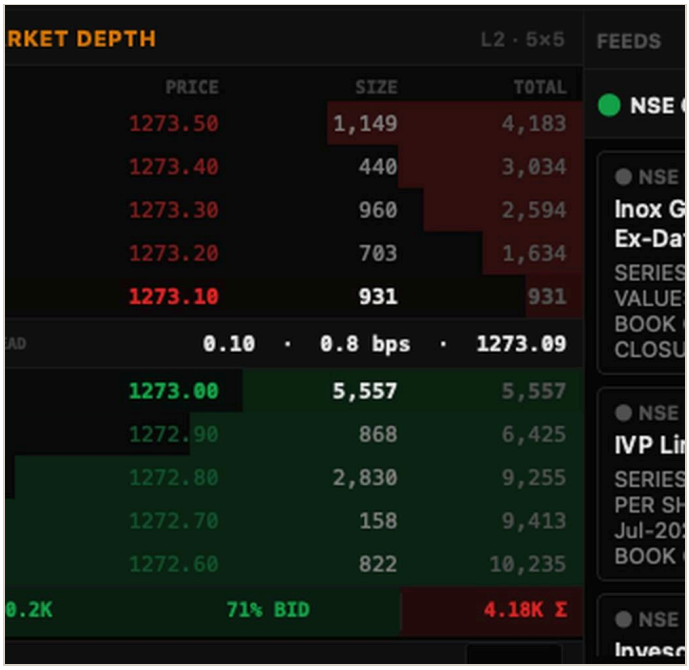


Fig 12 — The depth ladder: asks above, bids below, cumulative bars behind each level, the spread band across the middle and the imbalance footer at the bottom.

Element	Meaning
`PRICE / SIZE / total`	Each level's price, its resting size and the running cumulative total
`Depth bars`	Cumulative size drawn behind the rows, so a wall is visible as a shape rather than a number
`Best of book`	The top bid and ask are emphasised
`Spread band`	The absolute spread and its width in basis points, between the two sides
`Imbalance footer`	Total bid size, the bid share of the book as a percentage, and total ask size — a one-glance read of which way the book leans
`Level badge`	`L2 · 5×5` when the broker sends real depth; `L1 SYNTH` when only a top-of-book quote is available and the ladder is synthesised

Click a price to load it. Clicking any level puts that price into the ticket's limit field and switches the order type to **LMT**, which is the quickest way to place a limit at a level you can see resting in the book.

"No depth data" is normal in two cases: the broker does not provide a depth feed for that instrument, or the market is closed. The ladder fills as soon as the stream delivers levels — it does not need a refresh.

09 The order ticket: the core form

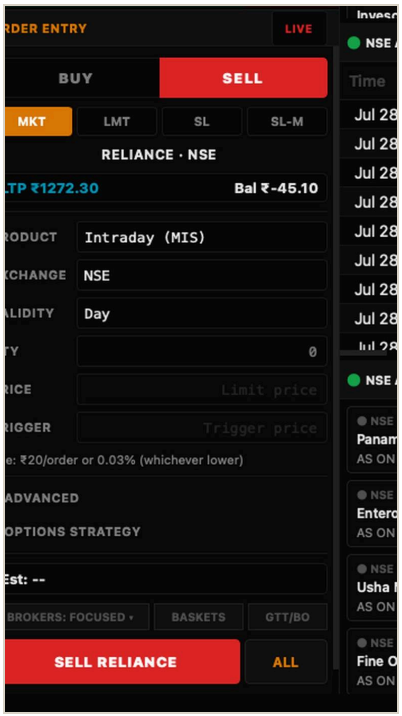


Fig 13 — The ticket with both collapsible sections closed. The chip beside **ORDER ENTRY** reads **LIVE**.

Field	What it does
`BUY / SELL`	The side. The submit button at the foot takes the side's colour and reads **BUY *SYMBOL*** or **SELL *SYMBOL***
`MKT / LMT / SL / SL-M`	Market, limit, stop-limit and stop-market. The price and trigger fields enable themselves to match
`Symbol · exchange line`	The instrument this ticket will trade — the authoritative label on the screen
`LTP / Bal`	The live last price, and the account's available balance in its own currency
`PRODUCT`	The broker's product types — for an Indian broker *Intraday (MIS)*, *Delivery (CNC)*, *Margin (NRML)*. Rebuilt per broker, so a US broker shows its own set
`EXCHANGE`	The venue to route to, from the broker's list
`VALIDITY`	*Day* or *IOC*
`QTY`	Quantity. An empty or non-numeric value is refused with *"Enter a valid quantity"*
`PRICE`	Limit price — enabled for LMT and SL
`TRIGGER`	Trigger price — enabled for SL and SL-M
`Fee line`	The broker's brokerage rule for this order (e.g. *₹20/order or 0.03%, whichever lower*)
`Est. / margin`	Estimated order value, and the margin the broker reports as required — fetched shortly after you stop typing

The ticket remembers what you typed. Side, type, quantity, price, trigger, product, exchange, validity and the advanced fields all survive a tab switch and a restart, so a half-prepared order is still there when you come back. Changing the focused account rebuilds the combos first, then re-applies your inputs.

Where else the ticket opens. The same form is used by the chart's right-click Buy/Sell, by **SELL** and **Buy/Sell** on a position or holding row (pre-filled with the held quantity), and by other screens such as Equity Research — so there is exactly one order path in the terminal, not several.

10 Advanced, GTT and option strategies

Two collapsible sections sit under the core form. Both are closed by default.

+ ADVANCED

Fig 14 — The advanced section.

Field	What it does
`SL`	Stop-loss price attached to the entry
`TP`	Take-profit / target price attached to the entry
`DISCLOSED QTY`	Iceberg disclosure. The hint states the rule: it must be more than 30% of the quantity
`After-Market Order (AMO)`	Queue the order for the next session, where the broker supports it
`GTT / Forever`	Place as a resting trigger order instead of a normal one

> **GTT / Forever is live-only.** Tick it on a paper account and the ticket refuses with *"GTT / Forever orders are live-only (not paper)"*. It also needs a trigger price — without one you get *"GTT needs a trigger price (TRIGGER field)"*. Existing resting orders are listed and cancelled from the **GTT/BO** button.

+ OPTIONS STRATEGY

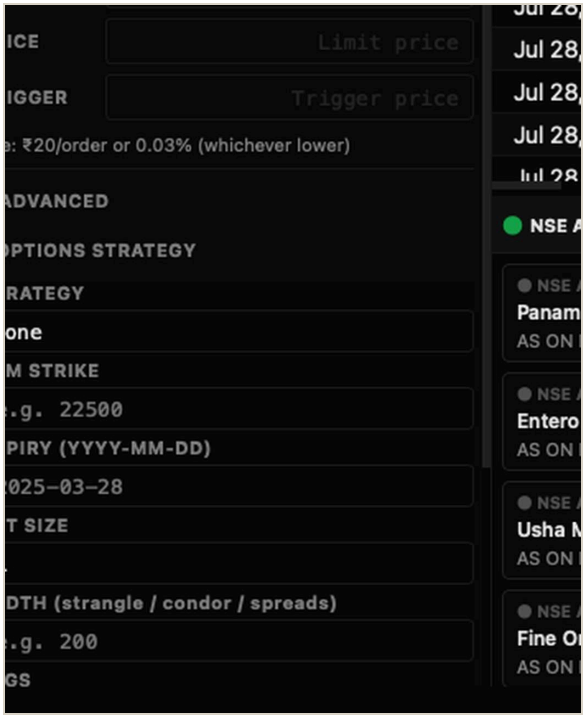


Fig 15 — The strategy builder.

Field	What it does
`STRATEGY`	One of seven: *Long Straddle, Short Straddle, Long Strangle, Iron Condor, Bull Call Spread, Bear Put Spread* — or *None*
`ATM STRIKE`	The centre strike to build around
`EXPIRY`	Contract expiry, `YYYY-MM-DD`
`LOT SIZE`	Lots per leg
`WIDTH`	Strike distance for strangles, condors and spreads
`LEGS`	A preview of the legs the template will produce
`PLACE STRATEGY`	Sends the legs as one basket order

> **At 1920×1080 the strategy builder runs past the bottom of the column.** With the section expanded you can scroll the ticket down to **LEGS**, but **PLACE STRATEGY** stays below the fold — collapse **+ ADVANCED**, drag the depth/ticket divider up, or use a taller window to reach it. The **F&O** tab is the roomier place to build multi-leg positions.

11 Sending one order to several accounts

Under the ticket sit three buttons that turn a single form into a multi-account instrument.

Control	What it does
`BROKERS: FOCUSED ▾`	A checkable list of your accounts. Leave it on **FOCUSED** and the order goes to the focused account only; tick two or more and the **same order is placed on every ticked account**.
`ALL`	The broadcast shortcut beside the submit button — sends the prepared order to the selected set.
`BASKETS`	Opens the basket-order dialog: several orders assembled and sent together, which is also what an option strategy produces.
`GTT/BO`	Lists this account's active GTT (Forever) and bracket orders, and cancels them.

A broadcast multiplies the position, not the risk-per-account. Three ticked accounts means three orders of the stated quantity — one per account — so the desk's total exposure is three times what the ticket shows. Check the ticked set before you submit; the button label tells you when it is more than the focused account.

Approval applies per account. With Semi-Auto set on an account, a broadcast to it queues a pending order for approval rather than sending, while accounts in Auto mode send immediately — one broadcast can therefore be part-sent and part-queued. The Action Center in the status bar is where the queued ones wait.

Replicating a book. The Holdings tab's **REPLICATE** action copies a live account's portfolio into a paper account, which is the cleanest way to rehearse a rebalance on your real positions without touching them.

Practical sequence for a multi-account entry

1. Focus the account whose product types and exchange you want the ticket built from.
2. Prepare the order: side, type, quantity, price.
3. Open **BROKERS** and tick the accounts to include.
4. Confirm the mode chip and the submit label read what you expect.
5. Submit, then watch the Orders tab — each account's acknowledgement arrives separately.

12 Blotter: Positions and Holdings

Positions

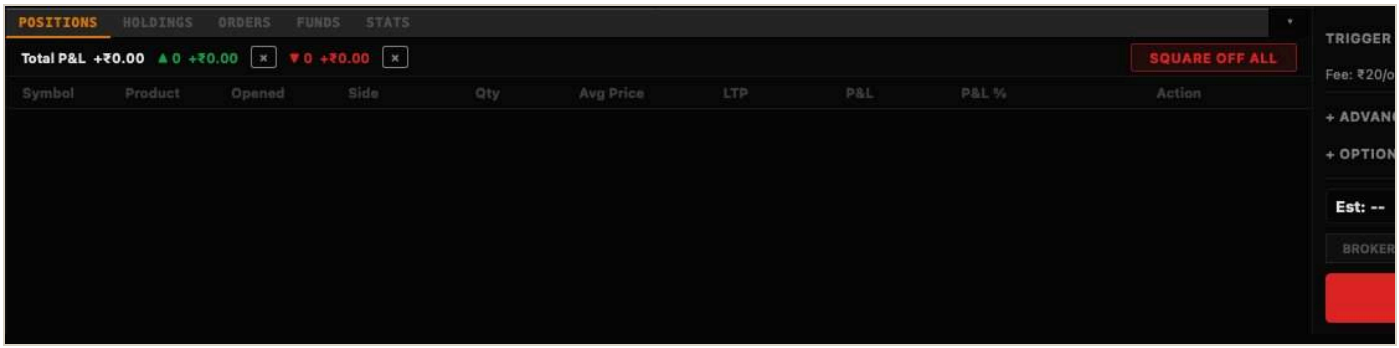


Fig 16 — The Positions tab: the action bar with net P&L, winner and loser chips and **SQUARE OFF ALL**, above the position rows.

Column / control	Meaning
`Symbol · Product · Opened · Side · Qty · Avg Price · LTP · P&L · P&L % · Action`	The ten columns. LTP and both P&L columns are patched live per tick rather than rebuilt
`Total P&L`	Net across open positions, in the account's currency
`▲ winners / ▼ losers`	Counts, each with its own group square-off button — close all the winners, or all the losers, in one confirmed action
`SQUARE OFF ALL`	Closes every open position, after a confirmation
`Action ▸ SELL`	Opens the ticket pre-filled with the held quantity
`Action ▸ → CNC`	On an intraday row, converts the position to delivery in place
`Right-click`	Buy (add) or Sell (reduce) the row's symbol
`Row click`	Loads that symbol into the chart

Holdings

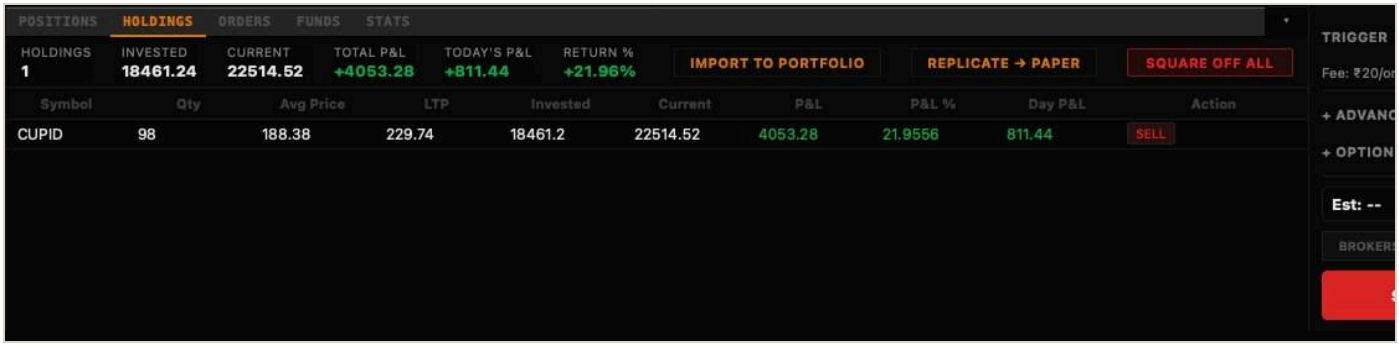


Fig 17 — The Holdings tab, with its own summary strip and bulk actions.

Column / control	Meaning
`Symbol · Qty · Avg Price · LTP · Invested · Current · P&L · P&L % · Day P&L · Action`	The ten columns, live-patched like Positions
`Summary strip`	Holding count, invested, current value, P&L, P&L % and today's P&L
`IMPORT`	Brings the broker's holdings into a Fincept portfolio for analysis
`REPLICATE`	Copies this portfolio into a paper account
`SQUARE OFF ALL`	Market-sells **every holding** — delivery only; open positions are untouched
`Action ▸ SELL`	Closes that single holding, after a confirmation

> **Two different "square off all" buttons.** The one on Positions closes intraday and margin positions; the one on Holdings sells your delivered stock. They act on different books and neither touches the other — read the tab heading before you press either.

13 Blotter: Orders, Funds and Stats

Orders

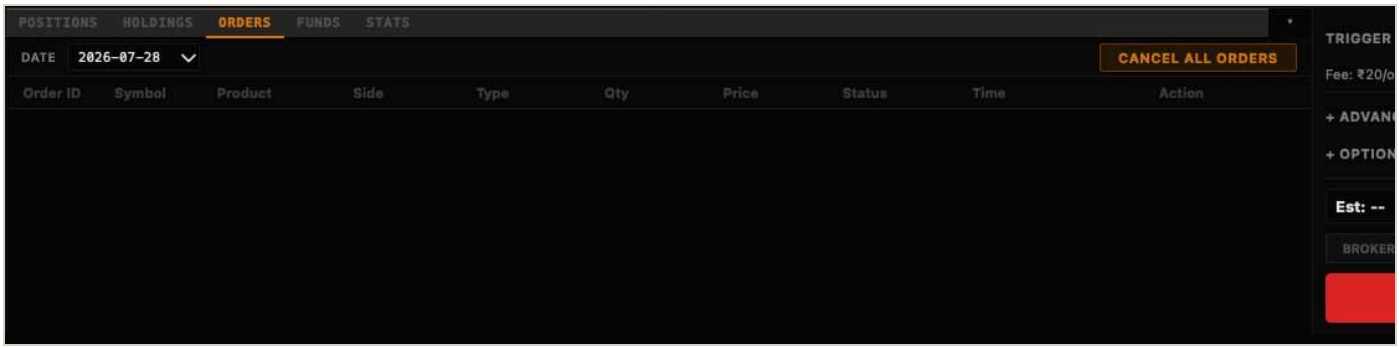


Fig 18 — The Orders tab with its day selector and bulk cancel.

Column / control	Meaning
`Order ID · Symbol · Product · Side · Type · Qty · Price · Status · Time · Action`	The ten columns
`Date selector`	Shows a chosen session's order book — the paper book keeps history, so you can look back at a past day. Positions and holdings stay current
`Action ▸ cancel / modify`	Cancel a pending order, or modify its quantity, price, type or trigger
`CANCEL ALL ORDERS`	Cancels every open order on the account, after a confirmation
`Right-click`	Fetches that order's full state history and fills from the broker

Funds — nine cards

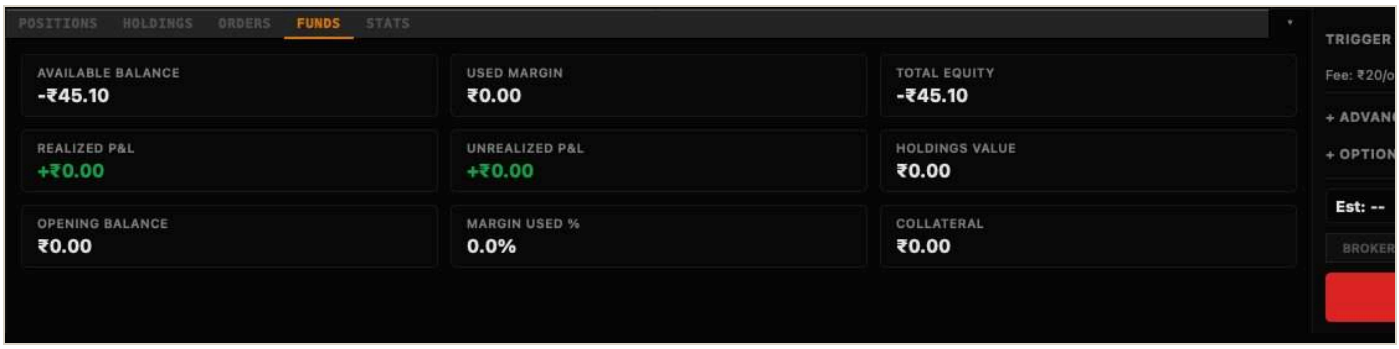


Fig 19 — The Funds tab. Live accounts fill the subset the broker reports; paper accounts fill the full set from the paper engine.

AVAILABLE BALANCE · USED MARGIN · TOTAL EQUITY · REALIZED P&L · UNREALIZED P&L · HOLDINGS VALUE · OPENING BALANCE · MARGIN USED % · COLLATERAL

Stats — twelve cards

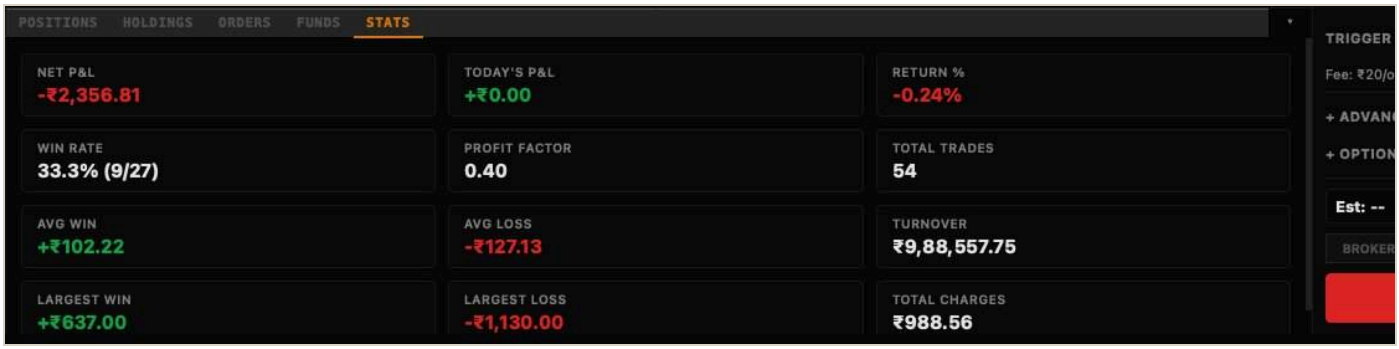


Fig 20 — The Stats tab: the trading record for this account.

NET P&L · TODAY'S P&L · RETURN % · WIN RATE · PROFIT FACTOR · TOTAL TRADES · AVG WIN · AVG LOSS · TURNOVER · LARGEST WIN · LARGEST LOSS · TOTAL CHARGES

> **The whole sheet collapses.** The ▾ control at the right of the tab strip folds the blotter down to its tab bar, giving the chart the full height — and restores the previous split when you expand it again.

14 US brokers, feeds and the unified monitor

Three extra tabs for US brokers

When the focused account is a US-market broker the blotter grows from five tabs to eight. They are hidden for other brokers because only US venues supply the data.

Fig 21 — Eight tabs with a US broker focused.

Fig 22 — Five tabs with an Indian broker focused.

Tab	Columns	What it is
TIME & SALES	Time · Price · Size · Exchange · Conditions · Tape	The trade tape for the selected symbol, appended live, with the exchange condition codes spelled out
AUCTIONS	Date · Type · Time · Price · Size · Exchange	Opening and closing auction prints
CALENDAR	Date · Open (ET) · Close (ET) · Pre-Market · After-Hours	Session times, including half-days, plus a market-status banner

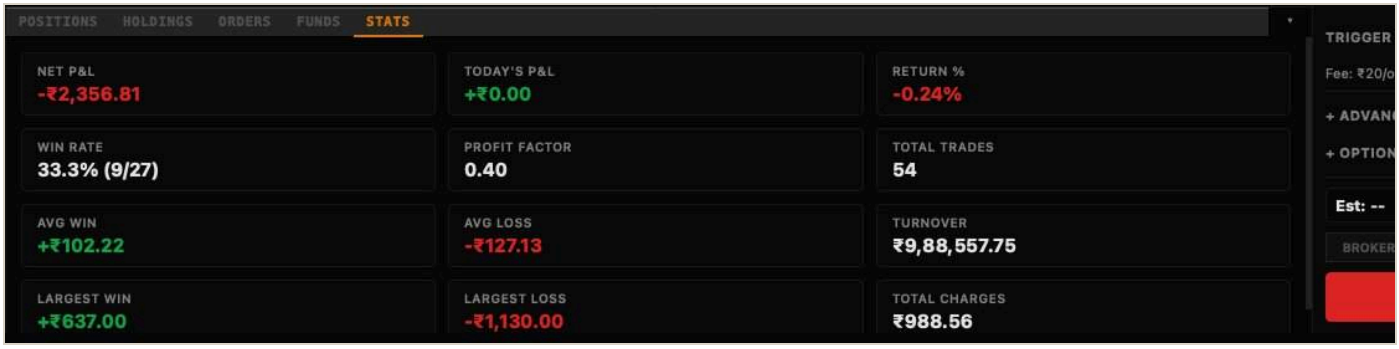


Fig 23 — The Time & Sales tab.

The two side columns

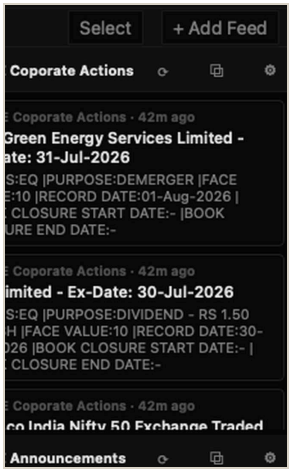


Fig 24 — The feed column.

Column	What it gives you
FEEDS	Up to three chosen news or filings feeds beside the desk — corporate actions, announcements, annual reports. **Select** picks which feeds; the choice and the column width are remembered
UNIFIED	The all-accounts portfolio monitor: positions and holdings from *every* connected account in one list, rather than the focused one. It can be **detached into its own window** (and closing that window re-docks it), which suits a second monitor

> **Why the unified column exists.** The rest of this tab is deliberately single-account — the ticket has to belong to one broker. UNIFIED is the answer to “what do I actually hold across all four accounts”, without leaving the order desk.

15 The F&O desk

Options and futures get their own screen. It is not a separate product — it uses the **same broker accounts** you configured in section 04, sends orders down the **same path** as the ticket in section 09, and its positions land in the **same paper blotter**. What it adds is the surface an options desk needs and a price ladder cannot give you: a full chain, a strategy builder with a payoff curve, open-interest analytics, and institutional flow.

Fig 30 — The Equity command bar's right-hand end. **FO** opens the F&O screen with this tab's context.

Opening it

Route	How
From this tab	The **FO** button at the right of the command bar — the intended route
Saved layout	**View ▸ Layouts ▸ Options & F&O Desk** puts the chain in the centre with pricing, the vol surface, this tab and the portfolio around it
Navigate menu	**Not there.** F&O is deliberately absent from the toolbar nav and from the command palette — it needs a parent tab's context, so a standalone entry point would land you on an empty screen

Fig 31 — Seven sub-tabs across the top. Each is built the first time you open it, so the first switch to a chart-heavy tab takes a moment.

Sub-tab	What it is
CHAIN	The live option chain for one underlying and expiry — the work surface
BUILDER	Multi-leg strategy construction with a payoff curve and Greeks
OI	Open-interest analytics: OI by strike, max pain, build-up, intraday OI
MULTI-STRA	Synthetic straddle / strangle premium through the day
FII / DII	Daily institutional buy, sell and net flows for the Indian market
SCREENER	The same chain, filtered by IV, distance from ATM and open interest
POSITIONS	The shared paper blotter — every paper position, order and trade, with square-off

> **It needs a connected broker.** With none, the broker picker offers only `databento` and every panel reads *Pick a broker, underlying, and expiry*. The chain is assembled from your broker's own quotes, so an account has to be connected and its NFO instrument master loaded first — the screen downloads that itself the first time, showing *Loading … instruments* while it does. US options come from **Databento** instead, which needs an API key in **Settings ▸ Credentials**.

16 The option chain

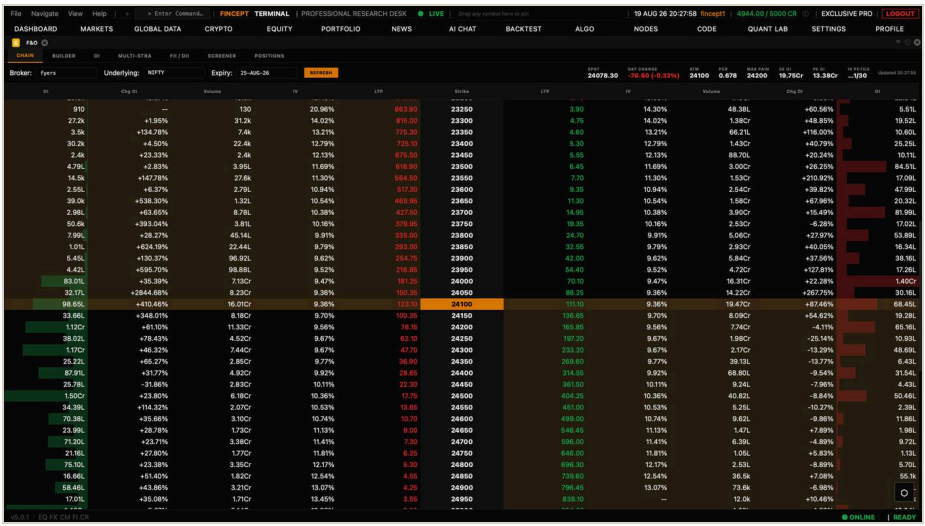


Fig 32 — The chain on a live fyers account: NIFTY, 25-AUG-26. Calls on the left, puts on the right, the strike ladder down the middle, and the ATM row picked out in amber.

The three pickers

Fig 33 — Broker, underlying, expiry, and a REFRESH that forces a re-fetch rather than waiting for the next scheduled poll.

Control	Behaviour
Broker	Every connected account's broker, plus `databento` last. It re-points itself at a real broker as soon as one connects, and remembers an explicit choice
Underlying	**Type to search** — it is an editable box with a contains-match completer over every F&O name your broker lists. Changing it also publishes the symbol to the tab's link group, so a linked watchlist or research panel follows
Expiry	Weeklies first, then monthlies and the long-dated series. For Dhan and Fyers the list comes from the broker's own expiry API, because those two reject any expiry they did not advertise
REFRESH	Forces an immediate re-fetch of this chain

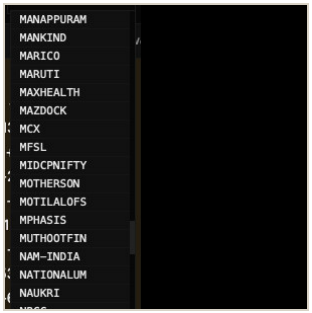


Fig 34 — The underlying list, top: every NSE F&O name, indices and stocks together.

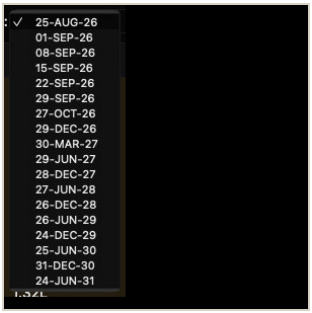


Fig 35 — Expiries, nearest first — weeklies, then monthlies, then the long-dated series.

The chain — the ribbon and the ladder

The eight-value ribbon

The right of the header carries the numbers you would otherwise compute, refreshed with the chain and stamped *Updated hh:mm:ss*.

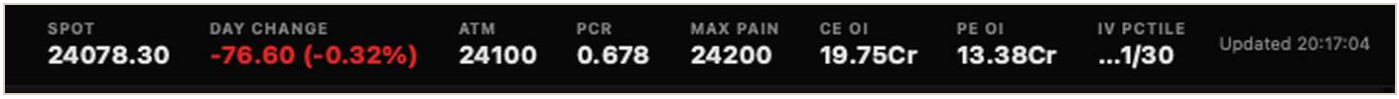


Fig 36 — Spot, day change, ATM, PCR, max pain, total CE and PE open interest, and an IV percentile.

Read-out	What it is
SPOT	The underlying's last price, from the broker's quote
DAY CHANGE	Its absolute and percentage move, green up, red down
ATM	The strike nearest spot — the row highlighted in the ladder
PCR	Put-call ratio across the whole chain, by open interest
MAX PAIN	The strike at which the least option value would be paid out at expiry
CE OI · PE OI	Total call and put open interest, compacted to k / L / Cr
IV PCTILE	Where today's ATM implied vol ranks against the last 90 days. It needs **30 days of history** before it can answer, and counts up — `…12/30` — until then

> **The IV percentile is built on this machine.** Daily ATM IV is recorded locally as you use the tab, so a fresh install starts at zero and climbs; it is not downloaded.

Reading the ladder

4.42L	+595.70%	98.88L	9.52%	216.85	23950	54.40	9.52%	4.72Cr	+127.81%	17.26L
83.01L	+35.39%	7.13Cr	9.47%	181.25	24000	70.10	9.47%	16.31Cr	+22.28%	1.40Cr
32.17L	+2844.68%	8.23Cr	9.36%	150.35	24050	88.25	9.36%	14.22Cr	+267.75%	30.16L
98.65L	+410.46%	16.01Cr	9.36%	123.10	24100	111.10	9.36%	19.47Cr	+87.46%	68.45L
33.66L	+348.01%	8.18Cr	9.70%	100.35	24150	136.65	9.70%	8.09Cr	+54.62%	19.28L
1.12Cr	+61.10%	11.33Cr	9.56%	78.15	24200	165.85	9.56%	7.74Cr	-4.11%	65.16L
38.02L	+78.43%	4.52Cr	9.67%	62.10	24250	197.20	9.67%	1.98Cr	-25.14%	10.93L

Fig 37 — Rows around the money. The amber row is the ATM strike; in-the-money cells carry a warm tint; the OI columns are shaded by size, calls growing leftward in green and puts rightward in red.

Column (each side)	Meaning
OI	Open interest, with a proportional bar behind it
Chg OI	Change in open interest, signed percentage
Volume	Contracts traded today
IV	Implied volatility. Dhan and Fyers supply it; for other brokers the terminal solves it in-process from the premium
LTP	Last traded premium, coloured by the day's move
Strike	The pivot column, bold and centred

> **The chain opens centred on the money and stays there.** A refresh that keeps the same strikes merges in place, so your scroll position survives; only a new underlying or expiry re-centres, and live ticks patch single cells.

17 Trading from the chain

Two gestures on a chain cell, and they do very different things.

Gesture	Result
Left-click an LTP cell	Adds that leg to the **Builder** at one lot. Nothing is ordered — it is a construction step, and the cheapest way to assemble a custom structure
Right-click any call or put cell	Opens a three-item menu: **Buy**, **Sell**, or **Add to Builder**

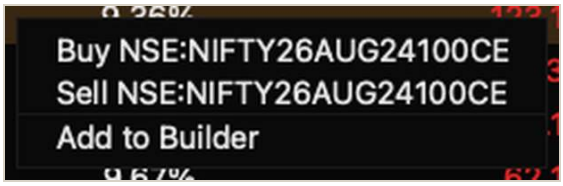


Fig 38 — The right-click menu, labelled with the broker's own contract symbol.

The order ticket

Buy or Sell opens a small ticket rather than firing immediately. It resolves the contract, its lot size and its last premium from the chain snapshot, and asks for four things.

Field	Detail
Type	**Market** or **Limit**. The price row is hidden unless you pick Limit
Qty (lots)	In lots, with the resulting quantity spelled out beneath — *= 1,800 qty (lot size 75)* — so the number you send is never a surprise
Limit price	Pre-filled with the current premium
Product	**NRML (positional)** or **MIS (intraday)**

A footer line names the account, the last price and the mode — `[PAPER]` or `[LIVE]` — and the confirm button reads **Place Buy** or **Place Sell**.

> **This is a live order path.** If the focused account's trading mode is LIVE, pressing Place sends the order to your broker. The ticket is the confirmation — there is no second one. Check the `[PAPER]` / `[LIVE]` tag in the footer before you press it, exactly as you would check the mode chip on the Equity ticket (section 05).

> **Paper and live store the symbol differently, on purpose.** A live order goes out in the broker's native symbol; a paper order is stored under the bare contract name so the position marks to market against the live quote feed and lands in the same account portfolio as everything else — which is what lets one blotter and one square-off cover the whole desk.

18 The strategy builder

The Builder turns a set of legs into a picture: a payoff curve, breakevens, the Greeks and a probability of profit — all computed on your own machine, instantly, and free however many times you change your mind.

Fig 39 — An Iron Condor built from the live NIFTY chain: four legs, the analytics ribbon above, the payoff below.

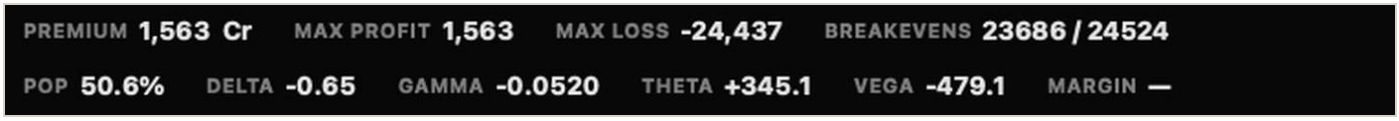


Fig 40 — The ten read-outs. The top row is the trade; the bottom row is its risk.

Read-out	Meaning
PREMIUM	Net premium of the structure — paid if positive, received if it reads as a credit
MAX PROFIT · MAX LOSS	The best and worst outcome of the payoff curve at expiry
BREAKEVENS	Where the curve crosses zero — one figure per crossing
POP	Probability of profit, from the current spot, time to expiry and implied vol
DELTA · GAMMA · THETA · VEGA	Aggregated across the active legs
MARGIN	Blank until a basket margin comes back from the broker or the risk model runs

The builder — templates and legs

Twenty templates

Fig 41 — The template strip: the catalogue, then W, S and L, then + ADD LEG and USE.

Fig 42 — The catalogue, grouped by outlook: six bullish, six bearish, four neutral, two volatility and two others.

On	B/S	Type	Strike	Lots	Entry	IV	LTP	Delta	P&L	
☑	SELL	PE	23700	-1	14.80	10.38%	14.95	-0.10	-9.75	×
☑	BUY	PE	23300	1	4.25	14.02%	4.75	-0.03	32.50	×
☑	SELL	CE	24500	-1	17.68	10.36%	17.75	0.11	-4.87	×
☑	BUY	CE	24900	1	4.17	13.07%	4.25	0.03	4.88	×

Fig 43 — The leg table after USE. Lots and Entry are editable in place; the checkbox drops a leg out of the maths without deleting it; ✕ removes it.

Control	What it does
TEMPLATE	Picks one of the twenty structures. Each is a recipe of legs at offsets from the ATM strike
W	Width — how many strikes apart the wings sit. Only the spreads, condors, butterflies and strangles use it
S	Shift — moves the whole structure up or down the ladder, for a directional lean
L	Lots — multiplies every leg
USE	Instantiates the template against the current chain, replacing whatever is in the table
+ ADD LEG	Appends a blank leg you fill in yourself

Outlook	Templates
Bullish (6)	Long Call · Short Put · Bull Call Spread · Bull Put Spread · Call Ratio Back Spread · Synthetic Long
Bearish (6)	Long Put · Short Call · Bear Put Spread · Bear Call Spread · Put Ratio Back Spread · Synthetic Short
Neutral (4)	Iron Condor · Iron Butterfly · Short Straddle · Short Strangle
Volatility (2)	Long Straddle · Long Strangle
Others (2)	Call Butterfly · Put Butterfly

Leg column	Detail
On	Include this leg in the payoff and the Greeks
B/S	Derived from the sign of Lots — a negative quantity is a sell
Type · Strike	CE or PE, and the strike, from the chain
Lots · Entry	**Editable.** Change either and everything above recomputes
IV · LTP · Delta · P&L	Live from the chain: implied vol at entry, current premium, the leg's delta and its running P&L

The builder — the payoff and saving

> **A template out of range is refused, not fudged.** Ask for wings the chain does not reach — a large W near the end of the ladder — and the Builder says so and leaves your legs alone, rather than silently substituting the nearest strike.

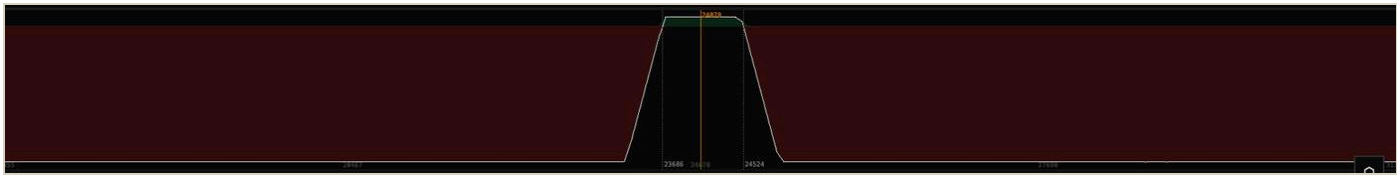


Fig 44 — The payoff curve for the condor above. Green is profit, red is loss, the vertical line is spot, and the two labelled crossings are the breakevens.

The controls under it

Fig 45 — SAVE, the charged RISK PROFILE, LOAD and the target-day spinner.

Control	What it does
SAVE	Stores the structure by name. Free — describing what you want is never the charge
LOAD ▾	Re-opens a saved structure, or deletes one (with a confirmation). Empty on a fresh install
TARGET +*n* d	Draws a second, dashed curve for the P/L *n* days from today rather than at expiry. 0 = today
RISK PROFILE · 2 CR	The one charged control on the whole screen — section 19
PCR · CE OI · PE OI	The same chain totals as the header, repeated where you are working
TRADE ALL (PAPER)	Sends every active leg as a paper order — section 19

> **The payoff draws across the whole strike ladder.** A tight structure — a condor one strike wide — therefore renders as a narrow spike with flat wings either side, because the x-axis spans every strike in the chain rather than the interesting range. Widen **W**, or read the breakevens in the ribbon instead of eyeballing the curve.

> **Everything in this section is free and instant.** Dragging a leg, retyping a quantity, switching template, moving the target day and redrawing the payoff and Greeks all happen on your machine. Only the server risk model and actual orders cost anything.

19 The risk model, and trading a structure

RISK PROFILE — 2 credits

Everything the Builder shows is computed locally at one valuation. **RISK PROFILE** asks the server for the part it cannot do: a two-dimensional scenario grid across spot *and* volatility, Greeks aggregated at a common valuation, and a margin estimate taken from the worst cell of that grid.

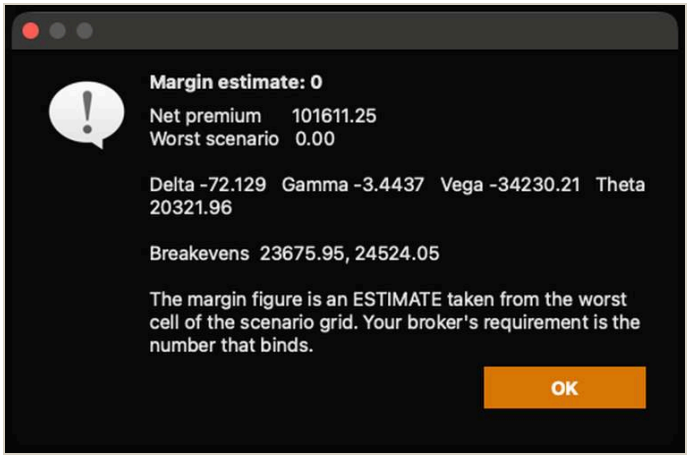


Fig 46 — The result. Net premium, worst scenario, aggregated Greeks and the breakevens, with the margin caveat stated in the dialog itself.

Fig 47 — The tab's own price list, opened from the ⓘ beside the button. Two things in F&O cost credits; nothing else on the screen does.

	Detail
Price	**2 CR**, fixed, printed on the button. Measured: the balance moved 4,946.00 → 4,944.00 on one run
What is free	Building the structure, dragging strikes, refreshing the chain, and redrawing the payoff and Greeks — however many times
Refusals	A run that produces no profile is not charged; a credit refusal opens the top-up sheet instead of failing silently

Read the margin figure as an estimate. The dialog says so itself: it comes from the worst cell of the scenario grid, and your broker's requirement is the number that binds. On the run photographed above it came back as **0** with a zero worst-case, which is not a usable margin — treat a zero here as “no answer”, not as “no margin needed”, and get the real number from the basket-margin line in the order dialog below.

The server Greeks are on a different scale from the ribbon's. The ribbon shows the structure's Greeks per share at the current valuation; the server model returns them aggregated across the whole position. Both are right; they are not the same number, and comparing them directly will look like a bug when it is not.

Trading a structure

Fig 48 — The right-hand end of the footer. TRADE ALL is disabled until at least one active leg and a loaded chain exist.

TRADE ALL (PAPER) sends every ticked leg as a market order into the **same paper account portfolio** as the chain's own Buy and Sell — one blotter, one square-off, one set of positions. It opens a confirmation first, *Confirm Paper Orders*, which is the only place the whole basket is priced together.

In the dialog	What it shows
Header	The structure's name, its underlying, its expiry and the current spot
Legs table	Symbol · B/S · Qty · Type · Entry — one row per leg that will actually be sent
Net Premium	What the basket costs or collects
Max Profit / Loss	Carried through from the Builder
Basket Margin	Requested from the broker for the whole structure, with initial and post-netting figures in its tooltip. This is the real margin number
Cost line	The per-fill paper charge, stated before you place
Buttons	**CANCEL · PLACE PAPER ORDERS**

> **Legs are sent one at a time, and the result is reported honestly.** If some fill and others do not, the message names how many of how many went through and why each failure failed, rather than reporting the basket as placed.

> **Paper fills cost 0.05 credits each.** A four-leg condor is four fills — 0.2 credits — flat, whatever the quantity. Placing an order that does not fill, modifying it or cancelling it is free, and **live broker orders are free too**, because those route through your own broker rather than the terminal's simulator.

20 Open interest and straddles

The **OI** sub-tab is four views of the same chain, arranged in resizable quadrants.

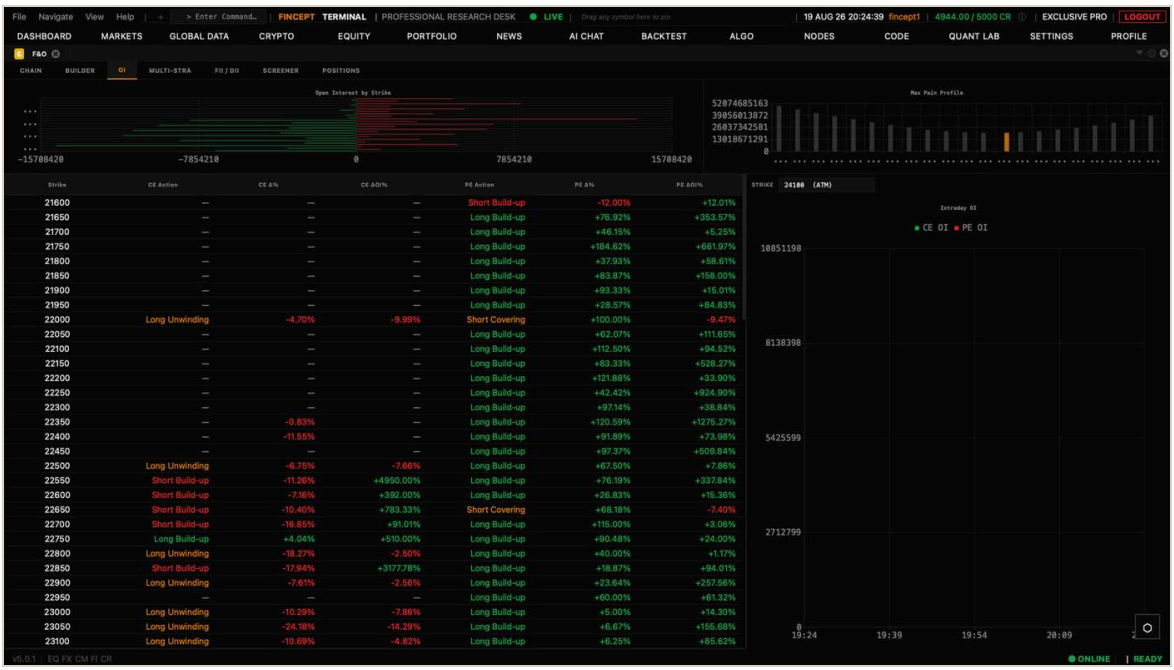


Fig 49 — OI: open interest by strike and the max-pain profile across the top, the build-up table and an intraday chart for one strike underneath.

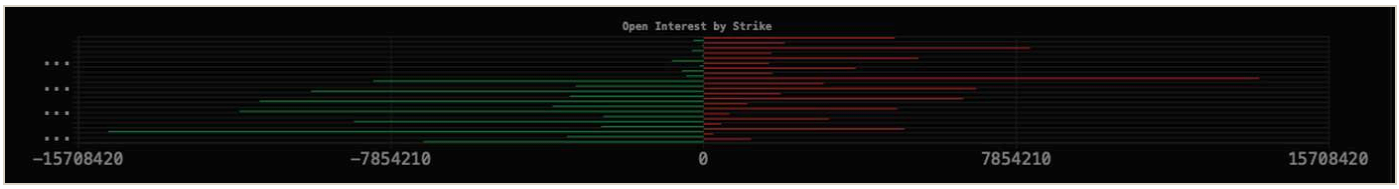


Fig 50 — Open interest by strike. Calls run left in green, puts right in red, so the strikes carrying the most written option are immediately visible.

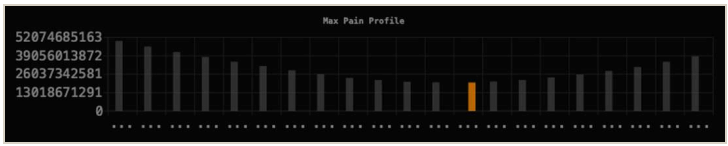


Fig 51 — The max-pain profile: total payout at each strike, with the minimum picked out in amber.

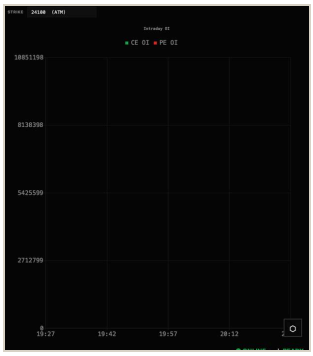


Fig 52 — Intraday OI for the selected strike, CE against PE.

OI — build-up, and the intraday history

Strike	CE Action	CE Δ%	CE ΔOI%	PE Action	PE Δ%	PE ΔOI%
21600	—	—	—	Short Build-up	-12.00%	+12.01%
21650	—	—	—	Long Build-up	+76.92%	+353.57%
21700	—	—	—	Long Build-up	+46.15%	+5.25%
21750	—	—	—	Long Build-up	+184.62%	+661.97%
21800	—	—	—	Long Build-up	+37.93%	+58.61%
21850	—	—	—	Long Build-up	+83.87%	+158.00%
21900	—	—	—	Long Build-up	+93.33%	+15.01%
21950	—	—	—	Long Build-up	+28.57%	+84.83%
22000	Long Unwinding	-4.70%	-9.99%	Short Covering	+100.00%	-9.47%
22050	—	—	—	Long Build-up	+62.07%	+111.65%
22100	—	—	—	Long Build-up	+112.50%	+94.52%

Fig 53 — The build-up table. Each side of each strike is classified from the sign of its price move against the sign of its OI move.

Classification	Price	Open interest	Reading
Long Build-up	Up	Up	New longs — fresh money going with the move
Short Build-up	Down	Up	New shorts — fresh money going against it
Short Covering	Up	Down	Shorts closing — a squeeze rather than conviction
Long Unwinding	Down	Down	Longs closing — an exit rather than fresh selling

> **The two intraday-history charts are built from snapshots this machine records.** The terminal writes a minute-aligned OI/LTP/volume/IV snapshot to local storage every sixty seconds while a chain is publishing, and keeps seven days. Both the intraday OI chart and the Multi-Straddle chart plot *those* rows — not a server history — so on a freshly connected desk, or after a gap, they show correctly scaled axes with no series on them. That is what Figs 52 and 55 are showing: the desk was photographed after the Indian close, on a broker connected the same hour.

MULTI-STRA — synthetic premium

A straddle is a call and a put at one strike; a strangle is a call above and a put below. This sub-tab plots the **combined premium** of such a pair through the day, so you can watch the price of volatility itself rather than either leg.

Fig 54 — Pick a type and an anchor strike, then ADD. Straddle at the money, or a strangle one, two or three strikes wide.

The four shapes.

MULTI-STRA — comparing structures

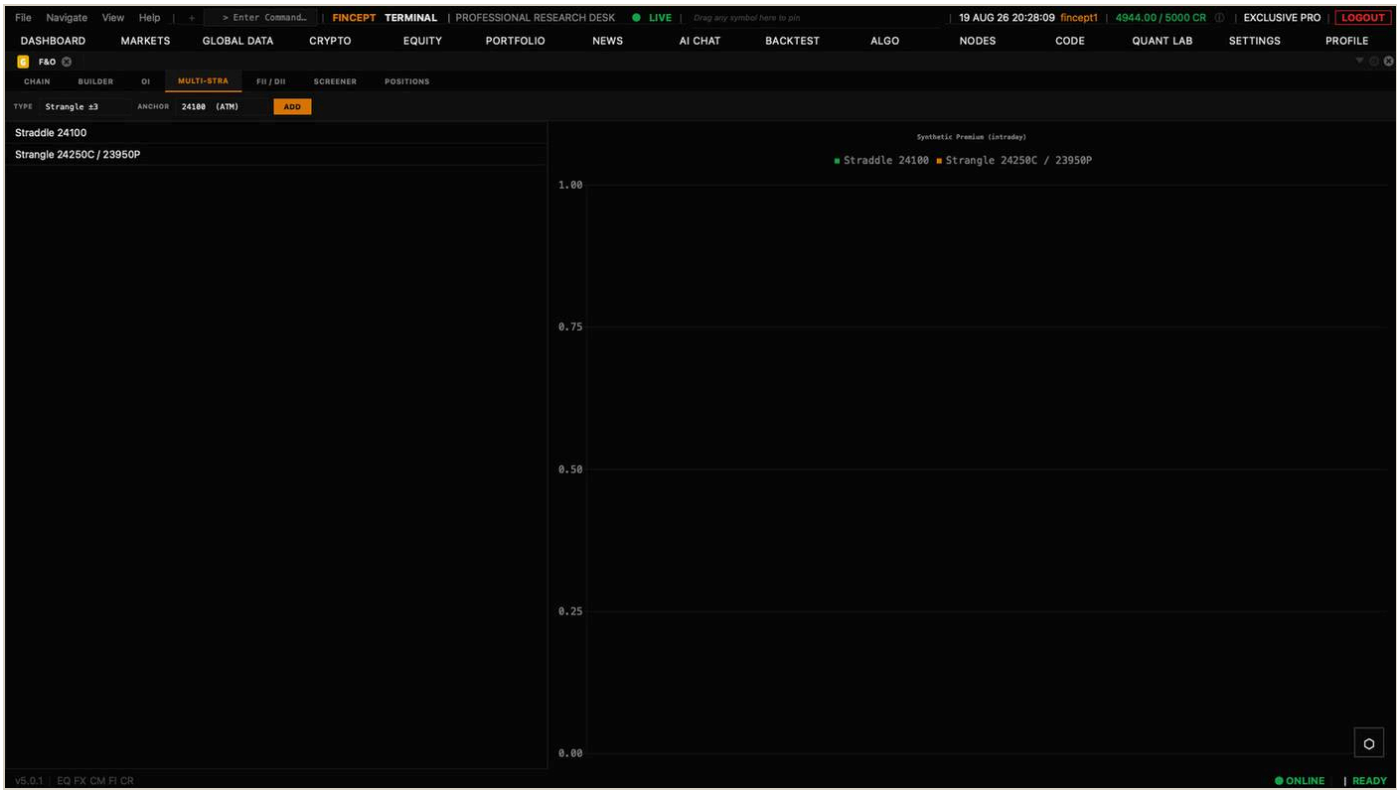


Fig 55 — Two selections added: an ATM straddle and a ±3 strangle. Each becomes a series on the shared chart; double-click an entry in the list to remove it.

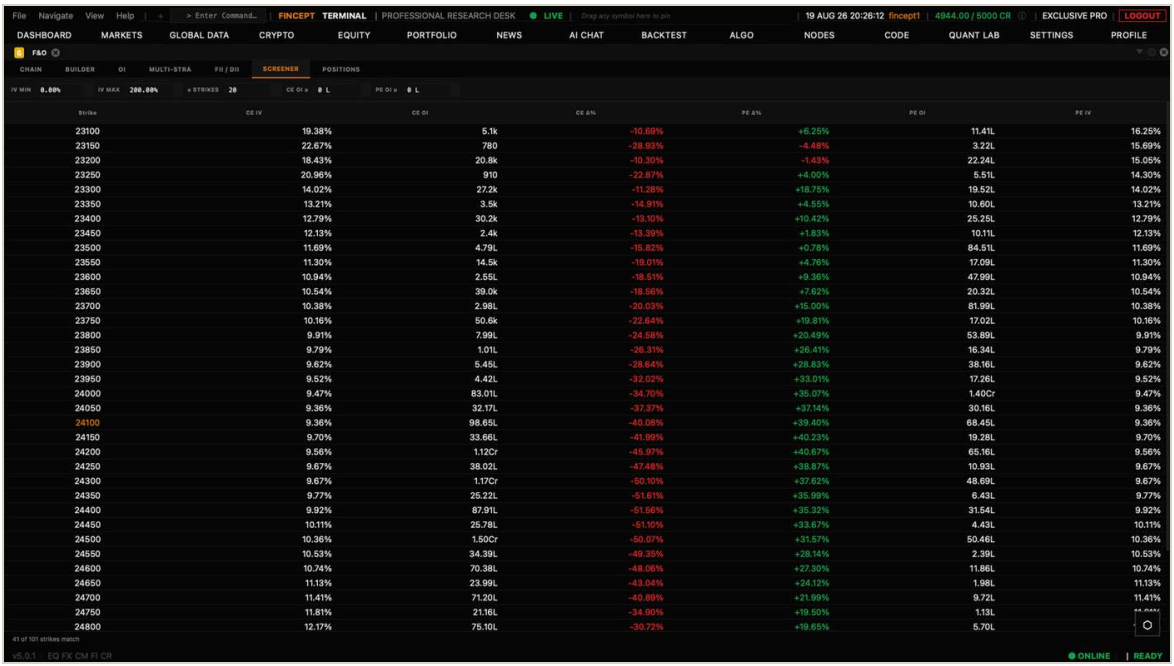
Control	Behaviour
TYPE	Straddle (ATM) · Strangle ±1 · Strangle ±2 · Strangle ±3
ANCHOR	The strike the pair is built around; the ATM row is marked and selected by default
ADD	Adds the pair to the list and subscribes to both legs' histories. Duplicates are ignored
The list	One row per selection. **Double-click to remove** — there is no delete button

> **Selections are dropped when the underlying changes.** A straddle on one index means nothing on another, so switching the chain's underlying clears the list rather than leaving stale rows that would silently plot the wrong contracts.

21 Screening, flows, positions and cost

SCREENER — the chain, narrowed

The same rows as the chain, reduced to the strikes worth looking at, with a count of how many survived.



Strike	CE IV	CE OI	CE Δ%	PE Δ%	PE OI	PE IV
23100	19.38%	5.1k	-10.69%	+6.25%	11.41L	16.25%
23150	22.67%	780	-28.93%	-4.48%	3.22L	15.69%
23200	18.43%	20.8k	-10.30%	-1.43%	22.24L	15.05%
23250	20.96%	910	-22.87%	+4.00%	5.51L	14.30%
23300	14.02%	27.2k	-11.28%	+18.75%	19.52L	14.02%
23350	13.21%	3.5k	-14.91%	+4.55%	10.60L	13.21%
23400	12.79%	30.2k	-13.10%	+10.42%	25.25L	12.79%
23450	12.13%	2.4k	-13.39%	+1.83%	10.11L	12.13%
23500	11.69%	4.79L	-15.82%	+0.78%	84.51L	11.69%
23550	11.30%	14.5k	-19.01%	+4.76%	17.09L	11.30%
23600	10.94%	2.55L	-18.51%	+9.36%	47.99L	10.94%
23650	10.54%	39.0k	-18.56%	+7.62%	20.32L	10.54%
23700	10.38%	2.98L	-20.03%	+15.00%	81.99L	10.38%
23750	10.16%	50.6k	-22.64%	+19.81%	17.02L	10.16%
23800	9.91%	7.99L	-24.58%	+20.49%	53.89L	9.91%
23850	9.79%	1.01L	-26.31%	+26.41%	16.34L	9.79%
23900	9.62%	5.45L	-28.64%	+28.83%	38.16L	9.62%
23950	9.52%	4.42L	-32.02%	+33.01%	17.26L	9.52%
24000	9.47%	83.01L	-34.70%	+35.07%	1.40Cr	9.47%
24050	9.36%	32.17L	-37.37%	+37.14%	30.16L	9.36%
24100	9.36%	98.65L	-40.08%	+39.40%	68.45L	9.36%
24150	9.70%	33.66L	-41.99%	+40.23%	19.28L	9.70%
24200	9.56%	1.12Cr	-45.97%	+40.67%	65.16L	9.56%
24250	9.67%	38.02L	-47.48%	+38.87%	10.93L	9.67%
24300	9.67%	1.17Cr	-50.10%	+37.62%	48.69L	9.67%
24350	9.77%	25.22L	-51.61%	+35.99%	6.43L	9.77%
24400	9.92%	87.91L	-51.56%	+35.32%	31.54L	9.92%
24450	10.11%	25.78L	-51.10%	+33.67%	4.43L	10.11%
24500	10.36%	1.50Cr	-50.07%	+31.57%	50.46L	10.36%
24550	10.53%	34.39L	-49.35%	+28.14%	2.39L	10.53%
24600	10.74%	70.38L	-48.06%	+27.30%	11.86L	10.74%
24650	11.13%	23.99L	-43.04%	+24.12%	1.98L	11.13%
24700	11.41%	71.20L	-40.89%	+21.99%	9.72L	11.41%
24750	11.81%	21.16L	-34.90%	+19.50%	1.13L	[illegible]
24800	12.17%	75.10L	-30.72%	+19.65%	5.70L	[illegible]

Fig 56 — 41 of 101 strikes matching the default filters. The ATM strike stays marked in amber.

Fig 57 — Five filters, applied as you change them.

Filter	What it keeps
IV MIN · IV MAX	Strikes whose higher side sits inside the implied-vol band
± STRIKES	How far from the money to look — the default 20 keeps 41 rows of a 101-strike ladder
CE OI ≥ · PE OI ≥	A floor on open interest, in lakhs, so illiquid strikes drop out

Fig 58 — Narrowing ± STRIKES to 6 leaves thirteen rows. The count is always in the footer.

Column	Meaning
Strike	Amber when it is the ATM row
CE IV · PE IV	Implied volatility each side
CE OI · PE OI	Open interest each side
CE Δ% · PE Δ%	The day's premium move each side, green up, red down

> **The filters persist.** They are part of the screen's saved state, so a band you set today is still set tomorrow — worth knowing if the table looks emptier than you expect.

FII / DII flows, and the paper blotter

FII / DII — institutional flows

Daily buy, sell and net figures for foreign and domestic institutions in the Indian market, in crore. It is the one sub-tab that needs no chain and no broker.

Date	FII Buy	FII Sell	FII Net	DII Buy	DII Sell	DII Net
2026-08-18	—	—	+1,651.53	—	—	+2,579.31
2026-08-17	—	—	-2,535.10	—	—	+5,101.46
2026-08-14	—	—	+508.12	—	—	+356.40
2026-08-13	—	—	-510.69	—	—	+4,353.09
2026-08-12	—	—	-1,002.50	—	—	+5,841.66
2026-08-11	—	—	+258.55	—	—	+24.77
2026-08-10	—	—	+1,974.76	—	—	-1,290.29
2026-08-07	—	—	+480.24	—	—	+235.56
2026-08-06	—	—	-17.86	—	—	+4,013.60
2026-08-05	—	—	-943.42	—	—	+2,883.17
2026-08-04	—	—	+2,446.47	—	—	-936.14
2026-08-03	—	—	+922.26	—	—	+1,571.18

Fig 59 — Newest day at the top. Net columns are coloured; the chart beside the table plots FII against DII net flow for the same days.

	Detail
Where it comes from	A free public daily-flow feed, cached locally. No credits, no broker, no key
How current	The header states the last date it holds and how many days are cached. Figures land after the close — the tab says so if you refresh too early
REFRESH	Forces a re-fetch rather than waiting for the cache to expire

POSITIONS — the shared paper blotter

Fig 60 — Positions · Orders · Trades, and SQUARE OFF ALL.

This is not an F&O-only view: it is the **same paper blotter** the rest of the terminal uses, showing every paper position, working order and trade across every active paper account, with account, symbol, product, side, quantity, average, last price and P&L. It marks to market on its own and needs no wiring — which is the point of routing the chain, the Builder and the Equity ticket all through one path.

> **Its three view buttons are clipped** — they read *Posit…*, *Ord…* and *Tra…* because the buttons are narrower than their labels. They are Positions, Orders and Trades, in that order.

What F&O costs, and its agent tools

Two things on the whole screen charge credits, and the terminal's own price list says so in as many words: *2 things in F&O cost credits. They are listed first below; nothing else on that tab charges anything.*

What	Price	When
Option strategy model	**2 CR** fixed	Each press of **RISK PROFILE** in the Builder
Paper trade	**0.05 CR** per fill	Every paper order that actually fills — from the chain, the Builder or anywhere else
Everything else	Free	The chain, every refresh, the OI analytics, the screener, FII/DII, the payoff, the Greeks, saving a structure, and live broker orders

Eleven agent tools

The screen is fully reachable from AI Chat and from agents, under the `fno` category. All eleven are read-or-compute; **none of them places an order**.

Tool	What it does
list_option_underlyings	Every F&O name tradable through a broker
list_option_expiries	Expiries for one underlying
fetch_option_chain	A full chain: spot, ATM, every strike, OI, IV and greeks
get_cached_option_chain	The last assembled chain, without re-fetching
list_option_strategy_templates	The twenty templates with their categories, outlooks and leg recipes
analyse_option_strategy	A multi-leg structure against the cached chain — premium, breakevens, max profit and loss
screen_option_chain	The Screener's filters, run headlessly
get_oi_analytics	PCR, max pain and build-up for the cached chain
get_oi_history	The minute-aligned intraday history for one leg
get_multi_straddle	Synthetic straddle / strangle premium history
fetch_fii_dii_flows	The institutional daily flows

> **Rehearsing with an assistant is safe here.** Because no F&O tool can trade, an agent can read the chain, screen it, build a structure and price it without any of the approval questions section 22's live trading tools raise. Execution stays on the buttons.

22 Data, persistence and agent tools

Where the numbers come from

Stream	Cadence	Notes
`Quotes`	Live, per tick	One stream per account feeds the ribbon, watchlist, depth, position rows and the ticket's LTP, so they always agree
`Subscription set`	On change	The active watchlist ∪ open-position symbols ∪ holding symbols — so anything you hold is priced even when it is not on the list
`Portfolio snapshot`	Every few minutes	After you place an order the tab pulls the book immediately and again a moment later, so a fill is not invisible until the next poll
`Paper engine`	Server-side	Placement is a request; the fill, rejection and updated book arrive over the paper stream. A reconciliation pull is scheduled behind every paper command in case a frame is missed
`Option marks`	Live	Held option positions are marked by reconciling the broker's tick symbol with its position symbol — the two spellings differ, and this is what keeps option P&L moving

What the tab remembers

Focused account · selected symbol and exchange · active watchlist · compare pane state and its symbol · the feed and unified columns with their widths · the chosen feeds · and the **entire order ticket** including side, type, quantity, price, trigger, product, exchange, validity and the advanced fields.

Agent tools — 28 of them

Category	Count	What they cover
`live-trading`	17	Symbol search, place / smart / modify / cancel / cancel-all, close position and close-all, positions, holdings, orders, trades, funds, quote, multi-quote, depth, option chain
`paper-trading`	11	Create / list / get / rename / reset / delete a paper portfolio, place and cancel orders, positions, orders and statistics

> **The live tools place real orders.** They are the same path as the button, so the same approval rule applies: an account in Semi-Auto queues an agent's order for your approval in the Action Center, while an account in Auto executes it. If you intend to let an assistant trade, put that account in Semi-Auto first — and rehearse on a paper portfolio, where the eleven paper tools behave identically at no risk.

23 Quick reference

The fifteen questions this desk gets asked most.

Question	Answer
…know whether I am about to trade real money?	The command-bar button and the ticket chip both read **PAPER** or **LIVE**
…add a review step before live orders?	ACCOUNTS ▸ that account ▸ **Order Approval** → **Semi-Auto**
…switch broker?	Click the account button and pick one — the whole screen re-points
…find a symbol?	Type in the command-bar symbol box and choose from the picker
…place a limit at a level in the book?	Click that price in the depth ladder — it fills PRICE and switches to LMT
…attach a stop and target?	+ **ADVANCED** ▸ SL and TP
…rest an order until a price?	+ **ADVANCED** ▸ GTT / Forever, with a TRIGGER — live accounts only
…trade an option structure?	+ **OPTIONS STRATEGY** here, or the full **F&O** desk — sections 15–21
…send one order to several accounts?	**BROKERS** ▸ tick the accounts, then submit
…exit one position?	Its **SELL** action, or **EXIT** on the chart's position card
…close only my losers?	Positions ▸ the ▼ losers chip's square-off button
…convert intraday to delivery?	Positions ▸ → **CNC**; an order's history is a right-click in Orders
…see everything I hold across brokers?	The **UNIFIED** column
…practise safely?	Focus a paper account; every control behaves the same

24 Honest notes on this build

Everything below was verified against the shipping source and on screen. A guide that hides these is no use to a desk that has to trade from the software.

- **In Auto mode — the default — a live order is sent with no confirmation.** Semi-Auto adds the inline review; only entry is immediate, since square-offs and cancels always confirm.
- **The primary chart pane header can lag the traded symbol** after the tab settles on another account; the ticker ribbon and the ticket's symbol line are authoritative.
- **With OPTIONS STRATEGY expanded, PLACE STRATEGY sits below the fold at 1080p.**
- **GTT / Forever is live-only** and needs a trigger price.
- **A market order is refused until the first quote arrives** for the symbol.
- **Product and exchange lists are broker-specific** and are rebuilt when you switch account — an order prepared for one broker may not describe the same thing on another.
- **The watchlist also lists what you hold**, and **Remove** does nothing on those rows.
- **Depth may read** `L1 SYNTH` when the broker sends only top-of-book; the ladder is then synthesised, not real market depth.
- **Fees and margin are estimates** from the broker's rule and margin API, not a contract note, and **portfolio snapshots poll on a multi-minute cadence** — a fill can show up a beat late.
- **The credential form shows the API key in clear text** while you configure an account, and **US-only tabs appear and disappear with the focused broker**.

In the F&O desk

- **The Builder's payoff curve is drawn across the whole strike ladder**, so a one-strike-wide structure renders as a narrow spike between two flat wings. Widen **W** or read the breakevens in the ribbon.
- **The server risk model returned a margin estimate of 0**, with a zero worst case, on the run in section 19. Take the binding margin from the order dialog's basket-margin line.
- **Its Greeks are aggregated across the position** and will not match the Builder ribbon's per-share figures. Both are correct; they answer different questions.
- **The two intraday-history charts** — OI ▸ Intraday OI and Multi-Stra — **plot minute snapshots this machine recorded**, not a server history, so a fresh desk shows scaled axes with no series on them.
- **The Positions view buttons are clipped** to *Posit… / Ord… / Tra…* — the buttons are narrower than their labels.
- **The served price list still describes the risk model as charging "on SAVE".** The shipped build separates them: **SAVE is free**, and the 2 credits are on the **RISK PROFILE** button. The price is right; the path in that description is stale.
- **A chain needs the broker's NFO instrument master**, which the screen downloads on first use — expect a *Loading … instruments* pause the first time you pick a new broker.

> Everything in this guide was read from the shipping source and confirmed on a live four-broker desk. No order was placed, modified or cancelled to produce it.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Crypto Trading

A complete, screen-by-screen walkthrough of the Crypto Trading tab — a full trading workstation over 14 exchanges, with a live chart, order book, ticket and blotter, in paper or live mode.

`>_ 14 native exchanges · chart · order book · depth · tape · order ticket · paper & live · 18 AI tools`

Covers · **the workspace & command bar** · **chart · order book · tape** · **the order ticket** the blotter · **paper & live · connecting an exchange** · AI tools

Demonstrated live on the Fincept Terminal **Crypto Trading** tab.

Contents

11 SECTIONS

GETTING ORIENTED

01 What the Crypto Trading tab is 3

02 Anatomy — the six regions 4

THE WORKSPACE

03 Command bar, exchanges & modes 5

04 The watchlist 6

05 The chart 7

06 Order book, depth & tape 9

07 The order ticket 10

08 The blotter 12

REFERENCE

09 Paper, live & connecting an exchange 13

10 AI tools & the data engine 14

11 Quick reference & honest notes 15

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal, taken on the live Crypto Trading tab streaming Kraken market data (ADA/USDT) in paper mode. Control labels are quoted exactly as they appear on screen.

01 What the Crypto Trading tab is

OVERVIEW

The **Crypto Trading** tab (the **CRYPTO** button on the top tab bar) is a full crypto trading workstation. One screen gives you a live chart, an order book, a depth chart, a time-&-sales tape, an order ticket, and a positions blotter — all streaming from your chosen exchange. It speaks **14 exchanges natively** (Binance, Bybit, OKX, Kraken, Coinbase and more) with no third-party bridge.

You can trade in **paper** mode (a simulated per-exchange book, no keys, no risk) to practice and test, or connect your API keys and trade **live**. Public market data — charts, books, tapes — is always free and keyless; only live order routing needs your keys, which are stored in your operating system's secure keychain.

AT A GLANCE, IN CRYPTO TRADING YOU CAN…

- Stream live prices from **14 native exchanges**.
- Chart any pair with a full indicator & drawing toolkit.
- Read a live **order book**, depth chart and trade tape.
- Place **market, limit, stop and stop-limit** orders.
- Attach **stop-loss / take-profit** brackets.
- Trade perpetuals with **leverage & margin** where supported.
- Practice risk-free in **paper** mode, per exchange.
- Compare two charts side by side; drive it all from **AI chat**.

How to get here

Click `CRYPTO` on the top tab bar. The tab reopens on your last exchange, symbol, mode and layout.

02 Anatomy — the six regions

LAYOUT

The workspace is a command bar across the top and three resizable columns below it. Every divider drags, so you can size each panel to taste.

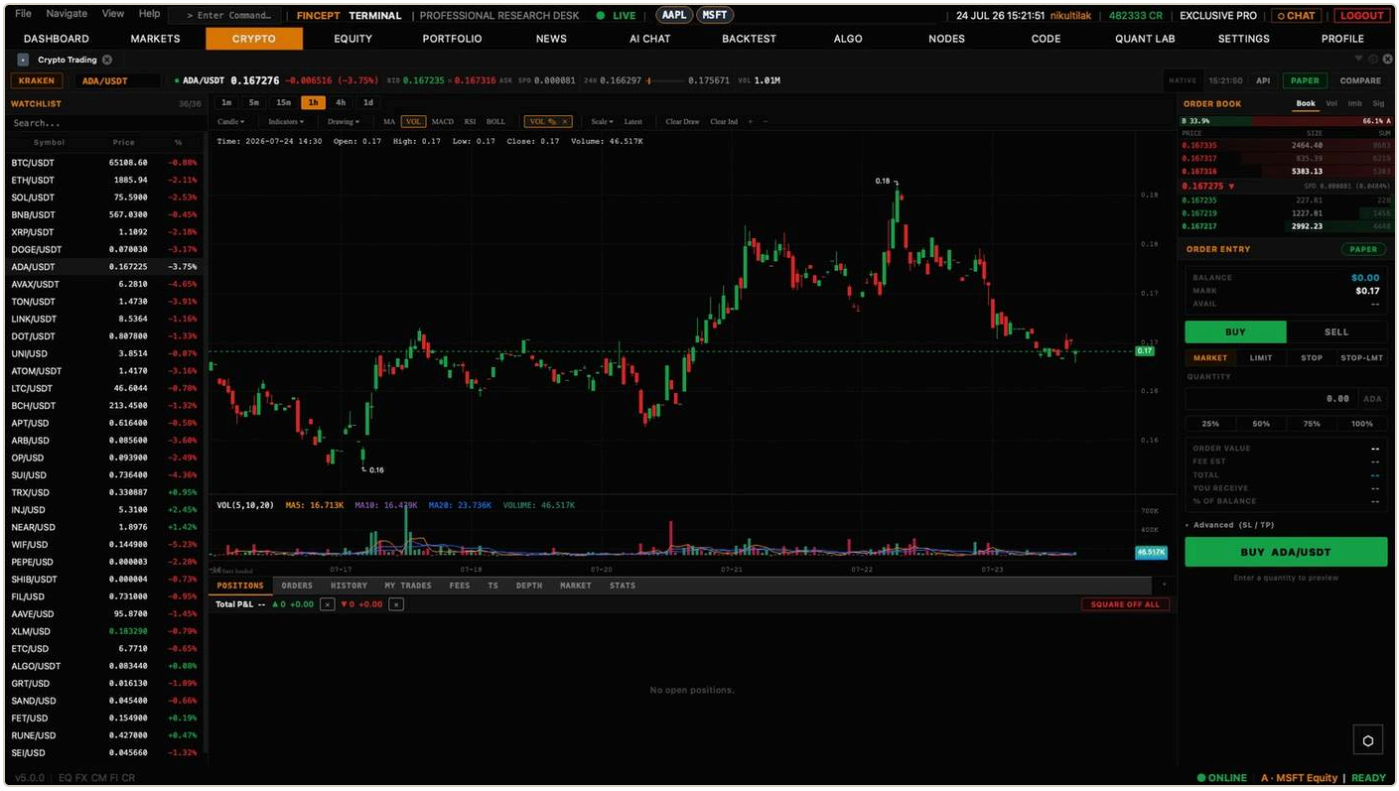


Fig 1. The Crypto Trading workstation — Kraken, ADA/USDT, paper mode. Left: the watchlist. Center: the chart (top) and blotter (bottom). Right: the order book (top) and order ticket (bottom). Across the top: the command bar with the live ticker.

Region	What it is
Command bar	Exchange picker, symbol input, the live ticker ribbon, transport chip, clock, **API**, the **PAPER/LIVE** mode toggle, and **COMPARE** (§3).
Watchlist (left)	Your per-exchange list of pairs with live price & % (§4).
Chart (center-top)	Candles, indicators, drawings, your position line & P&L card; a second COMPARE chart when enabled (§5).
Blotter (center-bottom)	Positions, orders, history, fees, tape, depth, market & stats — nine tabs (§8).
Order book (right-top)	The live ladder with imbalance gauge, mid & spread (§6).
Order ticket (right-bottom)	Buy/sell, order types, sizing, brackets, submit (§7).

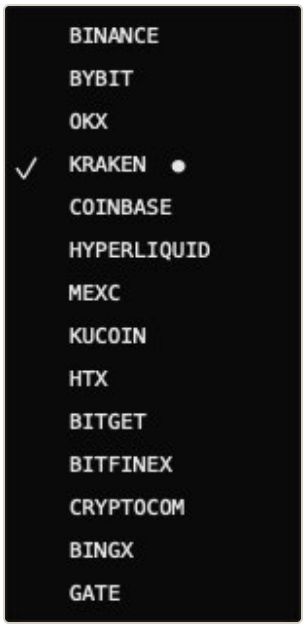
> **ALWAYS-LIVE, WS-FIRST**
>
> Market data streams over WebSockets the moment the tab opens — there's no "refresh." Switching exchanges keeps the previous one warm in the background, so flipping back is instant.

03 Command bar, exchanges & modes

COMMAND BAR

Fig 2. The command bar: exchange · symbol · live ticker · transport · clock · API · mode · COMPARE.

Control	What it does
Exchange (e.g. KRAKEN)	Opens the exchange menu (below). The label is the current venue.
Symbol (e.g. ADA/USDT)	Type any listed pair; a completer suggests from the venue's full catalog. Enter loads it into the focused chart.
Ticker ribbon	Live BID / ASK / SPD / 24H range / VOL for the selected pair (§6).
NATIVE chip	Transport indicator — the exchange is driven by a native connector (not a bridge).
Clock	Live wall clock.
API	Opens the credentials dialog for the current exchange (§9).
PAPER / LIVE	Toggles simulated vs real trading (below).
COMPARE	Shows a second live chart beside the first (§5).



The 14 exchanges

The exchange menu lists every supported venue and rebuilds each time you open it, so the status marks are always current:

- ● — streaming live right now.
- ✓ — API keys saved (but not currently streaming).
- *no mark* — no keys saved.

Pick one to switch; the whole workspace repoints to that venue.

Supported: Binance · Bybit · OKX · Kraken · Coinbase · Hyperliquid · MEXC · KuCoin · HTX · Bitget · Bitfinex · Crypto.com · BingX · Gate.

Paper vs Live

The **PAPER/LIVE** toggle switches the whole ticket and blotter between two worlds. **Paper** is a simulated book — each exchange has its own "Crypto Paper" account, funded with a starting balance, so you can practice on real prices with no keys and no risk. **Live** routes real orders to the exchange; if you flip to LIVE without saved keys, the terminal warns you and stays read-only until you add them (§9). Market depth varies by venue and pair — if a book looks thin, try the exchange's primary quote (some venues quote alts in USD, not USDT, and the terminal auto-remaps common cases like APT/USDT → APT/USD on Kraken).

04 The watchlist

WATCHLIST

The left column is your **per-exchange watchlist** — three columns: **Symbol · Price · %**, with a live count ("36/36") in the header. Prices flash green/red on each tick; the % is colored by direction; the active pair is highlighted.

Working with it

- **Click a row** to load that pair into the focused chart (and drive the book, ticker and ticket).
- **Search** — type in the box to search the venue's full market list.
- **Right-click** a search result to **add** it; right-click a list row to **remove** it.
- **Drag a row out** to any other panel to link that symbol.

Each exchange keeps its own list, saved across restarts. A pair that hasn't traded yet shows `--` until a tick or the periodic backfill fills it in — crypto tickers are trade-driven, so very quiet pairs can sit blank briefly.

> **WHY SOME ROWS READ "--"**
>
> WebSocket tickers only update when a trade prints. An illiquid or delisted pair with no trades has nothing to show until the 30-second REST backfill runs — it's not an error.

05 The chart

CHART

Fig 3. The chart — candles, the OHLC readout, a volume pane with moving averages, the last-price tag, and the dashed entry line when you hold a position.

The chart is a professional candlestick view. Across the top: a **timeframe** strip and a rich toolbar; below the price pane, a **volume** pane with its own moving averages.

Control	What it does
1m 5m 15m 1h 4h 1d	Timeframe (default 1h). Amber = active.
Candle ▾	Chart type: candle solid / stroke / up-stroke, OHLC bars, or area.
Indicators ▾	Overlays (MA, EMA, SMA, Bollinger, Parabolic SAR, VWAP) and sub-panels (Volume, MACD, RSI, KDJ, Williams %R, OBV, DMI and more).
Drawing ▾	Trend/horizontal/ray lines, channels, Fibonacci, shapes, annotations.
MA · VOL · MACD · RSI · BOLL	One-click toggles for the five most-used indicators.
Scale ▾ · Latest	Y-axis scale (normal / %/ log); jump to the latest bar.
Clear Draw · Clear Ind · + −	Remove drawings / indicators; zoom in / out.

Indicators & drawings

The **Indicators** menu splits into **OVERLAY** items (drawn on the candles) and **SUB-PANEL** studies (their own pane below). Each active indicator becomes a chip you can edit (✎ , e.g. change MA periods) or remove (×).

Right-click the chart for **Buy** / **Sell** at the clicked price — it prefills the order ticket with that price. When you hold a position, a dashed **entry line** and a draggable **P&L card** (with an **EXIT** button) appear.

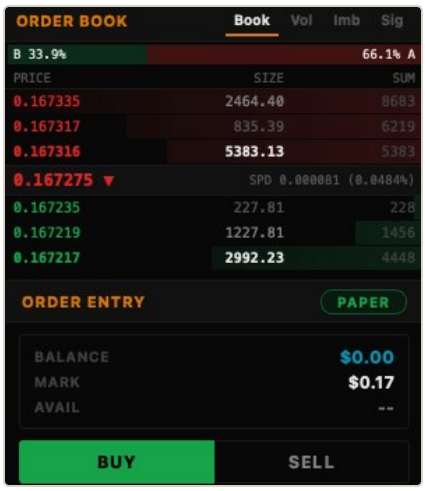
Compare two pairs

Fig 4. COMPARE shows a second live chart beside the first. Click a pane to focus it (the next symbol pick lands there); ◀ **TRADE** promotes the compared pair into the primary, traded slot.

> **GOOD TO KNOW**
>
> The compare chart is **view-only** — trading, the position card and right-click Buy/Sell live on the primary chart. Use ◀ **TRADE** to swap a compared pair into the primary slot when you want to act on it.

06 Order book, depth & tape

MARKET DATA



The order book

A live ladder: red **asks** above, green **bids** below, the **mid** and **spread** in the center. Columns are **PRICE · SIZE · SUM** (cumulative), with a depth bar behind each row and larger resting sizes shown in bolder text.

The **imbalance gauge** at the top (e.g. `B 33.9% / 66.1% A`) shows the bid/ask weight of the visible book. Price and size decimals adapt to the pair automatically, so a sub-cent coin isn't flattened to "0.00".

Click any level to drop its price into the order ticket.

The **Book** and **Vol** tabs show the ladder; the **Imb** and **Sig** tabs are reserved for imbalance/signal analytics.

The ticker ribbon

Across the command bar: the pair, last price (flashing on ticks), the day's change, **BID × ASK**, **SPD** (spread), a **24H** low–high range bar with a marker at the last price, and 24-hour **VOL**. Perpetuals also show mark / index.

Depth chart & tape

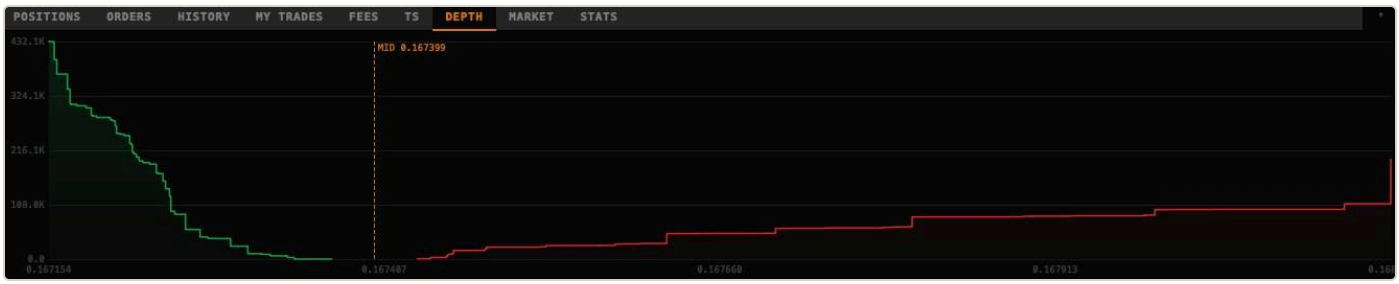


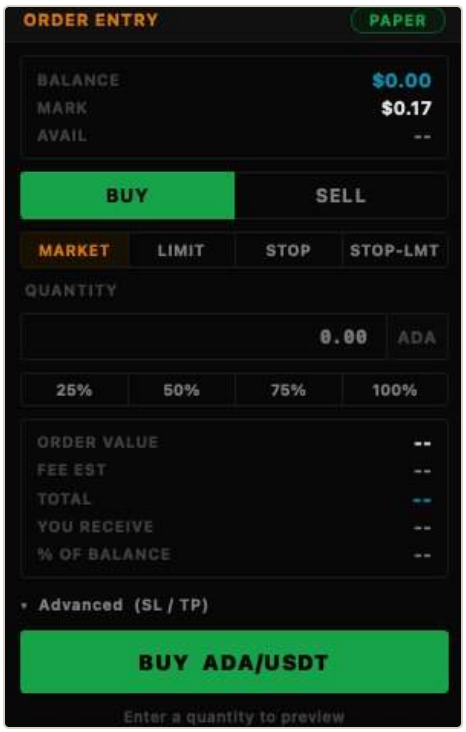
Fig 5. The **DEPTH** blotter tab: cumulative bid (green) / ask (red) depth with an amber mid line; hover for the cumulative size at any price.

The **T&S** (time & sales) tab is a live trade tape — **TIME · PRICE · SIZE**, colored green (buys) / red (sells), with larger prints emphasized. It fills as trades print; a quiet pair shows "Waiting for trades..." until the first one lands.

> **THE HEADLINE PRICE IS THE MID**
>
> When the book is live, the big price (ticker, ticket, marks) uses the **mid** = (bid+ask)/2, which is steadier than the last trade on thin pairs — so it can differ slightly from the newest tape print.

07 The order ticket

ORDER ENTRY



The ticket runs top to bottom: a **PAPER/LIVE** badge, an account card (**BALANCE · MARK · AVAIL**), the **BUY/SELL** toggle, the order-type tabs, quantity + quick-size, the order breakdown, and the big submit button.

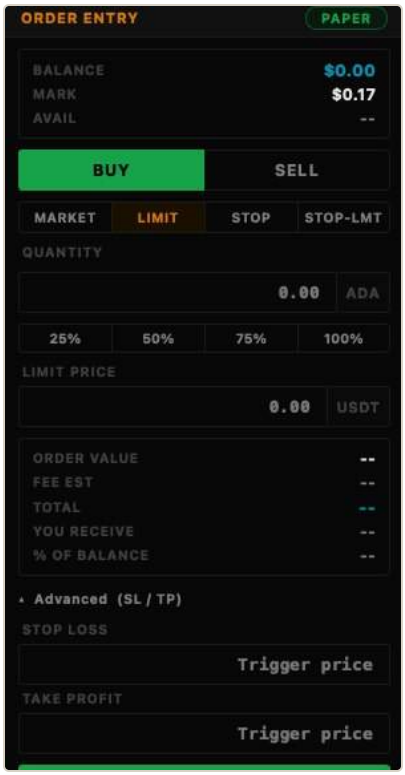
Order types

Type	Reveals
MARKET	— (fills now)
LIMIT	Limit price
STOP	Stop trigger
STOP-LMT	Stop + limit price

Sizing & the read-outs

Enter a **quantity** in the base asset (e.g. ADA), or tap **25 / 50 / 75 / 100%** to size from your balance (fee-aware, so a 100% order isn't rejected for the fee). As you type, the ticket previews:

Read-out	What it is
BALANCE / MARK / AVAIL	Your quote balance · the current mark price · balance left after this order.
ORDER VALUE	Notional = quantity × price.
FEE EST	Estimated fee (a taker-fee assumption; the real fee comes back on fill).
TOTAL	Margin + fee.
YOU RECEIVE	Base bought (buy) or quote received (sell).
% OF BALANCE	How much of your balance this order commits.



Brackets & futures

Open **Advanced (SL / TP)** to attach a **stop-loss** and **take-profit** trigger. On perpetual markets the ticket also shows **leverage**, **margin mode** and **reduce-only** — but only on venues that support them (§9).

Submit is a big colored **BUY / SELL <pair>** button (shortcut ⌘/Ctrl+Enter). Before it sends, the ticket validates quantity, min lot/notional, stop direction, bracket sanity and affordability, flagging any problem inline.

> **PAPER FILLS ARE HONEST**
>
> A paper market order fills at the current cached price — and is **rejected** (not filled at a made-up price) if no price has streamed yet. Resting limit/stop orders and SL/TP brackets are matched client-side against the live tick stream.

08 The blotter

BLOTTER

The panel under the chart is a nine-tab blotter. Its ▾ corner button collapses it to just the tab strip (giving the chart more room); clicking a tab expands it again.

Tab	What it shows
POSITIONS	Open positions — Symbol · Side · Qty · Entry · Mark · P&L · Lev · ✕ exit. Row click loads the symbol on the chart.
ORDERS	Active (pending/partial) orders — Symbol · Side · Type · Qty · Price · Status · ✕ cancel.
HISTORY	Your paper trade history (last 50) — Symbol · Side · Price · Qty · Fee · P&L · Time.
MY TRADES	Exchange-side fills (live only, needs keys) — Symbol · Side · Price · Amount · Cost · Fee · Ccy · Time.
FEES	The venue's fee schedule — Symbol · Maker % · Taker %.
T&S	The live trade tape (§6).
DEPTH	The cumulative depth chart (§6).
MARKET	Derivatives cards — Funding · Mark · Index · Open Interest · Maker/Taker · Next Funding.
STATS	Performance cards — Total P&L · Win Rate · Total Trades · Best/Worst · Balance · Equity · Used Margin.

The positions action bar

Above the positions table: a running **Total P&L**, plus **▲ winners** and **▼ losers** chips each with a one-click square-off (close only the profitable, or only the losing, positions — confirmed). A **SQUARE OFF ALL** button closes everything (confirmed), and each row's ✕ exits that position. Right-click any table to **export CSV**.

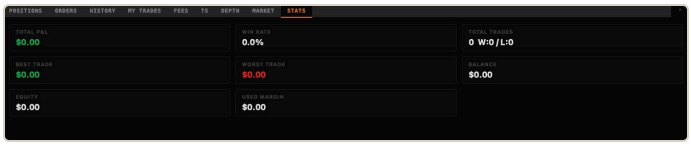


Fig 6. STATS — performance & funds cards.

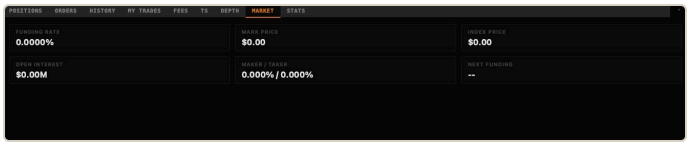


Fig 7. MARKET — funding & contract cards (populate for perpetuals).

09 Paper, live & connecting an exchange

PAPER · LIVE

To trade **live**, click **API** and enter that exchange's keys. The dialog adapts per venue — most show **API KEY**, **API SECRET**, an optional **PASSPHRASE** (OKX / KuCoin / Bitget / Coinbase) and a **TOTP** 2FA secret (with a live code); Hyperliquid instead asks for a wallet address + private key.

Keys are stored in your **operating-system keychain** — never on Fincept's servers. **Clear** blanks the form; saved keys stay until you save empty ones over them.

What each venue supports

The order ticket adapts to each exchange's capabilities — leverage, margin mode, reduce-only and native stop-loss/take-profit appear only where the venue offers them. Paper mode always supports SL/TP (matched client-side); in live mode, brackets appear only on venues with native support.

Capability	Venues
Perps + leverage + margin	OKX, Bybit, Binance, Bitget, BingX, Gate, Crypto.com, Hyperliquid*
Native SL / TP (live)	OKX, Bybit, Binance, Bitget, BingX, Kraken (single SL *or* TP)
Spot only	Coinbase, HTX, Bitfinex, MEXC, KuCoin

*Hyperliquid streams public data and reports as a perp venue, but live order signing is not yet enabled — see honest notes.

START IN PAPER

Every exchange has its own funded **Crypto Paper** book, so you can rehearse the full workflow — chart, book, ticket, brackets, blotter — on real prices before you ever connect a key.

10 AI tools & the data engine

AI · ENGINE

Driving crypto from AI chat

Eighteen tools (category `crypto-trading`) let the assistant read markets and — with your keys — trade for you. Read-only tools are free; anything that moves money is flagged destructive and needs API keys.

Tool group	Tools
Market data (free)	`get_ticker` · `get_order_book` · `get_candles` · `get_exchange_info` · `crypto_search_symbols` · `crypto_get_funding_rate` · `crypto_get_open_interest`
Account (keys)	`crypto_get_balance` · `crypto_get_positions` · `crypto_get_open_orders` · `crypto_get_my_trades` · `crypto_get_trading_fees`
Trading (keys · destructive)	`crypto_place_order` · `crypto_cancel_order` · `crypto_close_position` · `crypto_set_leverage` · `crypto_set_margin_mode`
Session	`crypto_set_exchange`

Under the hood

- **Native connectors.** Each of the 14 exchanges has a hand-written C++ connector speaking that venue's own REST + WebSocket — no Python/ccxt bridge. REST runs on a worker thread; streams run live.
- **One data plane.** All panels read a shared stream (ticker, order book, trades, candles) so the chart, book, tape and ticket never disagree.
- **Paper marking is independent.** A background service marks your paper positions and fills resting orders even when the tab is closed, so P&L stays honest.
- **Secure keys.** API secrets live in the OS keychain; live orders carry an idempotency key so a network hiccup can't double-send.

11 Quick reference & honest notes

REFERENCE

How do I…

Task	Do this
Switch exchange	Click the exchange name → pick from the menu.
Change pair	Click a watchlist row, or type it in the symbol box.
Change timeframe	The **1m … 1d** strip on the chart.
Add an indicator	Chart **Indicators ▾**, or the MA/VOL/MACD/RSI/BOLL quick toggles.
Place an order	Set BUY/SELL → type → quantity → **BUY/SELL <pair>** (⌘/Ctrl+Enter).
Set a price from the book	Click a book level (in **Book** mode).
Attach SL / TP	Open **Advanced (SL / TP)** in the ticket.
Close a position	Position card **EXIT**, or the ✕ in the POSITIONS tab.
Close many at once	Winners/losers chips, or **SQUARE OFF ALL**.
Compare two pairs	**COMPARE**; ◀ **TRADE** to promote one.
Go live	**API** → keys → flip to **LIVE**.

Honest notes & gotchas

- **Public data is free; live trading needs keys.** Charts/books/tapes stream with no keys; keys stay in your OS keychain.
- **Hyperliquid live trading is not yet enabled** — its public data works, but signed orders are disabled pending a signing fix.
- **MEXC has no public WebSocket**, so it streams by fast REST polling — slightly less granular than a true WS feed.
- **Order-book Imb / Sig tabs are placeholders** — the Book/Vol ladder and imbalance gauge are live; the Imb/Sig analytics tabs aren't populated yet.
- **The headline price is the mid**, not the last trade, so it can differ from the newest tape print.
- **Live SL/TP only where supported** (native-SL/TP venues); paper brackets work everywhere. **Fee estimates are approximate** until fill.
- **The chart engine renders prices to 2 decimals**, so very-low-priced coins can look like "0.00" on the chart — the order book still shows full precision.
- **The compare chart is view-only** — trade from the primary chart or promote with ◀ TRADE.
- **Some pairs stream thinly** — a quiet pair may show "--" or an empty tape until a trade prints; try the venue's primary quote for depth.

> **FINCEPT RESEARCH DESK**
>
> Prepared for institutional clients · Professional Edition v5.0 · Captured live on the Crypto Trading tab (Kraken · ADA/USDT · paper). Control labels quoted verbatim from the running application.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Algorithmic Trading

A complete, control-by-control walkthrough of the ALGO tab — author a strategy in plain sentences, test it over history, scan the market for it, deploy it to paper or real money, and watch every fill it ever books.

`>_ 7 tabs · sentence rules · baskets · F&O legs · paper & live · 34 agent tools`

Covers · **Builder** · **Strategies** · **Backtest Lab** · **Scan** · **Alerts**
Deploy · Dashboard · History · the engine · the safety rails in numbers

Demonstrated live on a working desk — real strategies, a real 50-symbol basket backtest, a real 30-member F&O fleet and a 771-trade ledger.

Contents

21 SECTIONS

GETTING ORIENTED

01 What the ALGO tab is 3
02 Anatomy: the frame and the seven tabs 4

BUILDING A STRATEGY

03 Builder — the four-step flow 5
04 Builder — writing rules in sentences 6
05 Builder — the safety net and F&O legs 7
06 Builder — the AI strategy assistant 8
07 Strategies — your library 9

TESTING AND FINDING

08 Backtest Lab — one symbol, one report 10
09 Backtest Lab — baskets, and what it does not model 11
10 Scan — the one-time check 12
11 Scan — the live universe sweep 13
12 Alerts — background watches 14

GOING LIVE

13 The deploy dialog, field by field 15
14 Baskets and F&O options baskets 16
15 Dashboard — the fleet 17
16 Dashboard — positions, fills and controls 18
17 History — the permanent ledger 19

UNDER THE HOOD

18 How a deployment actually runs 20
19 The safety rails, in numbers 21
20 Driving the desk from AI Chat 22
21 Quick reference and honest notes 23

About this guide. Every screenshot is a live capture of the ALGO tab in the shipping build, on a working desk. The basket backtest in sections 09–10 was run for this guide; **nothing was deployed and no order was ever placed** — every deploy dialog shown here was cancelled.

01 What the ALGO tab is

ORIENTATION · 01

ALGO is the terminal's **strategy desk**: one screen where you write a trading rule in plain sentences, prove it over history, hunt the market for it, hand it real or simulated money, and then watch every position and every fill it produces — with the engine running inside your own terminal rather than on someone else's server.

Seven tabs follow the natural order of the work: **BUILDER** to author, **STRATEGIES** to keep, **BACKTEST** to prove, **SCAN** to find, **ALERTS** to watch, **DASHBOARD** to monitor, **HISTORY** to account for. You can stop at any of them — plenty of desks use the Builder and Backtest Lab as a research tool and never deploy anything.

Where to find it

Route	How
Tab bar	**ALGO** — the ninth button on the top tab bar
Command bar	Type `algo`, then pick the row

At a glance, in the ALGO tab you can…

- Write entry and exit rules as readable sentences
- Use indicators, price, volume and fundamentals as operands
- Set stop-loss, target, trailing stop and scale-in
- Build multi-leg F&O structures from leg rules
- Have an AI assistant write the whole configuration
- Backtest one symbol — or a 50-name basket
- Grid-search stop-loss × take-profit
- Scan a symbol list once, or a whole exchange continuously
- Save conditions as background alert watches
- Deploy to paper money or a connected broker
- Run a basket of thirty option contracts as one fleet
- Monitor P&L, positions and fills in real time
- Keep a permanent, exportable ledger of every trade

This tab can place real orders. Deployments run in one of two modes — **paper** (simulated fills, no broker contact) or **live** (real orders at your connected broker). The dialog defaults to paper and says so, and the engine enforces order-rate breakers, exposure caps and per-deployment loss limits (section 19). Everything in this guide was captured without deploying anything.

02 Anatomy: the frame and the seven tabs

A title bar of seven coloured pills, the active tab's panel, and a status strip. The pills are the whole navigation — each carries its own accent colour, and the active one is filled.

Fig 1 — The tab bar. Right of the pills, a live count of running deployments.

Tab	What it is for	Guide
BUILDER	Author a strategy: market, entry rules, exit rules, risk, F&O legs, AI assistant	03–06
STRATEGIES	Your saved strategies plus the built-in library — edit, backtest, deploy, version history	07
BACKTEST	Test any strategy or basket over history; grid-search SL/TP	08–09
SCAN	Find matching symbols — a one-time check, or a continuous live sweep of a whole exchange	10–11
ALERTS	Save conditions as background watches and get notified when they fire	12
DASHBOARD	Monitor running deployments: P&L, positions, fills, fleet controls	15–16
HISTORY	The permanent ledger of every fill, with KPIs, analytics and CSV export	17

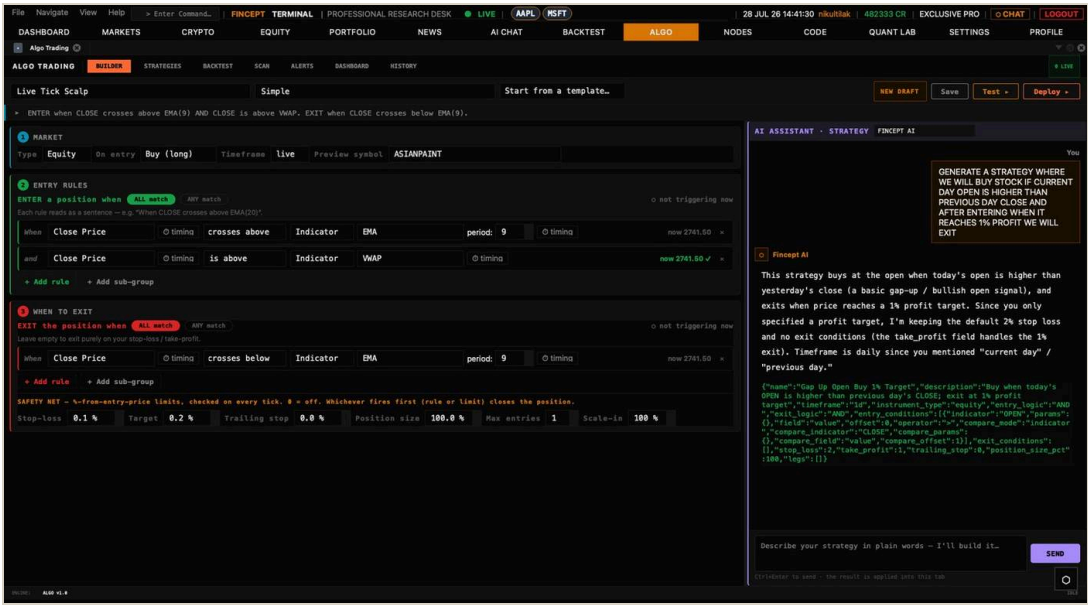


Fig 2 — The ALGO tab on the Builder, with a real draft loaded: the numbered authoring flow on the left, the AI strategy assistant on the right.

Frame element	Meaning
N LIVE badge	Deployments currently *running* or *starting*, across the whole fleet
ENGINE: ALGO v1.0	The status strip along the bottom — the engine build
IDLE / RUNNING	Bound to real state: RUNNING (green) whenever at least one deployment is live

03 Builder — the four-step flow

BUILDER FLOW · 03

The Builder is a numbered flow, top to bottom: **1 MARKET → 2 ENTRY RULES → 3 WHEN TO EXIT** → the **SAFETY NET** row. Above it sits the header: the strategy's name, a description, a template picker, and the four actions.

Header control	What it does
`Strategy Name`	Required before Save or Deploy — the name you will see everywhere else
`Description`	Optional free text
`Start from a template…`	Loads a ready-made strategy as a starting point
`NEW DRAFT`	Clears the editor and starts again
`Save`	Saves to **My Strategies**. Re-saving the same editor updates that strategy rather than making a copy
`Test ▸`	Hands the current draft — saved or not — to the Backtest Lab and switches tabs
`Deploy ▸`	Opens the deploy dialog for this strategy (section 13)

Fig 3 — The sentence line. The Builder narrates your rules back to you in English, so a mis-set operator is visible before you ever run anything.

Fig 4 — Step 1, MARKET: what to trade and how the rules are evaluated.

Step 1 field	Options	Meaning
`Type`	Equity · Option · Future	Instrument class. Option and Future reveal the F&O leg editor (section 05)
`On entry`	Buy (long) · Sell (short)	Which side the entry signal takes. Exits, stops and targets mirror automatically for a short
`Timeframe`	live · 1m · 5m · 15m · 1h · 4h · 1d · 1w	The bar the rules evaluate on. **live** evaluates on every tick
`Preview symbol`	any ticker	Used for the live values shown beside each rule, and pre-fills the deploy dialog

> **Direction lives on the strategy, not on the deploy.** Choosing *Sell (short)* here is what makes a short strategy short — the deploy dialog no longer asks, so the same strategy always trades the same way whoever deploys it.

04 Builder — writing rules in sentences

A rule is a row you read left to right: *when* a left operand, a comparison, a right operand, and an optional timing qualifier. Rows combine under **ALL match** (every rule must be true) or **ANY match** (one is enough).

Fig 5 — Step 2, ENTRY RULES. Two rules under **ALL match**: price crossing above a 9-period EMA, and price above VWAP. The live value of each operand is printed on the right — *now 2741.60* — and a chip states whether the rule set is triggering at this moment.

Element	What it gives you
`Left / right operand`	Price fields (open, high, low, close), an **Indicator** with its own period, a plain **Value**, volume, or a fundamental
`Comparison`	crosses above · crosses below · is above · is below · equals, and the numeric comparators
`⏱ timing`	Restricts when the rule may fire — an offset in bars, so you can say *"…one bar ago"*
`ALL / ANY match`	The logic joining the rules — `AND` or `OR`
`+ Add rule`	Another row at the same level
`+ Add sub-group`	A nested bracket, so you can write *A and (B or C)*
`now …` `value`	The operand's current value on the preview symbol, refreshed live — the fastest way to sanity-check a threshold
`not triggering now`	Whether the whole set is true at this instant

Fig 6 — Step 3, WHEN TO EXIT. Leave it empty to exit purely on the stop-loss and target; add rules for a signal-based exit.

> **Read the sentence, not the rows.** The line above the flow always states the finished strategy — *"ENTER when CLOSE crosses above EMA(9) AND CLOSE is above VWAP. EXIT when CLOSE crosses below EMA(9)."* If that sentence is not what you meant, the rules are wrong no matter how right they look.

> **An empty exit block is legitimate but consequential.** With no exit rules, the position is closed only by the stop-loss, the target, the trailing stop or a manual exit — so if all three risk numbers are zero, nothing will ever close the trade.

05 Builder — the safety net and F&O legs

Under the flow sits a single row of six numbers — the strategy's own risk. The caption is exact about how they interact: *%-from-entry-price limits, checked on every tick. 0 = off. Whichever fires first (rule or limit) closes the position.*

Fig 7 — The SAFETY NET row.

Field	Units	What it does
`Stop-loss`	% from entry	Closes the position when price moves this far against you. 0 = no stop
`Target`	% from entry	Closes the position at this much profit. 0 = no target
`Trailing stop`	% pullback	A stop that follows price in your favour and closes on a pullback of this size
`Position size`	% of capital	How much of capital each entry uses. **Backtest sizing only** — live size comes from the deploy dialog's Quantity
`Max entries`	1–10	How many stacked entries the engine may take before it stops adding (1 = a single entry)
`Scale-in`	10–100%	Size of each additional entry as a percentage of the base quantity. Only meaningful when Max entries > 1

F&O: options and futures

Set **Type** to Option or Future and three more controls appear: the **Underlying**, an **Expiry** selection rule, and the **OPTION / FUTURE LEGS** editor where each leg carries its own side, strike selection and — optionally — its own expiry.

F&O control	Options	Meaning
`Underlying`	e.g. NIFTY	The instrument the structure is written on
`Expiry`	Weekly · Monthly · Nearest DTE · Absolute	How the concrete expiry is chosen at entry time. A leg may override it
`Legs`	BUY / SELL per leg	The structure itself. The aggregate side is derived from the legs: all-sell reads as **SELL** (short premium), all-buy as **BUY**, mixed as **MULTI**

> **Why the aggregate side matters.** A short straddle is two SELL legs; labelling it “BUY” because buy is the scalar default would misreport the whole position. The engine derives the side from the legs, so a premium-selling structure is recorded as selling. The concrete contracts are resolved when the entry fires, not at deploy, so a restart re-attaches to the positions actually held.

06 Builder — the AI strategy assistant

The right-hand dock is a chat that **writes and edits the configuration**. Describe a strategy in words and it fills in the market, the rules and the risk numbers; ask for a change and it edits the draft in place, then tells you to review before saving.

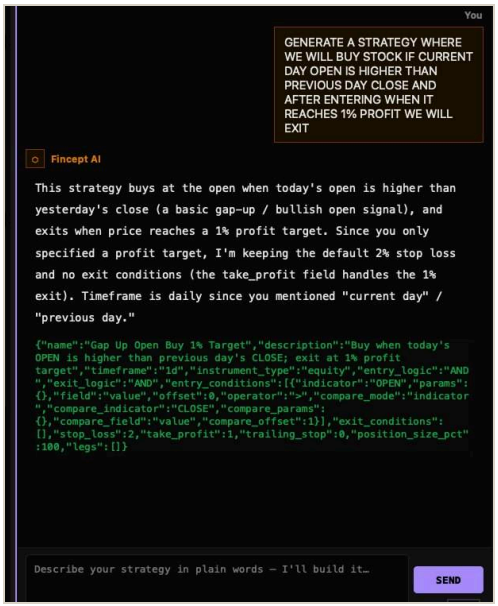


Fig 8 — The assistant turning *"buy stock if current day's open is higher than previous day's close and after entering when it reaches 1% profit we will exit"* into a complete configuration, and explaining the choices it made.

Aspect	Detail
What it edits	The whole draft — instrument, timeframe, entry and exit conditions, logic, stop-loss, target, position size, F&O legs
What it returns	A short explanation of its reasoning, then the configuration JSON it applied
How you accept	It applies to the draft directly; the Builder shows *"AI updated the draft — review, then Save / Test / Deploy."*
Cost	Each turn is a metered request to the managed Fincept model — it **spends credits**, like AI Chat
Where else it appears	The same dock is available in **SCAN** and **ALERTS** mode, where it writes scan and watch conditions instead

Everything the assistant produces is ordinary Builder state — nothing is hidden or locked. You can rewrite any row it wrote, and the sentence line above the flow still narrates the result.

> **Review before you trust it.** The assistant is good at turning a description into rules, but it chooses defaults you did not state — in the example above it kept a 2% stop because only a profit target was specified, and picked a daily timeframe from the words "current day". Read the sentence line and the safety-net row after every AI edit.

> **It is the fastest way to a first draft, not a substitute for the backtest.** Take what it writes straight to the Backtest Lab (**Test ▸**) before it goes anywhere near money.

07 Strategies — your library

A dense grid of everything you can trade: your own saved strategies merged with the terminal's built-in library, searchable, sortable and filterable by category.

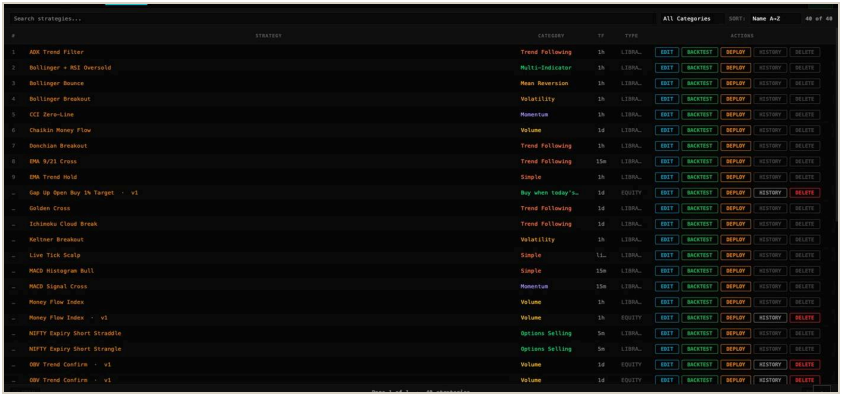


Fig 9 — The strategy grid. Built-in library rows and your own saved strategies sit in one list; the count on the right reads *40 of 40*.

Column / control	What it is
`All Categories`	Category filter — Trend Following, Mean Reversion, Volatility, Momentum, Volume, Options Selling, Simple, Multi-Indicator…
`SORT`	Name A→Z · Name Z→A · Category
`STRATEGY`	Name, with `· vN` appended for your own versioned strategies
`CATEGORY / TF`	Category, and the timeframe the rules were written for
`TYPE`	**LIBRARY** for a built-in, **EQUITY** (or the instrument class) for your own

Fig 10 — Per-row actions.

Action	What it does
`EDIT`	Opens the strategy in the Builder, keeping its identity so a save updates it
`BACKTEST`	Opens the Backtest Lab pre-loaded with it and runs immediately
`DEPLOY`	Opens the deploy dialog (section 13)
`HISTORY`	Lists saved versions; restoring one loads it into the Builder as a draft. Disabled for built-ins

> **DELETE** removes your own strategies only — built-ins refuse. **Saving is versioned:** Each save of one of your strategies creates a new server-side version, so **HISTORY** can walk back to what the strategy said last week — useful when a change stops working and you need the previous rule set.

08 Backtest Lab — one symbol, one report

Pick a strategy, a symbol list, capital and a window, then run. The lab is deliberately separate from the Builder so authoring stays focused and testing has room for a full report.

Fig 11 — The setup row: strategy picker, symbol chips with basket presets, capital, the date window, and the two actions.

Control	What it does
`Strategy`	Any saved strategy, the built-in library, or **✎ … (Builder draft)** — the unsaved draft handed over by **Test ▸**
`Symbols`	One or many. Chips make a basket; **+ Add basket…** offers NIFTY 50, NIFTY Bank and the whole F&O universe
`Capital`	Starting capital the sizing is applied to
`From / To`	The backtest window
`▶ RUN BACKTEST`	Runs the test — one symbol, or the whole basket sequentially
`⚙ OPTIMIZE SL / TP`	Grid-searches stop-loss × take-profit over the same window

The single-symbol report

Metric	Reading it
`TOTAL RETURN`	Percentage and the final capital figure
`SHARPE / SORTINO`	Risk-adjusted return, with a plain-English rating (*Excellent*) and the downside-adjusted variant
`MAX DRAWDOWN / CALMAR`	Worst peak-to-trough, and return ÷ drawdown
`WIN RATE / PROFIT FACTOR`	Share of winning trades, and gross profit ÷ gross loss (∞ when there are no losers)
`EXPECTANCY`	Average P&L per trade — the number that tells you whether the edge survives repetition
`TRADES / AVG BARS HELD`	Count with a W/L split, and the average holding period
`Equity Curve`	Strategy against buy-and-hold on the same window
`Drawdown %`	The underwater curve
`MONTHLY RETURNS %`	A month-by-month grid — where the return actually came from
`Trades table`	Every trade: entry and exit time and price, quantity, P&L, bars held, and the **reason** it closed

09 Backtest Lab — baskets, and what it does not model

Give the lab a list of symbols and it runs them one after another, filling a comparison table as it goes, then states a verdict for the basket as a whole.

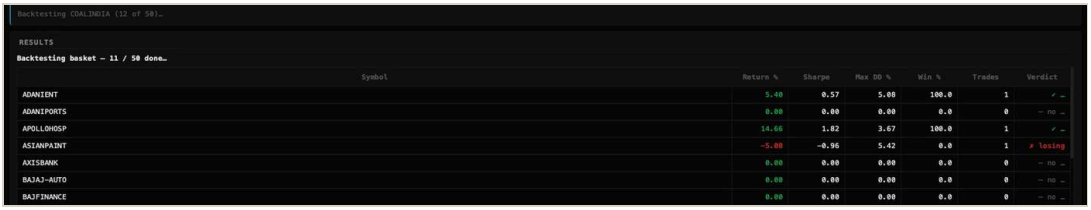


Fig 12 — A 50-symbol basket mid-run: *Backtesting basket — 11 / 50 done…* with results landing as each symbol finishes.

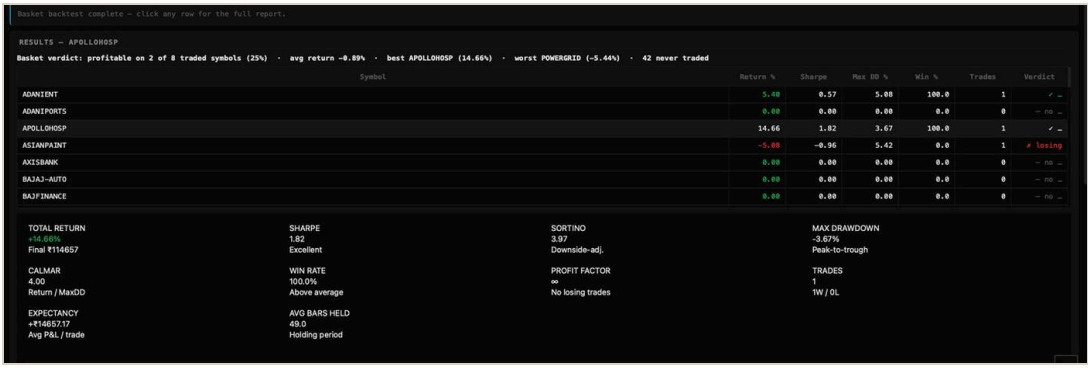


Fig 13 — The finished basket. The verdict line is the most honest sentence in the tab: *profitable on 2 of 8 traded symbols (25%) · avg return –0.89% · best APOLLOHOSP (14.66%) · worst POWERGRID (–5.44%) ·* ***42 never traded***. Click any row for that symbol's full report.

Comparison column	Meaning
`Return % · Sharpe · Max DD % · Win % · Trades`	The same metrics as the single-symbol report, per symbol
`Verdict`	A one-glance mark: profitable, losing, or *no …* where the strategy never traded

> **"42 never traded" is the number people miss.** The verdict percentage is computed over the symbols that actually produced a trade, not over the basket — here, 2 profitable out of *8 traded*, from 50 submitted. A strategy that fires on four names in a year is not necessarily bad, but it is a very different thing from one that trades all fifty.

> **What the backtest does not model.** The lab tells you itself, above the results: *"⚠ Backtest ignores: custom execution style (backtested at bar open)."* Fills are taken at the bar open, so a strategy whose live behaviour depends on limit-order placement, slippage budgets or intrabar timing will trade differently in the market than it does here. SHORT direction and scale-in *are* modelled.

10 Scan — the one-time check

SCAN answers "which of these symbols matches my conditions?" in two very different ways. The mode switch says which, in words.

Fig 15 — The mode switch. **ONE-TIME SCAN** (amber) checks a list once; **LIVE SCAN** (violet) watches a whole market continuously.

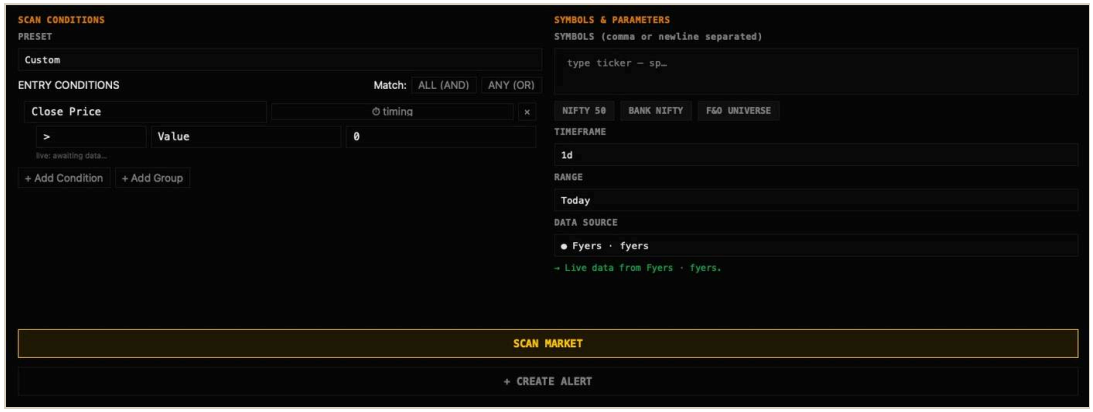


Fig 16 — One-time mode: the same condition builder as the Builder, a symbol list with universe shortcuts, and the data-source picker.

Control	What it does
PRESET	A ready-made condition set, or **Custom**
ENTRY CONDITIONS	The same sentence-rule editor as the Builder, with **ALL (AND)** / **ANY (OR)** and sub-groups
SYMBOLS	Comma- or newline-separated tickers, plus one-click **NIFTY 50**, **BANK NIFTY** and **F&O UNIVERSE** chips
TIMEFRAME / RANGE	The bar size, and how much history to pull for the check
DATA SOURCE	**Auto** (best available), a connected broker account, or **YFinance** (free, delayed). A hint under it states what you actually get
SCAN MARKET	Runs the check and fills the results table
+ CREATE ALERT	Hands these exact conditions to the ALERTS tab as a background watch (section 12)

Result column	Meaning
SYMBOL	The matching ticker
SIGNAL	**BULLISH**, **BEARISH** or **NEUTRAL**
MATCH / DETAILS	Which rules matched, and the values that made them true

> **The data source is the whole answer's provenance.** A scan on YFinance uses free delayed data; the same scan on a connected broker account uses that broker's live feed. The hint line — *"→ Live data from Fyers · fyers."* — is worth reading before you act on a match.

11 Scan — the live universe sweep

LIVE SCAN · 11

Live mode is a different engine: instead of checking a list once, it runs a **saved strategy** continuously across an entire universe on the live feed, and every match is actionable on the spot.

Fig 17 — Live mode: strategy, universe, broker account, sweep interval, cooldown, and the match table with a **DEPLOY** button per row.

Control	Options / meaning
STRATEGY	Any saved strategy — its entry conditions are what the sweep looks for
UNIVERSE	All NSE Equity · All BSE Equity · NIFTY 50 · Custom symbols
CUSTOM SYMBOLS	Your own list, when the universe is set to Custom
BROKER ACCOUNT	**Required** — a live tick feed comes from a connected broker
EVAL SWEEP (ms)	How often the whole universe is re-evaluated
COOLDOWN (min)	Minimum gap before the same symbol can match again — stops one choppy name flooding the list
Send matches to external providers	Also push each match to your configured notification channels (e.g. Telegram)
START SCAN	Begins the sweep. The status line reads *Warming history — matches will appear below…* then *Scanning live*

Match column	Meaning
SYMBOL · SIGNAL · PRICE	What matched, which way, and at what price
MATCHED RULES	The specific rules that fired
DEPLOY	Opens the deploy dialog for that symbol with the scanning strategy already chosen

> **Live scan needs a connected broker and refuses without one** — *"Select a connected broker account (live feed required)."* It also refuses a strategy with no entry conditions. Both are guards, not bugs: a universe sweep with nothing to look for would return everything.

> **Match → deploy is one click, so treat it carefully.** The DEPLOY button opens the normal deploy dialog, which still defaults to paper mode and still asks for size and limits. Nothing goes live from a scan without you completing that dialog.

12 Alerts — background watches

An alert is a condition set that keeps watching after you have moved on. Build it here — or send it over from a scan with **+ CREATE ALERT** — and the terminal polls it in the background.

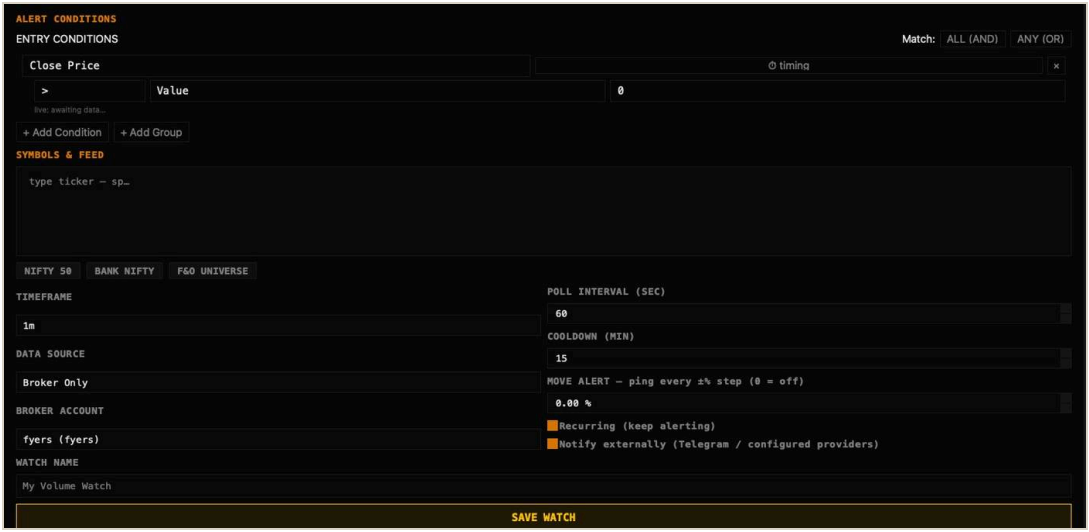


Fig 18 — The watch form: conditions, symbols and feed, poll interval, cooldown, a move-alert option, recurrence, external notification and a name.

Control	What it does
`ALERT CONDITIONS`	The same sentence-rule editor, with ALL / ANY logic
`SYMBOLS & FEED`	The tickers to watch, plus the universe shortcut chips
`TIMEFRAME`	The bar size for indicator conditions — *not used on the live tick feed*
`DATA SOURCE`	Broker Only · Auto (Broker → YFinance) · YFinance Only
`BROKER ACCOUNT`	Which connected account provides the feed, or none to force the YFinance fallback
`POLL INTERVAL (SEC)`	How often the watch is evaluated
`COOLDOWN (MIN)`	Minimum gap between two firings of the same watch
`MOVE ALERT — ± % step`	A price-move alert independent of the conditions: ping every ±N% from an anchor. 0 = off
`Recurring`	Keep alerting after the first fire, instead of firing once
`Notify externally`	Also send to Telegram or your other configured providers
`WATCH NAME`	How the watch appears in the list
`SAVE WATCH`	Creates it — or updates the one you are editing

> **Check now vs Test. Check now** evaluates the watch against real current data — a genuine answer. **Test** fires it artificially so you can confirm the notification actually reaches you, which is the part people forget to verify.

13 The deploy dialog, field by field

Every route to going live — the Builder's **Deploy ▸**, a strategy row's **DEPLOY**, a live scan match — opens this one dialog. It is the last gate before the engine starts placing orders, and it is organised as three questions: what to trade, how to execute, and what limits to respect.

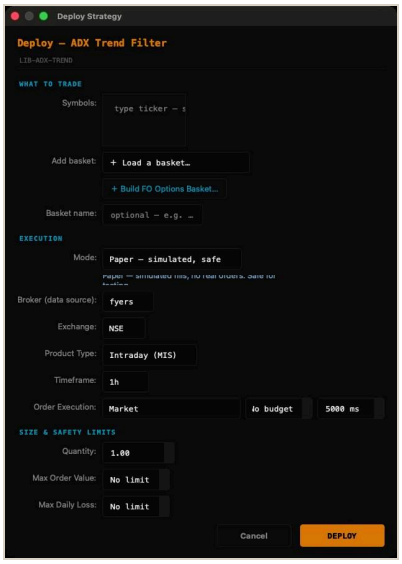


Fig 20 — The deploy dialog for a library strategy, in its default state: **Paper — simulated, safe**.

Field	Meaning
`Symbols`	One ticker = a single deploy; several = a basket, one deployment per symbol under a shared basket id
`Add basket`	Load a preset symbol list instead of typing
`Build F&O Options Basket…`	Turn an underlying and an expiry into a fleet of option contracts (section 14)
`Basket name`	Optional label, e.g. *F&O Universe*
`Mode`	**Paper — simulated, safe** or **Live — real money**. Paper is the default and carries the hint *"simulated fills, no real orders"*
`Broker (data source)`	Which connected account provides data and, in live mode, receives the orders
`Exchange · Product Type · Timeframe`	From the broker profile (e.g. NSE, *Intraday (MIS)*) and the strategy; a timeframe mismatch is flagged
`Order Execution`	**Market**, or **Limit** with a slippage budget in basis points and a chase/cancel timeout in ms
`Quantity`	Shares or units per order — this, not the Builder's position-size %, is live sizing
`Max Order Value`	Rejects any single order above this notional. *No limit* by default
`Max Daily Loss`	Halts the deployment for the day once realised losses reach it

> **Two fields deserve a deliberate answer every time. Mode** — because Live means real orders at your broker — and **Quantity**, because the Builder's *Position size %* governs backtest sizing only. A strategy tested on 100% of ₹100,000 deployed at quantity 1 is a different strategy.

14 Baskets and F&O options baskets

A basket is many deployments created together and grouped afterwards: each member runs its own copy of the strategy on its own symbol, and the Dashboard rolls them up under one header with an aggregate P&L.

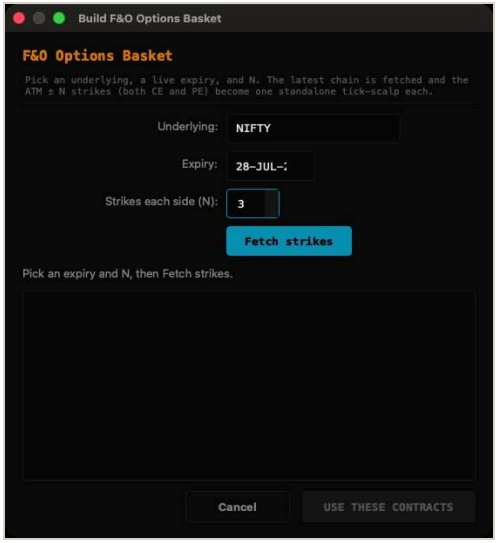


Fig 21 — The F&O options basket builder.

The dialog states its own contract: *"Pick an underlying, a live expiry, and N. The latest chain is fetched and the ATM ± N strikes (both CE and PE) become one standalone tick-scalp each."*

Field	Meaning
`Underlying`	e.g. NIFTY
`Expiry`	A live expiry from the fetched chain
`Strikes each side (N)`	How far either side of at-the-money to go
`Fetch strikes`	Pulls the current chain and lists the contracts it will use
`USE THESE CONTRACTS`	Returns them to the deploy dialog as the basket's members

With N = 7 you get 15 strikes × 2 option types = **30 deployments** — exactly the fleet shown in section 15.

Basket behaviour	Detail
`One deployment per member`	Each has its own status, position, P&L and trade history
`Shared basket id and label`	What lets the Dashboard group them and total them
`Grouped controls`	The basket header carries aggregate P&L, an in-position count and its own actions
`Backtesting a basket`	Sequential, with a comparison table and a basket verdict (section 09)

> **A basket multiplies everything — including order flow.** Thirty members each inside their own order-rate budget can still be thirty entries in the same second when the condition turns true across the whole chain, which is exactly why the engine also enforces a fleet-wide entry ceiling (section 19). Size a basket for the market you expect, not the one you hope for.

15 Dashboard — the fleet

The Dashboard is the monitoring screen: a KPI strip across the top, the fleet on the left, and positions, fills and a detail pane on the right.

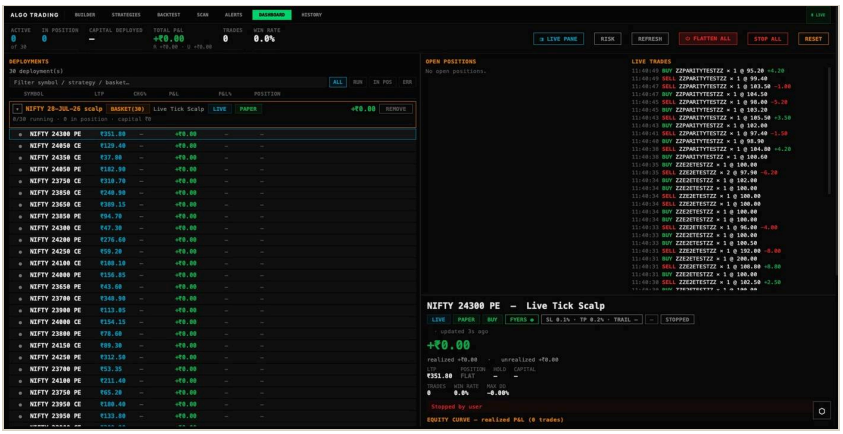


Fig 22 — The Dashboard on a real 30-member F&O basket, with a live fill feed on the right.

KPI	Exactly what it counts
ACTIVE	Deployments running or starting, out of the total loaded
IN POSITION	Deployments holding an open position at this moment
CAPITAL DEPLOYED	Σ |quantity| × entry across open positions
TOTAL P&L	Realised + unrealised, with the two split out beneath
TRADES / WIN RATE	Fill count and win rate across the fleet

Fig 24 — A basket header row: name, member count, strategy, mode chips, aggregate P&L and a REMOVE action, with *0/30 running · 0 in position* beneath — then the members indented below it.

Fleet list element	Meaning
Filter box	Narrow by symbol, strategy or basket
ALL / RUN / IN POS / ERR	Status filters — everything, running, holding a position, or errored
SYMBOL · LTP · CHG%	The instrument and its last traded price
P&L · P&L%	This deployment's realised + unrealised, and the same as a percent of position cost

> **Basket members are ordinary deployments.** Selecting one shows its own detail pane, its own equity curve and its own trades — the grouping is presentational, so you can still stop, exit or inspect a single leg of a thirty-member fleet.

16 Dashboard — positions, fills and controls

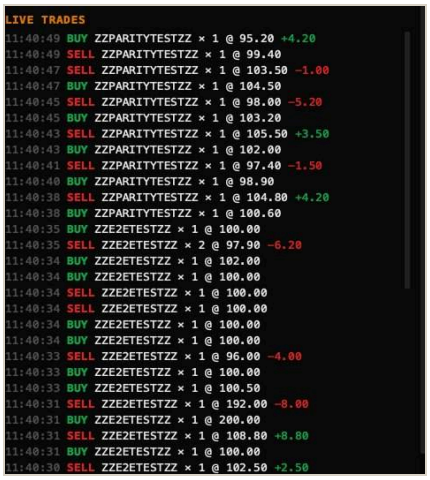


Fig 25 — The LIVE TRADES feed: every fill as it books, with side, quantity, price and P&L.

The right-hand panes

Pane	What it shows
OPEN POSITIONS	Every open position attributed to its deployment — including one row per leg for an F&O structure. This is the authoritative view of what you are holding
LIVE TRADES	The rolling fill feed across the fleet
◨ LIVE PANE	Hides or shows the pair, giving the fleet list the full width

The pane is deliberately split this way: positions answer *"what do I own?"* and the feed answers *"what just happened?"* — two questions that need different answers during a fast market.

Fig 26 — The detail pane for the selected deployment: mode and side chips, the risk numbers it was deployed with, its status, big P&L with the realised/unrealised split, LTP, position, capital, trade count, win rate, max drawdown — and a banner stating why it stopped.

Fleet controls

Control	What it does	Care
REFRESH	Re-reads the deployment list	Safe
RISK	Opens the **Algo Risk Caps** dialog (section 19)	Safe
⏻ FLATTEN ALL	Closes **every open position** across the fleet at market, then reports how many were flattened	**Destructive — confirms first**
STOP ALL	Stops every deployment on this machine	**Confirms first**
RESET	Clears the dashboard's local deployment and trade state	**Confirms first**

> **"This machine" is literal.** The confirmation for the fleet-wide actions says so: *"Deployments running on another machine are not affected."* Algo deployments run inside the terminal that started them, so stopping everything here does not stop a deployment you started on a second desk.

17 History — the permanent ledger

LEDGER · 17

The Dashboard shows what is happening; HISTORY is the record of everything that ever happened. Every fill the engine has booked, paper or live, newest first.

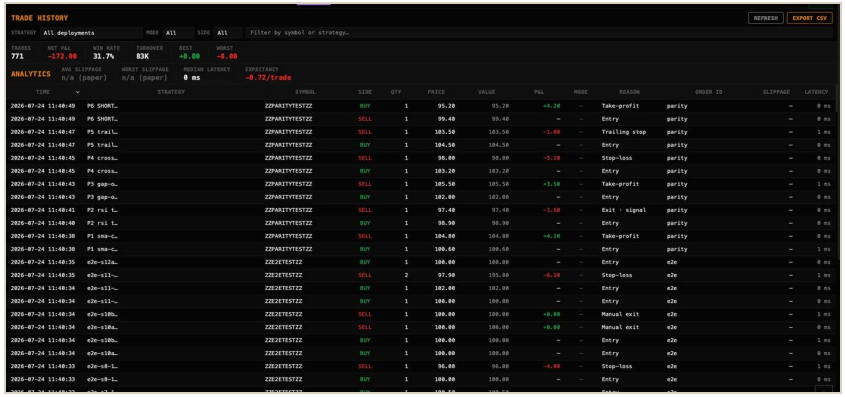


Fig 27 — The ledger on a real desk: 771 trades, filtered and totalled.

Strip	Fields
KPIs	TRADES · NET P&L · WIN RATE · TURNOVER · BEST · WORST — all over the **filtered** set, not the whole ledger
ANALYTICS	AVG SLIPPAGE · WORST SLIPPAGE (most adverse single live fill vs its signal price) · MEDIAN LATENCY (signal→fill) · EXPECTANCY per trade. **Slippage and latency are live-only** — a paper desk reads *n/a (paper)*, because a simulated fill has no broker round-trip to measure

Column	Meaning
TIME · STRATEGY · SYMBOL	When, which strategy, which instrument
SIDE · QTY · PRICE · VALUE	The fill itself
P&L	Realised on the closing side of a trade
MODE	paper or live
REASON	Why it happened — see below
ORDER ID · SLIPPAGE · LATENCY	Broker order reference and per-fill telemetry

REASON	What closed (or opened) the position
Entry / Entry · recovered	A new position; *recovered* means the engine re-attached to it after a restart
Exit · signal	Your exit rules fired
Stop-loss / Take-profit / Trailing stop	A risk limit fired
Daily-loss cap	The deployment's Max Daily Loss halted it
Manual exit · Safety halt	You closed it from the Dashboard, or a safety rail stopped the deployment (section 19)

18 How a deployment actually runs

Worth understanding before you trust one with money, because it explains both the guarantees and the limits.

Stage	What happens
`1 · Freeze`	At deploy time the engine takes a **snapshot of the strategy** — conditions, legs, risk numbers — and stores it on the deployment. A running algo is immune to later edits of the source strategy
`2 · Feed`	Candles or ticks arrive for the deployment's symbol at its timeframe. **live** evaluates per tick; a bar timeframe evaluates per completed bar
`3 · Evaluate`	The condition evaluator tests the entry rules (or, when in position, the exit rules) against the current data
`4 · Risk`	On every tick the stop-loss, target and trailing stop are checked as %-from-entry-price limits. Whichever fires first — rule or limit — closes the position
`5 · Route`	An order goes to **paper** (a simulated per-deployment book) or to your **broker**. F&O resolves its concrete contracts here and remembers them
`6 · Record`	The fill is booked to the ledger with its reason, order id and telemetry, and pushed to the Dashboard

Property	What it means for you
`Runs locally`	The engine lives in your terminal, not on a Fincept server. Close the terminal and your deployments stop trading
`Survives a restart`	On relaunch the engine recovers orphaned deployments, re-attaches to open positions and marks their fills *recovered*
`Per-deployment isolation`	One deployment per symbol, each with its own position, book, breaker and ledger entries
`Paper is a real simulation`	Paper fills are marked, booked and totalled like live ones — the whole ledger and analytics work in paper, minus slippage and latency
`Exits are privileged`	Rate limiting never blocks an exit — position protection always gets through

Because it runs locally, uptime is yours. A deployment does not keep trading while your machine sleeps, and the fleet-wide controls only reach the terminal you are sitting at. For a strategy that must not miss a session, keep the terminal awake and check the Dashboard's **N LIVE** badge at the open.

Native broker brackets are off by default. When enabled and supported by the broker, a live position can be opened with the stop and target resting *at the broker*, which keeps protection in place even if the terminal disappears. By default the terminal keeps its own client-side stops — simpler and broker-independent, but only enforced while the engine runs.

19 The safety rails, in numbers

The engine ships with layered limits, on the assumption that the thing most likely to go wrong is not a bad strategy but a runaway one. These are the defaults.

Rail	Default	Scope and effect
`Orders per minute`	**10**	Per deployment. Past this the deployment is treated as a runaway loop and **halted** — a legitimate one-minute strategy places a handful
`Orders per day`	**200**	Per deployment — the second stage of the same breaker
`Entries per minute (fleet)`	**75**	Engine-wide, across every deployment. Catches the case where a 50-member basket is inside every individual budget and still floods the broker. **Entries only**
`Entries per day (fleet)`	**500**	Engine-wide daily ceiling
`Entry cooldown`	**5 s**	Minimum time between an exit fill and the next entry, so a *live*-timeframe strategy with an always-true rule cannot re-enter on the very next tick
`Max total exposure`	No limit	Set in **RISK** on the Dashboard — enforced before every new entry
`Max concurrent positions`	No limit	Also in RISK, also checked before each entry
`Max Order Value`	No limit	Per deployment, from the deploy dialog — rejects any single oversized order
`Max Daily Loss`	No limit	Per deployment — halts it for the day, and the fill is booked with reason *Daily-loss cap*
`Slippage budget`	None	Per deployment, in basis points, when Order Execution is Limit; with a chase/cancel timeout in ms
`Duplicate guard`	Always on	Refuses a second identical active deployment

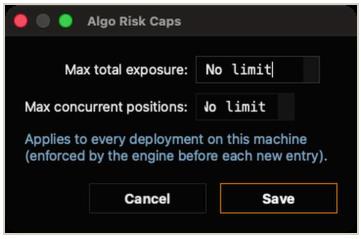


Fig 29 — The Algo Risk Caps dialog.

The dialog is explicit about its reach: *"Applies to every deployment on this machine (enforced by the engine before each new entry)."* Both caps default to **No limit**, which is the one place this guide would suggest changing a default before trading live — an exposure cap is the cheapest protection against a strategy that is right about direction and wrong about size.

> **The rails are settable.** The three order-rate values live in the terminal's configuration and can also be read and changed through the agent tool `algo_safety_limits` (section 20) — useful for tightening a desk's defaults before anyone deploys.

20 Driving the desk from AI Chat

ALGO publishes **34 agent tools** in the `algo-trading` category — the largest trading surface exposed to the assistant, covering the whole life of a strategy.

Group	Tools	What you can ask for
`Strategies (7)`	algo_strategy_catalog, algo_list_strategies, algo_get_strategy, algo_create_strategy, algo_update_strategy, algo_create_strategy_from_template, algo_delete_strategy	“Write me a strategy that…”, “what does my Golden Cross strategy say?”
`Backtesting (1)`	algo_run_backtest	“Backtest it on NIFTY 50 over the last year and summarise”
`Deployments (8)`	algo_deploy_strategy, algo_list_deployments, algo_get_deployment, algo_pause_deployment, algo_resume_deployment, algo_stop_deployment, algo_exit_deployment, algo_get_deployment_metrics	“What is running?”, “stop the losing ones”, “exit NIFTY 24300 PE”
`Monitoring (3)`	algo_engine_status, algo_get_equity_curve, algo_list_deployment_trades	“How is the fleet doing?”, “show me today's fills”
`Safety (1)`	algo_safety_limits	“What are my order-rate limits?”, “tighten them to 5 a minute”
`Scanning (3)`	algo_scan_market, algo_scanner_catalog, algo_scan_preset	“Scan the NIFTY 50 for this condition”
`Scanners & watches (11)`	algo_create_scanner, algo_update_scanner, algo_create_universe_scanner, algo_list_scanners, algo_get_scanner, algo_set_scanner_active, algo_delete_scanner, algo_scanner_history, algo_test_fire_scanner, algo_clear_fired_scanners, algo_clear_scanner_history	“Watch these names for a breakout and tell me on Telegram”

Some of these tools trade. `algo_deploy_strategy` can start a deployment and `algo_exit_deployment` can close a position — without a confirmation dialog, because there is nobody to confirm. Be specific with the assistant: name the strategy, the symbol and the mode, and prefer *paper* unless you mean otherwise.

The most useful safe combination is read-only: `algo_list_deployments` + `algo_get_deployment_metrics` + `algo_list_deployment_trades`. “Summarise how each deployment did today, worst first” is a genuinely better morning report than reading the Dashboard row by row.

21 Quick reference and honest notes

How do I…?

Question	Answer
…write my first strategy?	BUILDER: name it, set Type/direction/timeframe, add one entry rule, set a stop and a target, **Save**
…check what my rules actually say?	Read the sentence line above the flow, and the *now* … value beside each rule
…test it?	**Test** ▸ from the Builder, or **BACKTEST** on its row in STRATEGIES
…test it on many symbols?	Backtest Lab ▸ add symbol chips or + **Add basket…**, then RUN
…tune the stop and target?	⚙ **OPTIMIZE SL / TP** — a grid search over the same window
…find symbols that match today?	SCAN in one-time mode
…watch a whole exchange?	SCAN ▸ LIVE SCAN (needs a connected broker)
…get told when a condition happens?	ALERTS — build it, name it, SAVE WATCH; use **Test** to prove the notification arrives
…paper-trade it?	DEPLOY, leave Mode on **Paper — simulated, safe**, set Quantity
…go live?	Same dialog, Mode = **Live — real money**, pick the broker account, set Quantity and both limits
…run an option chain as one fleet?	Deploy ▸ + **Build F&O Options Basket…** ▸ underlying, expiry, N ▸ Fetch strikes
…see what I'm holding?	DASHBOARD ▸ OPEN POSITIONS (per-leg for F&O)
…close everything now?	DASHBOARD ▸ ⏻ **FLATTEN ALL** (confirms; this machine only)
…cap my risk across all algos?	DASHBOARD ▸ **RISK** — exposure and concurrent-position caps
…account for every trade?	HISTORY ▸ filter, then **EXPORT CSV**

Honest notes on this build

- **Deployments run in your terminal.** Close it and they stop; fleet controls reach only this machine.
- **Backtests fill at the bar open** and do not model execution style — the lab says so above the results. SHORT and scale-in *are* modelled.
- **A basket verdict counts only symbols that traded** — read the *"N never traded"* tail before believing a percentage.
- **Native broker brackets are off by default**, so live stops are client-side and enforced only while the engine runs.
- **Exposure and concurrent-position caps default to "No limit"** — set them before trading live.
- **The AI assistant spends credits** and fills in defaults you did not state; review the sentence line and safety-net row after every edit.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Watchlist Workstation

A complete, screen-by-screen walkthrough of the Watchlist tab — many named lists, a live table or card grid, and prices streamed straight from your broker.

`>_ multiple lists · table & cards · live broker streaming · CSV & one-click backtest`

Covers · **the watchlist sidebar** · **the table & cards views** · **adding & removing symbols** live pricing · **CSV import / export** · backtest & AI tools

Demonstrated live on the Fincept Terminal **Watchlist** tab.

Fincept Research Desk · Professional Edition v5.0 · Prepared for institutional clients

Contents

7 SECTIONS

GETTING ORIENTED

01 What the Watchlist tab is 3

02 Anatomy — the four regions 4

USING IT

03 Your watchlists — the sidebar 5

04 Adding symbols & the two views 6

05 Live prices, CSV & backtest 7

REFERENCE

06 AI tools & where lists appear 8

07 Quick reference & honest notes 9

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal, taken on the live Watchlist tab with a 53-symbol "Nifty 50" list streaming live prices from a connected broker. Control labels are quoted exactly as they appear on screen.

01 What the Watchlist tab is

OVERVIEW

The **Watchlist** tab is where you keep the symbols you care about and watch them move. You can hold **many named lists** side by side — say "Nifty 50", "Tech", "Earnings this week" — and view each as a dense sortable **table** or a scannable grid of **cards**, with prices streaming live from your connected broker.

It's built for speed: paste a batch of tickers to build a list in seconds, drag a symbol out to any other panel, export to CSV, or send the whole list to the Backtester in one click. When a broker is connected, quotes tick in real time; otherwise they fall back to Fincept's market-data feed.

AT A GLANCE, IN WATCHLIST YOU CAN…

- Keep **multiple named lists**, each with its own color.
- Add many symbols at once (comma-separated) or one at a time.
- Switch each list between a **table** and a **cards** grid.
- Watch **live prices** stream from your broker.
- Sort the table by price, change, %, volume — any column.
- **Import / export** a list as CSV.
- **Backtest** an entire list with one click.
- Drag a symbol out to charts, tickets and other panels.

How to get here

Open the **Navigate** menu (or the command bar, top-left) and type `watchlist`, then pick `Watchlist`. The tab remembers which list you had open and whether you were in table or cards view, and reopens exactly there.

02 Anatomy — the four regions

LAYOUT

The screen has four parts: a **sidebar** of your lists on the left, and — filling the rest — a **toolbar**, an **add bar**, and the **content area** (table or cards).

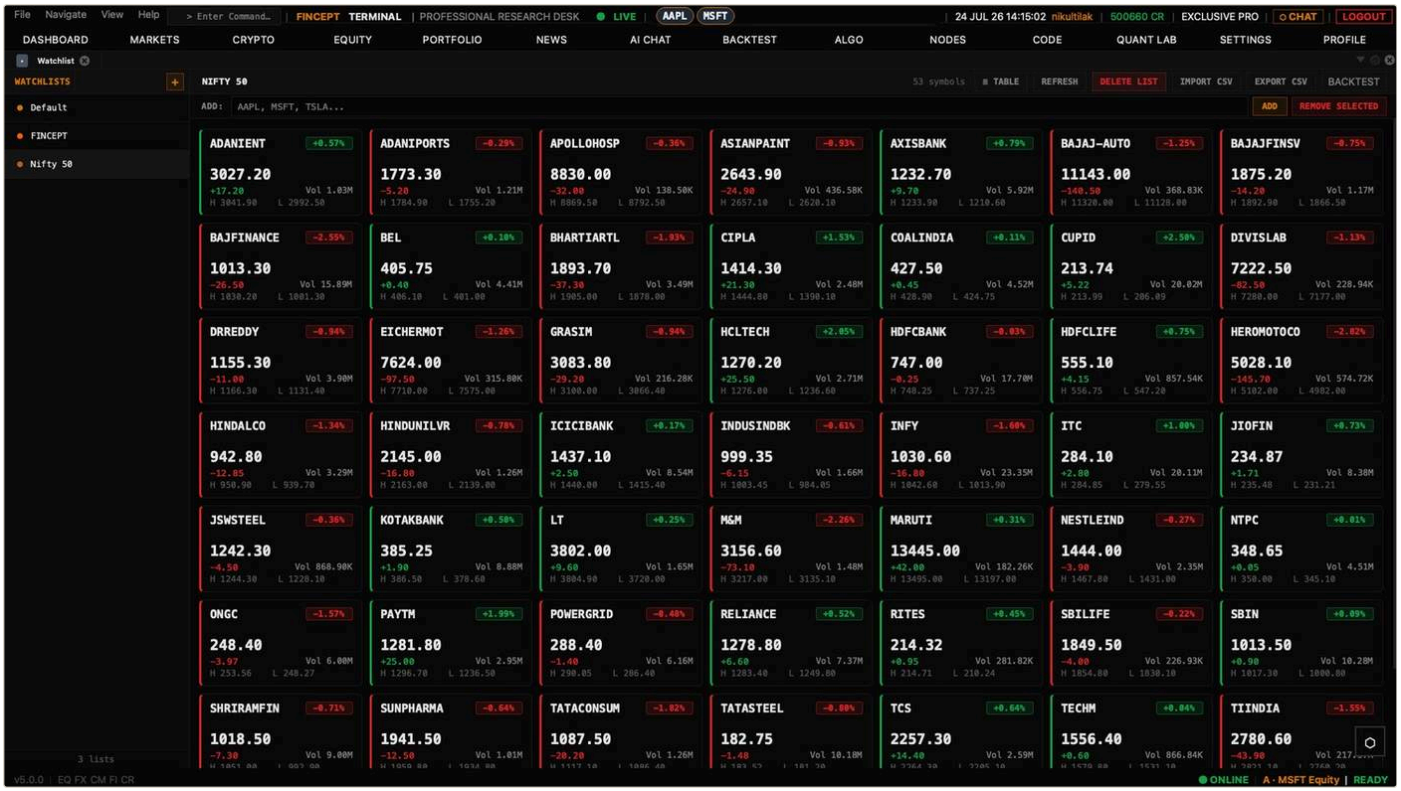


Fig 1. The Watchlist tab in cards view — the "Nifty 50" list (53 symbols) streaming live. Left: the sidebar of three watchlists. Top: the toolbar and the add bar. Center: the responsive card grid.

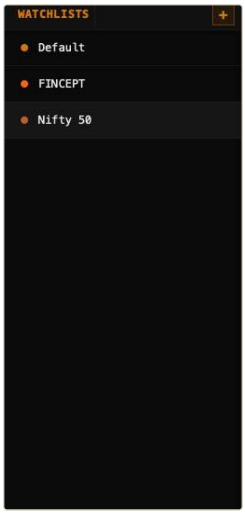
Region	What it holds
Sidebar	Your watchlists, each with a color dot; a + button to make a new one; a "**N lists**" count at the bottom (§3).
Toolbar	The active list's name, its symbol count, the **table / cards** toggle, and **REFRESH · DELETE LIST · IMPORT CSV · EXPORT CSV · BACKTEST** (§3, §5).
Add bar	The **ADD** input for new symbols, plus **ADD** and **REMOVE SELECTED** (§4).
Content	The active list as a **table** or a **cards** grid (§4).

Fig 2. The toolbar: list name · symbol count · view toggle · REFRESH · DELETE LIST · IMPORT/EXPORT CSV · BACKTEST.

> **THE VIEW TOGGLE SHOWS THE *TARGET***
>
> The toggle button names the view you'll *switch to*, not the one you're in: it reads ≡ **TABLE** while you're looking at cards, and ▦ **CARDS** while you're looking at the table.

03 Your watchlists — the sidebar

SIDEBAR



The sidebar holds every list you've made. Each row shows a **color dot** and the list name; the selected list is highlighted, and a running **"N lists"** count sits at the bottom.

Making & switching lists

Press the + button (top-right of the sidebar) to create a new list — a small **New Watchlist** dialog asks for a name. Click any row to switch to that list; its symbols load instantly and the toolbar's DELETE / IMPORT / EXPORT buttons light up.

The terminal always keeps at least one list — if you've never made one, a **Default** list is created for you.

Deleting a list

DELETE LIST (in the toolbar) removes the active list and all its symbols, after a confirmation. It's the one destructive action here and it can't be undone.

> GOOD TO KNOW
>
> **No rename, no color picker in the UI.** A list's name is set when you create it — there's no rename control here, and new lists get a default orange dot (the auto-created Default list is amber). The color dot is a visual marker only. (The AI assistant *can* rename or re-color a list — see §6.)

04 Adding symbols & the two views

SYMBOLS · VIEWS

Adding & removing symbols

Fig 3. The add bar. Type one ticker or several separated by commas, then **ADD** (or press Enter).

Type a symbol — or a whole batch, comma-separated like `AAPL, MSFT, TSLA` — into the **ADD** box and press **ADD** or Enter; every ticker is added at once. To remove one, select its row in the table and press **REMOVE SELECTED**. You can also **drag a symbol in** from another panel to add it, or **drag one out** to a chart or ticket.

Table view

The table is dense and sortable — click any header to sort by that column. Columns: **SYMBOL · NAME · PRICE · CHANGE · CHG % · HIGH · LOW · VOLUME**. Change and percent are colored green (up) / red (down); a symbol with no quote yet shows `--`.

SYMBOL	NAME	PRICE	CHANGE	CHG %	HIGH	LOW	VOLUME
ADANIENT		3026.60	+16.60	+0.55%	3041.90	2992.50	1.03M
ADANIPORTS		1773.30	-5.20	-0.29%	1784.90	1755.20	1.21M
APOLLOHOSP		8821.50	-40.50	-0.46%	8869.50	8792.50	139.30K
ASIANPAINT		2643.50	-25.30	-0.95%	2657.10	2620.10	437.14K
AXISBANK		1232.90	+9.90	+0.81%	1233.90	1210.60	5.94M
BAJAJ-AUTO		11145.50	-138.00	-1.22%	11320.00	11128.00	369.28K
BAJAJFINSV		1875.50	-13.90	-0.74%	1892.90	1866.50	1.18M
BAJFINANCE		1013.80	-26.00	-2.50%	1030.20	1001.30	15.90M
BEL		405.75	+0.40	+0.10%	406.10	401.00	4.43M
BHARTIARTL		1894.60	-36.40	-1.89%	1905.00	1878.00	3.50M
CIPLA		1414.80	+21.80	+1.56%	1444.80	1390.10	2.48M

Fig 4. Table view — sortable columns, right-aligned figures, live green/red changes. (NAME is blank when the broker feed doesn't supply one.)

Cards view

Cards are easier to scan. Each card shows the **symbol**, a colored **percent pill**, the **price**, the day's **change** and **volume**, and a **H / L** footer, with a green or red accent stripe down the left edge. The grid reflows to fit the width, and each card updates in place as prices tick.

Fig 5. Three cards up close — symbol, percent pill, price, change, volume, H/L, and the green/red accent stripe.

> SELECTING & SORTING — TWO THINGS TO KNOW
>
> **REMOVE SELECTED and drag-out act on the table's selected row**, so do them in **table** view. And if you've clicked a header to re-sort, select the row again before removing — after a sort, "selected row" and the underlying symbol can drift apart.

05 Live prices, CSV & backtest

DATA · TOOLS

Where prices come from

When a broker is connected, the watchlist streams **live quotes straight from it** — the same real-time feed as your trading screens. For symbols the broker can't price (or when no broker is connected), it falls back to Fincept's market-data feed. Broker ticks always win, and the feed only runs while the tab is open (it detaches when you leave, so it never streams in the background).

REFRESH re-pulls quotes for the active list on demand.

Import & export (CSV)

Action	What happens
EXPORT CSV	Saves the list as `<name>.csv` with columns SYMBOL, NAME, PRICE, CHANGE, CHG %, HIGH, LOW, VOLUME — using the current live quotes.
IMPORT CSV	Reads a CSV, requires a **SYMBOL** column (NAME optional; other columns ignored), and adds each new symbol — **skipping duplicates**. It reports how many were imported and skipped.

Backtest the whole list

BACKTEST bundles every symbol in the active list into a portfolio and hands it to the **Backtester**, switching you straight to that screen with the list pre-loaded. (If the list is empty, nothing happens.) It's the fast path from "here's my universe" to "how would it have performed."

> **LISTS ARE YOURS, EVERYWHERE**
>
> Your watchlists back the droppable **Watchlist dashboard widget** and are readable by the AI assistant (§6), so the lists you build here show up across the terminal.

06 AI tools & where lists appear

AI · REUSE

Watchlists aren't confined to this tab. They're stored per-account and surfaced to the AI assistant and to other parts of the terminal.

Reading & editing lists from AI chat

The assistant can read and edit your watchlists through six dedicated tools — e.g. "add NVDA and AMD to my Tech list" or "make a new list called Earnings and add these five tickers."

Tool	What it does
`get_watchlists`	Lists every watchlist with its symbols, names and colors.
`create_watchlist`	Makes a new list (name, optional description & color).
`update_watchlist`	Renames or re-colors a list — the one way to rename (the UI can't).
`delete_watchlist`	Deletes a list.
`add_to_watchlist`	Adds a symbol (defaults to your first list; makes a Default if none).
`remove_from_watchlist`	Removes a symbol from one list — or from all if none is named.

The same lists, elsewhere

Surface	How lists appear
Dashboard — Watchlist widget	A droppable tile bound to one of your lists, with the same live quotes.
Symbol linking / drag	Selecting or dragging a symbol here broadcasts it to linked panels (charts, tickets).
Backtester	The BACKTEST button sends the whole list over as a portfolio (§5).

> **ONE SET OF LISTS**
>
> Whether you edit a list here, from the dashboard widget, or via the AI, you're editing the same underlying watchlists — changes show up everywhere.

07 Quick reference & honest notes

REFERENCE

How do I…

Task	Do this
Make a new list	Click + in the sidebar → name it → OK.
Switch lists	Click a row in the sidebar.
Add symbols	Type in **ADD** (comma-separate for several) → **ADD** or Enter.
Remove a symbol	Select its row in **table** view → **REMOVE SELECTED**.
Switch table / cards	The toggle button (shows the view you'll switch to).
Sort	Click a column header in table view.
Refresh prices	**REFRESH**.
Import / export	**IMPORT CSV** / **EXPORT CSV** in the toolbar.
Backtest a list	**BACKTEST** → opens the Backtester with the list loaded.
Delete a list	**DELETE LIST** → confirm.

Honest notes & gotchas

- **Remove / drag act on the table's selected row.** Do them in table view; in cards view there's no selection, so they use the table's last selection.
- **Sorting can shift the selection.** After clicking a header to sort, re-select a row before removing it, or you may remove a different symbol.
- **No rename, no color picker.** A list's name is fixed at creation; new lists get a default color dot.
- **REMOVE SELECTED has no confirmation** (unlike DELETE LIST) and removes one symbol at a time.
- **Symbols aren't validated on add.** A typo becomes a permanent `—` row until you remove it.
- **NAME can be blank** when the broker feed doesn't supply a company name.
- **Live streaming runs only while the tab is open** — it detaches when you leave, so it doesn't stream in the background.
- **BACKTEST on an empty list does nothing** (no message).

> **FINCEPT RESEARCH DESK**
>
> Prepared for institutional clients · Professional Edition v5.0 · Captured live on the Watchlist tab with a broker-streamed 53-symbol list. Control labels quoted verbatim from the running application.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Equity Research

A complete, screen-by-screen walkthrough of the Equity Research workstation — a single-symbol analyst desk spanning price, fundamentals, valuation, quality, technicals, peers, ownership, macro, scenarios, SEC filings, news, sentiment and an AI research copilot.

`>_ fifteen sections · one symbol · live data`

Covers · **15 sections + research dock** · **live yfinance & SEC EDGAR** · **valuation & quality** **technicals · peers · ownership · macro** · news, sentiment & AI copilot · 45 agent tools

Demonstrated live on the Fincept Terminal Equity Research workstation.

Contents

11 SECTIONS

GETTING ORIENTED

Overview — what the Equity Research workstation is 3

1 Anatomy of the workstation (title bar, quote bar, rail, dock) 4

ANALYSING THE COMPANY

2 OVERVIEW — the one-screen snapshot 6

3 FUNDAMENTALS — Financials · Valuation · Analysis · Quality 8

4 MARKET — Technicals · Peers · Ownership · Cross-Asset 12

5 CONTEXT & RESEARCH — Macro · Quant Lab · Scenario · Filings 16

6 FLOW — News · Sentiment 19

DOCK, ACTIONS & REFERENCE

7 The research dock — Notes · Copilot · Alerts 21

8 Toolbar actions — Buy/Sell · Backtest · CSV export 21

9 Where the numbers come from (data & provenance) 23

10 AI agent tools (45 MCP tools) 23

11 Quick reference & important notes 25

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal, pictured with live NVDA data. Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

What the Equity Research workstation is

OVERVIEW

Equity Research is a **single-symbol analyst workstation**. You load one ticker and the whole screen — fifteen switchable sections plus a research dock — re-frames that company from every angle: its price and fundamentals, valuation and cash quality, technical signals, peers, ownership, macro exposure, forward scenarios, SEC filings, news and sentiment, and an AI copilot that reads the terminal's own data.

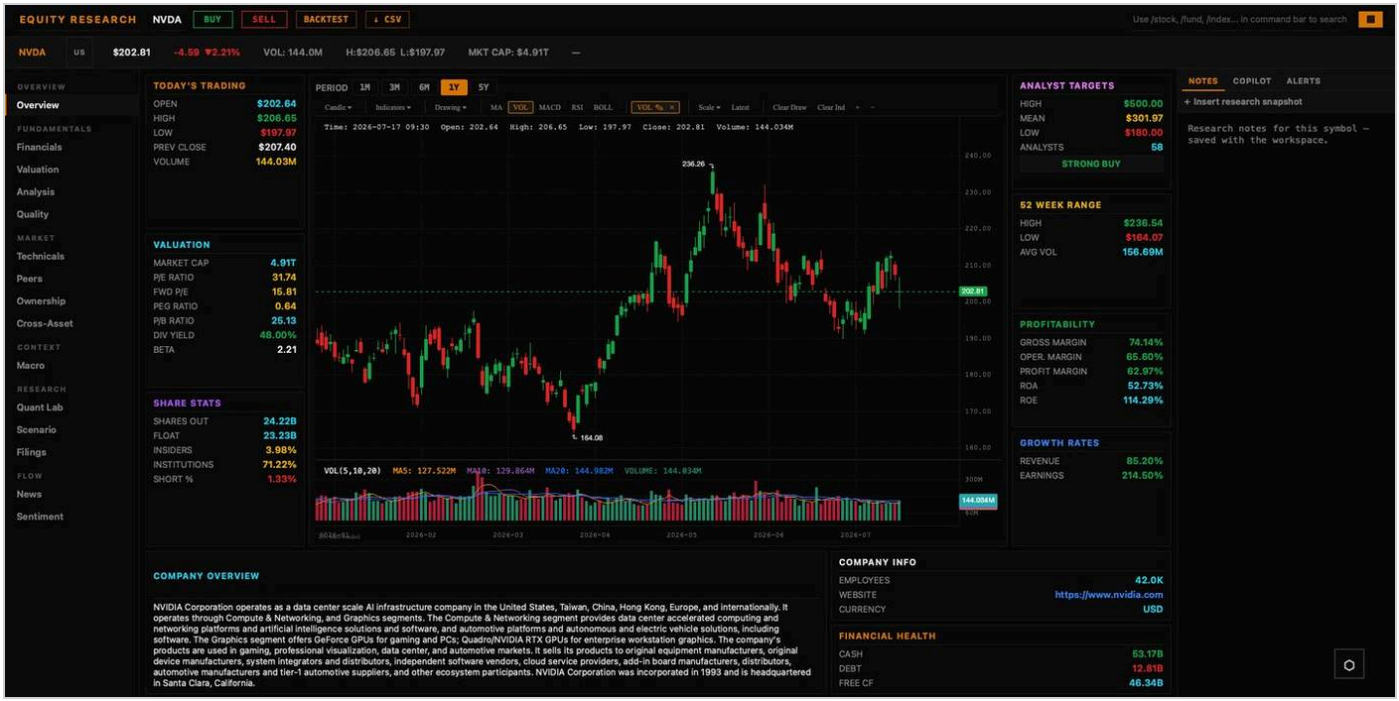
The data is live, and mostly local: quotes, fundamentals, statements, technicals, peers and news are pulled from yfinance through the app's embedded Python — not a Fincept server — with SEC EDGAR filings fetched directly from the SEC. Where a matching broker is connected, the price ticks live and technical candles come from the broker. The only action that consumes AI credits is the Copilot analyst. Load a symbol from the command bar (/stock AAPL), by dragging a ticker onto the screen, or from any linked panel; the default on open is AAPL (this guide pictures NVDA).

AT A GLANCE, IN EQUITY RESEARCH YOU CAN…

- See a full **one-screen snapshot** of any stock — session prices, valuation multiples, share structure, analyst targets, profitability, growth, a price chart and the company profile.
- Read the **three annual statements** (income / balance / cash flow) with derived ratios, DuPont, and trend charts — and export any statement to CSV.
- Run a live, editable **DCF & reverse-DCF** (Valuation Lab), a plain-English six-card verdict (Analysis), and forensic scores — **Altman Z, Piotroski F, Beneish M** (Quality).
- Get a **TradingView-style technical consensus** from ~24 indicators, compare 16 fundamentals across peers, and inspect institutional/insider ownership.
- Measure **macro beta & capture** vs the home index, realized-vol term structure with a theoretical option, quant factor stats, and DCF stress scenarios.
- Browse recent **SEC EDGAR filings**, the latest news (provider-switchable), and a blended market-sentiment read.
- Keep per-symbol **research notes**, set price alerts, and ask an **AI analyst copilot** — then trade, backtest, or export from the toolbar.

1 Anatomy of the workstation

ORIENTATION



The workstation, Overview section. ① Title bar ② Quote bar (identity strip) ③ Grouped section rail ④ Canvas ⑤ Research dock (Notes / Copilot / Alerts).

Region	What it shows / does
① **Title bar**	Amber **EQUITY RESEARCH** · the current **symbol** (drag it to broadcast to other panels) · **BUY/SELL** (appear only when a connected broker can trade that market) · **BACKTEST** (hands the symbol to the Backtesting screen) · ↓ **CSV** (price-history export dialog) · a hint (`Use /stock, /fund, /index… to search`) · the ▤ dock toggle.
② **Quote bar**	Symbol · a **market chip** (US / IN / GLOBAL, tooltip shows the fundamentals source) · price · change% · VOL · H/L · MKT CAP. Refreshes every 30 s (or on a live broker tick for connected Indian brokers).
③ **Section rail**	Fifteen sections in six labelled groups — **OVERVIEW**; **FUNDAMENTALS** (Financials, Valuation, Analysis, Quality); **MARKET** (Technicals, Peers, Ownership, Cross-Asset); **CONTEXT** (Macro); **RESEARCH** (Quant Lab, Scenario, Filings); **FLOW** (News, Sentiment). Click to switch the canvas.
④ **Canvas**	The active section. Every section is driven by the one loaded symbol and fetches its own data on first view; a loading overlay covers it while data arrives.
⑤ **Research dock**	A collapsible right panel (toggle ▤) with three tabs: **Notes** (per-symbol, saved with the workspace), **Copilot** (AI analyst), **Alerts** (price alerts).

> **How data loads.** Every section fetches on demand from live yfinance (local Python) and caches briefly (quote 30 s, info 5 min, candles 2 min, news 3 min, EDGAR 30 min). Sections are read-only except **Valuation** (editable DCF), **Peers** (editable peer list), the dock **Notes/Alerts**, and the toolbar actions. Full provenance is in §9.

Loading a symbol & keeping context

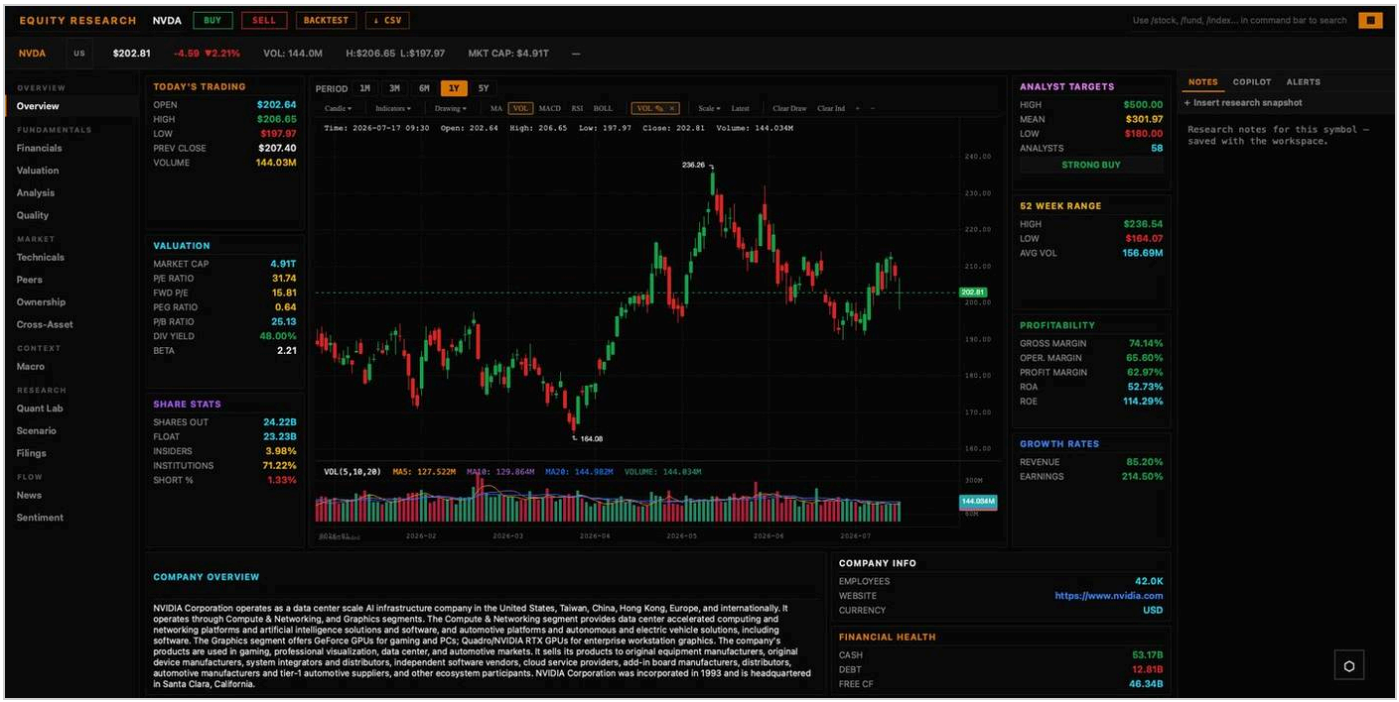
- **Command bar** — `/stock`, `/fund`, `/index` search (server-side), then the pick loads here.
- **Drag & drop** — drop any ticker chip onto the screen to jump the research view.
- **Linked group** — the loaded symbol broadcasts to Watchlist / Equity Trading / News so they follow along.

- **State** — the symbol, active section, per-symbol notes, peer list, news provider and alerts persist to the cloud (JWT-synced), not to a local file.

2 OVERVIEW — the one-screen snapshot

DEFAULT VIEW

The default landing view: a single scrollable snapshot of one stock's day trading, valuation, share structure, analyst targets, ranges, profitability, growth, price chart, company profile and balance-sheet health.



OVERVIEW (NVDA) — left column (Today's Trading / Valuation / Share Stats), the candlestick chart with period buttons, right column (Analyst Targets / 52-Week / Profitability / Growth), and the company panels below.

Panel	Contents
TODAY'S TRADING	Open · High · Low · Prev Close · Volume (session).
VALUATION	Market Cap · P/E · Fwd P/E · PEG · P/B · Div Yield · Beta.
SHARE STATS	Shares Out · Float · Insiders % · Institutions % · Short %.
Chart	Candlestick history; **PERIOD** buttons **1M · 3M · 6M · 1Y** (default) · **5Y** reload candles. (Interactive KLineChart or a painted canvas depending on the build.)
ANALYST TARGETS	High / Mean / Low targets · analyst count · a colour-coded consensus badge (STRONG BUY … STRONG SELL).
52 WEEK RANGE	52-week High / Low · Average Volume.
PROFITABILITY	Gross / Operating / Profit margin · ROA · ROE.
GROWTH RATES	Revenue & Earnings growth.
COMPANY OVERVIEW / INFO / FINANCIAL HEALTH	Business description · employees / website / currency · cash / debt / free cash flow.

Controls & states. The only interactive controls are the five period buttons — everything else is read-only. Each field shows – until data lands; a per-leg overlay clears once quote+info+historical resolve, and shows `NO DATA FOR <symbol>` if every leg fails. The website is plain text (not a link, truncated to 40 chars).

At a glance, in OVERVIEW you can…

- Read a full session + valuation + ownership snapshot, chart the price over five lookbacks, and gauge sell-side consensus and company health — all on one screen.

3 FUNDAMENTALS

RAIL GROUP

Four sections that turn filed statements and fundamentals into statements, ratios, valuation and forensic quality scores.

3.1 Financials — the three statements

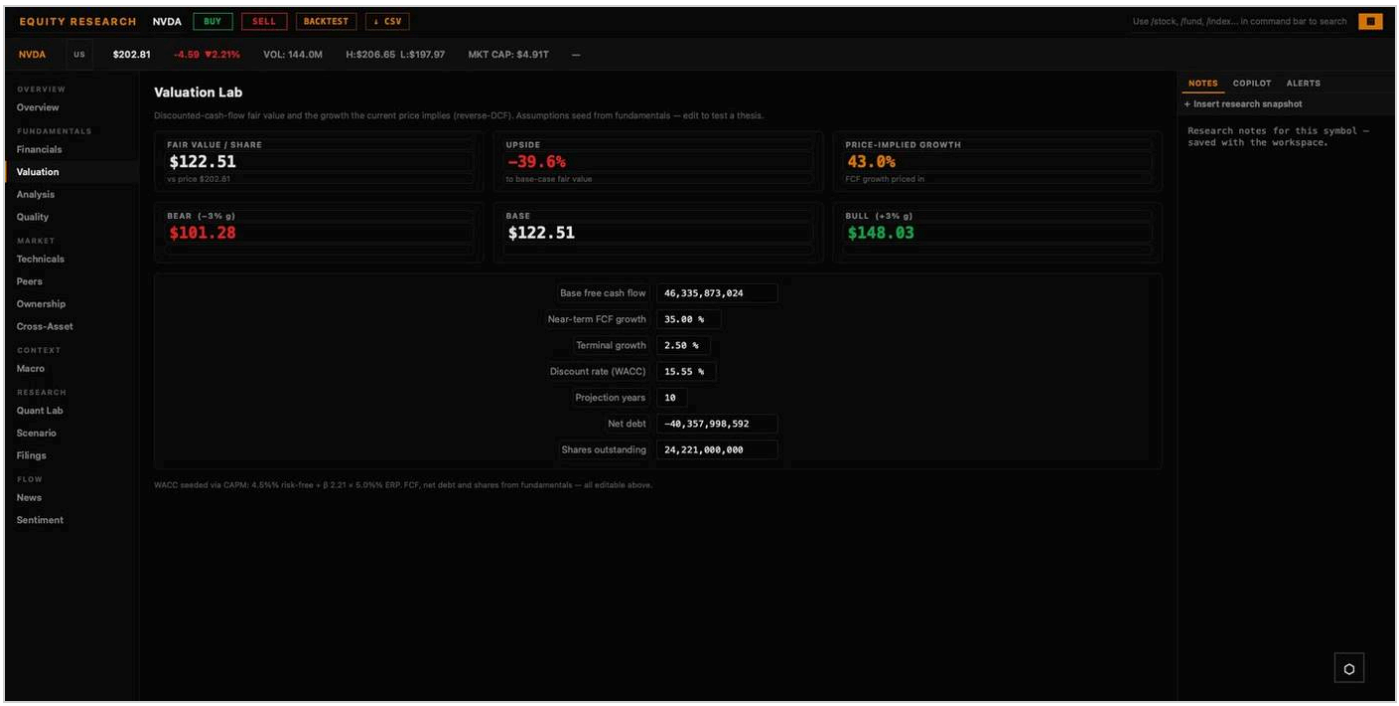
FINANCIALS (NVDA) — Income Statement view: key-metric cards, revenue/margin trend charts, return metrics + DuPont, and the full raw statement table. Toggle to Balance Sheet / Cash Flow; EXPORT CSV top-right.

Control	Type	What it does
Income Statement / Balance Sheet / Cash Flow	toggle	Switches the visible annual statement page (mutually exclusive).
EXPORT CSV	button	Writes the currently visible statement table to a CSV in the managed file store.

Each page shows headline metric cards (revenue, income, EBITDA, assets/liabilities/equity, operating/investing/financing CF, FCF, capex, dividends, buybacks), derived margins & ratios, ROE/ROA/ROIC/ROCE with a **DuPont** decomposition, liquidity/leverage ratios, up to five trend charts, and the full raw statement table (every yfinance line item, negatives in red).

> **Honest caveats.** Statements are **annual only** (no quarterly toggle). If a fetch fails the loading overlay spins indefinitely (no error path). The CSV filename is prefixed `equity_`, not the ticker. ROIC uses a hardcoded 0.75 after-tax factor; interest coverage uses EBITDA/interest.

3.2 Valuation — the Valuation Lab (live DCF)

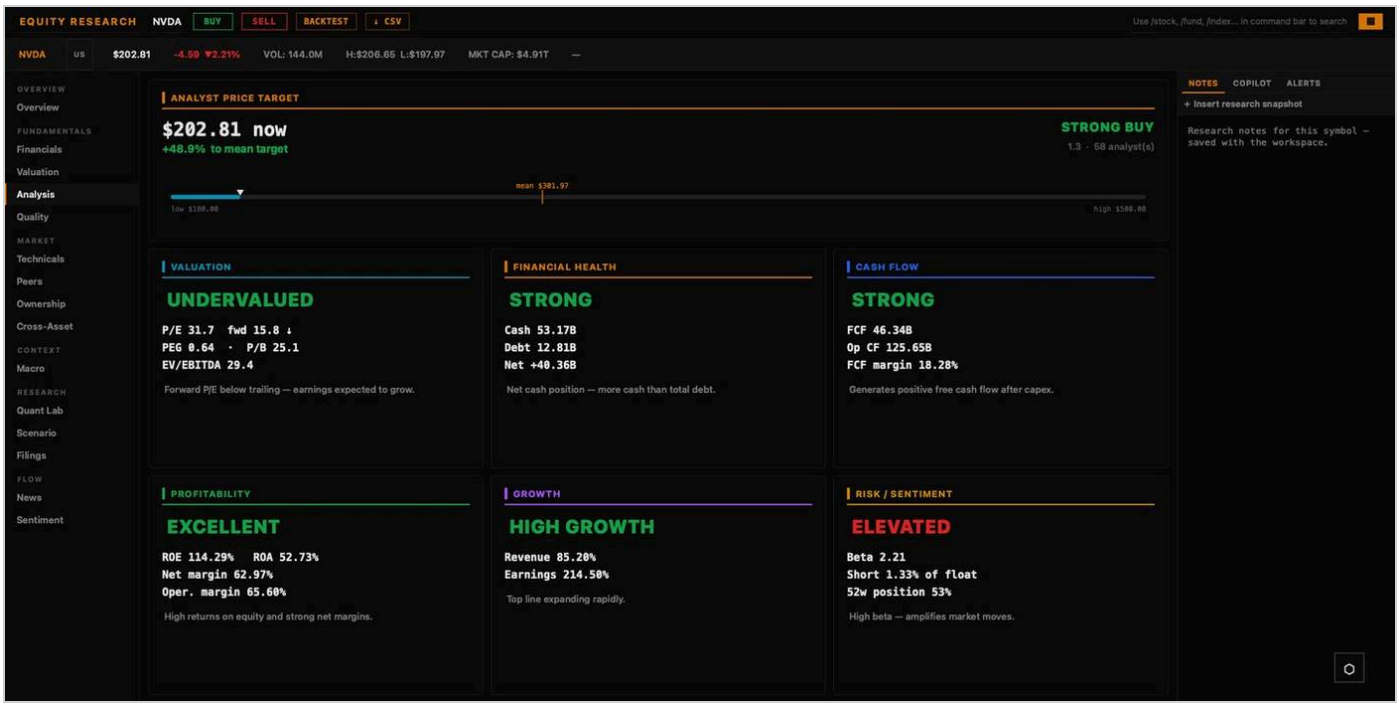


VALUATION Lab (NVDA) — fair value / upside / price-implied-growth tiles, a bear/base/bull spread, and seven editable DCF assumptions that recompute live.

A self-contained two-stage DCF + reverse-DCF, **seeded from fundamentals but fully editable**. Seven **QDoubleSpinBox** inputs — Base free cash flow · Near-term FCF growth · Terminal growth · Discount rate (WACC) · Projection years · Net debt · Shares outstanding — recompute all six tiles on every keystroke. Tiles: **Fair Value / Share**, **Upside** vs live price, **Price-Implied Growth** (reverse-DCF), and a **Bear (–3% g) / Base / Bull (+3% g)** spread. WACC is seeded via CAPM (4.5% risk-free + β × 5% ERP, clamped 6–20%).

> **Note.** The "football field" is three tiles, not a plotted range; it swings growth ±3 pp only (no WACC/multiples scenarios). Terminal growth (2.5%) and 10 projection years are fixed seeds. No comps/multiples method and no save/export of a scenario.

3.3 Analysis — six plain-English verdicts

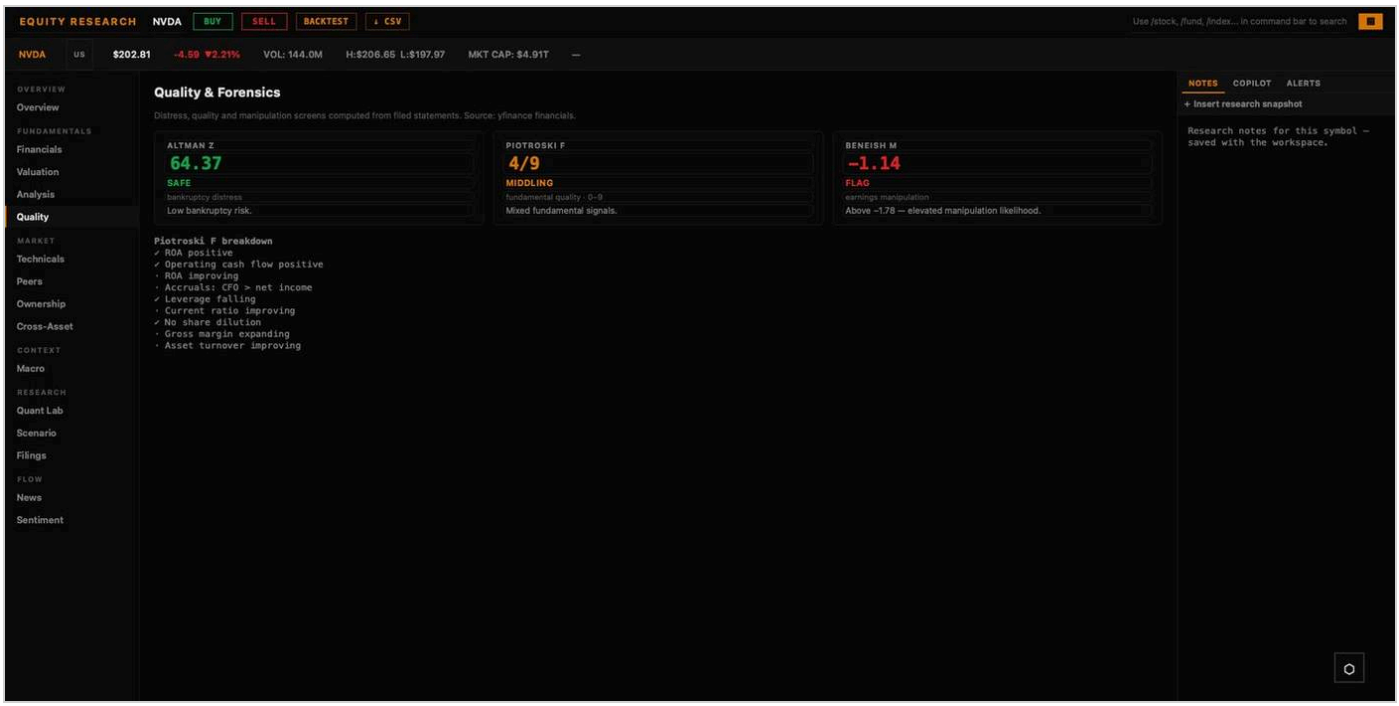


ANALYSIS (NVDA) — an analyst price-target gauge plus six colour-coded rating cards (Valuation, Financial Health, Cash Flow, Profitability, Growth, Risk/Sentiment).

A pure read-out (no controls): a painted **price-target gauge** (low→price bar, mean tick, "now" marker, % upside to mean, consensus badge + analyst count) and six **verdict cards**, each a one-word rating (green/amber/red), up to three driving metrics, and a plain-English rationale — Valuation (UNDERVALUED…EXPENSIVE), Financial Health (STRONG…STRETCHED), Cash Flow, Profitability, Growth, Risk/Sentiment.

> **Caveat.** Thresholds are absolute screening signals — **not sector- or size-adjusted** (the code itself notes "not investment advice"); on a failed info fetch the overlay spins (no error path).

3.4 Quality — forensic accounting scores



QUALITY & Forensics (NVDA) — Altman Z (distress), Piotroski F (fundamental quality, 0–9), Beneish M (earnings-manipulation), each with a colour-coded verdict, plus the 9-point Piotroski checklist.

Three classic forensic screens computed in-process from the filed statements: **Altman Z** (SAFE / GREY ZONE / DISTRESS at 2.99 / 1.81), **Piotroski F** (STRONG / MIDDLING / WEAK at 7 / 4, with the full 9-check pass/fail list), and **Beneish M** (FLAG / CLEAN at −1.78). Display-only, no controls.

At a glance, in FUNDAMENTALS you can…

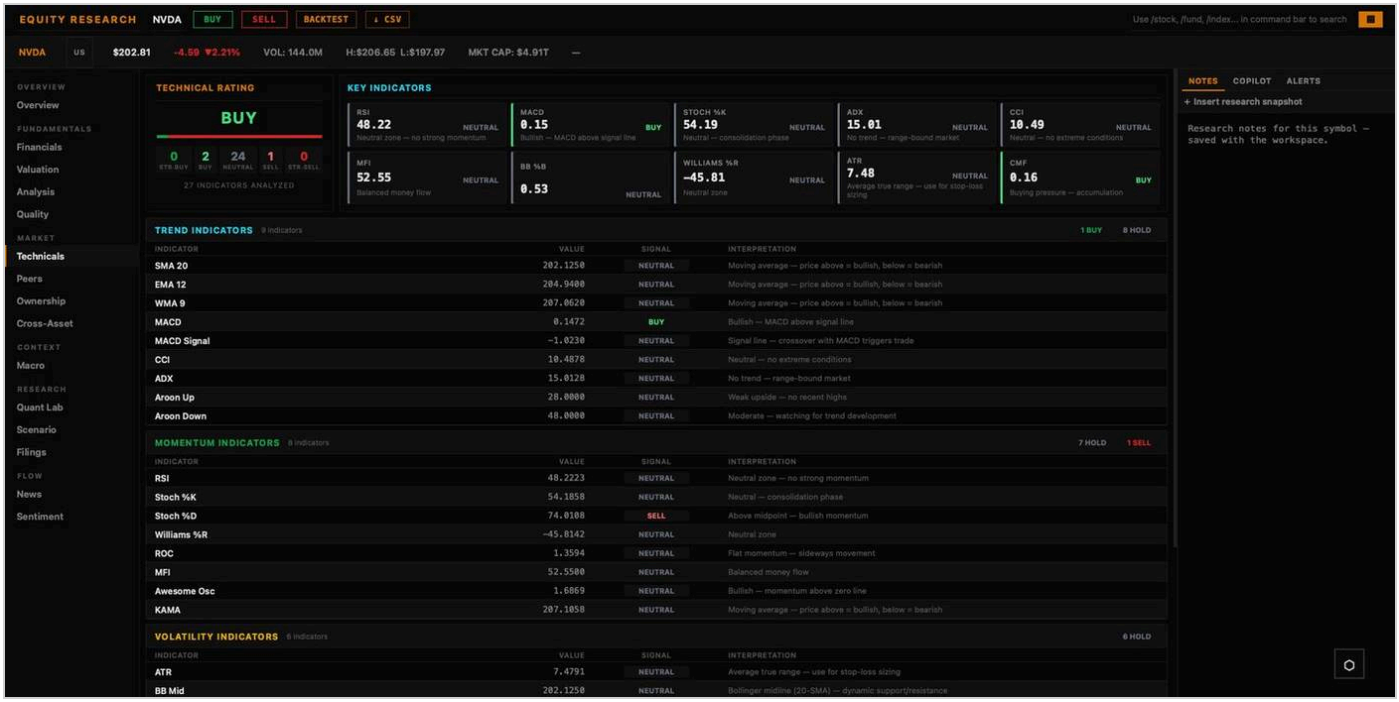
- Read the three annual statements with ratios, DuPont and trends (and export to CSV); run a live editable DCF with reverse-DCF and a bear/base/bull spread.
- Get six plain-English fundamental verdicts and three forensic scores (Altman Z, Piotroski F, Beneish M) at a glance.

4 MARKET

RAIL GROUP

Four sections on how the stock trades and who owns it — technical signals, peer multiples, ownership, and a cross-asset volatility/option read.

4.1 Technicals — indicator consensus

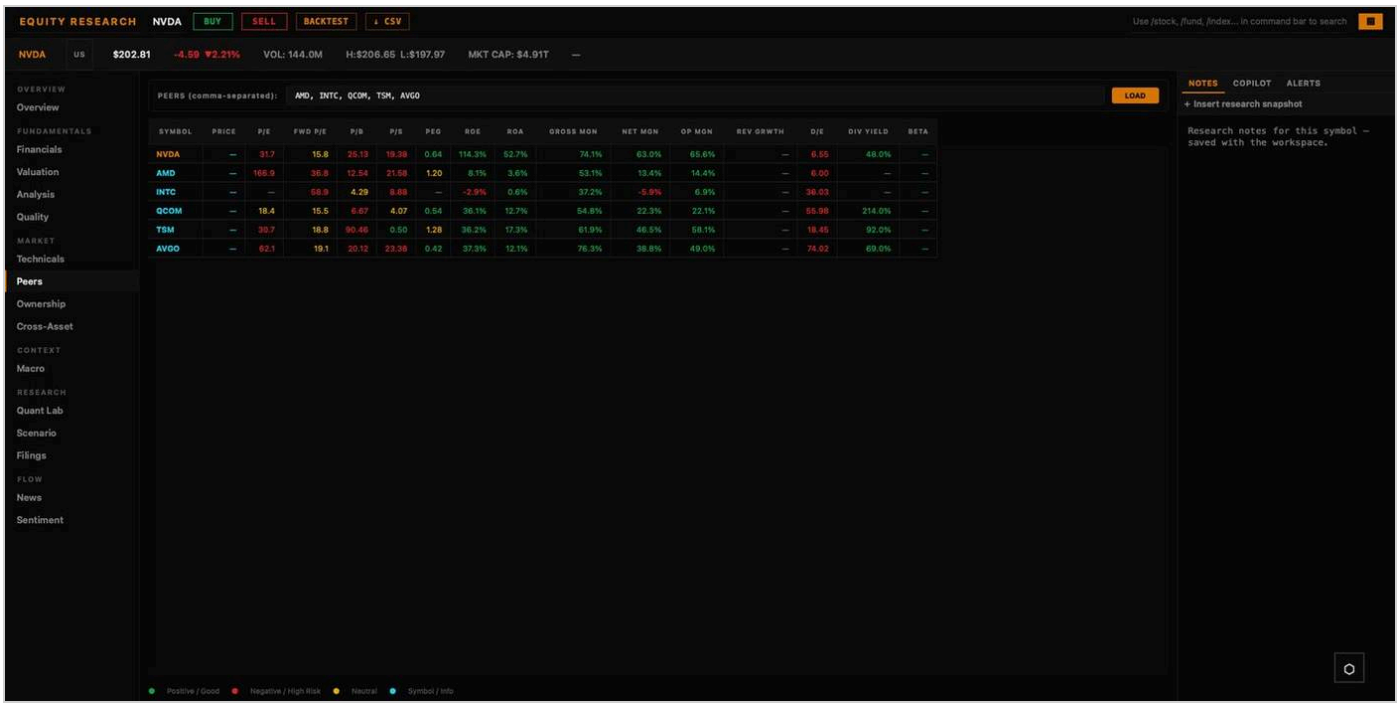


TECHNICALS (NVDA) — a TradingView-style rating (STRONG BUY…STRONG SELL) with a bull/bear tally, 10 key-indicator cards, and full Trend / Momentum / Volatility / Volume tables (value · signal · interpretation).

~24 indicators computed from 1-year daily candles and scored into one **consensus rating** plus a bull-ratio gauge and a 5-bucket count (Str.Buy / Buy / Neutral / Sell / Str.Sell). Ten **key-indicator cards** (RSI, MACD, Stoch %K, ADX, CCI, MFI, BB %B, Williams %R, ATR, CMF) sit above four category tables. Read-only.

> **Caveat.** Despite the name there is **no price chart** here; the lookback (1y/1d) and the indicator set are fixed. The headline gauge (unweighted) and the rating word (weighted) can differ slightly.

4.2 Peers — cross-company comparison

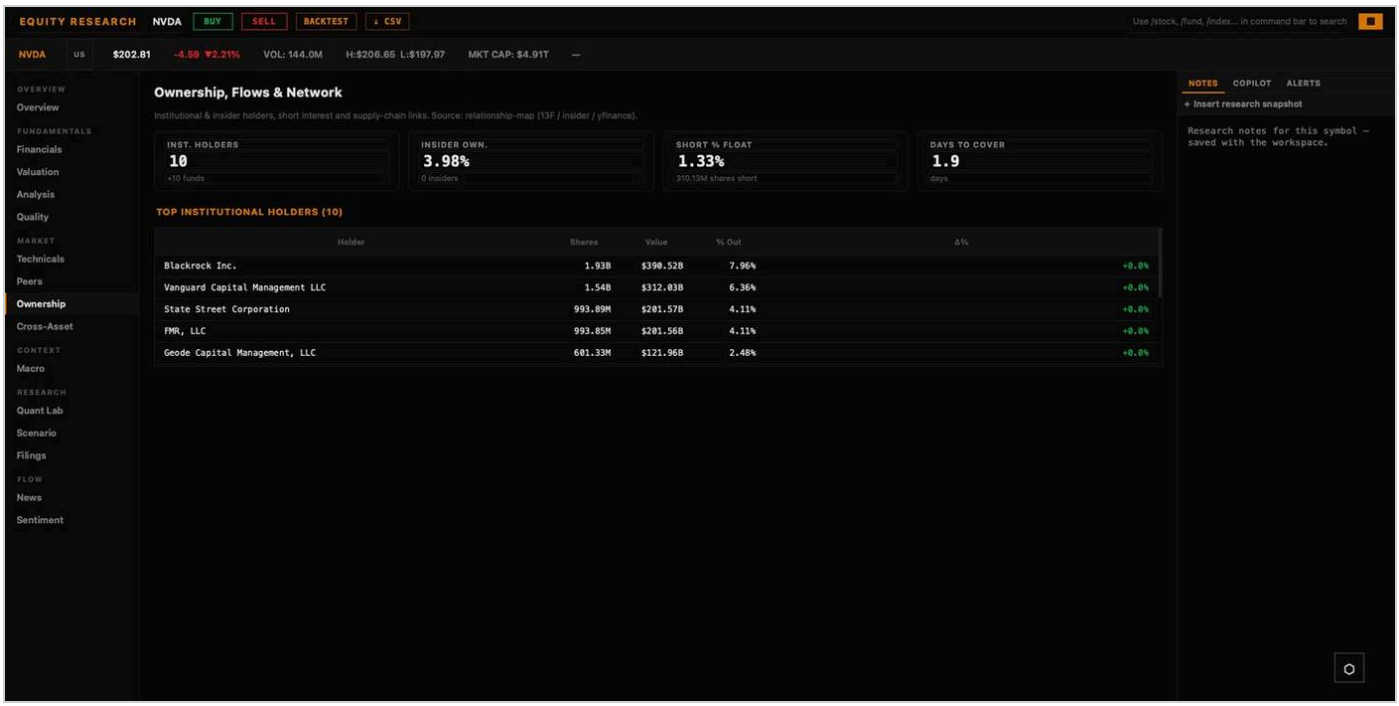


PEERS (NVDA vs its set) — an editable comma-separated peer list + LOAD, and a 16-metric colour-coded comparison grid with a good/neutral/risk legend.

Control	Type	What it does
PEERS (comma-separated)	text	Edit the peer set (auto-filled with a default list on load).
LOAD	button	Recomputes the comparison for the subject + parsed peers.
Column headers	sort	Click any of the 16 columns to sort.

16 columns: SYMBOL · PRICE · P/E · FWD P/E · P/B · P/S · PEG · ROE · ROA · GROSS/NET/OP MGN · REV GRWTH · D/E · DIV YIELD · BETA. Cells are green/amber/red by health; the subject ticker is amber.

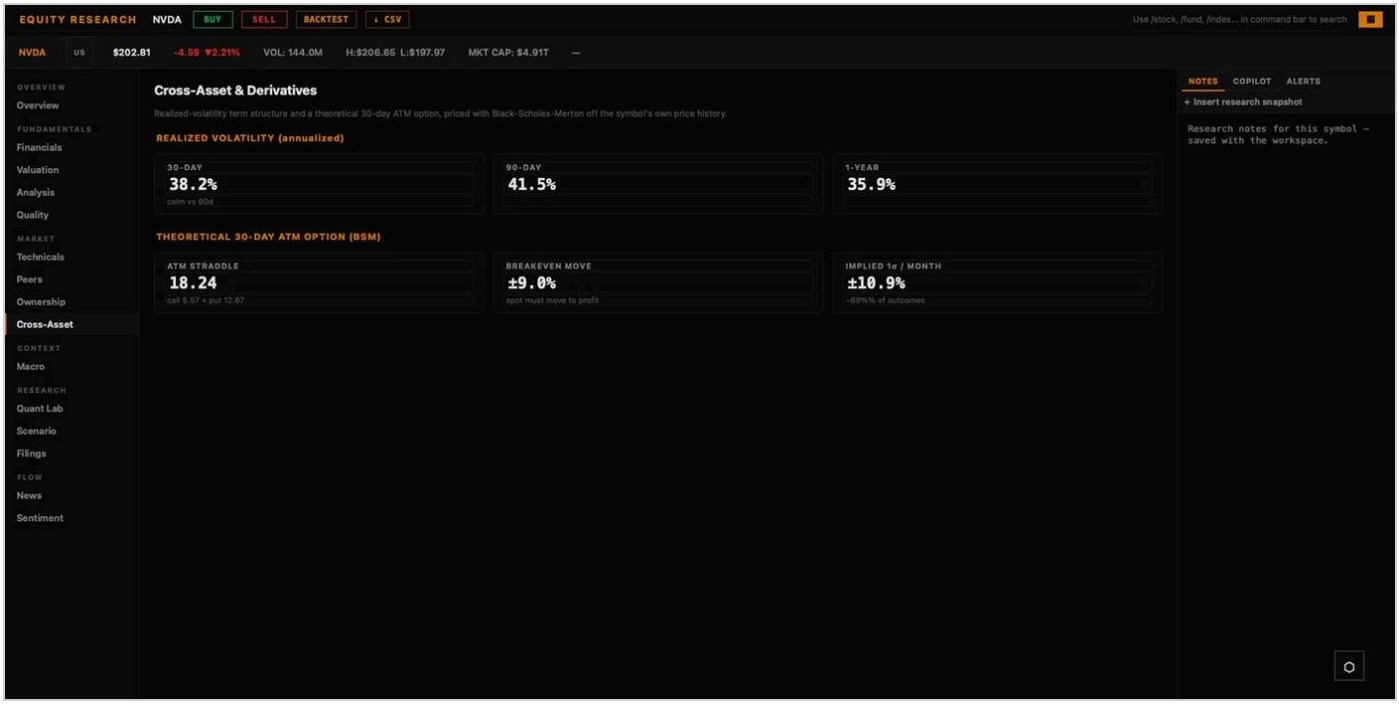
4.3 Ownership — holders, flows & network



OWNERSHIP (NVDA) — four summary tiles (institutional holders, insider ownership, short % float, days-to-cover) plus Top Institutional Holders, Insider Holders and Supply-Chain tables.

Read-only tiles + three tables from a single relationship-map fetch (13F / insider / yfinance): **Top Institutional Holders** (top 15 by value — shares, value, % out, Δ%), **Insider Holders** (title, shares, % out, last buy/sell), and **Supply Chain** (entity, relationship, revenue %). No filters or sorting.

4.4 Cross-Asset — volatility & a theoretical option



CROSS-ASSET (NVDA) — 30/90/365-day realized volatility and a theoretical 30-day ATM straddle priced by Black-Scholes-Merton off the symbol's own history.

Two tile rows, no controls: **Realized Volatility** (annualized 30-day / 90-day / 1-year, with an elevated-vs-90d note) and a **Theoretical 30-Day ATM Option** — ATM straddle (call+put), breakeven move, and implied ~1σ monthly range, computed locally by BSM.

> **Caveat.** Cross-Asset uses **realized vol as a proxy for implied** — it is a theoretical model off price history, not a live option-chain quote (risk-free 4.5% hardcoded).

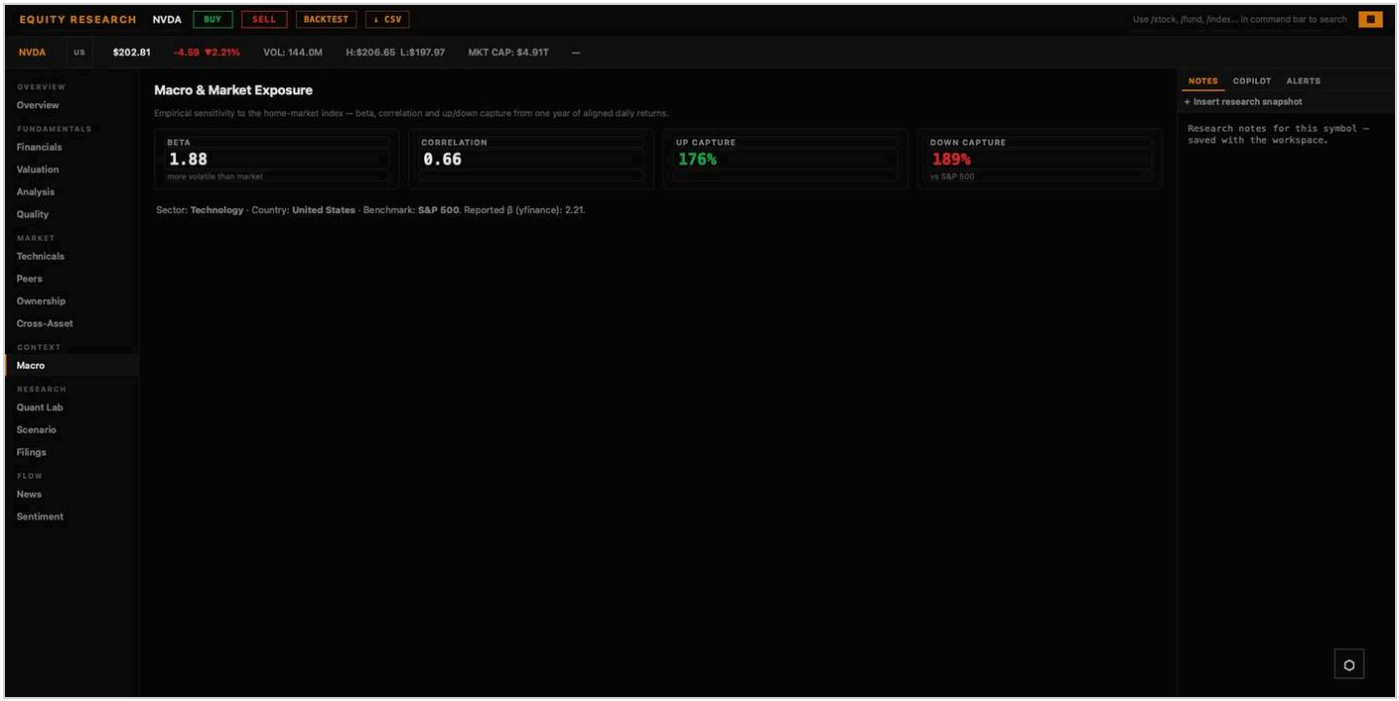
> **At a glance, in MARKET you can…**
>
> - Read a technical consensus from ~24 indicators, compare 16 fundamentals across an editable peer set, inspect institutional/insider ownership and supply-chain links, and gauge realized volatility + a theoretical option.

5 CONTEXT & RESEARCH

RAIL GROUPS

Macro exposure, quant factor stats, DCF stress scenarios, and SEC EDGAR filings.

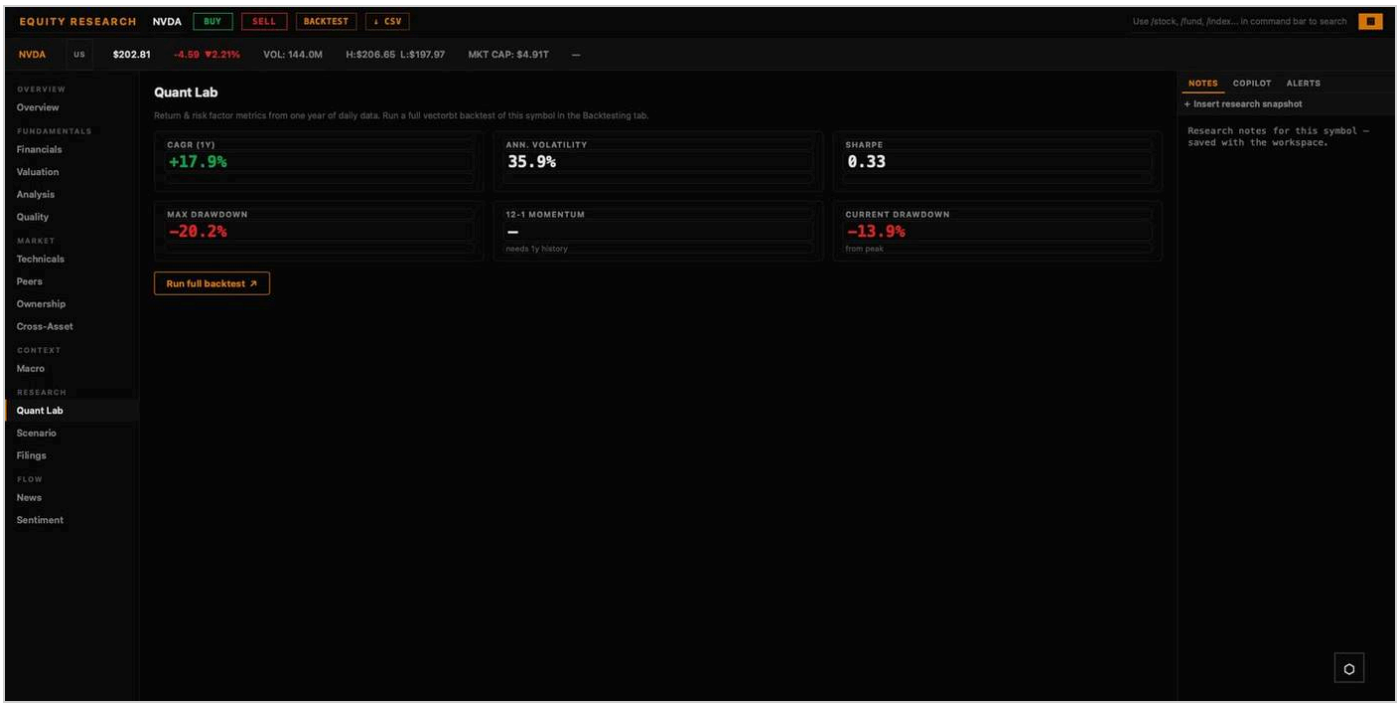
5.1 Macro — market exposure



MACRO (NVDA) — empirical beta, correlation, and up/down capture vs the home index (S&P 500 for US, NIFTY 50 for India), plus sector/country context.

Four tiles from one year of aligned daily returns — **Beta**, **Correlation**, **Up Capture**, **Down Capture** vs the home benchmark — plus a context line (sector · country · benchmark · yfinance-reported β). Read-only.

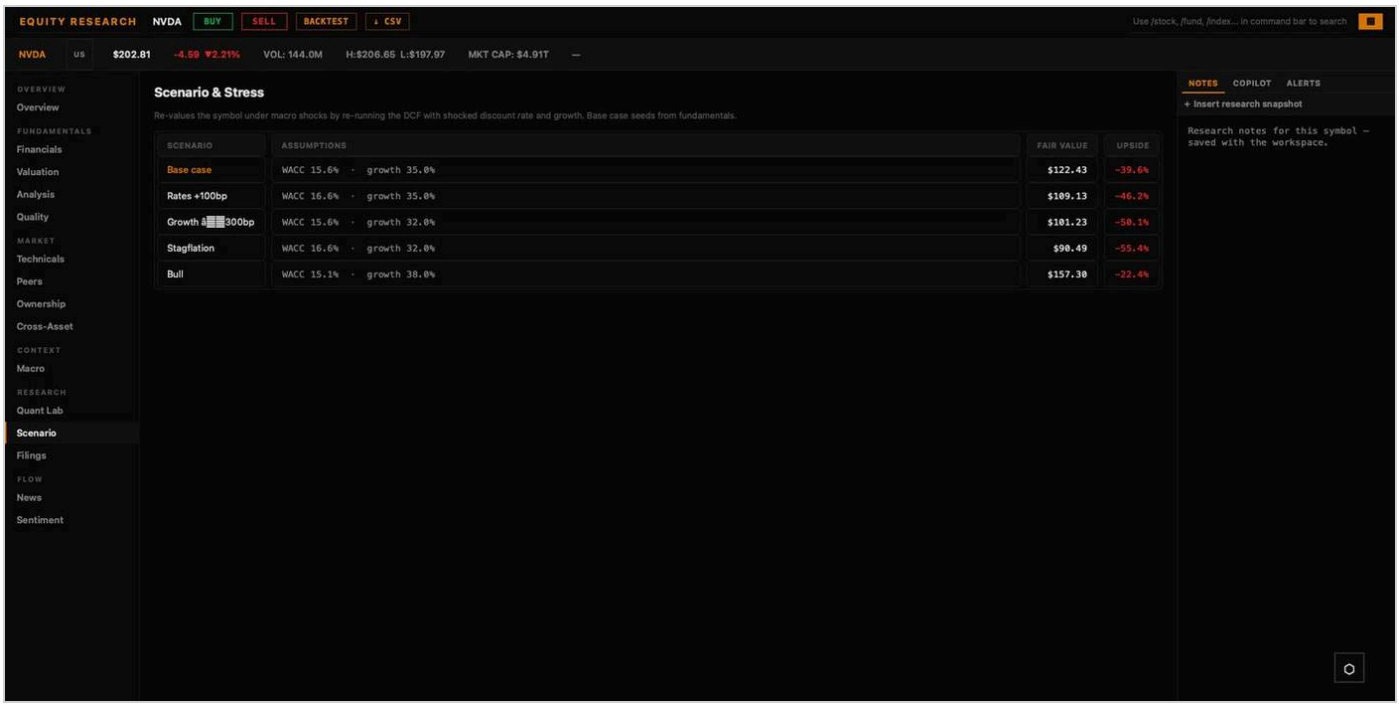
5.2 Quant Lab — factor statistics



QUANT LAB (NVDA) — CAGR, annualized volatility, Sharpe, max drawdown, 12-1 momentum, current drawdown, and a "Run full backtest ↗" hand-off.

Six factor tiles from one year of daily data (CAGR, Ann. Volatility, Sharpe, Max Drawdown, 12-1 Momentum, Current Drawdown), colour-coded for quality. One control: **Run full backtest** ↗ pre-loads the symbol and auto-runs it in the Backtesting screen.

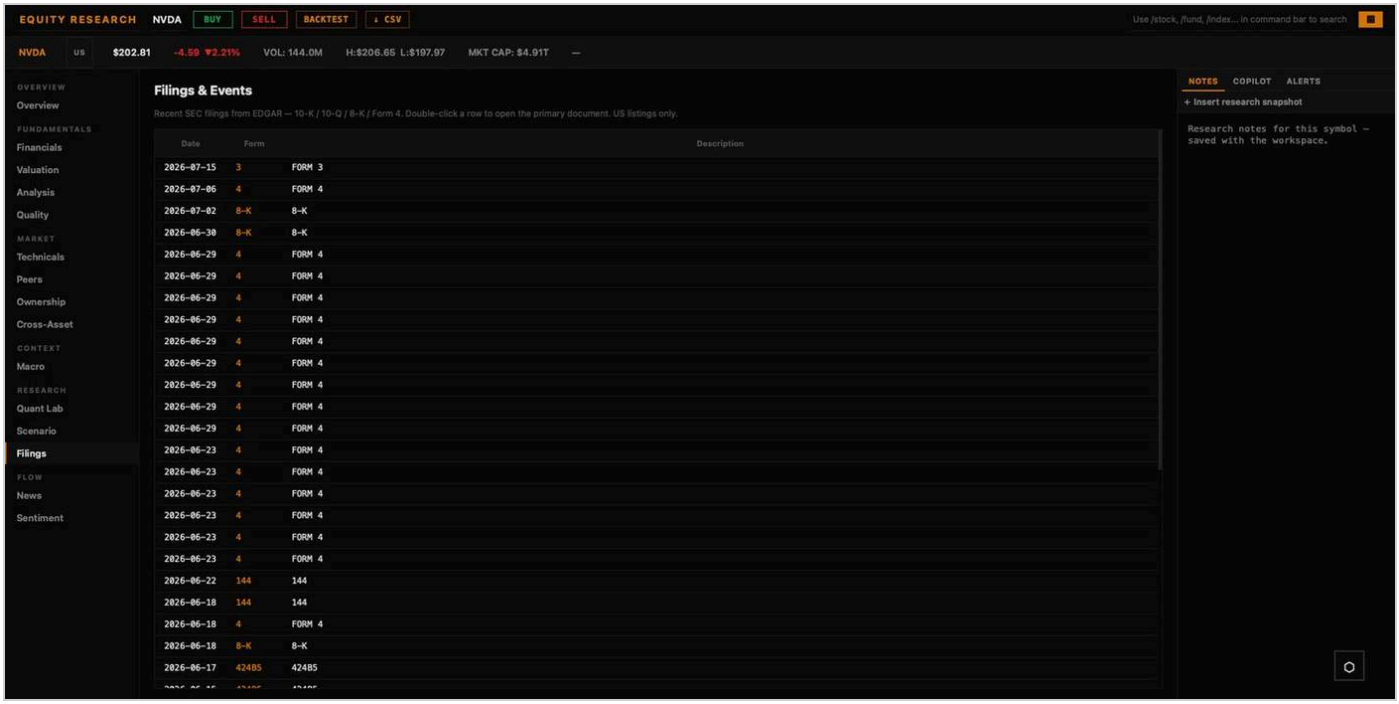
5.3 Scenario — DCF stress



SCENARIO & Stress (NVDA) — DCF fair value and upside re-derived under Base case, Rates +100bp, Growth –300bp, Stagflation and Bull shocks.

A read-only table: five named macro shocks (**Base case · Rates +100bp · Growth –300bp · Stagflation · Bull**), each showing the applied WACC & growth assumption, the resulting DCF fair value, and colour-coded upside vs the current price. Base assumptions seed from fundamentals; the shocks and DCF constants are fixed (not editable here — use the Valuation Lab to tune).

5.4 Filings — SEC EDGAR



FILINGS & Events (NVDA) — recent SEC EDGAR filings (10-K / 10-Q / 8-K / Form 4 / 144 / 424B5…). Double-click a row to open the primary document.

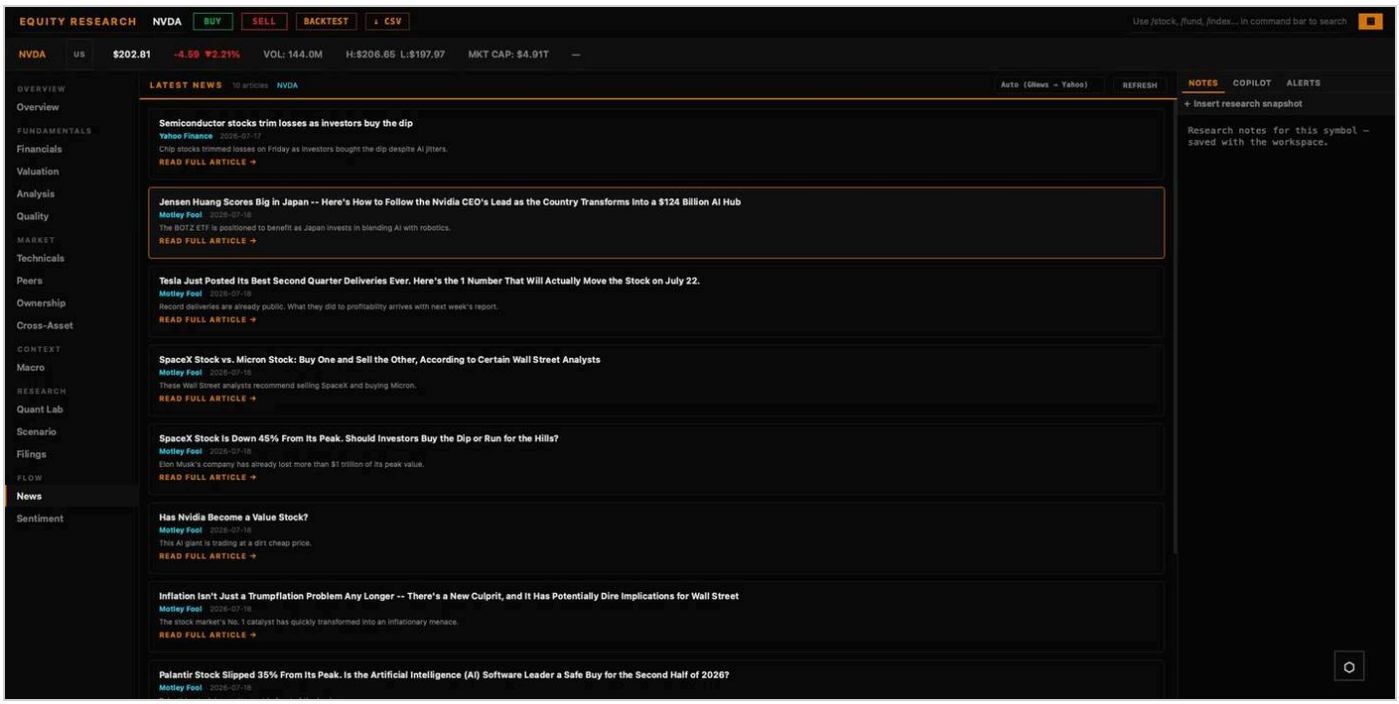
A read-only table (Date · Form · Description) of up to ~40 recent EDGAR filings; **double-click a row** to open its primary document in the browser. **US listings only** — non-US symbols show a clear "EDGAR is US-only" message.

> **At a glance, in CONTEXT & RESEARCH you can…**
>
> - Measure the stock's beta / correlation / capture vs its home index; read quant factor stats and launch a full backtest; stress the DCF under five macro shocks; and browse recent SEC filings.

6 FLOW — News & Sentiment

RAIL GROUP

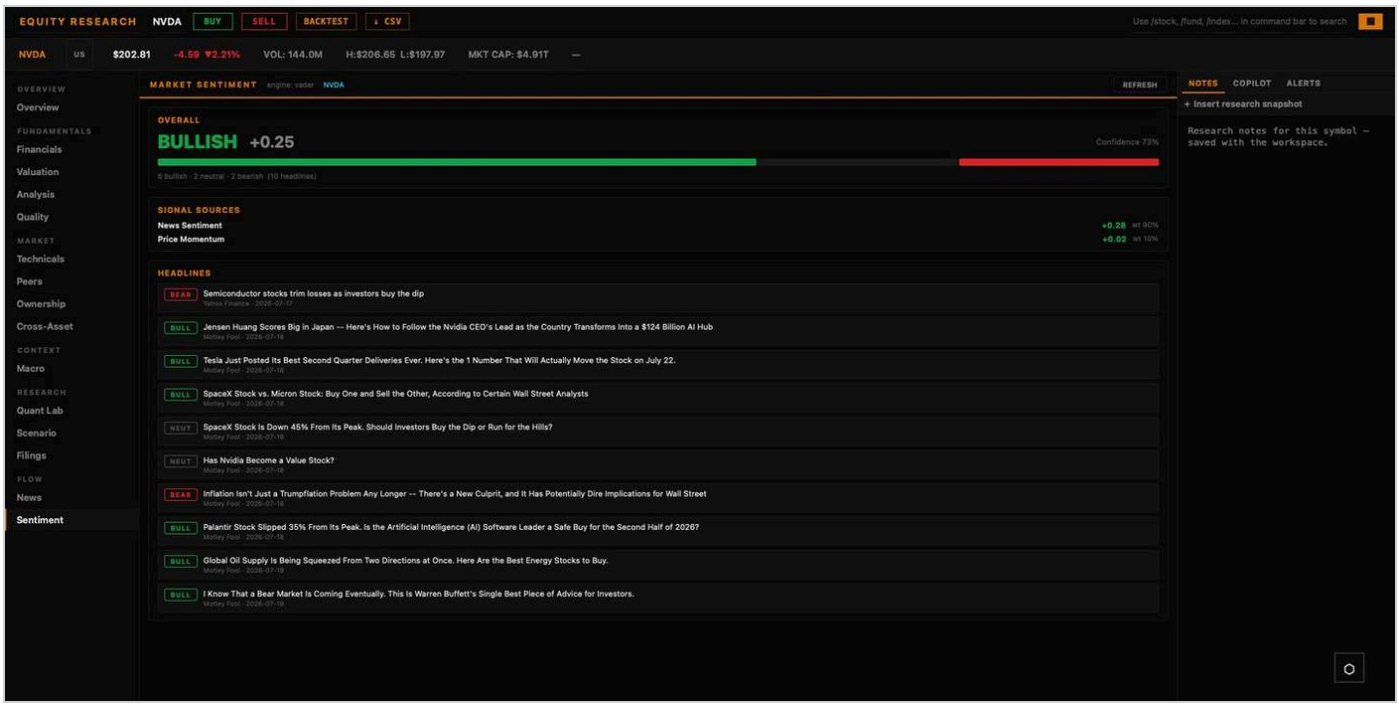
6.1 News



NEWS (NVDA) — clickable headline cards with a provider selector (Auto: GNews→Yahoo, or NewsAPI) and REFRESH.

Control	Type	What it does
Provider selector	combo	**Auto (GNews → Yahoo)** or **NewsAPI** (enabled only with a key configured in Data Sources).
REFRESH	button	Re-pulls the current symbol with the selected provider.
Article card	click	Opens the full story in your browser (title · publisher · date · description · READ FULL ARTICLE →).

6.2 Sentiment



SENTIMENT (NVDA) — a blended verdict (BULLISH/BEARISH/NEUTRAL) with score & confidence, a bull/neutral/bear distribution bar, per-source breakdown, and scored headlines.

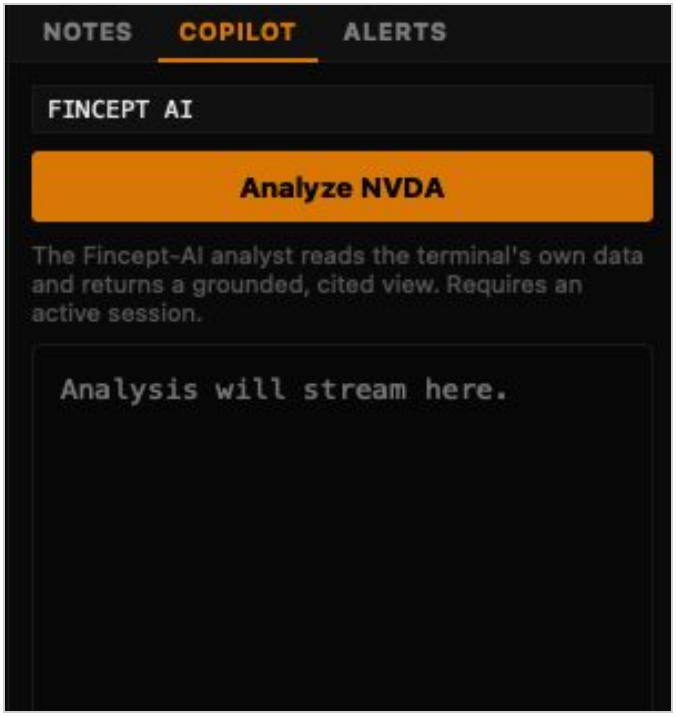
The terminal's own keyless sentiment read: an **Overall** verdict (score + confidence + distribution bar + counts), a **Signal Sources** breakdown (news NLP + price momentum, and Adanos when keyed, each with score & weight), and a **Headlines** list of individually scored articles. One control: **REFRESH**.

> **At a glance, in FLOW you can…**
>
> - Read provider-switchable headlines (open any in the browser), and a blended market-sentiment verdict broken down by news + price momentum with the scored headlines behind it.

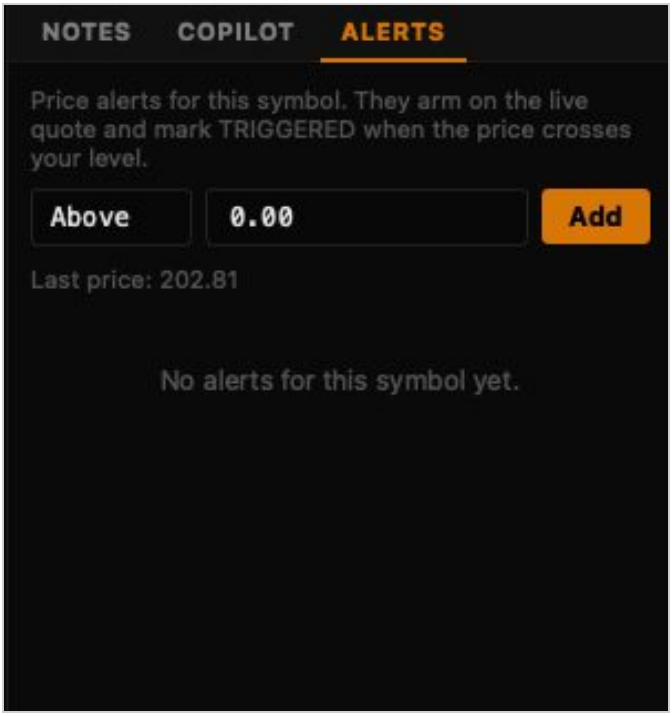
7 The research dock

NOTES · COPILOT · ALERTS

A collapsible right panel (toggle ▤ in the title bar) with three tabs, all scoped to the current symbol and saved with the workspace.



Copilot — an AI research analyst.



Alerts — per-symbol price alerts.

Dock tab	What it does
Notes	A plain-text research editor saved per symbol with the workspace. **Insert research snapshot** stamps the symbol, timestamp, price, market and a thesis/fair-value template into the note.
Copilot	An AI analyst. A **persona picker** (default Fincept-AI stock-analysis agent, or a bound Agent/Team persona) + an **Analyze <symbol>** button stream a grounded, cited answer that reads the terminal's own data tools. A live "thinking" line and status (Starting… → Done. / Analysis failed.) show progress.
Alerts	Per-symbol **Above/Below price alerts**: a direction combo + price field + **Add** arm against the live quote and flip to **TRIGGERED** when crossed. Each row shows ▲/▼ level, ARMED/TRIGGERED, and a ✕ delete. A banner announces fires.

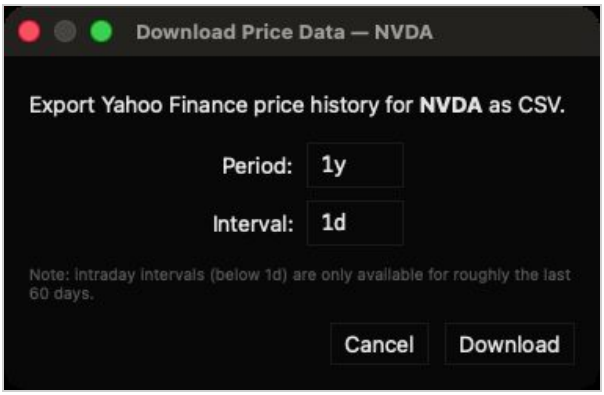
> **Copilot & credits.** The Copilot analyst is the **only credit-spending action** on this tab — it calls the Fincept-AI LLM/agent backend. There is no model picker or cancel; it needs an active session. Alerts are in-pane only (no OS/push notification) and a triggered alert can only be deleted, not re-armed.

8 Toolbar actions & CSV export

TITLE BAR

Action	What it does
BUY / SELL	Appear only when a connected broker (paper or live) can trade the symbol's market. Open the same order ticket as the Equity Trading tab, pre-seeded with the symbol and last price.

Action	What it does
BACKTEST	Hands the current symbol to the Backtesting screen and auto-runs a basic backtest.
↓ **CSV**	Opens the **Download Price Data** dialog (right) — choose Period & Interval, then Download to save Yahoo Finance OHLCV as a CSV.
▤	Toggles the research dock open/closed.



The ↓ CSV export dialog.

9 Where the numbers come from

PROVENANCE

Institutional users care where a number originates. On this tab, almost everything is **live yfinance run through the app's embedded local Python** — not a Fincept server — plus SEC EDGAR direct for filings. The exceptions are listed below.

Data	Source	Cache
Quote (price/change/OHLC/vol)	yfinance `quote` (local Python) — or a **live broker tick** when a region-matched broker is connected	30 s
Company info, valuation, analyst targets	yfinance `info`	5 min
Historical candles (chart)	yfinance `historical_period`	2 min
Financial statements	yfinance `financials` (annual)	—
Technicals candles	**Connected broker** (region-matched) preferred → yfinance fallback → `compute_technicals.py`	2 min
Peers	yfinance `multiple_ratios`	—
News	Google News (`fetch_company_news.py`) → Yahoo fallback; or NewsAPI.org when a key is set	3 min
Sentiment	News NLP + price momentum (+ Adanos when keyed) — computed locally	—
Filings	**SEC EDGAR** direct (local Python)	30 min
Command-bar `/stock` search	Fincept edge API `/market/search`	—
Copilot analysis	**Fincept-AI LLM/agent** (consumes credits)	—

The **market chip** resolves US / IN / GLOBAL from the ticker suffix and its tooltip names the primary fundamentals source (EDGAR XBRL for US, yfinance otherwise); today the tab's statement numbers come from yfinance for all markets, with EDGAR reached through the Filings section and the agent tools below.

10 AI agent tools (45 MCP tools)

AGENTIC

The terminal's AI agent can drive equity research and pull SEC filings through 45 tools — the same live pipelines the UI uses.

Equity research (category `equity-research`, 10 tools)

`search_equity_symbols` · `load_equity_symbol` · `get_equity_quote` · `get_equity_info` · `get_equity_historical` · `get_equity_financials` · `get_equity_technicals` · `get_equity_peers` · `get_equity_news` · `get_equity_sentiment`

SEC EDGAR / filings (category `sec-edgar`, 35 tools)

Resolve & metadata: `edgar_resolve_cik` · `edgar_resolve_ticker` · `edgar_find_company` · `edgar_get_metadata` · `edgar_get_company_facts`. Financials: `edgar_get_financials` · `edgar_get_financial_metrics` · `edgar_get_xbrl_statements` · `edgar_calc_multiples`. 10-K: `edgar_10k_sections` · `edgar_10k_full_text` · `edgar_10k_markdown` · `edgar_10k_search` · `edgar_10k_exhibits` · `edgar_10k_metadata`. 10-Q: `edgar_10q_sections` · `edgar_10q_full_text` · `edgar_10q_markdown` · `edgar_10q_search` · `edgar_10q_metadata`. 8-K: `edgar_8k_events` · `edgar_8k_events_categorized` · `edgar_8k_full_text` · `edgar_8k_search`. Insider & institutional: `edgar_insider_transactions` · `edgar_insider_transactions_detailed` · `edgar_insider_summary` · `edgar_13f_holdings` · `edgar_13f_top_holdings` · `edgar_13f_summary` · `edgar_13f_manager_info`. Filings & docs: `edgar_get_filing_text` · `edgar_search_filings` · `edgar_get_filing_documents` · `edgar_find_merger_agreement`.

> **Non-credit.** All 45 tools run on local Python / yfinance / SEC EDGAR and do **not** spend AI credits — only the Copilot *analysis* (which may call these tools) does.

11 Quick reference & important notes

HOW DO I...

I want to…	Go to
Load a company	Command bar `/stock AAPL`, drag a ticker onto the screen, or pick from a linked panel.
See a full snapshot	OVERVIEW (default) — chart period 1M–5Y.
Read the statements / export	FUNDAMENTALS ▸ Financials ▸ Income/Balance/Cash Flow ▸ EXPORT CSV.
Value it & test a thesis	FUNDAMENTALS ▸ Valuation — edit the seven DCF inputs live.
Get a quick verdict / quality read	FUNDAMENTALS ▸ Analysis (six cards) / Quality (Z, F, M scores).
Check technical signals	MARKET ▸ Technicals — consensus rating + indicator tables.
Compare to peers	MARKET ▸ Peers — edit the list, LOAD, sort columns.
See who owns it	MARKET ▸ Ownership — institutions, insiders, supply chain.
Gauge volatility / an option	MARKET ▸ Cross-Asset — realized vol + theoretical straddle.
Measure market/macro exposure	CONTEXT ▸ Macro — beta, correlation, capture.
Stress the valuation	RESEARCH ▸ Scenario — five macro shocks.
Read SEC filings	RESEARCH ▸ Filings — double-click to open (US only).
Read news / sentiment	FLOW ▸ News (switch provider) / Sentiment.
Ask an AI analyst / take notes / set alerts	Research dock (▤) ▸ Copilot / Notes / Alerts.
Trade / backtest / export prices	Title bar ▸ BUY·SELL / BACKTEST / ↓ CSV.

Important notes for institutional users

- **Live & mostly local.** Research data is live yfinance via embedded Python (not a server), plus SEC EDGAR direct; a connected broker upgrades the quote to a live tick and supplies technical candles. See §9.
- **One credit-spending action.** Only the Copilot analyst calls the AI backend; everything else — quotes, statements, technicals, peers, ownership, news, sentiment and all 45 filing/agent tools — is non-credit.
- **Read-mostly.** Only Valuation (DCF inputs), Peers (list), the dock Notes/Alerts, and the toolbar actions change anything; every other section is a computed read-out.
- **Screening signals, not advice.** Analysis/Quality/Technicals thresholds are fixed and not sector- or size-adjusted; Cross-Asset uses realized vol as an implied-vol proxy; Scenario/Valuation DCF constants (terminal growth, projection years, CAPM inputs) are seeded and, in Scenario, fixed.
- **Known limitations.** Financials are annual-only and a failed fetch spins the overlay; the CSV filename is prefixed `equity_`; Filings is US-only (≤40 rows, no filters); Ownership tables cap at 15 rows; NewsAPI/Adanos need keys.
- **State is cloud-synced.** Symbol, section, per-symbol notes, peer list, news provider and alerts persist server-side (JWT), not to a local file.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Portfolio Workstation

A complete, screen-by-screen walkthrough of the Portfolio tab — the live position monitor, nine institutional analytics views, trade & dividend entry, and the built-in AI analyst.

Covers · **live dashboard** · **9 analytics views** · **8 dialogs & forms**
AI & Agent insights · **14 automation tools** · data sources & engine

Demonstrated live on a funded 5-position portfolio — **Kite Top Performer (INR)**.

Contents

18 SECTIONS

GETTING ORIENTED

01 What the Portfolio tab is 3

02 Workspace anatomy & the command bar 4

03 The live dashboard — holdings, chart, sectors, positions 6

THE NINE ANALYTICS VIEWS

04 Sectors & concentration 8

05 Performance & risk 9

06 Portfolio optimization 10

07 QuantStats report 12

08 Reports & attribution 14

09 Custom indices 15

10 Risk management 16

11 Financial planning 17

12 Economics & macro factors 18

ACTIONS & INTELLIGENCE

13 Trading — order entry, buy / sell / dividend 19

14 Managing portfolios — create, import, export 20

15 AI & Agent insights 21

REFERENCE

16 Analytics engine, live data & metrics 22

17 Agent automation tools 23

18 Quick reference & honest notes 24

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal, taken while driving a funded demonstration portfolio (Kite Top Performer, ₹560,716 across 5 Indian equities). Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

01 What the Portfolio tab is

OVERVIEW

The Portfolio tab is a **single-portfolio analyst workstation**. It holds one selected portfolio at a time, keeps its value and risk metrics streaming live, and layers nine institutional analytics views on top of the position book — from sector concentration and mean-variance optimization to a QuantStats tearsheet, Monte-Carlo simulation, retirement planning and a macro-factor lens. Trades, dividends, imports and exports are all recorded from the same screen, and a built-in AI analyst can narrate the book on demand.

Reach it from the top navigation bar (PORTFOLIO) or by typing `portfolio` into the command bar. It is available on every paid plan — there is no separate plan lock. Opening the tab spends **no credits**; only the optional AI / Agent analyst does.

AT A GLANCE, YOU CAN…

- Watch **portfolio value, P&L and today's move** update live over a WebSocket
- Read a **Sharpe / Beta / Volatility / Max-Drawdown / concentration** risk strip
- Scan every holding as a **color-coded heatmap** and a sortable **11-column blotter**
- Chart **NAV performance** vs a benchmark over 1W→ALL windows
- Break the book down by **sector** and inspect a **correlation matrix**
- Run **mean-variance optimization** with an efficient frontier
- Generate a full **QuantStats report** + a 1,000-path **Monte-Carlo** forecast
- Produce **summary, transaction & attribution reports**
- Build **custom indices** from your holdings
- Stress-test against **8 historical crises** and decompose risk by holding
- Plan **retirement, goals and savings rates**
- Record **buys, sells and dividends**; import / export portfolios
- Ask a built-in **AI analyst** (or a saved agent) to review the book
- Drive everything programmatically through **14 agent tools**

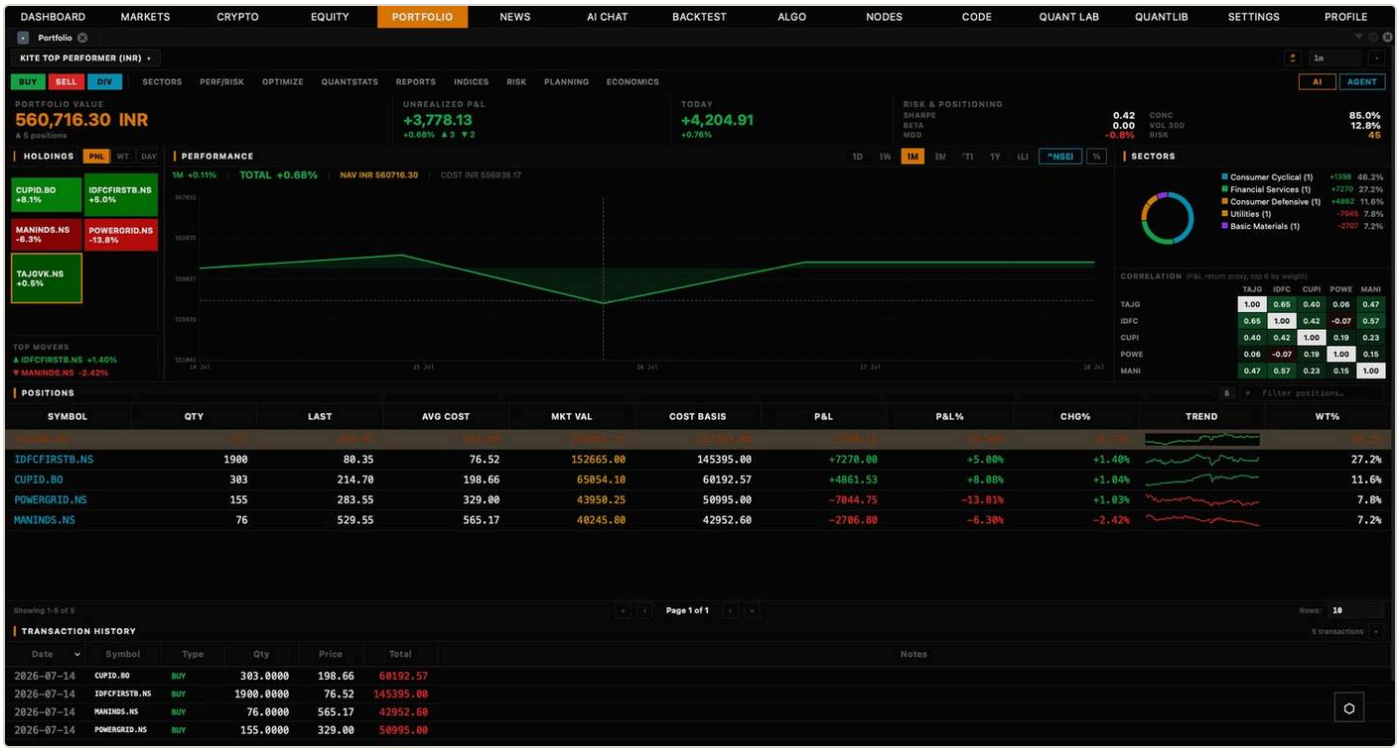
WHERE THE DATA COMES FROM — IN ONE LINE

Holdings and portfolios live on the **Fincept Exclusive API** (server-side, synced to your account); live prices come from your connected broker or market feed; the heavy analytics (optimization, QuantStats, Monte-Carlo, FFN, NAV history, macro) run **server-side** and stream results back; the risk metrics, correlations, sector maths, stress tests and planners compute **locally in the app**. Full map in §16.

02 Workspace anatomy & the command bar

LAYOUT

The workstation is four stacked bands. From top to bottom: a two-row **command bar**, a **KPI stats ribbon**, the **content area** (the live dashboard, or one of the nine analytics views), and a thin **status bar**. An order-entry panel and the AI dock slide in as overlays when summoned.



The live dashboard. Command bar (portfolio selector + trade buttons + nine view tabs + AI/Agent), KPI ribbon, then the working area: holdings heatmap, NAV chart, sector donut & correlation matrix, the positions blotter and the transaction history footer.

Command bar — row 1: identity & refresh

CONTROL	TYPE	WHAT IT DOES
Portfolio selector "KITE TOP PERFORMER (INR) ▾"	Dropdown	Opens a searchable list of your portfolios; also holds `+ CREATE NEW` and `DELETE`. Switches the whole workstation to the chosen book.
↻ Refresh	Icon button	Re-pulls the summary & live quotes for the current portfolio (shows ⏳ while working).
Interval	Combo	Auto-refresh cadence: `1m` · `5m` · `10m` · `30m` · `1h` · `3h` · `1d`. (In live-stream mode the WebSocket drives updates; the timer is the fallback.)
Overflow	Menu	Export CSV · Export JSON · Import JSON… · FFN Analysis · Backtest Portfolio (see §14, §16).

Command bar — row 2: trade · nine views · intelligence

CONTROL	TYPE	WHAT IT DOES
BUY / **SELL** / **DIV**	Buttons	Slide in the order-entry panel to buy or sell the selected holding, or open the **Record Dividend** dialog (§13).

Nine view tabs	Toggle pills	**SECTORS · PERF/RISK · OPTIMIZE · QUANTSTATS · REPORTS · INDICES · RISK · PLANNING · ECONOMICS.** Each replaces the dashboard with a full-screen analytics view (§04–§12); the active pill lights amber; click it again (or ← BACK) to return.
AI / AGENT	Buttons	Open the right-hand insights dock — AI analysis or a saved agent run (§15). *The only credit-spending action on the tab.*

KPI stats ribbon

CELL	SHOWS
PORTFOLIO VALUE	Total market value + currency, with the position count beneath.
UNREALIZED P&L	Open profit/loss in currency and %, with ▲ gainers / ▼ losers.
TODAY	Today's change in currency and %.
RISK & POSITIONING	A 2×3 chip grid: **Sharpe · Conc · Beta · Vol 30D · MDD · Risk** (Risk is a 0–100 composite, colored green/amber/red).

HOW THE NINE VIEWS OPEN — WORTH KNOWING

Clicking a view pill **replaces** the dashboard entirely (the KPI ribbon hides; the command bar and a ← BACK header stay). You are never more than one click from any view or from the dashboard. Selecting a view does not disturb your position book — analytics are computed, never written back.

03 The live dashboard

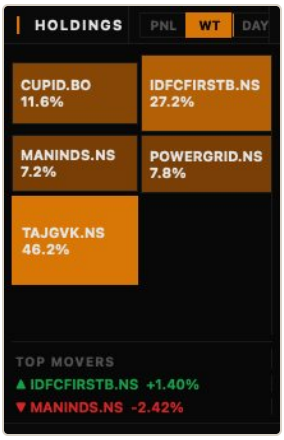
DASHBOARD

The default content area is a five-region cockpit: the holdings heatmap (left), the NAV performance chart (center), the sector donut & correlation matrix (right), the positions blotter (middle) and the transaction history footer.

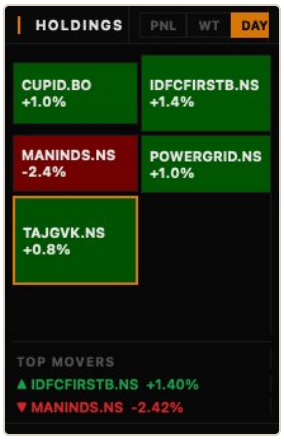
Holdings heatmap & top movers

Every holding is a colored block sized by weight. A three-way toggle recolors the grid; a **TOP MOVERS** footer names the day's best and worst.

PNL — by open P&L %



WT — by weight



DAY — by today's move

MODE	COLORS BLOCKS BY
PNL	Unrealized P&L % — green→red, saturating at ±20%
WT	Portfolio weight — amber gradient, saturating at 40%
DAY	Today's % change — green→red

Click any block to select that holding (a 2 px amber border marks it) and drive the order panel & group-linked panels. Top movers always rank by today's move, regardless of the active color mode.

NAV performance chart

A time-series of portfolio value with a dashed cost-basis reference line, an optional benchmark overlay and a hover crosshair. The info bar reads period change, **TOTAL** return, **NAV** and **COST**.

CONTROL	BEHAVIOR
1D · 1W · 1M · 3M · YTD · 1Y · ALL	Time window (default 1M). **1D is disabled** — the book is captured as daily snapshots, so there is no intraday series; 1W needs ≥5 days of history.
Benchmark (e.g. `^NSEI` / `SPY`)	Toggles a benchmark overlay, auto-selected by the portfolio's currency.
%	Rebases the portfolio and benchmark to base 100 for a clean like-for-like comparison.

Sectors & correlation (right rail)

A weight-sorted **allocation donut** with a clickable legend (click a slice or row to filter the blotter to that sector), above an **N×N correlation matrix** of the top-6 holdings by weight — Pearson correlation of daily returns over the trailing 30 trading days, green for positive, red for negative.

Positions blotter

The primary book: a sortable, filterable, paginated table with a per-row trend sparkline.

COLUMNS (IN ORDER)
`SYMBOL · QTY · LAST · AVG COST · MKT VAL · COST BASIS · P&L · P&L% · CHG% · TREND · WT%`

- **Sort** by clicking a header — SYMBOL, LAST, MKT VAL, P&L, P&L%, CHG% and WT% are sortable (default: weight, descending).
- **Filter** with the `Filter positions…` box above the table, or by clicking a sector.
- **Right-click a row** → Edit Transaction or Close / Delete Position.
- **Paginate** with «‹ ›» and set page size (`10 / 20 / 50 / 100`).

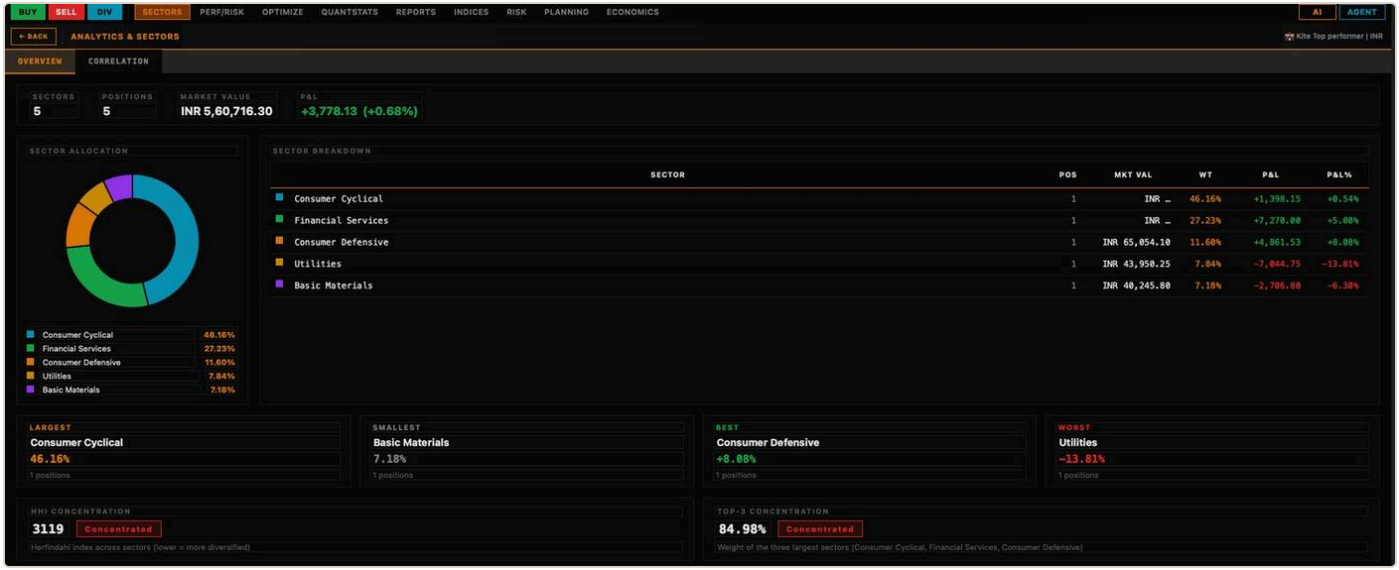
Transaction history

A collapsible footer listing recent transactions — columns `Date · Symbol · Type · Qty · Price · Total · Notes`. Type is color-coded (BUY green, SELL red, DIVIDEND cyan) and the chevron collapses it to a single strip.

04 Sectors & concentration

VIEW 1 · AUTO-RUNS

The SECTORS view (titled **Analytics & Sectors**) answers "how is the book distributed, and how concentrated is it?" It populates instantly on open — no run button — across two tabs: **Overview** and **Correlation**.



Overview tab. A KPI strip, the allocation donut with legend, a sector breakdown table, LARGEST / SMALLEST / BEST / WORST cards, and two concentration diagnostics.

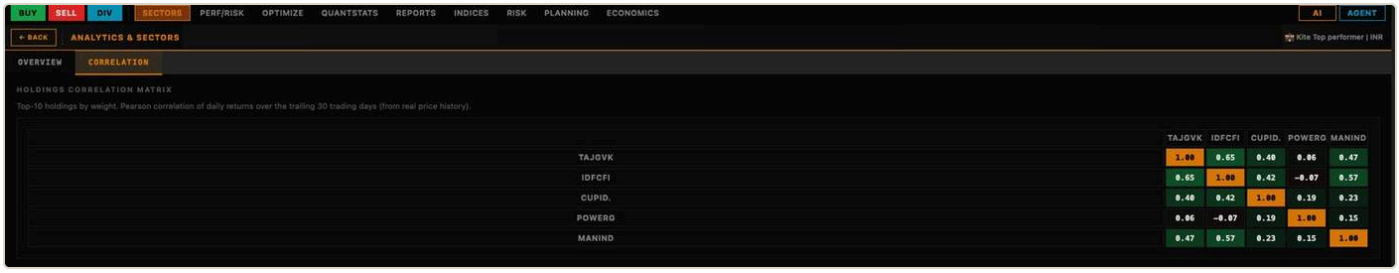
SECTOR BREAKDOWN TABLE

COLUMN	MEANING
SECTOR	GICS sector (click to filter blotter)
POS	Positions in the sector
MKT VAL	Sector market value
WT	Weight of the book
P&L / P&L%	Unrealized sector P&L

CONCENTRATION DIAGNOSTICS

METRIC	VERDICT THRESHOLDS
HHI	Herfindahl index of sector weights. `<1500` Diversified · `<2500` Balanced · else **Concentrated**
Top-3	Weight of the three biggest sectors. `<50%` · `<75%` · else **Concentrated**

The demo book scores HHI 3119 and Top-3 84.98% — flagged **Concentrated**, as expected for five holdings.



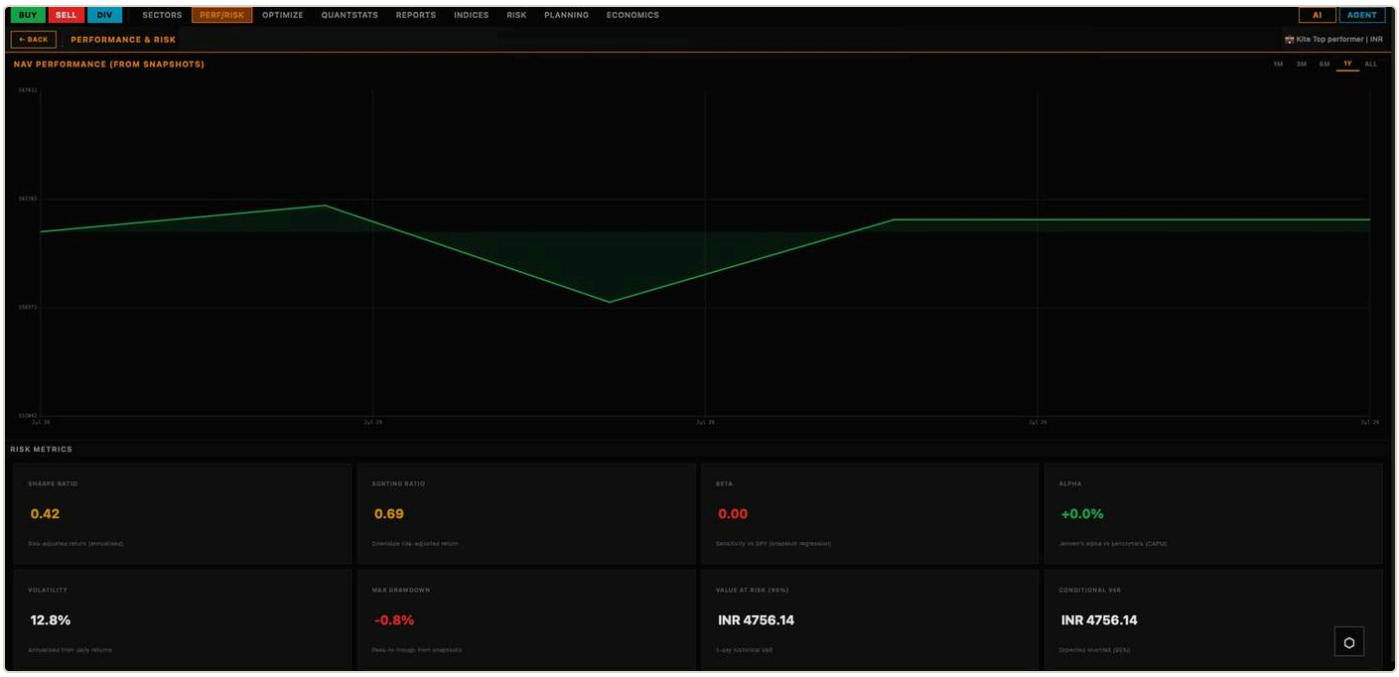
Correlation tab. Pearson correlation of daily returns over the trailing 30 trading days for the top-10 holdings by weight; diagonal 1.00 on amber, positive green, negative red.

> **READING NOTE**
>
> Sector labels come from an automatic resolver; anything it cannot classify lands in one "Unclassified" bucket. Correlation needs ≥5 price points per name — a thinly-traded holding shows "—" cells.

05 Performance & risk

VIEW 2 · AUTO-RUNS

The PERF/RISK view charts NAV performance from daily snapshots and prints eight risk / performance metric cards. It auto-runs; the period buttons re-filter the chart (the metrics always reflect full history).



NAV performance (1M–ALL) over an eight-card risk panel: Sharpe, Sortino, Beta, Alpha, Volatility, Max Drawdown, VaR 95% and Conditional VaR.

CARD	DEFINITION
Sharpe Ratio	Excess return ÷ volatility, annualised, using the live risk-free rate.
Sortino Ratio	Excess return ÷ downside deviation, annualised.
Beta	OLS sensitivity of daily returns vs **SPY** (needs ≥5 aligned days).
Alpha	Annualised Jensen's alpha (CAPM) vs benchmark.
Volatility	Annualised standard deviation of daily NAV returns.
Max Drawdown	Largest peak-to-trough decline across the snapshot series.
VaR 95%	1-day historical Value-at-Risk in currency.
Conditional VaR	Expected shortfall — the mean of the worst 5% of days.

READING NOTE

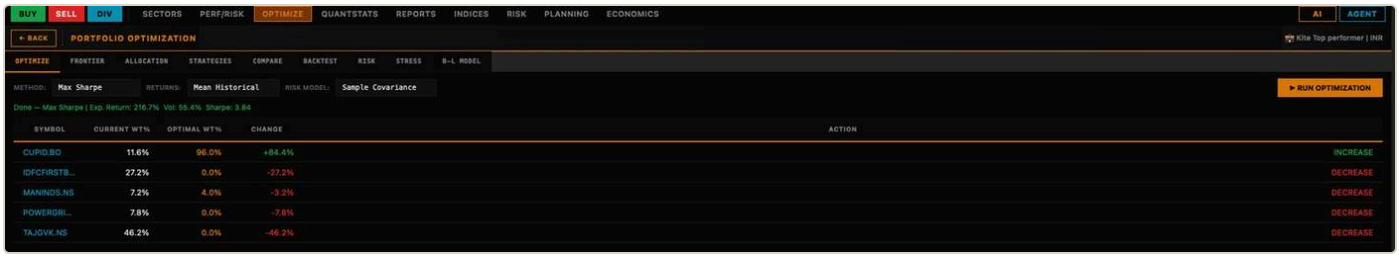
Beta and Alpha are always measured against **SPY**, and a metric shows "N/A" until enough aligned history exists (≥5 days for Beta/Alpha, ≥3 snapshots for the rest). NAV history is reconstructed from your transactions the first time each portfolio is opened.

06 Portfolio optimization

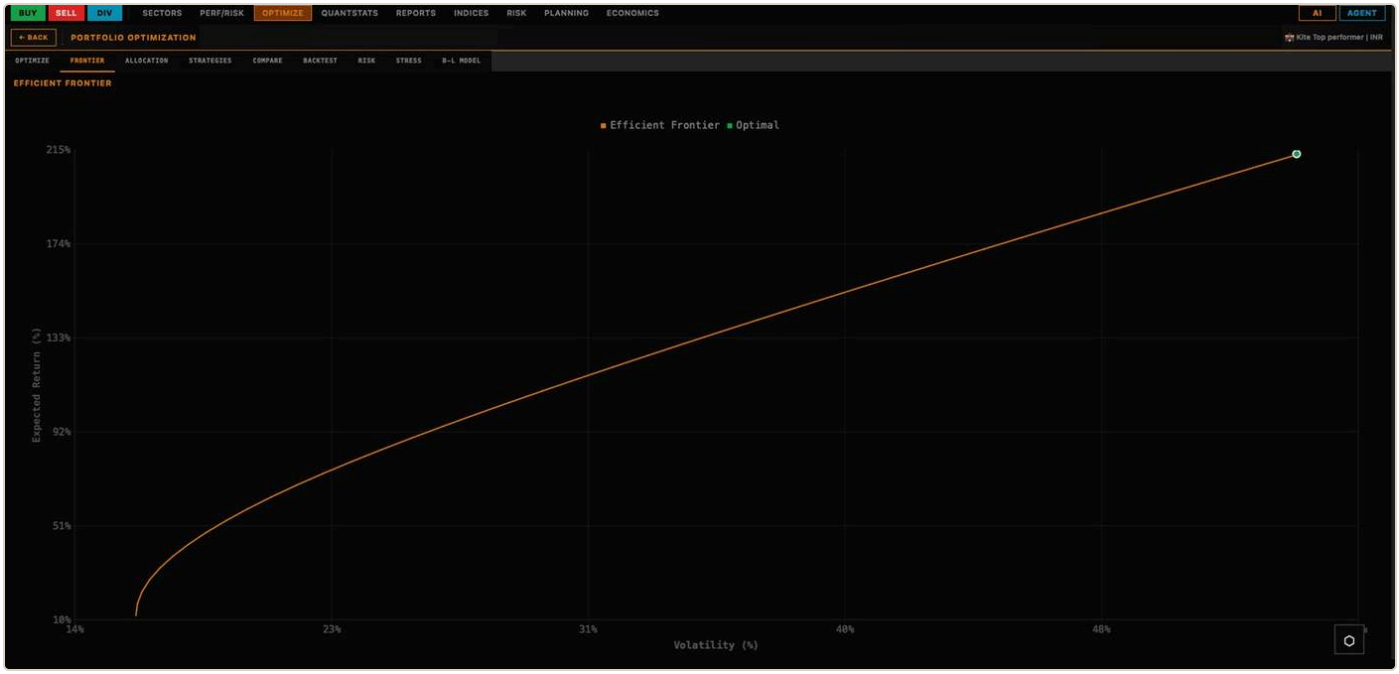
VIEW 3 · RUN

The OPTIMIZE view runs a mean-variance / risk-based optimizer and lays the results across nine tabs. Pick a **Method** and press ▶ RUN OPTIMIZATION; results are computed server-side and cached for 10 minutes.

Before running — Method / Returns / Risk-Model combos and the run button; the nine result tabs sit empty.



Optimize tab, post-run. A status line (method, expected return, vol, Sharpe) over the reweighting table: `SYMBOL` · `CURRENT WT%` · `OPTIMAL WT%` · `CHANGE` · `ACTION` (INCREASE / DECREASE / HOLD).



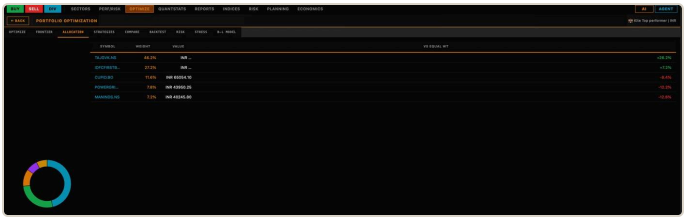
Frontier tab. The efficient frontier (amber) with the optimal max-Sharpe point (green). X = volatility %, Y = expected return %.

METHODS

Max Sharpe · Min Volatility · Risk Parity · Max Return · Equal Weight · HRP · Target Return · B-L Model.

THE NINE TABS

Optimize · Frontier · Allocation · Strategies · Compare · Backtest · Risk · Stress · B-L Model. Allocation & Stress fill on open; the rest populate after a run.



Allocation tab — current weights donut & table (auto-populates).

READING NOTE

The **Method** drives the result. The **Returns** and **Risk-Model** combos are recorded but do not currently change the server output, and **Target Return** has no separate target field yet. **Backtest** tab buttons run a buy-and-hold backtest of current or optimal weights, or hand off to the full Backtesting terminal.

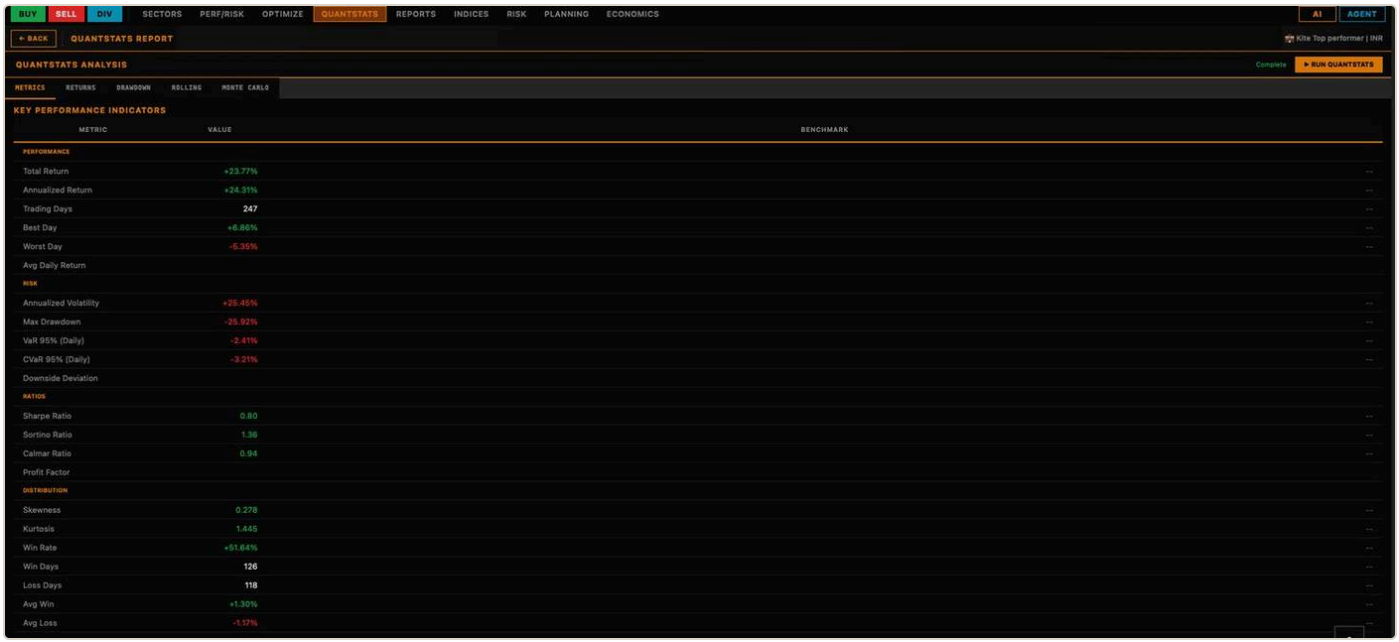
Monte Carlo — 1,000 GBM return paths over 252 trading days (bright line = median), with a percentile strip: median return, 5th / 95th percentile, probability of loss and expected max drawdown.

> **READING NOTE**
>
> Lookback is fixed at one year, the simulation at 1,000 paths over 252 days; the benchmark column is reserved for a future feed. These are server analytics — they need a reachable backend, but they spend **no LLM credits**.

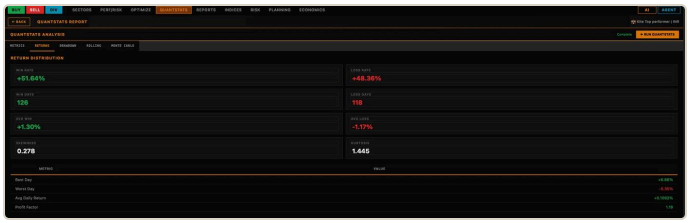
07 QuantStats report

VIEW 4 · AUTO + RUN

The QUANTSTATS view is an institutional tearsheet — ~30 performance/risk/distribution metrics plus a 1,000-path Monte-Carlo forecast, across five tabs. The metric tabs auto-run on open (from one year of price history); Monte-Carlo runs on its own button.

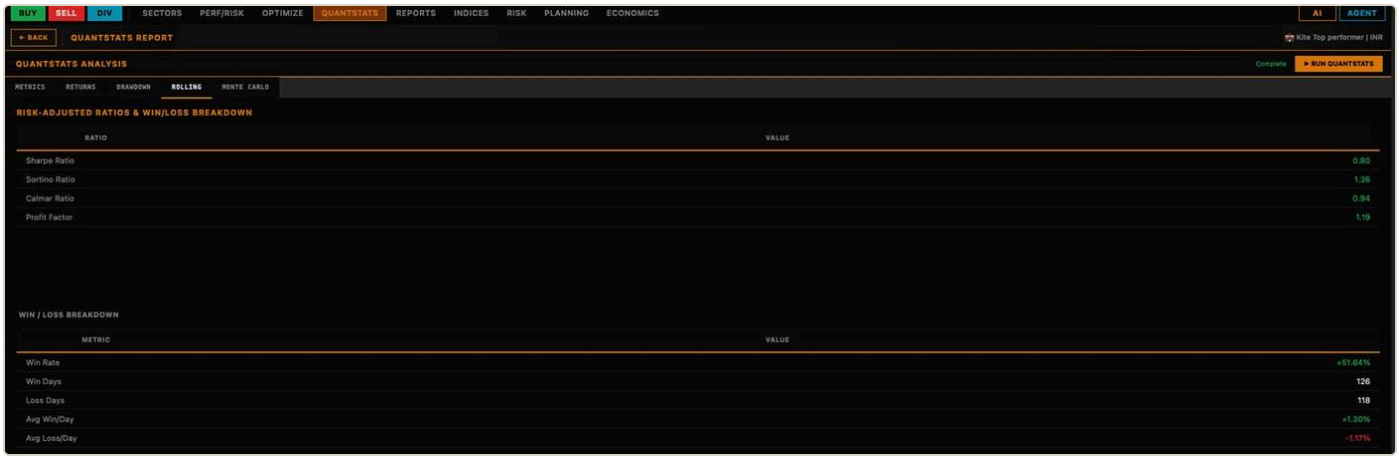


Metrics tab. Key performance indicators grouped Performance / Risk / Ratios / Distribution — total & annualised return, best/worst day, volatility, drawdown, VaR/CVaR, Sharpe/Sortino/Calmar, skew/kurtosis, win rate.



Returns — win/loss cards + distribution table.

Drawdown — max-drawdown hero + risk table.

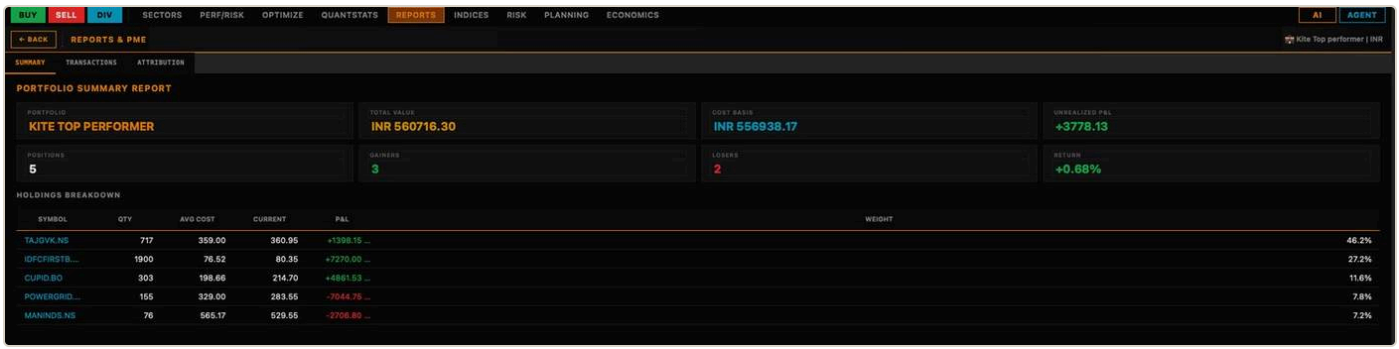


Rolling — risk-adjusted ratios (Sharpe / Sortino / Calmar / Profit Factor) and a win-/loss-day breakdown.

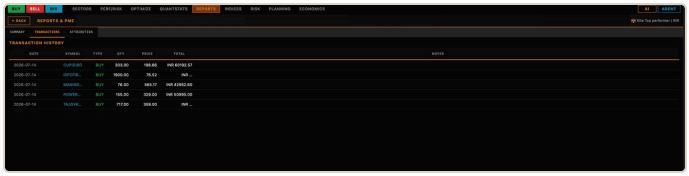
08 Reports & attribution

VIEW 5 · AUTO-RUNS

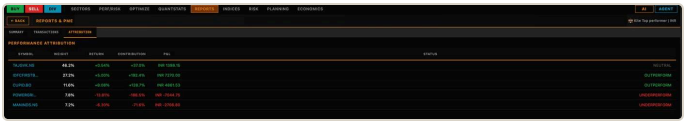
The REPORTS view assembles three formatted reports — a portfolio summary, the full transaction ledger, and per-holding performance attribution. All three populate on open.



Summary tab. Eight headline cards (value, cost basis, unrealized P&L, positions, gainers, losers, return) over a holdings breakdown: SYMBOL · QTY · AVG COST · CURRENT · P&L · WEIGHT.



Transactions — DATE · SYMBOL · TYPE · QTY · PRICE · TOTAL · NOTES.



Attribution — weight, return, contribution, P&L and an OUTPERFORM / NEUTRAL / UNDERPERFORM verdict.

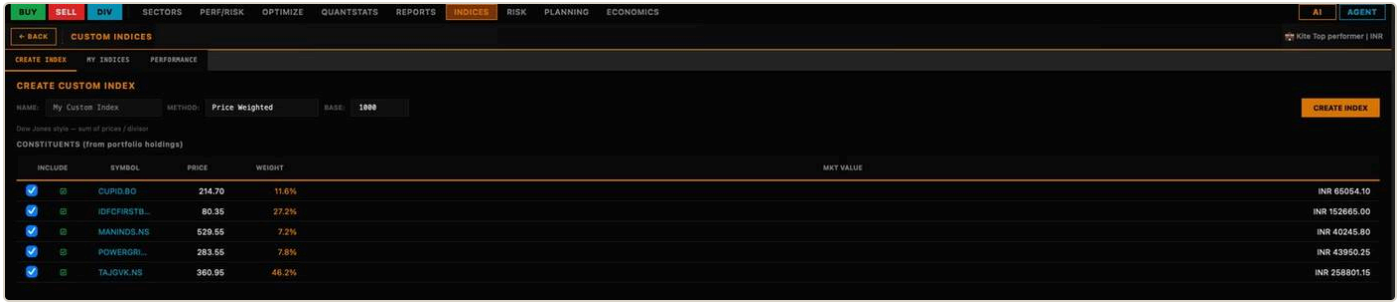
READING NOTE

"Contribution" is each holding's share of total portfolio P&L; the verdict uses a fixed ±5% band. The report set is text/table-based — export the underlying data from the ··· menu (Export CSV / JSON, §14).

09 Custom indices

VIEW 6 · BUILD

The INDICES view lets you construct a named index from the current holdings using one of ten weighting methods, then track it over time. Three tabs: **Create Index**, **My Indices** and **Performance**.



Create Index tab. Name, method & base value, then a constituents table drawn from your holdings (`INCLUDE · SYMBOL · PRICE · WEIGHT · MKT VALUE`) — tick the names to include and press CREATE INDEX.

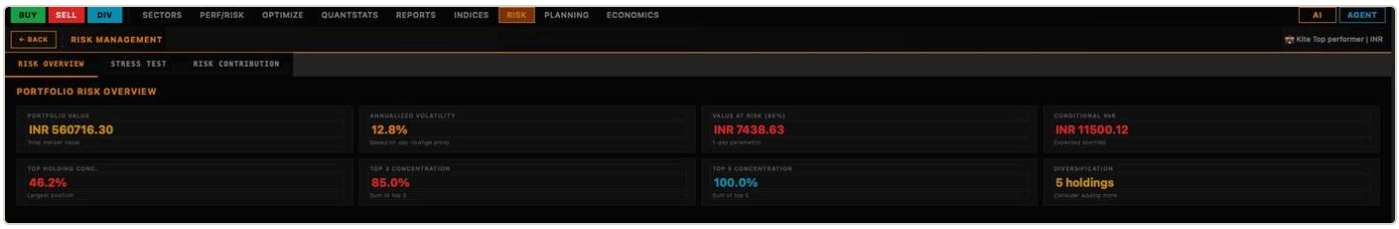
CONTROL	WHAT IT DOES
Name	Index name (required).
Method	Price Weighted · Market Cap · Equal Weighted · Float Adjusted · Fundamental · Modified Market Cap · Factor · Risk Parity · Geometric Mean · Capped.
Base	Starting index level (default 1000).
Include ☑	Per-holding checkbox — choose the constituents.
CREATE INDEX	Saves the index and its opening value; it then appears under **My Indices**.

> **READING NOTE**
> Indices are saved **locally** on this machine. Price Weighted, Equal Weighted and Geometric Mean use distinct maths; the other seven methods currently reduce to a weight-based formula. The Performance chart fills in as daily values accumulate.

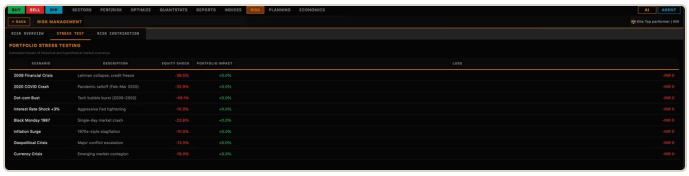
10 Risk management

VIEW 7 · AUTO-RUNS

The RISK view is a three-tab risk desk — a parametric VaR/CVaR & concentration overview, historical & hypothetical stress scenarios, and a per-holding risk decomposition. Everything computes locally and instantly.



Risk Overview. Portfolio value, annualised volatility, 1-day VaR & CVaR, plus top-holding / top-3 / top-5 concentration and a diversification read.



Stress Test — 8 scenarios (2008, COVID-2020, Dot-com, rate shock, Black Monday, inflation, geopolitical, currency), each with an equity shock and estimated impact/loss.

Risk Contribution — per holding: weight, vol proxy, risk & VaR contribution, and a HIGH/MEDIUM/LOW concentration flag.

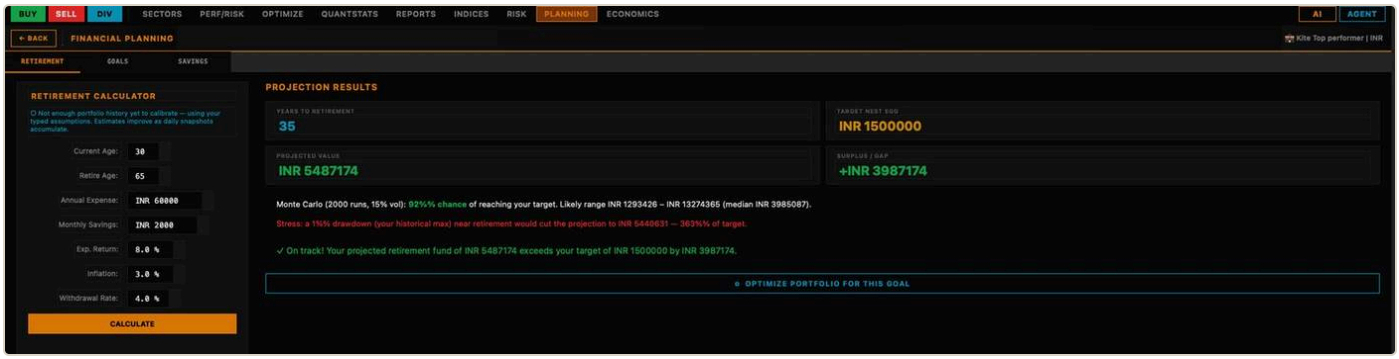
READING NOTE — STRESS IMPACT SCALES WITH BETA

Scenario impact is scaled by the portfolio's measured equity beta. A book whose beta reads near zero — common until a portfolio has enough return history aligned to the benchmark — will show muted (near-0%) scenario impacts. The scenario shocks and the concentration/VaR panels remain fully meaningful; the impact column sharpens as beta history builds.

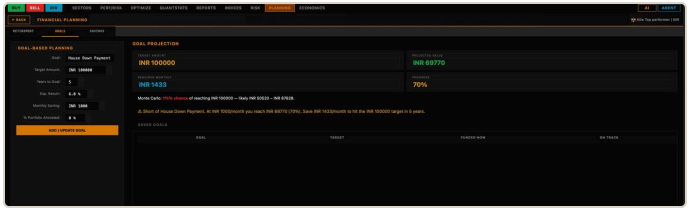
11 Financial planning

VIEW 8 · AUTO-RUNS

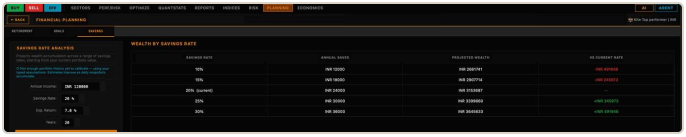
The PLANNING view holds three calculators — retirement, goal-based saving, and a savings-rate comparison — each seeded by the portfolio's own value and, where enough history exists, its measured return & volatility. All three fill on open with default assumptions; edit any input and press the tab's CALCULATE button to recompute.



Retirement tab. Inputs (ages, expense, savings, return, inflation, withdrawal rate) drive a projection: years to retirement, target nest egg, projected value and surplus/gap, plus a Monte-Carlo probability line and an on-track verdict. ⚙ OPTIMIZE PORTFOLIO FOR THIS GOAL jumps to the optimizer targeting the required return.



Goals — target, horizon & monthly saving → projected value, required monthly and a progress read; goals persist per-portfolio.



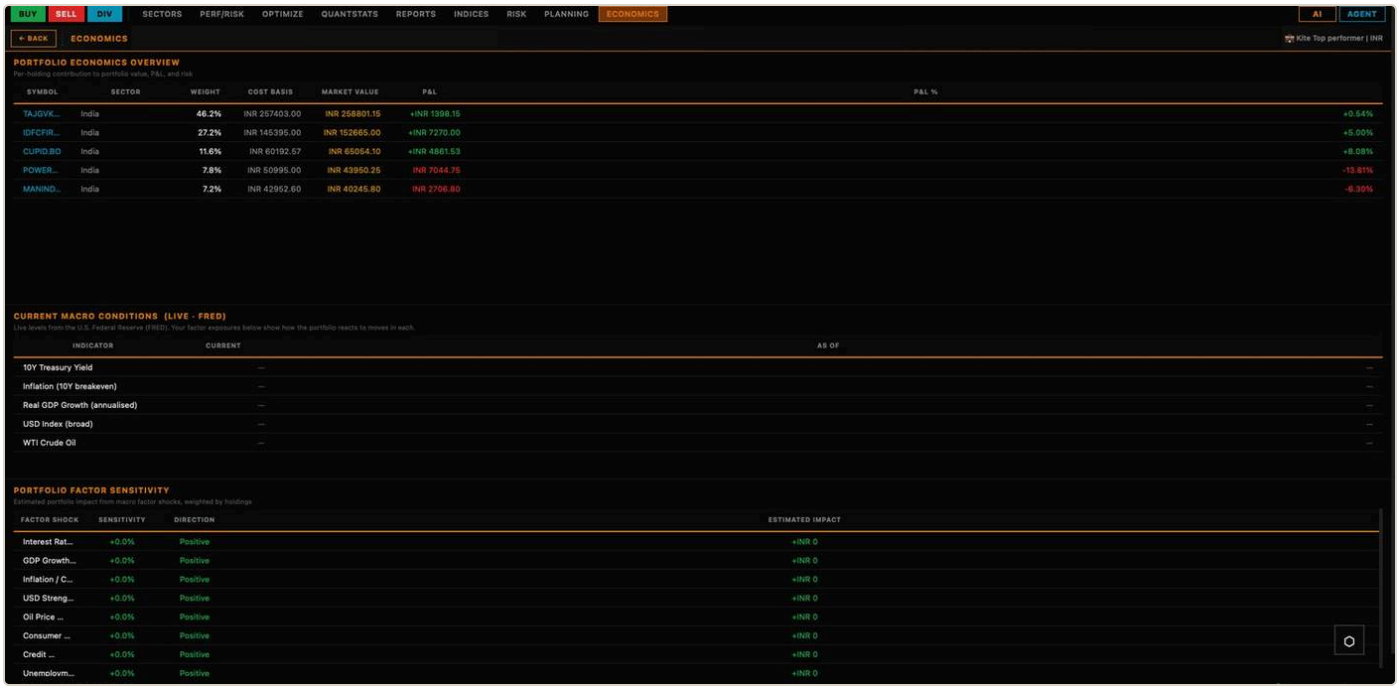
Savings — projects wealth across a band of savings rates vs your current rate.

> **READING NOTE**
>
> Once a portfolio has ≥20 days of return history, the calculators calibrate the expected-return & volatility inputs to your book (shown as a "Calibrated to your portfolio" note); until then they use your typed assumptions. All planning maths is local — no network, no credits.

12 Economics & macro factors

VIEW 9 · AUTO-RUNS

The ECONOMICS view puts the book in a macro context — a per-holding economic breakdown, live macro indicator levels from the U.S. Federal Reserve (FRED), and the portfolio's estimated sensitivity to macro-factor shocks.



Three stacked sections. Portfolio Economics Overview (per-holding sector, weight, cost, value, P&L), Current Macro Conditions (10Y yield, inflation, real GDP, USD index, WTI crude — live from FRED), and Portfolio Factor Sensitivity (rates, growth, inflation, USD, oil, spending, credit, unemployment).

SECTION	COLUMNS
Economics Overview	SYMBOL · SECTOR · WEIGHT · COST BASIS · MARKET VALUE · P&L · P&L%
Macro Conditions	INDICATOR · CURRENT · AS OF — 5 FRED series, live via the edge
Factor Sensitivity	FACTOR SHOCK · SENSITIVITY · DIRECTION · ESTIMATED IMPACT

READING NOTE

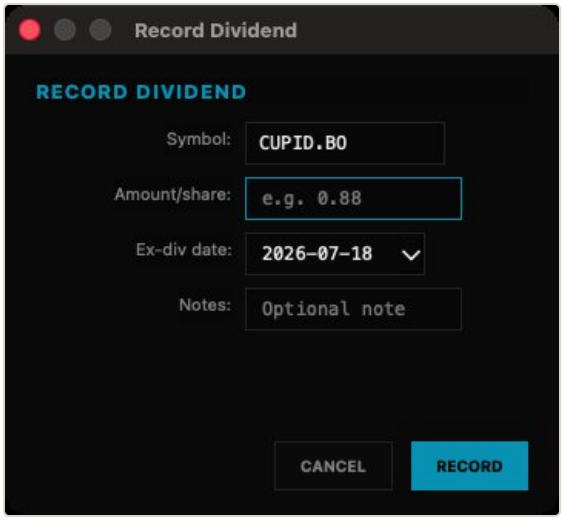
The macro levels are pulled live from FRED through the Fincept backend when the feed is configured (they display "—" if the server has no FRED data available). Factor sensitivities are keyed off each holding's GICS sector — holdings classified only by **country** (e.g. an Indian listing shown as sector "India") map to a neutral factor profile, so their estimated impact reads near zero.

13 Trading — order entry & dividends

ACTIONS

Trades are recorded straight from the command bar. **BUY** and **SELL** slide in an order-entry panel for the selected holding; **DIV** opens the dividend dialog.

Order-entry panel — BUY/SELL toggle, the selected symbol with its price, quantity held & market value, and OPEN BUY ORDER.



Record Dividend. Symbol, amount per share, ex-dividend date and notes.

HOW AN ORDER IS PLACED

1. Select a holding (heatmap or blotter).
2. Press BUY or SELL — the panel slides in from the right.
3. Confirm the side and press OPEN BUY ORDER / OPEN SELL ORDER.
4. Fill quantity & price in the **Add Asset** / **Sell Asset** dialog — the buy form has a live ticker search — and confirm.

A BUY upserts the holding at a weighted-average cost and records a BUY transaction; a SELL validates against the held quantity and records a SELL. A dividend records a DIVIDEND transaction (quantity × amount per share).

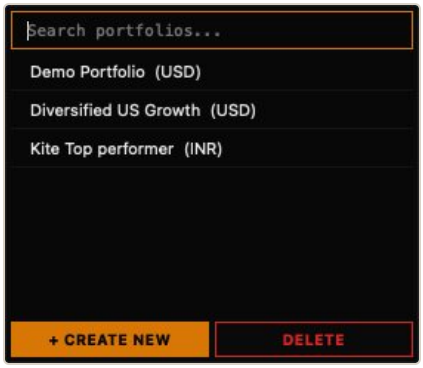
All eight portfolio forms at a glance

DIALOG	KEY FIELDS	CONFIRM
Create Portfolio	Name · Owner · Currency (36 ISO codes)	CREATE
Add Asset (Buy)	Symbol (live search) · Quantity · Price	ADD
Sell Asset	Quantity (≤ held) · Price	SELL
Record Dividend	Symbol · Amount/share · Ex-div date · Notes	RECORD
Edit Transaction	Quantity · Price · Date · Notes	SAVE
Import Portfolio	File · New / Merge · target portfolio	IMPORT
Sector Mapping	Per-holding sector picker (16 sectors)	SAVE
Delete / Close	Confirmation only	DELETE

14 Managing portfolios

ACTIONS

Create, switch, import and export portfolios from the selector dropdown and the ⋯ overflow menu.



Portfolio selector. Search, switch between books, and reach + CREATE NEW / DELETE.

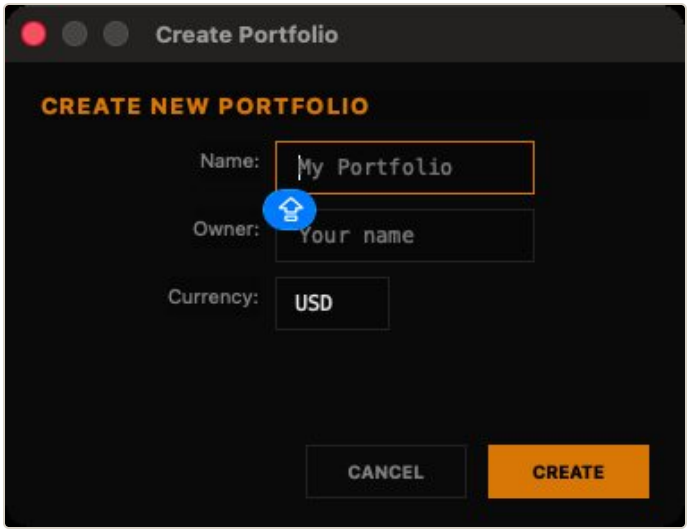
menu. Export CSV / JSON, Import JSON…, FFN Analysis, Backtest Portfolio.

CREATE & IMPORT

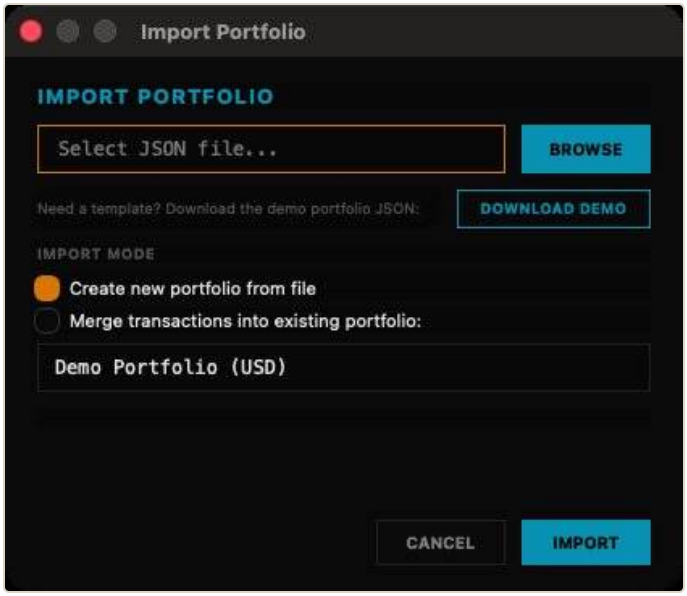
A new portfolio needs only a name (currency defaults to USD, chosen from 36 ISO codes). Import reads an exported JSON of transactions and replays them into a new or existing (merge) portfolio; a **Download Demo** button writes a sample file to start from.

EXPORT

CSV emits a holdings section (`Symbol, Quantity, AvgBuyPrice, CostBasis`) and a transactions section. **JSON** emits the transaction ledger for re-import.



Create Portfolio — name, owner and currency.



Import Portfolio — browse to a JSON file, choose **Create new** or **Merge into existing**, and import.

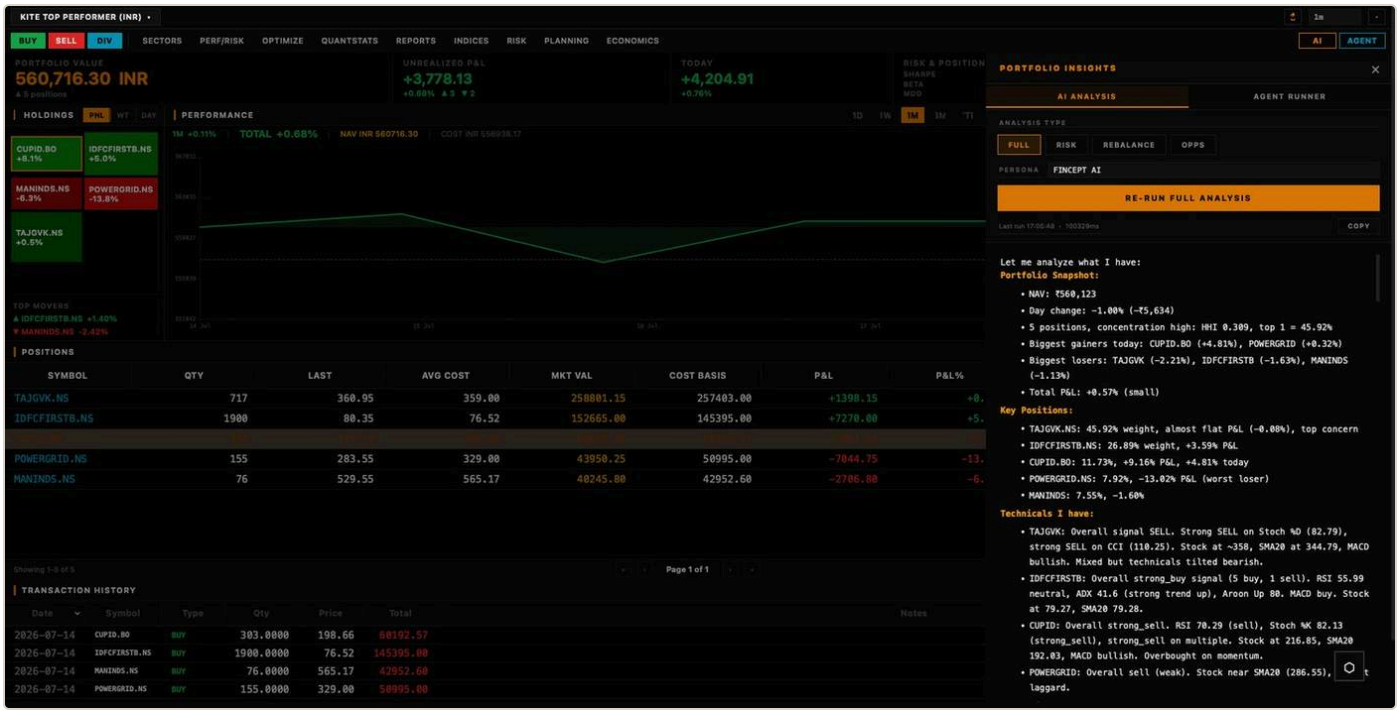
ALSO ON THE MENU

FFN Analysis opens a six-tab quantitative view (overview, benchmark, optimisation weights, rebased prices, drawdowns, rolling correlations — a server compute, no credits). **Backtest Portfolio** packages the current holdings and hands them to the full Backtesting terminal.

15 AI & Agent insights

INTELLIGENCE · CREDITS

The **AI** and **AGENT** buttons open a right-hand **Portfolio Insights** dock. The AI tab writes a natural-language review of the book; the Agent tab runs a saved specialist agent against it. Both stream their answer live and cache it per-portfolio, so re-opening the dock shows the last result without re-running.



AI Analysis. Choose an analysis type (FULL / RISK / REBALANCE / OPPS) and a persona, then run. The analyst is fed a precise, pre-computed context — NAV, concentration (HHI), per-holding P&L contribution, technicals — and asked to reason over it rather than invent numbers.

Agent Runner. Pick a saved agent (e.g. a Currency & FX strategist) and run it on the portfolio.

ANALYSIS TYPES

TYPE	FOCUS
FULL	Complete review
RISK	Risk & exposure
REBALANCE	Reweighting ideas
OPPS	Opportunities

CREDITS

This is the **only** credit-spending action on the tab — running the AI or Agent analysis calls the Fincept LLM. Opening the dock, switching tabs, or reading a cached result costs nothing. **Copy** exports the result as markdown.

16 Analytics engine, live data & metrics

REFERENCE

Three layers feed the Portfolio tab: a live WebSocket for streaming value & risk, request/response calls to the Exclusive API for data & heavy analytics, and local C++ maths for the everyday metrics.

Live stream & where each number comes from

DATA	SOURCE	MODE
Portfolio value / P&L / day move	WebSocket `portfolio:value`	Streaming (drives the ribbon & status bar)
Server risk metrics	WebSocket `portfolio:metrics`	Streaming
Portfolios / holdings / transactions	`/v1/portfolios` (Exclusive API)	Request/response, synced to your account
Live quotes	Connected broker, else market feed	Per refresh
Benchmark & correlation candles	`/v1/market/candles`	Request/response
NAV history	`/v1/analytics/…/nav`	Reconstructed from transactions, once per session
Risk-free rate	`/v1/analytics/risk-free-rate`	10Y Treasury, cached 24h
Optimize · QuantStats · Monte-Carlo · FFN	`/v1/analytics/…/{kind}/stream`	Server compute, streamed (no LLM credits)
Macro factors	`/v1/analytics/macro/factors`	Live FRED via the edge
AI / Agent analysis	`/v1/research/chat`	LLM — **spends credits**

Metrics computed locally

METRIC	METHOD
Concentration	Sum of top-3 holding weights; HHI across sectors
Volatility	Std-dev of daily NAV returns × √252
Sharpe / Sortino	Excess return over total / downside deviation, annualised
Beta / Alpha	OLS vs SPY; annualised Jensen's alpha (CAPM)
Max Drawdown	Peak-to-trough of the NAV series
VaR / CVaR 95%	Historical simulation; CVaR = mean of the worst 5% tail
Risk score	0–100 composite of vol, concentration, drawdown & beta
Correlation	Pearson of 30-day daily returns (computed in-app)

When the server pushes its own risk metrics over the stream, they take precedence over the local estimate — the display is identical either way.

17 Agent automation tools

REFERENCE · 14 TOOLS

The whole portfolio data model is exposed to the terminal's AI agents and MCP clients as **14 tools** in the `portfolio` category — five read-only and nine that create, edit or delete. The three delete-style tools are flagged so they always prompt for confirmation before running.

TOOL	DOES	TYPE
list_portfolios	List every named portfolio	Read
get_portfolio	Get one portfolio's metadata	Read
get_portfolio_assets	List a portfolio's holdings	Read
get_transactions	Transaction history	Read
get_portfolio_snapshots	Historical performance snapshots	Read
create_portfolio	Create a new portfolio	Create
update_portfolio	Rename / edit metadata	Edit
add_portfolio_asset	Add a holding	Create
update_portfolio_asset	Edit a holding (qty / avg price)	Edit
add_transaction	Record BUY / SELL / DIVIDEND / SPLIT	Create
update_transaction	Correct a transaction	Edit
delete_portfolio	Delete a portfolio + all its data	**Delete** ⚠
remove_portfolio_asset	Remove a holding by symbol	**Delete** ⚠
delete_transaction	Delete a transaction by id	**Delete** ⚠

WHY THIS MATTERS

An institutional desk can run the entire portfolio lifecycle — create books, add positions, record trades, pull snapshots for reporting — through the AI chat or an external MCP client, with the same server-side data the UI uses. Destructive operations always surface a confirmation prompt first.

18 Quick reference & honest notes

REFERENCE

How do I...

GOAL	DO THIS
Switch portfolios	Click the selector ▾ → pick / search a book
Create / import a portfolio	Selector ▾ → + CREATE NEW, or ⋯ → Import JSON...
See sector concentration	SECTORS → Overview
Check Sharpe / drawdown	KPI ribbon, or PERF/RISK
Find an optimal allocation	OPTIMIZE → pick a method → RUN OPTIMIZATION
Run a full tearsheet	QUANTSTATS (Monte-Carlo on its own button)
Stress-test the book	RISK → Stress Test
Plan for retirement / a goal	PLANNING
Record a trade / dividend	Select a holding → BUY / SELL, or DIV
Export for reporting	⋯ → Export CSV / Export JSON
Get an AI review	AI → choose type → run (spends credits)
Automate via agents	The 14 `portfolio` tools (§17)

Honest notes for accurate reading

- **Beta & Alpha are vs SPY** regardless of the portfolio's home currency, and need aligned history before they read meaningfully.
- **Stress-test impact is beta-scaled** — a near-zero measured beta produces muted scenario impacts until more return history accrues (§10).
- **Economics factor sensitivity** keys off GICS sector; holdings identified only by country (e.g. an Indian listing) map to a neutral profile and read near zero. Macro levels display "—" when the server has no live FRED data (§12).
- **Custom indices** save locally; seven of the ten methods currently reduce to a weight-based formula (§9).
- **QuantStats & Optimize** use fixed settings (1-year lookback, 1,000 MC paths; the Returns / Risk-Model combos and a Target-Return field are not yet wired to the server).
- **Analytics need a reachable backend** — Optimize / QuantStats / Monte-Carlo / FFN / NAV / macro all run server-side. They spend **no LLM credits**; only the AI / Agent analyst does.
- **Nothing here writes to your book** unless you explicitly trade, import, or delete — every view is read-only analytics computed on the fly.

THE ESSENTIALS IN ONE PARAGRAPH

Pick a portfolio and the workstation streams its value and risk live. Read the book as a heatmap, a sortable blotter, a NAV chart and a sector/correlation panel. Then press any of nine analytics pills — sectors, performance, optimization, a QuantStats tearsheet, reports, custom indices, risk & stress, planning, macro — each replacing the dashboard and

returning with one click. Trade, record dividends, import and export from the same bar, ask the built-in AI analyst for a review, and automate the lot through 14 agent tools.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Strategy Backtesting

A complete, screen-by-screen walkthrough of the Backtest tab — six engines, eleven commands, 110+ strategies, and five result windows.

`>_ 6 engines · 11 commands · 110+ strategies`

Covers · **the 3-panel workbench** · **6 engines** · **11 commands** **110+ strategies** · **5 result windows** · 4 agent tools

Demonstrated live with real runs on the Fincept Terminal Backtest tab.

Contents

10 SECTIONS

GETTING ORIENTED

Overview — what the Backtest tab is 3

1 Anatomy of the workbench (top bar, three panels, status bar) 4

2 Providers — the 6 backtest engines 6

3 Commands — the 11 operations & how the UI morphs 7

4 Configuring a run — market data, execution, advanced, benchmark, strategy 8

5 The five result windows (Summary · Equity · Metrics · Details · Raw JSON) 9

RUNNING STRATEGIES

6 Core commands — Backtest, Optimize, Walk-Forward 13

7 Analysis commands — Indicators, Signals, ML Labels, CV Splits, Returns, Sweep 15

8 Strategy catalog (per engine) + the Custom Combo builder 19

REFERENCE

9 Engine, data & AI agent tools 20

10 Quick reference — “How do I…?” & important notes 21

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal, taken while driving live backtest runs. Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

What the Backtest tab is

OVERVIEW

The Backtest tab is a **multi-engine strategy research terminal**. You pick a provider (a backtesting engine), a command (what to run — a backtest, an optimization, an analysis…), a strategy, and a market-data window, then press **RUN**. Results render across five result windows (Summary, Equity Curve, Metrics, Details, Raw JSON). The whole workbench reconfigures itself around the command you choose — that is the key idea to grasp before anything else.

Under the hood each provider is a real Python backtesting library run as an out-of-process worker (VectorBT, Backtesting.py, FastTrade, Zipline, BT, and the in-house Fincept engine). Market data comes from Yahoo Finance by default, with an automatic live-broker candle path for VectorBT when an Indian broker is connected. Everything runs asynchronously; the terminal stays responsive while a run executes.

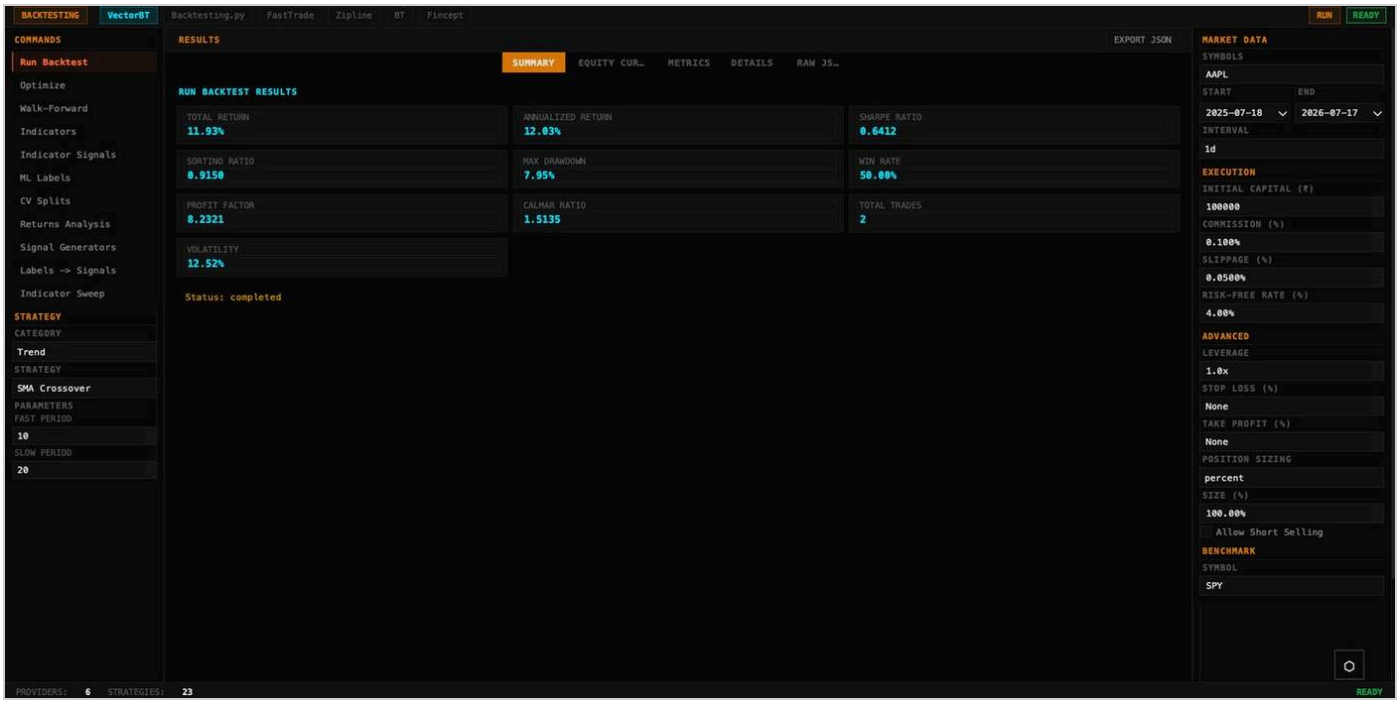
AT A GLANCE, IN BACKTEST YOU CAN…

- **Backtest 110+ built-in strategies** across 6 engines, on any symbol(s) and date range, with realistic execution (commission, slippage, leverage, sizing, stops, shorting)
- **Optimize** a strategy's parameters over a grid / random search and read the best-parameter set
- Run **walk-forward analysis** for out-of-sample robustness across rolling train/test splits
- Compute **indicators**, turn them into entry/exit signals, and generate **ML labels**
- Build **cross-validation splits** and analyse returns (cumulative, rolling, drawdown, distribution, benchmark)
- Generate random / probabilistic signals, convert labels to signals, and sweep an indicator parameter
- Read every result as headline cards, an equity / drawdown chart, a metrics table, a trades table and raw JSON — and export it
- Drive all of the above from the **AI agent via 4 MCP tools**

1 Anatomy of the workbench

ORIENTATION

Open Backtest from the main tab bar (**BACKTEST**) or the command bar. The screen is a top bar over three resizable panels (a splitter you can drag) over a status bar.



The Backtest workbench (Run Backtest command, Summary result). ① Top bar — brand · provider pills · RUN · status ② Left — commands, strategy & parameters ③ Center — the five result windows + EXPORT JSON ④ Right — market data & run configuration ⑤ Status bar — provider/strategy counts.

Region	What it holds / does
① **Top bar**	The **BACKTESTING** brand, the six **provider** pills (the active one is coloured), the **RUN** button, and a live status chip (`READY` / `EXECUTING…` / `ERROR`).
② **Left panel**	The **COMMANDS** list (11 buttons; unsupported ones grey out per provider) and — for the backtest family only — the **STRATEGY** block (category + strategy pickers) and **PARAMETERS** (auto-built per strategy).
③ **Center panel**	The **RESULTS** header with an **EXPORT JSON** button, over a five-tab result viewer: **SUMMARY** · **EQUITY CURVE** · **METRICS** · **DETAILS** · **RAW JSON**.
④ **Right panel**	Run configuration: **MARKET DATA** (always shown), and — depending on the command — **EXECUTION**, **ADVANCED**, **BENCHMARK**, plus a command-specific config block.
⑤ **Status bar**	`PROVIDERS: 6` , a live `STRATEGIES:` count (updates as each engine's strategies load), and the run-state chip.

> **The one concept that explains the whole tab.** The command you pick **morphs the panels**. The *backtest family* (Run Backtest, Optimize, Walk-Forward) shows the strategy picker, Execution and Advanced sections. The *analysis commands* (Indicators, ML Labels, CV Splits, Returns, Signals, …) hide those and show only Market Data + that command's own configuration. Optimize additionally reveals a MIN/MAX/STEP range editor under each strategy parameter. Market Data is the only section shown for every command.

Reading the workflow

Provider (which engine) → Command (what to do) → Strategy + Parameters (left) → Market data & execution config (right) → **RUN** → results (center). The status chip turns amber `EXECUTING…` during the run and green `READY` when results arrive.

2 Providers — the 6 backtest engines

TOP BAR

A "provider" is the backtesting **engine** that runs your test. Each is a Python library run as an out-of-process worker. Selecting a provider re-skins the pills, enables only the commands that engine supports, and reloads its strategy catalog. VectorBT is selected by default and is the most capable.

Provider	Engine / nature	Commands supported
VectorBT	Flagship, fully-featured vectorized engine; the only one wired for live-broker candle injection.	**All 11** commands.
Backtesting.py	The `backtesting` PyPI library. Needs `pip install backtesting`.	Backtest, Optimize, Walk-Forward, Indicators.
FastTrade	The `fast_trade` library. Needs `pip install fast_trade`.	Backtest only.
Zipline	The Zipline engine; near-full command set.	10 (all except Indicator Sweep).
BT	The `bt` portfolio library (strong on weighting/rebalancing).	Backtest, Optimize, Walk-Forward.
Fincept	In-house engine over a 418+ strategy registry.	Backtest, Optimize, Walk-Forward.

If a provider isn't installed the pill stays visible and RUN returns a clean install error (e.g. "needs pip install backtesting") rather than hiding the option — so you know exactly what to install. Unsupported command buttons are greyed for the active provider; picking one anyway returns "Command '...' is not supported by provider '...'".

At a glance, providers…

- Are interchangeable engines — the same strategy/symbol/date inputs run on whichever engine you select.
- Gate which commands and strategies are available; VectorBT unlocks everything, FastTrade only a basic backtest.
- Cache their strategy & option lists for 10 minutes, so re-selecting a provider is instant.

3 Commands — the 11 operations

LEFT PANEL

The command decides what RUN does and which configuration windows appear. There are 11, in a fixed order. The first three are the "backtest family"; the rest are quantitative-research utilities.

Command	What it does	Shows
Run Backtest	Run a strategy over the window; returns performance, trades and an equity curve.	Strategy · Exec · Adv · Bench
Optimize	Grid/random search over strategy parameters; returns the best set + its performance + a sweep table.	+ MIN/MAX/STEP · Optimization
Walk-Forward	Rolling in-/out-of-sample robustness test across N splits.	Strategy · Exec · Adv · Splits
Indicators	Compute one indicator over the series; stats + sample series.	Market data only + Indicator
Indicator Signals	Turn an indicator into entry/exit signals (crossover / threshold / breakout / mean-reversion / filter).	Market data + Signal config
ML Labels	Generate financial-ML target labels (FIXLB / MEANLB / LEXLB / TRENDLB / BOLB) + class distribution.	Market data + Labels
CV Splits	Build cross-validation splitters (Rolling / Expanding / Purged K-Fold).	Market data + Splitter
Returns Analysis	Return-series analytics: cumulative / rolling / drawdown / distribution / benchmark comparison.	Market data + Benchmark + Returns
Signal Generators	Random / probabilistic signal generators (RAND / RANDX / RANDNX / RPROB / RPROBX).	Market data + Generator
Labels → Signals	Convert ML labels into tradeable entry/exit signals.	Market data + Labels→Signals
Indicator Sweep	Run one indicator across a MIN/MAX/STEP grid; aggregated stats per value (VectorBT).	Market data + Sweep

Section-visibility rule. EXECUTION, ADVANCED and the STRATEGY picker appear only for Run Backtest / Optimize / Walk-Forward. BENCHMARK appears for those three plus Returns Analysis. MARKET DATA is always visible. The per-parameter MIN/MAX/STEP editor appears only under Optimize.

4 Configuring a run

RIGHT + LEFT

These are the inputs a Run Backtest consumes. The right panel holds the market/execution config; the left panel holds the strategy and its parameters.

MARKET DATA (always shown)

Field	Type	Meaning & default
SYMBOLS	text	Comma-separated tickers (default `SPY`, placeholder `SPY,AAPL,MSFT`). The first symbol is the "active" ticker for symbol-group linking.
START / END	date	Backtest window (defaults: one year ago → yesterday).
INTERVAL	combo	`1d · 1wk · 1h · 4h · 15m · 5m · 1m` (default 1d).

EXECUTION & ADVANCED (backtest family)

Field	Type	Meaning & default
INITIAL CAPITAL	number	Starting cash (default 100,000; currency-bound label).
COMMISSION (%)	number	Per-trade commission (default 0.1%). Sent as a fraction.
SLIPPAGE (%)	number	Execution slippage (default 0.05%).
RISK-FREE RATE (%)	number	For Sharpe/Sortino etc. (default 4.0%).
LEVERAGE	number	Gross leverage multiplier (default 1.0×).
STOP LOSS / TAKE PROFIT (%)	number	Optional exits — **None** at 0; only sent when > 0.
POSITION SIZING	combo	`percent · fixed · kelly · vol_target · risk`. The value field below relabels per method (SIZE % / SIZE $ / RISK PER TRADE % / TARGET %).
Allow Short Selling	checkbox	Permit short positions.
BENCHMARK · SYMBOL	text	Comparison baseline (default `SPY`) — drives alpha/beta/information-ratio.

STRATEGY & PARAMETERS (left panel)

- **CATEGORY** then **STRATEGY** combos — the categories and strategies are loaded from the active engine (e.g. Trend ▸ SMA Crossover). See §8 for the full catalog.
- **PARAMETERS** — a spin-box is auto-built for each of the strategy's tunable parameters (e.g. Fast Period 10, Slow Period 20), with the engine's ranges and defaults.
- Under **Optimize**, each parameter also gets a **MIN / MAX / STEP** row that defines its search grid.

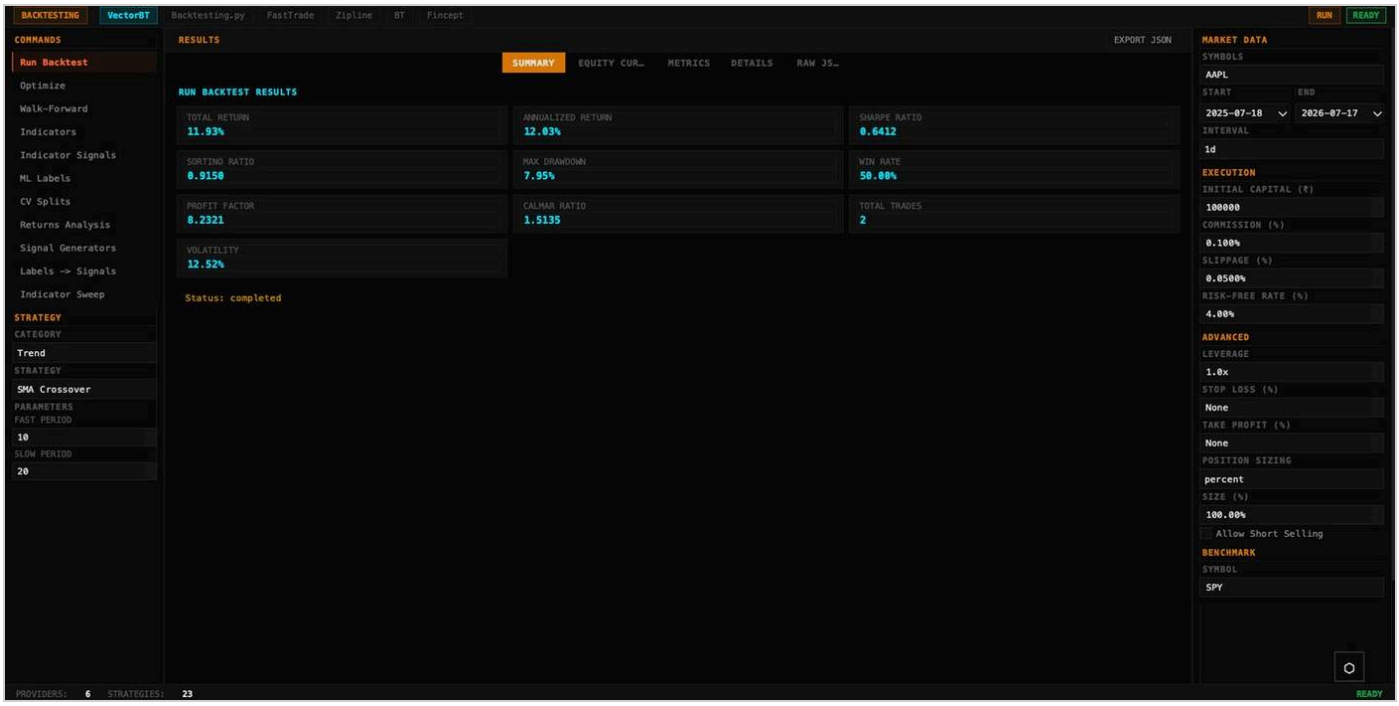
> **Symbol format matters.** Symbols are passed to the engine as typed. Non-US tickers must carry their exchange suffix (e.g. `RELIANCE.NS`, `PIDILITIND.NS`) or Yahoo will resolve them wrong. US symbols (`AAPL`, `SPY`) need no suffix.

5 The five result windows

CENTER

Every run populates five tabs in the center panel. After a run the view snaps to SUMMARY; the other four are one click away. Here is each, demonstrated on a Run Backtest of AAPL with the SMA Crossover strategy.

① SUMMARY — headline cards



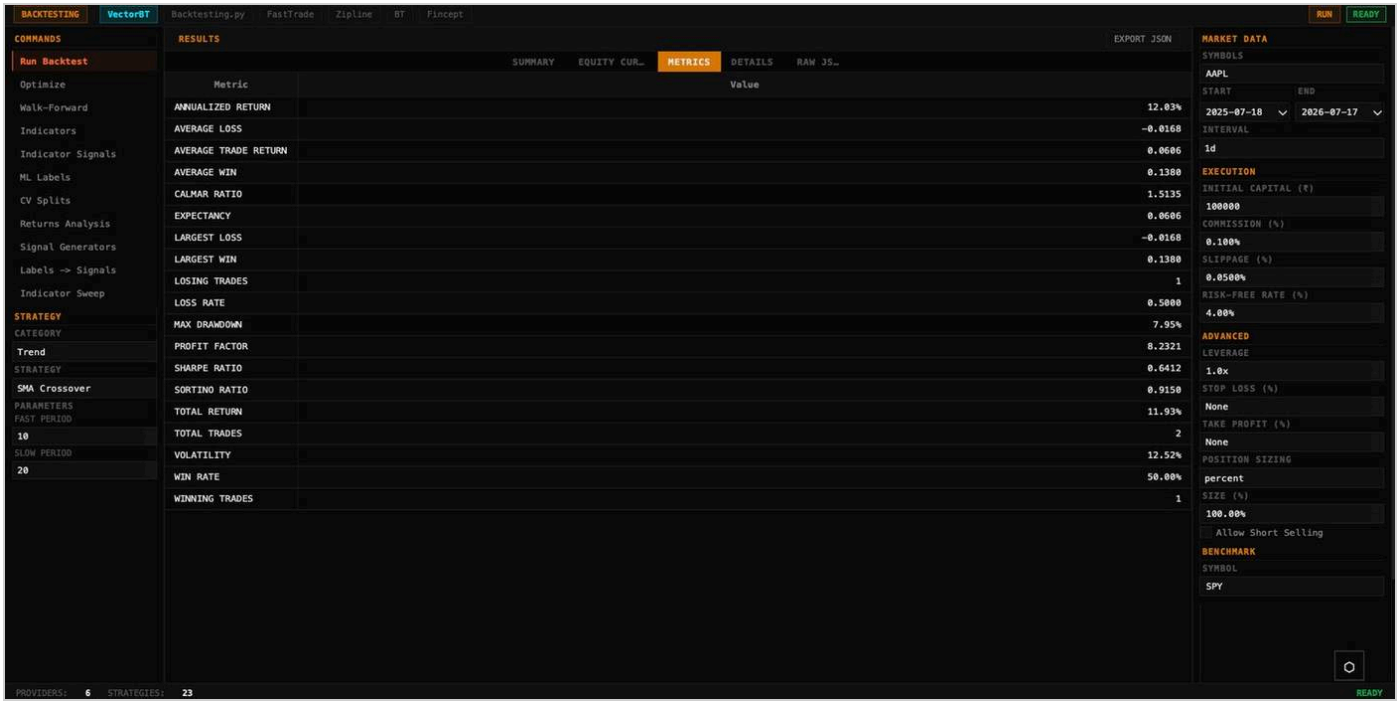
SUMMARY — a titled results header and a grid of headline metric cards (total return, annualized return, Sharpe, Sortino, max drawdown, win rate, profit factor, Calmar, total trades, volatility) plus a status line.

The card set adapts to the command: Optimize shows best-parameter & best-performance cards, Walk-Forward shows robustness cards, ML Labels shows a class distribution, and so on (see §6–§7).

② EQUITY CURVE — portfolio value & drawdown

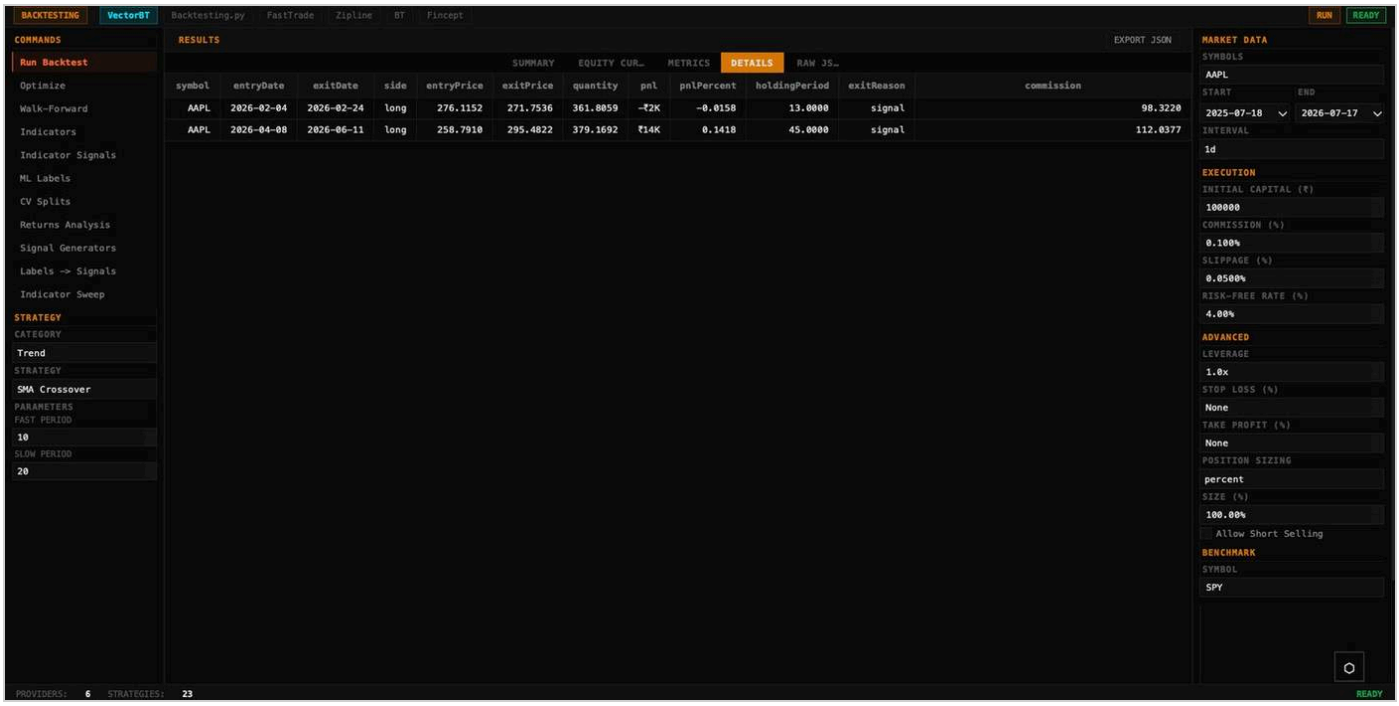
EQUITY CURVE — the portfolio value over time (left axis, engine-accent line) with an overlaid drawdown series (right axis, red %) when the drawdown is material.

③ METRICS — the full metric table



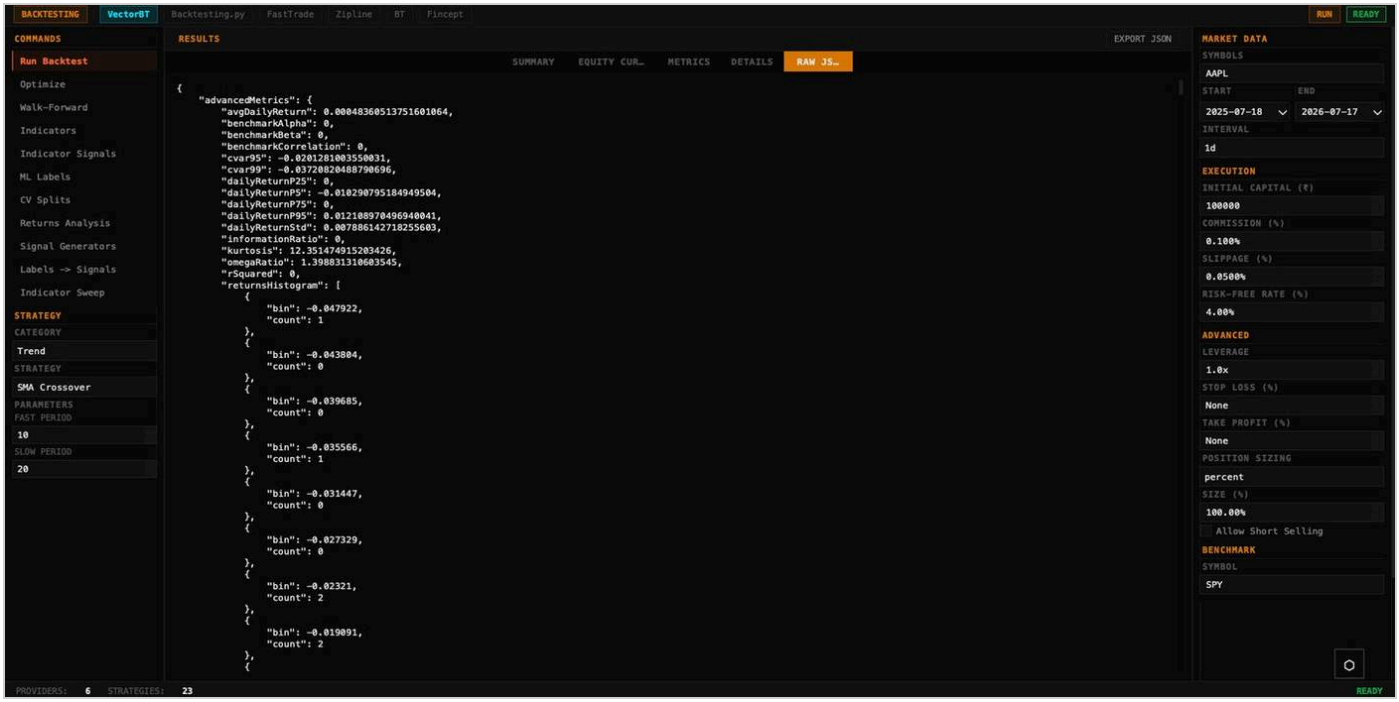
METRICS — every performance metric the engine emitted, as a Metric/Value table (annualized return, average win/loss, Calmar, expectancy, largest win/loss, loss rate, max drawdown, profit factor, Sharpe, Sortino, total return, total trades, volatility, win rate, winning/losing trades…).

④ DETAILS — the trades / records table



DETAILS — one row per closed trade for a backtest: symbol, entry/exit dates, side, entry/exit price, quantity, P&L, P&L %, holding period, exit reason, commission. For other commands this table holds the command's records (parameter combinations, walk-forward windows, CV folds, signals, sweep results…), capped at 200 rows.

⑤ RAW JSON — the full payload & EXPORT



RAW JSON — the complete, pretty-printed engine payload (always populated). The **EXPORT JSON** button in the header saves exactly this to a .json file and registers it with the File Manager.

At a glance, the result windows give you…

- A scannable headline (SUMMARY), the equity/drawdown picture (EQUITY CURVE), the complete metric set (METRICS), the trade-by-trade detail (DETAILS) and the machine-readable payload (RAW JSON) — for every command.

6 Core commands — Backtest, Optimize, Walk-Forward

BACKTEST FAMILY

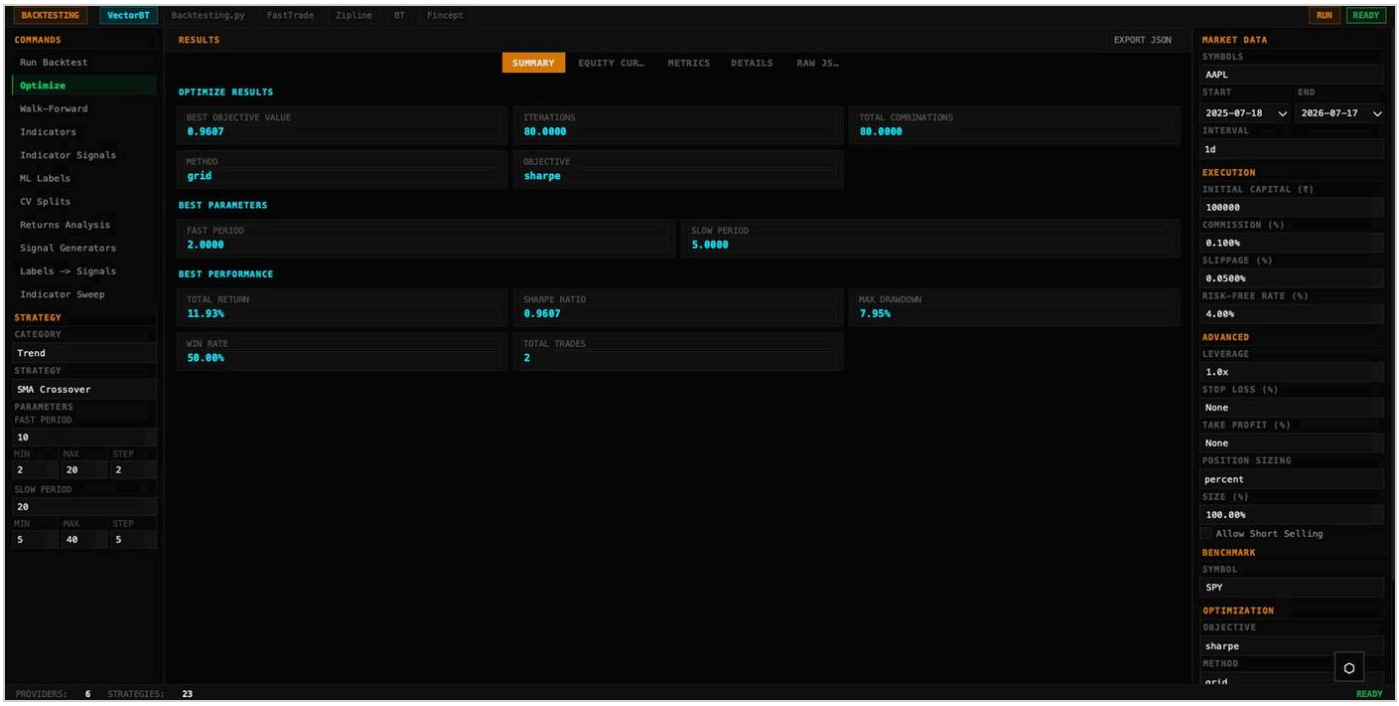
Run Backtest

The default command. Pick a strategy, set the window and execution config, RUN. Results: performance cards + a green/amber **Status** line, the metrics table (from the engine's `performance` object), the trades table, and the equity/drawdown chart. This is the workbench shown throughout §5.

Optimize

Searches the strategy's parameter space for the best objective. The left panel exposes a **MIN / MAX / STEP** editor under every parameter; the right panel adds an **OPTIMIZATION** block (objective, method, max iterations).

Control	Type	What it does
OBJECTIVE	combo	Metric to maximise — `sharpe · sortino · calmar · return`.
METHOD	combo	`grid` (exhaustive) or `random`.
MAX ITERATIONS	number	Cap on evaluated combinations (default 100).
Per-param MIN / MAX / STEP	number×3	The search grid for each strategy parameter (left panel).



Optimize — result of an SMA-crossover grid search: OPTIMIZE RESULTS (best objective 0.9607 across 80 combinations, method grid, objective sharpe), BEST PARAMETERS (Fast 2 / Slow 5) and BEST PERFORMANCE cards. The left panel shows each parameter's MIN/MAX/STEP grid; the DETAILS tab holds one row per parameter combination.

Walk-Forward

Tests robustness by repeatedly training on an in-sample window and evaluating out-of-sample, rolling forward across N splits. Right-panel config: **NUMBER OF SPLITS** (2–20, default 5) and **TRAIN RATIO** (0.5–0.95, default 0.7). Results: summary cards — N windows, average out-of-sample return & Sharpe, robustness score, degradation ratio — plus a DETAILS table with one row per window (train/test dates, in- and out-of-sample return & Sharpe).

> **At a glance, the core commands…**
>
> - Backtest a strategy, optimize its parameters over a grid, and validate it walk-forward — the standard "does this edge survive?" workflow, all from the same left-panel strategy picker.

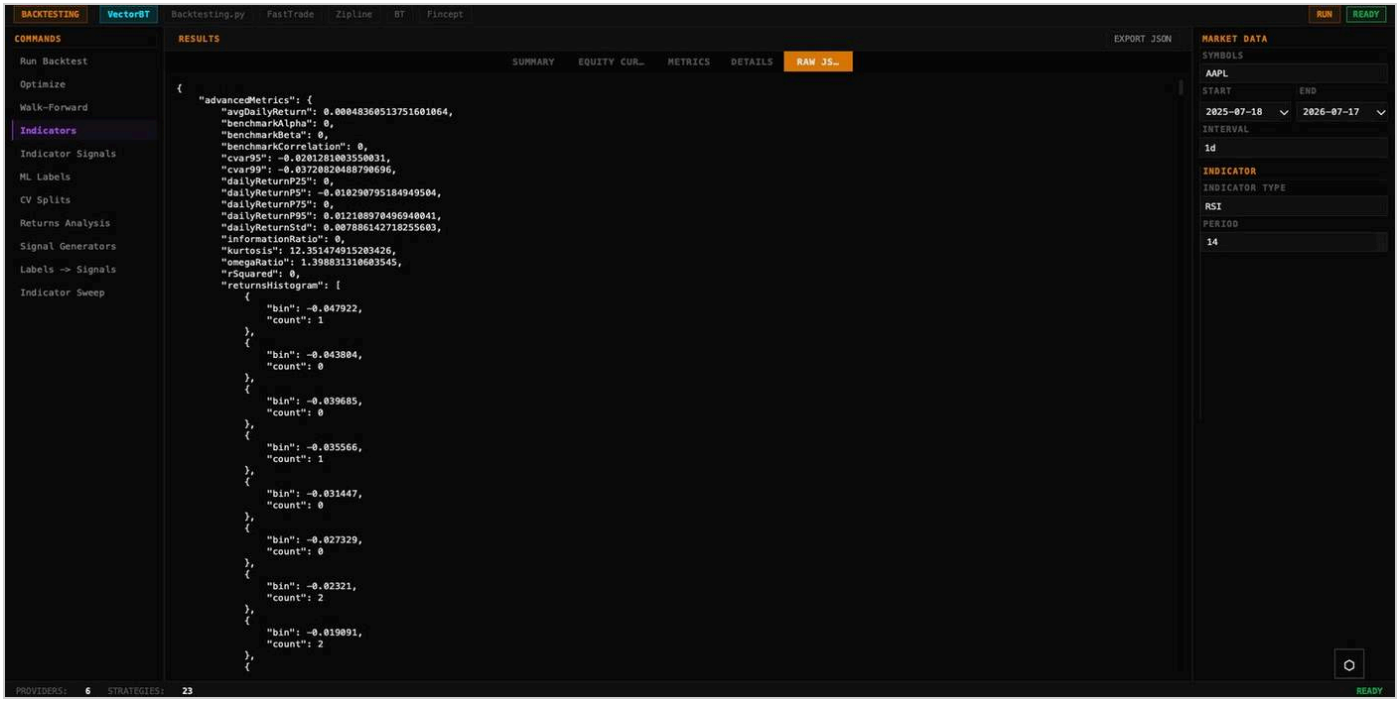
7 Analysis commands

RESEARCH UTILITIES

The remaining eight commands are quantitative-research utilities. They hide the strategy/execution sections and show only Market Data plus their own configuration window. The centre keeps showing the previous result until you RUN the new command.

Indicators

Compute a single indicator series and its statistics. Config: **INDICATOR TYPE** (loaded from the engine) and **PERIOD** (2–500, default 14). Results: indicator/symbol/period/bar cards, a STATISTICS section (last / mean / std / min / max), and a DETAILS table of the sampled indicator series.



Indicators — a minimal config (indicator type + period). Note the left panel: the strategy picker is gone (this isn't a backtest-family command), leaving just the command list.

Indicator Signals

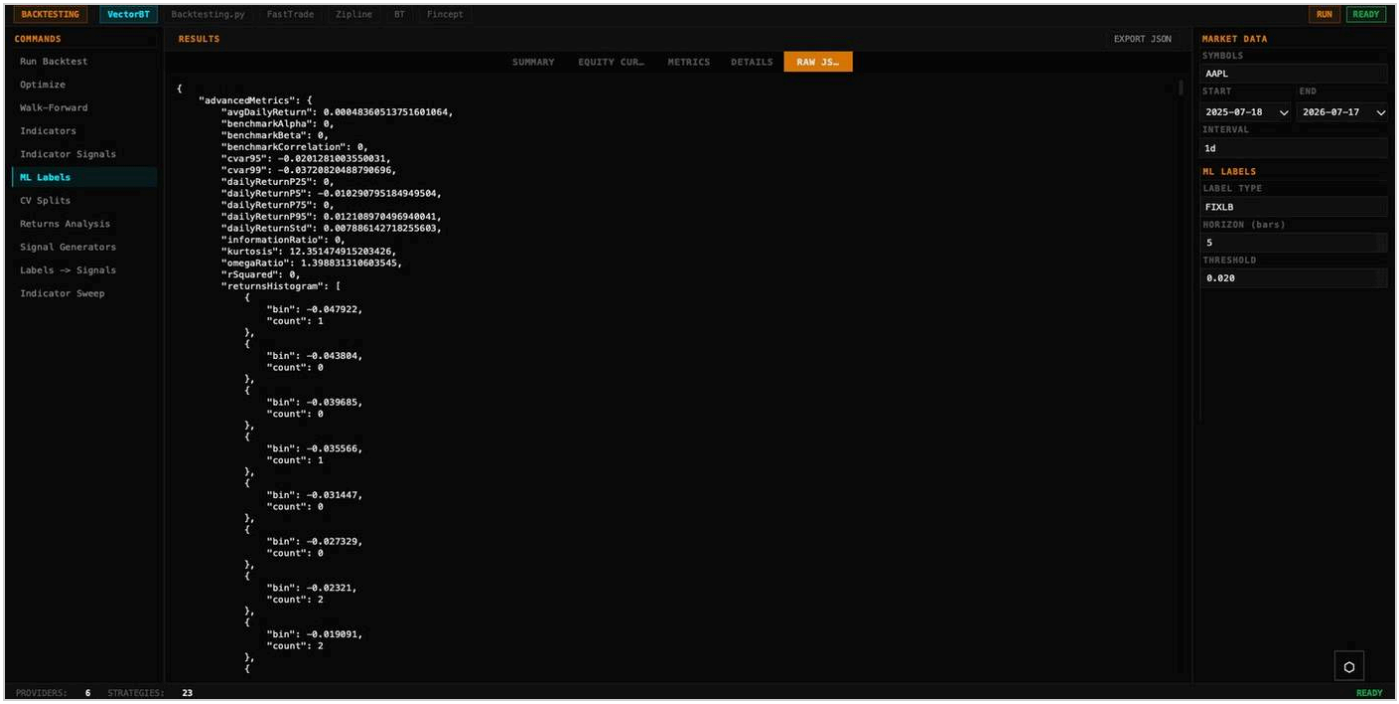
Turn an indicator into entry/exit signals. Config: **INDICATOR** + **SIGNAL MODE**, which reveals mode-specific fields:

Mode	Fields revealed
crossover	Fast Period, Slow Period, MA Type (ma / ema).
threshold	Period, Lower Threshold, Upper Threshold.
breakout	Period, Channel (donchian / bbands / keltner).
mean_reversion	Period, Z-Entry (σ), Z-Exit (σ).
filter	Filter Indicator (adx/atr/mstd/zscore/rsi), Filter Period, Filter Threshold, Filter Type.

Results: signal-quality cards (total signals, win rate, profit factor, avg/median return, best/worst trade, signal density, entry/exit counts) and a DETAILS table of entry/exit signals tagged ENTRY/EXIT.

ML Labels & Labels → Signals

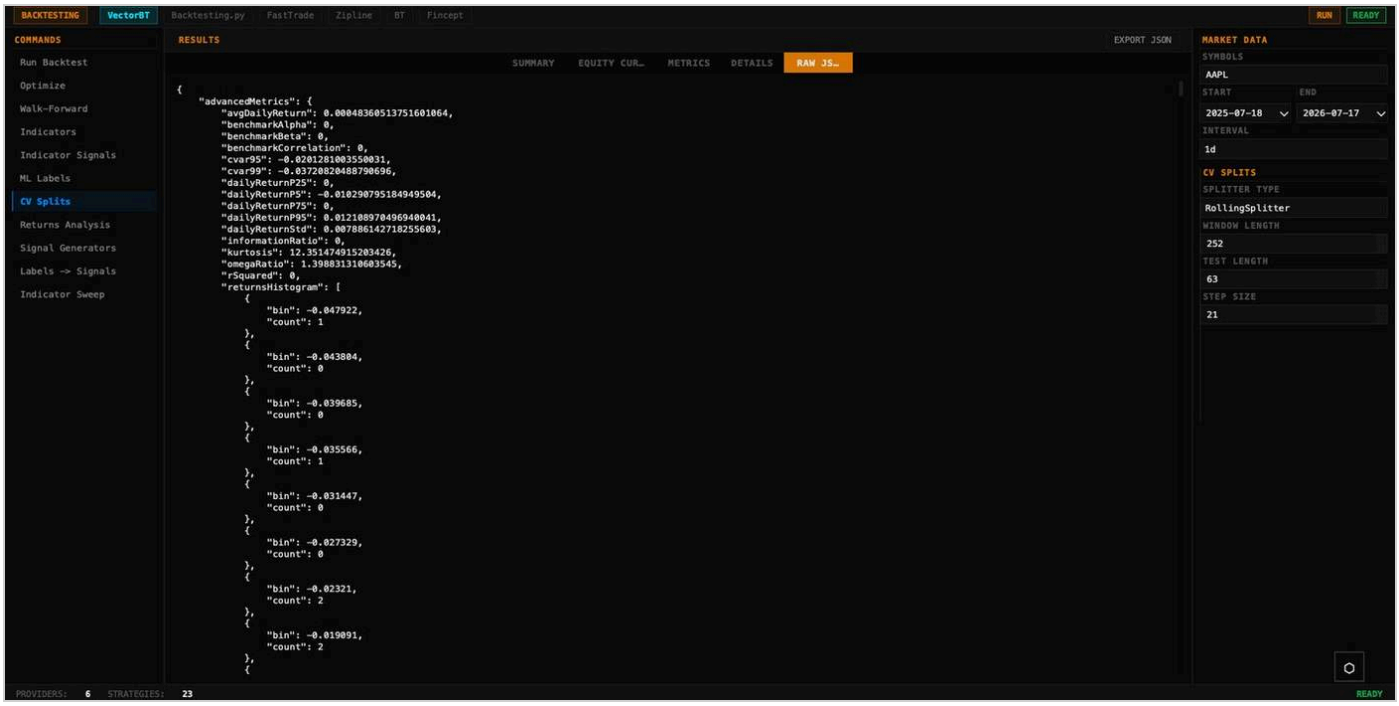
Generate financial-ML target labels. Config: **LABEL TYPE** reveals type-specific fields — **FIXLB** (horizon, threshold), **MEANLB / TRENDLB** (window, threshold), **LEXLB** (window), **BOLB** (window, alpha). Results: label-type/bar cards, a CLASS DISTRIBUTION section, and a DETAILS table of sample labels. **Labels → Signals** adds ENTRY LABEL / EXIT LABEL fields to convert those labels into tradeable signals.



ML Labels — the FIXLB generator (horizon + threshold). This shot demonstrates the morphing UI clearly: the entire strategy/execution/advanced/benchmark stack is hidden, leaving Market Data + the ML Labels config.

CV Splits

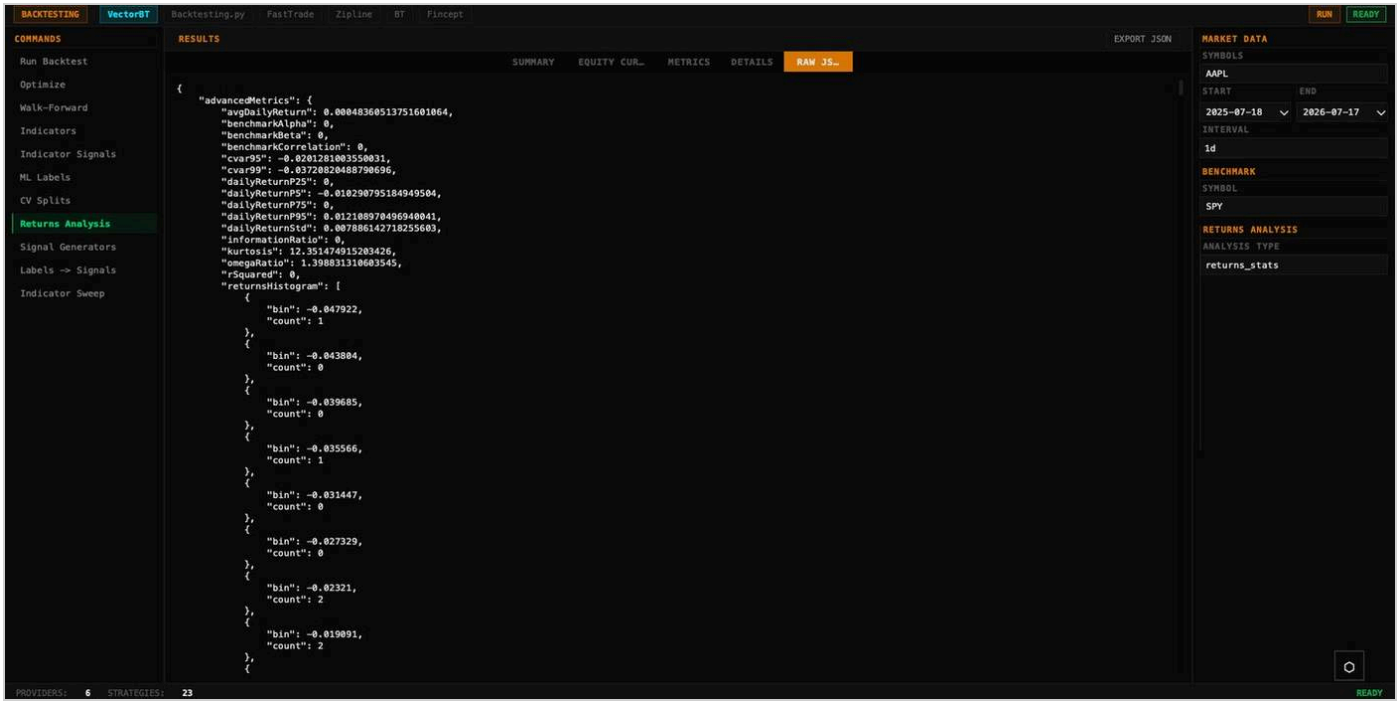
Build cross-validation splitters for ML workflows. Config: **SPLITTER TYPE** reveals fields — **Rolling** (window length, test length, step), **Expanding** (min train length, test length, step), **Purged K-Fold** (n splits, purge length, embargo length). Results: splitter/fold cards and a DETAILS table with one row per fold (train/test start/end & sizes).



CV Splits — the splitter configuration window (splitter type + its fold parameters).

Returns Analysis

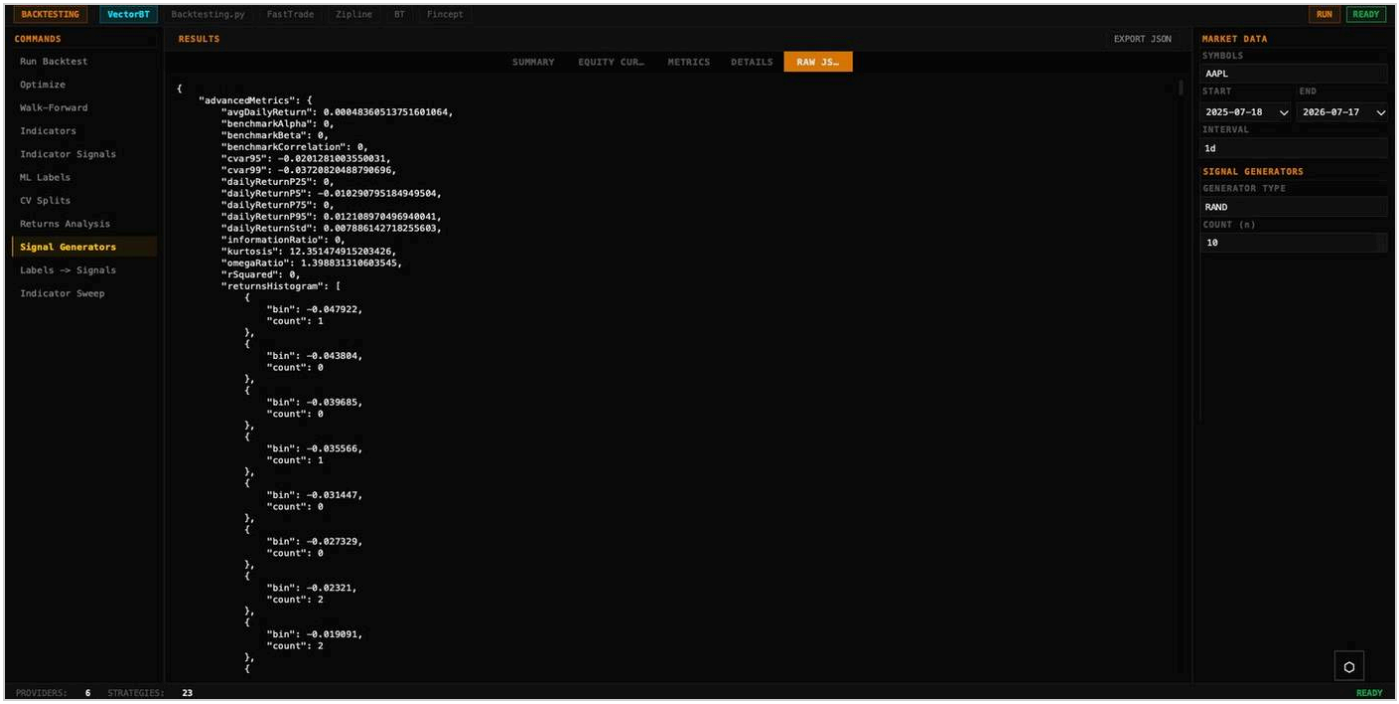
Return-series analytics — the only analysis command that keeps the BENCHMARK section. Config: **ANALYSIS TYPE** (cumulative / rolling / drawdown / distribution / benchmark comparison); **rolling** adds a window + metric. Results: a STATISTICS section (total & annualized return, Sharpe, Sortino, Calmar, max drawdown, annualized volatility, downside risk) or, for drawdown/ranges, drawdown-specific figures.



Returns Analysis — note BENCHMARK is still shown here (it is for this command), unlike the other analysis modes.

Signal Generators & Indicator Sweep

Signal Generators produce random/probabilistic signals for null-hypothesis testing. Config: **GENERATOR TYPE** reveals fields — **RAND/RANDX/RANDNX** (count n; RANDNX adds min/max hold), **RPROB/RPROBX** (entry prob; RPROBX adds exit prob). **Indicator Sweep** (VectorBT) runs one indicator across a MIN/MAX/STEP grid and aggregates stats per value.



Signal Generators — the random-signal generator window (generator type + its probability/count parameters).

At a glance, the analysis commands…

- Turn the tab into a full quant-research bench: compute indicators & signals, engineer ML labels, build CV splits, analyse returns and drawdowns, and benchmark strategies against random signals — all on the same market-data window.

8 Strategy catalog

PER ENGINE

Strategies are provided by the active engine and grouped into categories. The counts and coverage per engine:

Engine	Strategies	Categories
VectorBT	23	Trend, Mean Reversion, Momentum, Breakout, Multi-Indicator, Volume, Portfolio
Backtesting.py	26	Trend, Mean Reversion, Momentum, Breakout, Volume, Multi-Indicator, Risk
Zipline	23	Trend, Mean Reversion, Momentum, Breakout, Multi-Indicator, Other, Pipeline
BT	19	Portfolio, Momentum, Trend, Mean Reversion, Breakout
FastTrade	20	Moving Average, Oscillator, Volatility, Volume, Trend, Combined
Fincept	418+ (registry)	General Strategy (free-form params)

VectorBT catalog (the default engine)

Category	Strategies (key parameters)
Trend	SMA Crossover (fast 10, slow 20) · EMA Crossover · MACD Crossover (12/26/9) · ADX Trend (period 14, thr 25) · Keltner Breakout · Triple MA (5/20/50)
Mean Reversion	Z-Score Mean Reversion (period 20, z 2.0) · Bollinger Bands (20, 2σ) · RSI (14, 30/70) · Stochastic · Williams %R
Momentum	Momentum (lookback 20) · Dual Momentum (20/60) · CCI (20, ±100)
Breakout	Donchian Breakout (20) · Volatility Breakout (20, k 1.5)
Multi-Indicator	RSI + MACD · MACD + ADX · Bollinger + RSI · **Custom Combo** (builder)
Volume	OBV Trend (20)
Portfolio	Buy & Hold · Periodic Rebalance (63d)

Custom Combo builder (VectorBT)

Selecting the **Custom Combo** strategy replaces the parameter spin-boxes with a rule builder: up to three indicator rows, each with a **TYPE** (rsi / sma / ema / macd_hist / bbands_pctb / stochastic / atr), **PERIOD**, **CONDITION** (below / above / cross_up / cross_down) and **THRESHOLD**, joined by an AND/OR **COMBINE LOGIC**. Indicator 3 is behind an "Enable Indicator 3" checkbox. This lets you compose a bespoke entry rule without writing code.

> **Fincept engine footnote.** The Fincept provider exposes a 418+ strategy registry (ids like `FCT-…`) but those strategies take free-form parameters, so no tunable spin-boxes render for them. Also note FastTrade's runtime only wires a handful of its catalog ids; the rest fall back to buy-and-hold.

9 Engine, data & AI agent tools

UNDER THE HOOD

How a run executes

- **Python workers.** Each provider is a Python script run as an out-of-process subprocess; it prints a JSON envelope that the terminal parses. There is no native-C++ or edge compute path for the math.
- **Asynchronous.** Runs never block the UI — the status chip shows EXECUTING… and results arrive via a signal. Backtests can genuinely take from seconds to minutes (agent timeouts are 300 s for runs, 60 s for list calls).
- **Market data.** Yahoo Finance by default (`yfinance`). If yfinance is unavailable the engine falls back to *synthetic* data and the SUMMARY shows an amber warning — treat those results as illustrative only.
- **Live-broker candles (VectorBT).** When VectorBT is selected and a connected Indian broker is available, OHLC is pulled from the broker instead of Yahoo (daily/intraday intervals; weekly/monthly fall back to Yahoo). Symbols are canonicalised (`NSE:RELIANCE.NS → RELIANCE`) across the boundary.
- **Caching & state.** Strategy and option lists cache for 10 minutes per provider. The tab persists your provider, command, symbols, benchmark and last result — reopening restores them. Another tab (e.g. Equity Research) can hand off a symbol + capital and auto-run a backtest.

AI agent tools (MCP)

Four tools in the `backtesting` category let the terminal's AI agent drive this engine:

Tool	What it does
`list_backtest_providers`	List the 6 providers with their supported commands (synchronous).
`list_backtest_strategies`	List a provider's strategies with ids, categories and tunable parameters.
`run_backtest`	Run a strategy backtest — provider, strategy, params, symbols, dates, interval, capital, commission, extras — returns performance, equity curve and trades.
`run_backtest_command`	Generic dispatcher for any command (optimize, walk_forward, indicator, labels, splits, returns, signals, sweep…) with raw args.

10 Quick reference & important notes

HOW DO I...

I want to…	Do this
Run my first backtest	Keep VectorBT + Run Backtest, set SYMBOLS & dates, pick a strategy, press RUN.
Test an Indian stock	Add the exchange suffix — e.g. `RELIANCE.NS` — or it resolves wrong on Yahoo.
Find the best parameters	Optimize command → set each param's MIN/MAX/STEP and an objective → RUN → read BEST PARAMETERS.
Check a strategy isn't overfit	Walk-Forward → set splits & train ratio → read the robustness score & degradation ratio.
See the trades	DETAILS tab after a backtest.
Compare vs a benchmark	Set BENCHMARK · SYMBOL (backtest family) or use Returns Analysis.
Add realistic costs	EXECUTION → commission & slippage; ADVANCED → leverage, stops, position sizing.
Export results	EXPORT JSON (center header) or the RAW JSON tab.
Use a different engine	Click another provider pill — the command & strategy lists update to what it supports.
Build a custom rule	VectorBT → Custom Combo strategy → compose up to 3 indicator conditions with AND/OR.

Important notes for institutional users

- **Synthetic-data warning.** If Yahoo is unreachable, results are computed on synthetic prices and the SUMMARY flags it in amber — do not treat those as real-market results.
- **Symbol suffixes.** Non-US symbols must include `.NS`/`.BO` etc.; US symbols don't.
- **Provider install.** Backtesting.py and FastTrade need their PyPI packages installed; the tab tells you which. VectorBT is the most complete engine and the only one with live-broker candles.
- **The panels morph per command.** If a section (strategy, execution, benchmark) seems "missing", it's because the current command doesn't use it — switch to Run Backtest to see the full set.
- **Runs are asynchronous and can take minutes.** The status chip shows EXECUTING...; the previous results stay on screen until the new ones arrive.
- **State persists.** Your provider, command, symbols, benchmark and last result are restored when you reopen the tab.
- **Costs are percentages.** Commission/slippage/stop/take/risk-free are entered as percents and sent as fractions.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Derivatives

A complete, screen-by-screen walkthrough of the Derivatives tab — a native-C++ options & volatility workbench built around one underlying's live option chain: chain analytics, a vol surface, realized-vs-implied vol, a strategy lab, book risk, dealer positioning, event analytics and multi-asset pricers.

`>_ eight modules · one native engine`

Covers · **the 8-module workbench** · **a native C++ engine (41 models)** · **chain & vol surface** **23 strategies · book risk · dealer GEX** · 16 pricers · 35 agent tools

Demonstrated live on the Fincept Terminal Derivatives tab.

Contents

13 SECTIONS

GETTING ORIENTED

Overview — what the Derivatives tab is 3

1 Anatomy of the workbench (shell, rail + canvas, charts) 4

THE MODULES

2 CHAIN — the option chain 6

3 VOL SURFACE — SVI / SSVI calibration 8

4 VOL ANALYSIS — realized vs implied, VRP, VIX strip 10

5 STRATEGY LAB — 23 templates, payoff & greeks 12

6 RISK — greeks, scenarios & VaR 14

7 POSITIONING — dealer gamma (GEX) 16

8 EVENTS — earnings / event vol & density 18

9 PRICERS — 16 multi-asset calculators 20

REFERENCE

10 Data sources & the seeded demo 22

11 Analytics engine — capability catalog 22

12 AI agent tools (35 MCP tools) 23

13 Quick reference — “How do I…?” & important notes 24

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

What the Derivatives tab is

OVERVIEW

The Derivatives tab is a **workbench centred on a single subject**: one underlying at a time, plus one position on it. You pick an underlying in the header; the terminal loads its option-chain snapshot, and every sub-tab reads that same chain — pricing the chain, fitting a vol surface, measuring realized-vs-implied vol, building and risking multi-leg strategies, mapping dealer gamma, and analysing event/earnings vol. A shared position (your strategy legs) flows from the Strategy Lab to the header greeks badge and the Risk page.

All analytics run on a native C++ engine (the green NATIVE ENGINE badge) — 41 models spanning option pricing & greeks, SVI/SSVI surface calibration, realized vol, VRP & VIX-strip replication, strategy payoff/greeks/probability, book VaR & scenarios, dealer GEX, and event analytics. The math is in-process and deterministic. Chain data is free and public: a seeded demo, plus optional delayed Cboe (US) and Deribit (crypto) feeds — no API key and no credits.

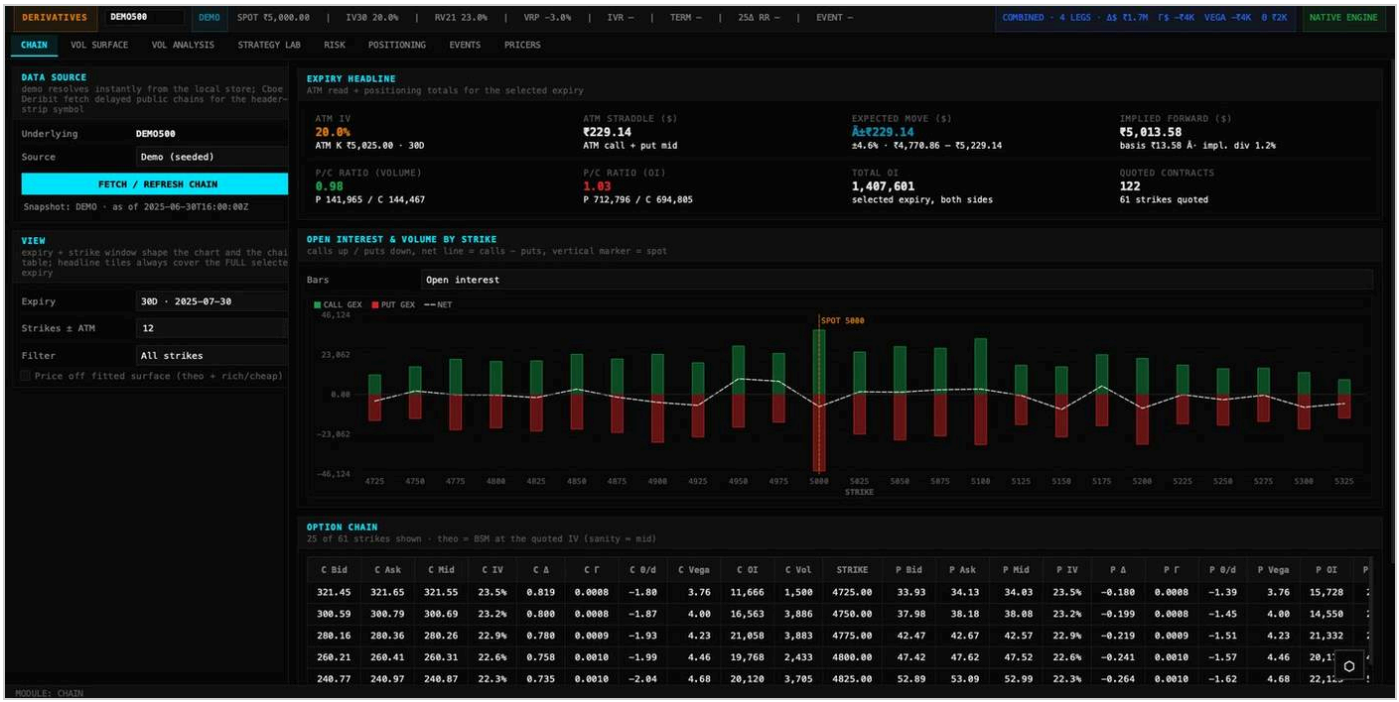
AT A GLANCE, IN DERIVATIVES YOU CAN…

- **Load an option chain** — the seeded DEMO500 index (or DEMOEV, a single stock with earnings in 5 days) with zero setup, or a free delayed Cboe/Deribit chain.
- **Read the chain**: ATM IV, straddle, expected move, put-call-parity forward, P/C ratios, OI, and a full calls/strike/puts greeks table.
- **Fit an implied-vol surface** (raw SVI per expiry or global SSVI) with no-arbitrage checks; read 25Δ risk-reversal, butterfly and skew.
- **Measure realized vol** (5 estimators), the vol cone, the variance-risk premium and a per-name synthetic VIX.
- **Build any of 23 option strategies**, see the payoff curve, greeks, probability of profit and margin, and push the position to Risk.
- **Risk the book**: net dollar greeks, a spot×vol scenario matrix, parametric & Monte-Carlo VaR, stress presets, and a futures hedge.
- **Map dealer positioning**: net GEX, the zero-gamma flip, call-resistance / put-support and pin risk.
- **Analyse events**: implied move, event vol from the term-structure kink, the risk-neutral density, and an IV-crush reprice.
- **Price 16 instruments** — bonds, swaps, CDS, FX, European/American options, Black-76, and exotics (digital / barrier / Asian).

1 Anatomy of the workbench

ORIENTATION

Open the tab from the command bar (type `deriv`) or **Navigate ‣ Trading & Portfolio ‣ Derivatives**. Every sub-tab shares the same header, the same two-pane layout, and the same underlying + position.



The workbench on the CHAIN page. ① Underlying picker ② Source badge ③ Live metrics strip ④ Position / greeks badge ⑤ Sub-tab strip ⑥ Rail (inputs) ⑦ Results canvas.

Region	What it shows / does
① **Underlying picker**	An editable dropdown seeded with the demo symbols (DEMO500, DEMOEV). Pick one — it loads that symbol's chain snapshot and drives every page. (Tip: use the dropdown; the field also accepts a typed symbol.)
② **Source badge**	`DEMO` / `CBOE` / `DERIBIT` — which feed the current snapshot came from, with the as-of time on hover.
③ **Metrics strip**	A live one-line read: `SPOT · IV30 · RV21 · VRP · IVR · TERM · 25Δ RR · EVENT`. Each figure is *published* by the page that computes it (Vol Analysis fills RV21/VRP/IVR, Vol Surface fills TERM/25Δ RR, Events fills the EVENT badge); an em-dash means "not computed yet".
④ **Position badge**	`NO POSITION`, or the active position's mode and net greeks — `<mode> · N LEGS · Δ$ Γ$ VEGA Θ`.
⑤ **Sub-tab strip**	Eight modules, fixed order: **CHAIN · VOL SURFACE · VOL ANALYSIS · STRATEGY LAB · RISK · POSITIONING · EVENTS · PRICERS**.
⑥ **Rail** / ⑦ **Canvas**	Left rail = inputs for that module; right canvas = KPI tiles, charts and tables, with a status strip (⏳ `Running…` / done / error) on top.

The shared underlying & position

Two things are shared by every page. The **underlying context** — the chosen symbol, its chain snapshot (spot, rate, dividend yield, per-expiry strike grids) and an optional fitted vol surface. And the **position (LegSet)** — your option/stock legs, in one of three modes:

- **SIMULATED** — the legs you build in the Strategy Lab.

- **BOOK** — a seeded/imported book (the demo underlying ships a demo book: a short strangle + long calls + stock).
- **COMBINED** — book legs + simulated legs together.

Two behaviours worth knowing. (1) The chain snapshot is the tab's only notion of "now" — every price is repriced off the snapshot's as-of time, never a live tick or wall clock. (2) Fitting a surface on VOL SURFACE unlocks surface-priced features elsewhere (chain rich/cheap, the event density, strategy market-implied probabilities); switching the underlying clears the fit.

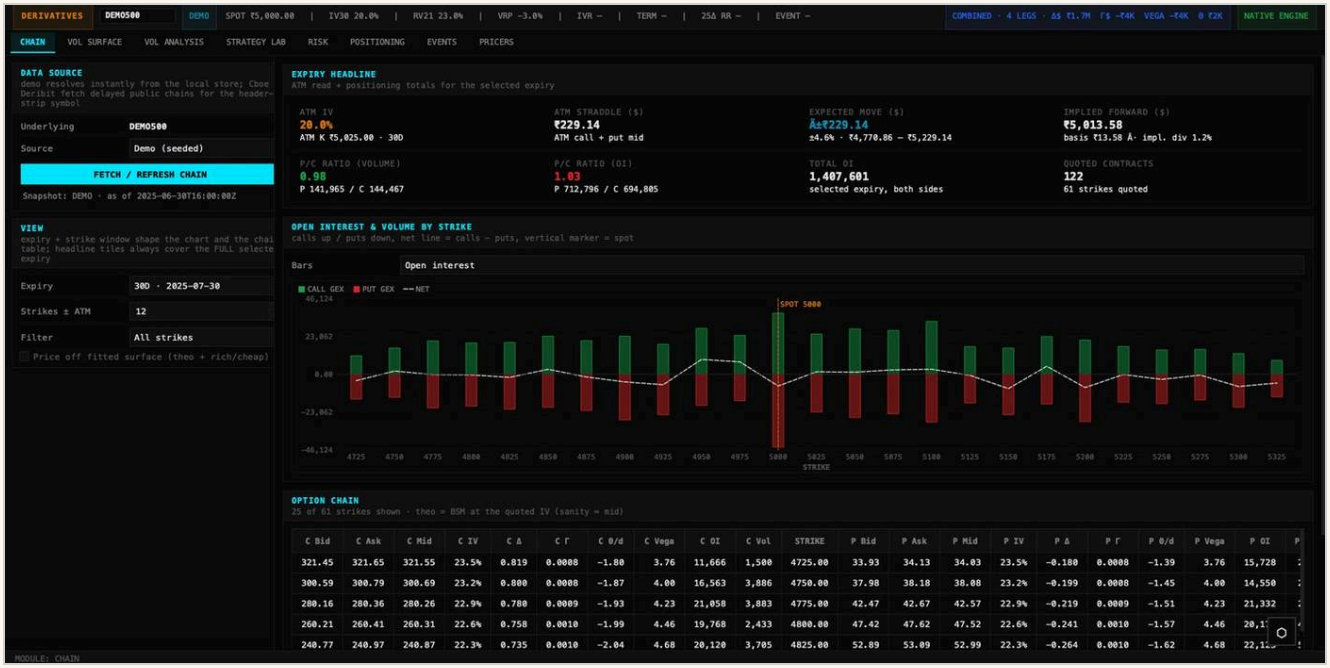
The chart library

- **Smile** — fitted IV curve + market dots per expiry.
- **Surface heatmap** — strike × tenor IV grid.
- **Term structure** — ATM IV vs days-to-expiry (with event kink).
- **Vol cone** — RV percentile bands + current RV + IV.
- **P&L curve** — payoff vs spot (expiry + T+n, breakevens, σ-bands).
- **P&L heatmap** — price × date reval grid.
- **GEX bars** — mirrored call/put gamma by strike + levels.
- **Density** — Breeden-Litzenberger implied PDF vs lognormal.
- **Greeks ladder** — one greek bucketed by expiry.
- **Histogram / heatmap** — VaR P&L, scenario grids.

2 CHAIN — the option chain

AUTO-LOADS · TAB 1

An analytics option-chain viewer (explicitly *not* a trading ticket). Loads a chain snapshot for the header underlying from a chosen source, then renders ATM/positioning tiles, an OI/volume-by-strike map, and the full calls/strike/puts greeks table — all repriced off the snapshot's as-of.



CHAIN — data-source & view rail, the expiry-headline tiles, the OI/volume-by-strike bars with a spot marker, and the full T-chain greeks table.

Rail controls

Control	Type	What it does
Source	combo	**Demo (seeded)** — instant, local. **Cboe (delayed)** — ~15-min US equity/index chains. **Deribit (crypto)** — BTC/ETH options. All free, no key, no credits.
FETCH / REFRESH CHAIN	button	Pulls the chain for the header symbol from the selected source and publishes it to every page.
Expiry	combo	Selects the expiry (e.g. `30D · 2025-07-30`) that shapes the chart & table; the headline tiles always cover the full expiry.
Strikes ± ATM	spin	How many strikes each side of ATM to show (1–500).
Filter	combo	All strikes / Near ATM (±10%) / By delta (|Δ| 0.05–0.60).
Price off fitted surface	checkbox	Reveals the **Theo** and **rich/cheap (R/C)** columns — enabled only after a surface is fit on VOL SURFACE.

Results

- **Expiry Headline** — ATM IV, ATM straddle, expected move, put-call-parity implied forward, P/C ratios (volume & OI), total OI, quoted contracts.
- **OI & Volume by Strike** — calls up / puts down, a net line, and a spot marker (toggle between open interest and volume).

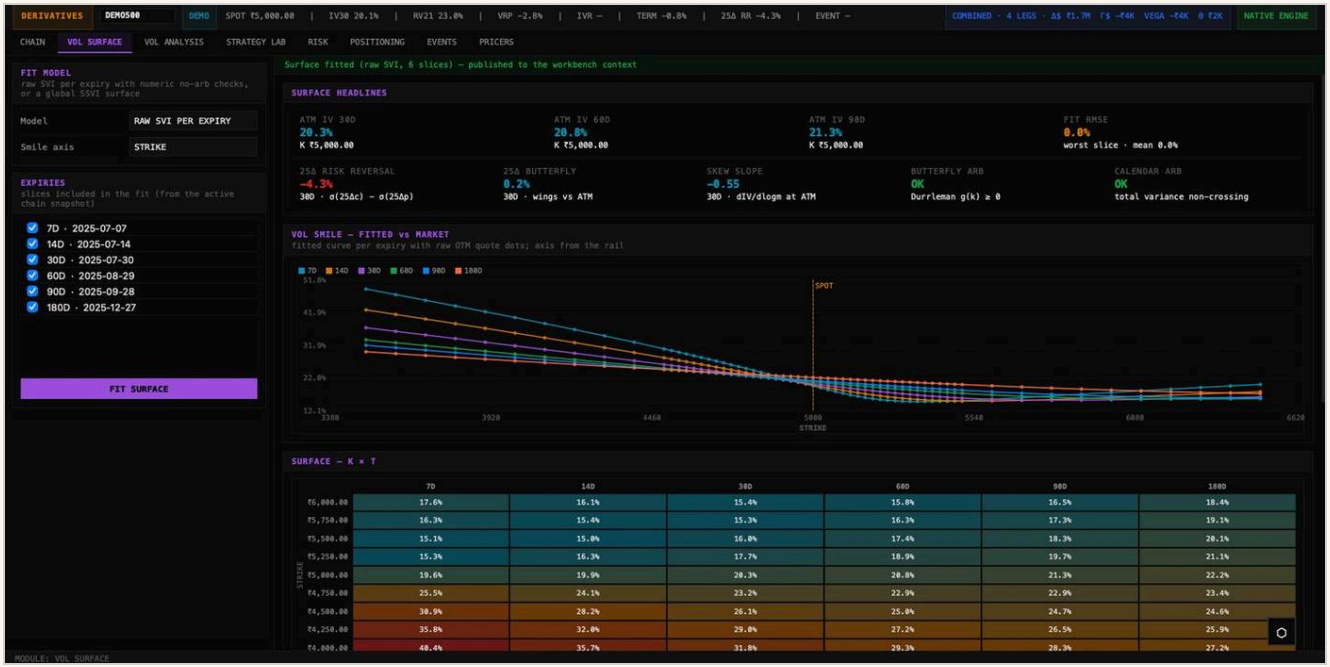
- **Option Chain** — the full 25-column T-chain: C Bid/Ask/Mid/IV/Δ/Γ/Θ/Vega/OI/Vol · STRIKE · P … (read-only; Theo & R/C appear only when priced off a fitted surface).

> **At a glance, in CHAIN you can…**
>
> - Load the seeded DEMO500 index chain instantly, or fetch a free delayed Cboe (US) or Deribit (crypto) chain — no key, no credits.
> - Read ATM IV, straddle, expected move and the implied forward for any expiry; gauge positioning via P/C ratios and OI; narrow the view by strike window or delta; and reveal rich/cheap theo once a surface is fitted.

3 VOL SURFACE — SVI / SSVI

AUTO-FITS · TAB 2

Calibrates the chain to an implied-vol surface — raw SVI per expiry or a global SSVI — with numeric no-arbitrage checks, and publishes the fit to the workbench so every other page can price off it.



VOL SURFACE — fit model & expiry rail, surface-headline tiles (ATM IV / RMSE / 25Δ RR & butterfly / skew / no-arb), the fitted-vs-market smile overlay, and the strike×tenor IV heatmap.

Rail controls

Control	Type	What it does
Model	combo	**Raw SVI per expiry** (independent slice fits) or **SSVI global** (one arbitrage-free surface).
Smile axis	combo	Draw the smile against Strike / Delta / Log-moneyness.
Expiries	checklist	Which chain expiries enter the fit (all checked by default).
FIT SURFACE	button	Recalibrates and re-publishes the surface.

Results

- **Surface Headlines** — ATM IV at 30/60/90D, fit RMSE, 25Δ risk-reversal & butterfly, skew slope, and **butterfly / calendar no-arb** flags (OK / VIOLATION).
- **Vol Smile — Fitted vs Market** — one fitted curve per expiry over the raw OTM quote dots.
- **Surface K×T** — the strike-tenor implied-vol heatmap.
- **ATM Term Structure** and a **Per-Expiry Fit Detail** table (SVI a/b/ρ/m/σ, RMSE, Durrleman g, N points).

> **Publishes to the header.** Fitting the surface fills the `25Δ RR` and `TERM` figures on the metrics strip, and unlocks the chain's rich/cheap columns and the event density on other pages.

At a glance, in VOL SURFACE you can…

- Fit a raw-SVI or global-SSVI surface over a chosen set of expiries with butterfly & calendar no-arb checks.
- Read ATM term IVs, 25Δ risk-reversal / butterfly and skew; view the fitted-vs-market smile, the K×T heatmap, the ATM term curve, and per-slice SVI parameters.

4 VOL ANALYSIS — realized vs implied

AUTO-RUNS · TAB 3

Runs the vol engine over the underlying's daily OHLC history and chain — realized vol (five estimators), the vol cone, the variance-risk premium, IV rank, a rolling IV-vs-RV series, and a per-name synthetic VIX.

VOL ANALYSIS — estimator & window rail, headline tiles (RV21 / IV30 / VRP / IV-rank / synthetic VIX), the realized-vol cone with current-RV dots & ATM-IV diamonds, and the rolling IV-vs-RV series.

Rail controls

Control	Type	What it does
Estimator	combo	Close-close / Parkinson / Garman-Klass / Rogers-Satchell / **Yang-Zhang** — drives the cone & headline (the matrix shows all five).
Windows	checkboxes	Rolling lookbacks: 10 / 21 / 42 / 63 / 126 / 252 days.
History Span	spin	How many trailing daily bars to feed (up to 504).
RUN VOL ANALYSIS	button	Recomputes the pack.

Results

- **Headlines** — RV21 (Yang-Zhang), IV30, **VRP** (IV30−RV21), IV rank/percentile, and a per-name **synthetic VIX 30D** (variance-swap replication off the chain).
- **Vol Cone** — realized-vol percentile bands per window, with current-RV dots and ATM-IV diamonds.
- **IV vs RV — Rolling** time series, the **estimators × windows RV matrix**, and the per-expiry **VIX-strip** table.

> **Honest caveats.** The **VOL INDICES** rail rows (^VIX / ^VIX3M / ^SKEW / ^INDIAVIX) are placeholders — live index quotes are not wired yet. **IV rank / percentile** and the IV line on the rolling chart need a dated IV history that isn't wired (point-in-time IV shows in the tiles). Realized-vol analytics need daily OHLC, which currently exists only for the demo underlyings (DEMO500 / DEMOEV).

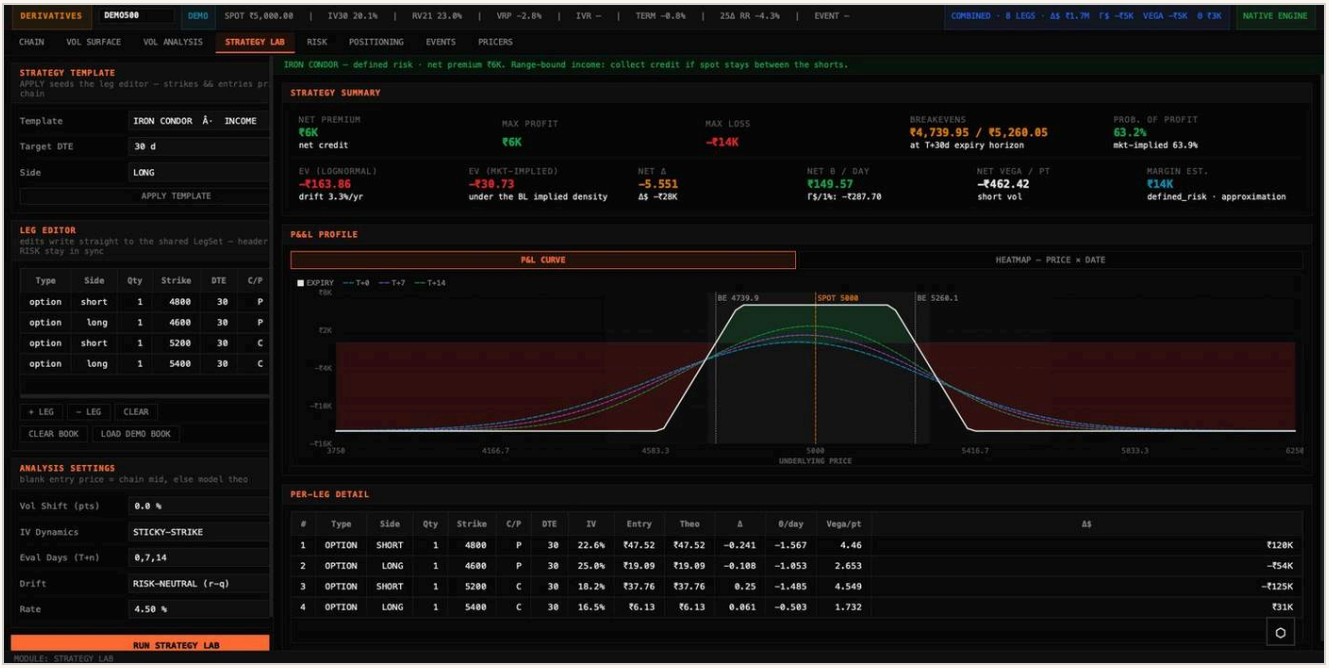
At a glance, in VOL ANALYSIS you can…

- Compute realized vol across five estimators and windows; read RV21, IV30, VRP, IV rank and a synthetic VIX.
- View the realized-vol cone, a rolling IV-vs-RV series, the estimators×windows matrix, and a per-expiry VIX-strip replication.

5 STRATEGY LAB — build & analyse

TAB 4

A multi-leg strategy workbench: pick from 23 prebuilt templates (or hand-edit legs), and it computes payoff, greeks, probability of profit and margin, and draws a P&L-vs-spot curve and a price×date heatmap. The leg editor writes straight through to the shared position, so the header badge and Risk page stay in sync.



STRATEGY LAB with an Iron Condor applied — summary tiles (net credit, max P/L, breakevens, PoP, greeks, margin), the leg editor, the P&L profile (expiry + T+n, breakevens, σ-bands), and the per-leg detail.

Rail controls

Control	What it does
Template · Target DTE · Side · APPLY	Pick a named strategy and one-click seed the legs at a target expiry, long or short — strikes chosen off the fitted smile.
Leg Editor (table + LEG / CLEAR buttons)	Add / remove / edit legs (type, side, qty, strike, DTE, C/P, entry) — edits publish to the shared LegSet.
CLEAR BOOK / LOAD DEMO BOOK	Empty or reseed the demo book (feeds Risk's book/combined mode).
Analysis Settings	Vol shift, sticky-strike vs sticky-delta IV, T+n eval days, drift (risk-neutral / zero), rate.
RUN STRATEGY LAB · ANALYZE AS BOOK → RISK	Recompute; or publish the LegSet to the Risk page.

The 23 strategy templates

Long Call · Long Put · Covered Call · Protective Put · Bull Call Spread · Bear Put Spread · Bull Put Spread · Bear Call Spread · Long Straddle · Long Strangle · Call Butterfly · Broken-Wing Butterfly · Iron Condor · Iron Butterfly · Call Condor · Call Ratio Spread · Calendar Spread · Diagonal Spread · Collar · Risk Reversal · Box Spread · Jade Lizard · ZEBRA.

Results

- **Strategy Summary** — net premium (credit/debit), max profit, max loss (or UNBOUNDED), breakevens, probability of profit (lognormal + market-implied), EV, net Δ/Θ/Vega, and an estimated margin.
- **P&L Profile** — payoff-vs-spot curve (expiry + T+0 + your T+n horizons, breakevens, ±1σ/±2σ bands) or a price×date P&L heatmap.
- **Per-Leg Detail** — each leg's resolved IV, entry, theo and greeks.

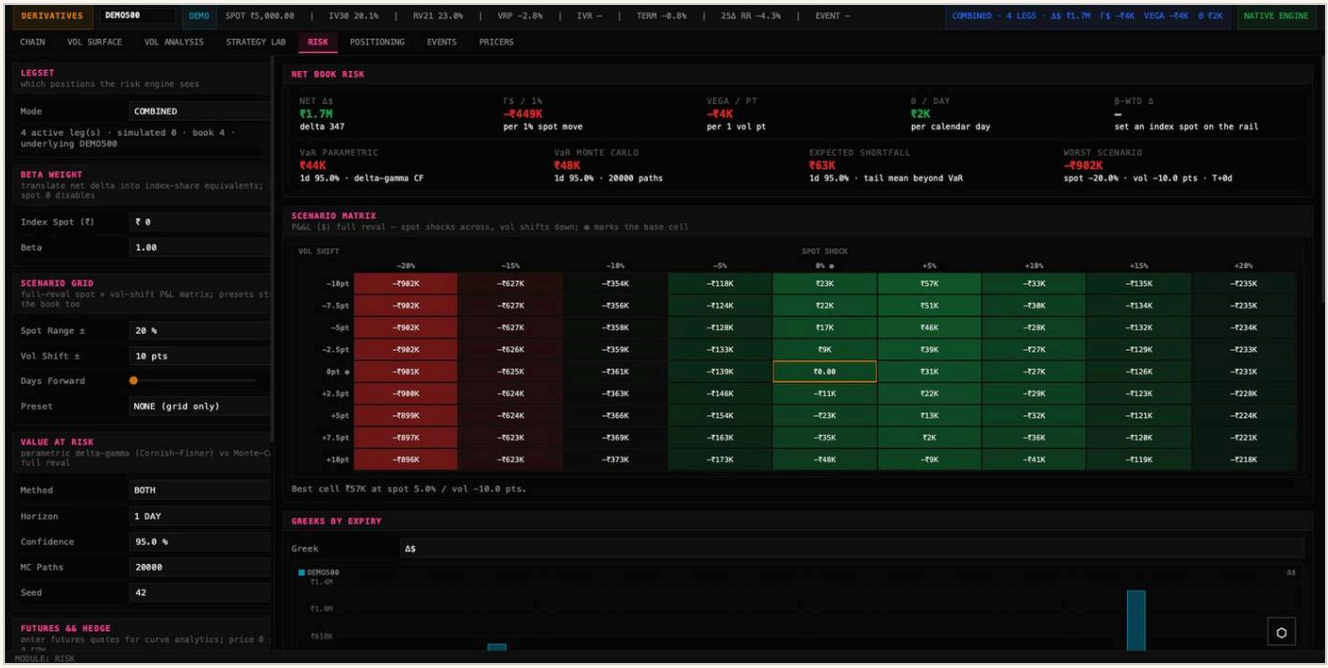
> **Note.** Margin, probability of profit and expected value are explicitly labelled approximations ("broker/exchange margin will differ"). Delta-based strikes and market-implied probabilities are most meaningful with a surface fitted on VOL SURFACE.

> ### At a glance, in STRATEGY LAB you can…
>
> - Auto-build 23 named strategies at any DTE (long or short), or hand-edit a write-through multi-leg book.
> - Read net credit/debit, max P/L, breakevens, PoP, EV, greeks and margin; see the payoff curve or heatmap; and publish the position to Risk.

6 RISK — greeks, scenarios & VaR

NEEDS A POSITION · TAB 5

A book-level risk cockpit over the shared LegSet — net dollar greeks with beta-weighted delta, a full-revaluation spot×vol scenario matrix, parametric and Monte-Carlo VaR/ES, and futures term-structure / hedge analytics.



RISK on the combined demo book — net-book-risk tiles (Δ$ / Γ$ / vega / θ / VaR / ES / worst scenario), the spot×vol P&L scenario heatmap, and greeks bucketed by expiry.

Rail controls

Card	What it does
LegSet — Mode	Point the engine at SIMULATED / BOOK / COMBINED positions (synced with the rest of the tab).
Beta Weight — Index Spot, Beta	Translate net delta into index-share equivalents.
Scenario Grid — Spot Range ±, Vol Shift ±, Days Forward, Preset	Configure the spot×vol P&L matrix (and an optional named stress overlay).
Value at Risk — Method, Horizon, Confidence, MC Paths, Seed	Parametric delta-gamma (Cornish-Fisher) vs Monte-Carlo full reval.
Futures & Hedge	Futures quotes (curve carry, rich/cheap, roll yield) and a minimum-variance hedge — independent of the option legs.
RUN ALL	Recompute everything.

Results

- **Net Book Risk** — NET Δ$, Γ$/1%, vega/pt, Θ/day, beta-weighted delta, parametric & MC VaR, expected shortfall, and the worst scenario cell.
- **Scenario Matrix** — full-reval P&L heatmap (spot shocks × vol shifts), with best/worst cells and stress presets (CRASH / GRIND / EVENT CRUSH / RATES UP / RATES DOWN).

- **Greeks by Expiry** ladder, a **Monte-Carlo P&L histogram**, a **per-position** greeks table, and a **futures curve & hedge** readout.

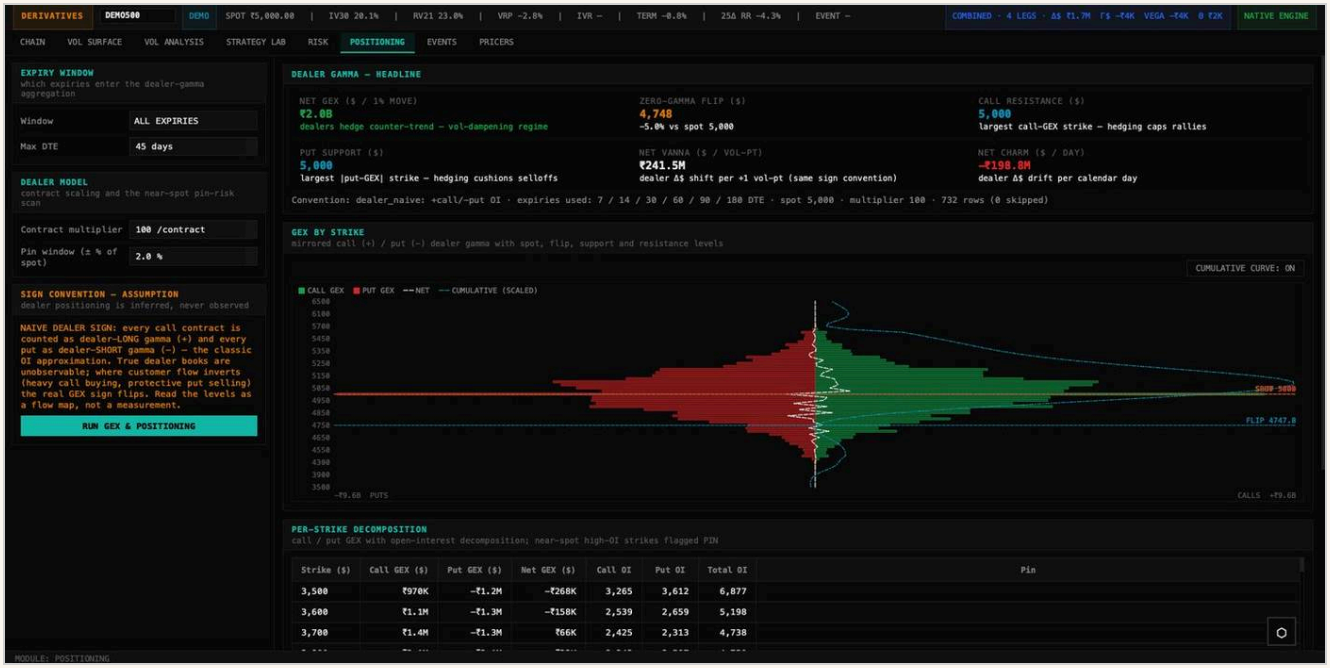
> **Note.** With no active legs (default SIMULATED mode starts empty) the page shows an empty state — switch the LegSet mode to BOOK / COMBINED, or build a strategy in the Strategy Lab. P&L is measured against the current theoretical mark, so the base (0/0) scenario cell is exactly zero.

> ### At a glance, in RISK you can…
>
> - Read net book dollar greeks and beta-weighted delta; run a full-reval spot×vol scenario matrix with stress presets.
> - Compute VaR & expected shortfall two ways at a chosen confidence/horizon; break greeks down by expiry and per position; and analyse a futures curve & hedge.

7 POSITIONING — dealer gamma

AUTO-RUNS · TAB 6

Maps inferred dealer gamma-exposure (GEX) across the chain — net GEX, the zero-gamma flip level, call-resistance / put-support strikes, vanna/charm, a by-strike GEX map, and a near-spot pin-risk scan — under an explicitly-labelled naive dealer sign assumption.



POSITIONING — dealer-gamma headline tiles, and the mirrored call/put GEX-by-strike map with spot, zero-gamma-flip, call-resistance and put-support markers plus a cumulative curve.

Rail controls

Control	What it does
Window / Max DTE	Which expiries enter the dealer-gamma aggregation (all / nearest 1–3 / max-DTE).
Contract multiplier / Pin window	Contract scaling, and the ± % of spot for the pin-risk scan.
RUN GEX & POSITIONING	Recompute from the active chain snapshot.

Results

- **Dealer Gamma Headline** — net GEX ($/1% move) and whether the regime is vol-dampening (positive) or vol-amplifying (negative), the zero-gamma flip, call-resistance / put-support strikes, and net vanna / charm.
- **GEX by Strike** — mirrored call/put gamma bars with spot / flip / support / resistance markers and a cumulative curve.
- **Per-Strike Decomposition** (call/put/net GEX + OI, PIN flags) and a **nearest-expiry Pin-Risk** table.

> **Read as a flow map, not a measurement.** Every call is counted dealer-long gamma and every put dealer-short — the classic OI approximation. True dealer books are unobservable; where customer flow inverts, the real GEX sign flips. The page flags this prominently.

At a glance, in POSITIONING you can…

- Read net dealer GEX and the vol regime, the zero-gamma flip, call-resistance / put-support, and vanna / charm.
- Chart mirrored call/put GEX by strike with key levels, and scan near-spot high-OI strikes for pin risk.

8 EVENTS — earnings & event vol

AUTO-RUNS · TAB 7

Turns the chain (and fitted surface + position) into event analytics — the ATM-straddle implied move, one-day event vol from the term-structure kink, the Breeden-Litzenberger risk-neutral distribution, a prob-ITM/touch ladder, and an IV-crush reprice of the live position.

EVENTS on the DEMOEV earnings demo (5 days out) — implied-move tiles (straddle, event vol 6.7%, days-to-event), the Breeden-Litzenberger density vs lognormal, and the ATM term structure with its event kink.

Rail controls

Control	What it does
Event Expiry / Crush Target / Post-Event IV	Pick the event expiry and the IV the crush snaps front vols to (auto base-IV, or a manual %).
Tenor / Strike Grid / P(S<x)	The density slice, grid resolution, and a probe price for the cumulative probability.
RUN EVENT ANALYSIS	Recompute the pack.

Results

- **Implied Event Move** — straddle & 0.85×straddle moves, **event vol (1D)** from the front-vs-base kink, days-to-event, implied skewness / kurtosis, and P(S<x).
- **Implied Distribution** — the Breeden-Litzenberger risk-neutral PDF vs lognormal (needs a fitted surface), with hover P(S<x).
- **ATM Term Structure — Event Kink** (raw vs ex-event clean curve), a **Prob-ITM / Prob-Touch ladder**, and an **IV-Crush reprice** of the active position.

> **Publishes the event badge.** When the front expiry shows a material vol kink, the header strip flashes `EVENT <n>D / ±<move>`. The Breeden-Litzenberger density needs a surface fitted on VOL SURFACE; everything else runs off the raw chain.

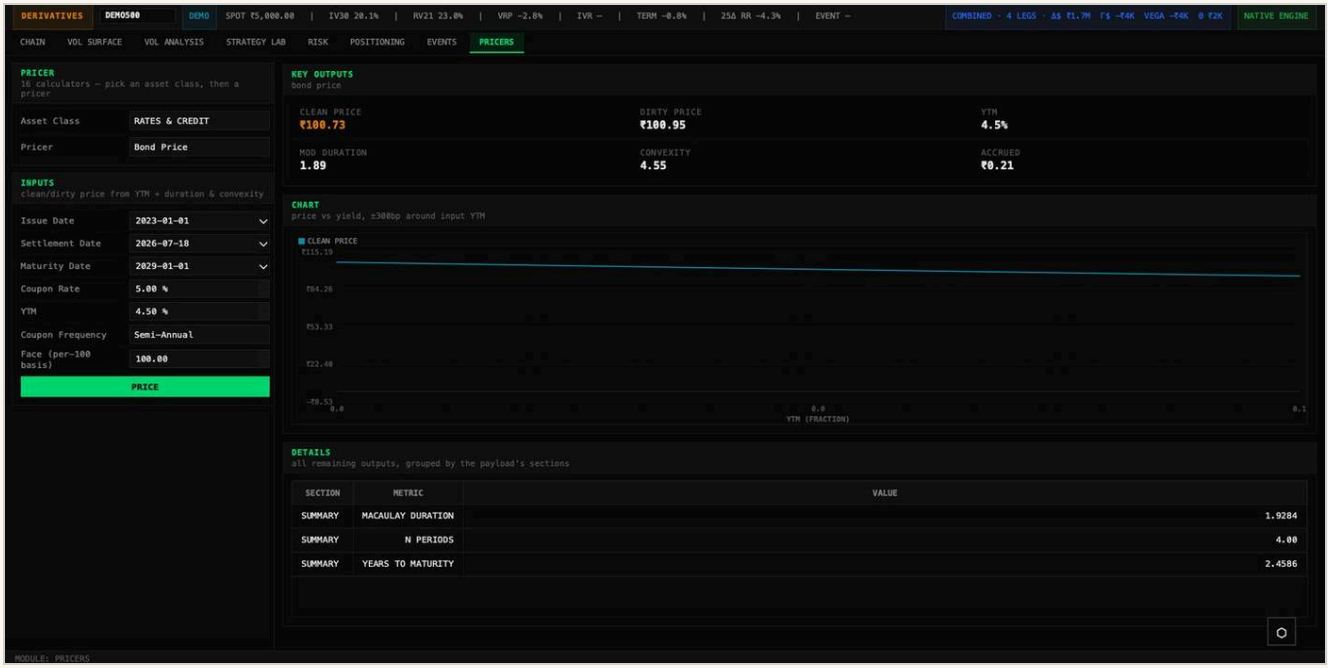
At a glance, in EVENTS you can…

- Read the implied move and extract one-day event vol from the term-structure kink, with a raw-vs-clean ATM curve.
- View the risk-neutral density vs lognormal, scan a prob-ITM/touch ladder, and reprice your position through an IV crush.

9 PRICERS — 16 calculators

STANDALONE · TAB 8

A multi-asset pricing workbench: one spec-table-driven form prices any of 16 standalone "what-if" calculators on the native engine — headline tiles, an optional value/payoff chart, and a full detail table. Nothing here touches your positions; every run is deterministic.



PRICERS — asset-class & pricer pickers, a form that adapts to the instrument, and the key-output tiles + chart + grouped detail table (a bond priced here).

The 16 pricers (by asset class)

Group	Pricers
Rates & Credit (5)	Bond Price (clean/dirty, duration, convexity) · Bond YTM (Brent solve) · Interest-Rate Swap (par rate, DV01) · Credit Default Swap (hazard, survival) · Rate Curve (zero rates, DFs, forwards)
FX (2)	FX Option (Garman-Kohlhagen) · FX Forward (covered-interest-parity)
Equity & Index (6)	European Option (Black-Scholes-Merton) · Full Greeks Ladder (incl. vanna/volga/charm/…) · Implied Volatility (Newton/Brent) · American Option (CRR binomial) · Option on Forward (Black-76) · Forward / Futures Price (cost-of-carry)
Exotics (3)	Digital (cash-/asset-or-nothing) · Barrier (Reiner-Rubinstein knock-in/out) · Asian (geometric & arithmetic average-price)

Each pricer's form adapts to its contract; percent inputs are entered as percents, spot/rate/dividend prefill from the active underlying where sensible, and shared inputs carry across pricers (e.g. the bond definition is reused by the YTM solver). Results are 3–6 headline tiles, a value/payoff/convergence chart where the model provides one, and a grouped detail table of every other output.

> **Deterministic.** All 16 pricers are closed-form or tree-based — no Monte Carlo, no seed on this page. Identical inputs always give identical output, and nothing here reads or writes your positions.

At a glance, in PRICERS you can…

- Price bonds, swaps, CDS and rate curves; FX options & forwards; European/American equity options and the full greeks ladder; Black-76 and forwards; and digital / barrier / Asian exotics.
- Back out implied volatility, read a value/payoff chart, and get a full detail table — all as standalone what-ifs.

10 Data sources & the seeded demo

REFERENCE

Every page is populated from the moment you open the tab, because two deterministic demo underlyings ship with it. The math never needs the network; only live chain quotes do.

The two seeded underlyings (as-of 2025-06-30)

Symbol	What it is	Details
DEMO500	Index-like (the default)	Spot 5000, r 4.5%, q 1.2%, 6 expiries (7/14/30/60/90/180 DTE), strikes 3500–6500. Quotes generated from a known SSVI surface (≈20% ATM IV at 30d) so the SVI fit recovers ground truth. Ships a demo book (short 30d strangle + long 90d calls + 100 shares) and 2 years of synthetic daily OHLC.
DEMOEV	Single stock, earnings in 5 days	Spot 200, 4 expiries (5/33/61/124 DTE), front ATM IV ≈55% vs ≈30% later — a textbook event vol kink for the EVENTS page.

Live data sources (all free, unmetered — no API key, no credits)

Source	Coverage	Endpoint
Demo	The two seeded chains	Local, synchronous — no network.
Cboe (delayed)	US equities & indices, ~15-min delayed	Cboe public delayed-quotes API (index tickers auto-prefixed, e.g. `_SPX`).
Deribit	Crypto (BTC / ETH) options	Deribit public API (coin prices converted to USD).
FRED (rates)	Treasury curve for discounting	Optional key; flat 4.5% fallback when absent.

11 Analytics engine — capability catalog

NATIVE C++

A single in-process C++ engine (41 functions) powers every module — no Python, no server, deterministic. Highlights by area:

Area	Key models
Pricing	Black-Scholes-Merton, Black-76, American (CRR binomial), implied-vol solver, full greeks ladder, digital / barrier / Asian exotics, cost-of-carry forward, put-call-parity forward, batch chain theo.
Vol / surface	SVI & SSVI calibration (with no-arb), surface query, realized vol (5 estimators), vol cone, IV-vs-RV / VRP / IV-rank, 25Δ skew metrics, Cboe VIX-strip replication.
Strategy	Payoff (expiry + T+n, breakevens, σ-bands), probability (PoP / EV / prob-ITM / prob-touch), greeks (net + per-leg + dollar), margin estimate, P&L heatmap, 23 templates.
Risk	Book greeks (+ beta-weighted delta), scenario matrix + stress presets, VaR/ES (parametric Cornish-Fisher + Monte-Carlo), futures curve & min-variance hedge.
Positioning	Dealer GEX (net GEX, zero-gamma flip, call-resistance / put-support, vanna / charm, pin risk).
Events	Implied move (3 methods), event vol from the term-structure kink, IV-crush reprice, Breeden-Litzenberger density.
Rates / FX / credit	Bond price / YTM, IRS value, CDS value, FX option (GK) / forward, rate-curve interpolation.

12 AI agent tools (35 MCP tools)

AGENTIC

The terminal's AI agent can drive this engine through 35 tools in the `derivatives` category — the same store/context the UI uses. Chain/leg tools default to the seeded demo, so the agent analyses what's on your screen. Every tool is pure deterministic compute (none mutate market state).

- **Pricing:** price_equity_option, implied_volatility, deriv_american_option, deriv_exotic_option, deriv_greeks_full, deriv_forward_price, deriv_chain_snapshot
- **Vol/surface:** deriv_svi_fit, deriv_surface_query, deriv_realized_vol, deriv_vol_cone, deriv_iv_rv, deriv_skew_metrics, deriv_vix_strip
- **Strategy:** deriv_strategy_templates, _payoff, _probability, _greeks, deriv_margin_estimate
- **Risk:** deriv_book_greeks, deriv_scenario_matrix, deriv_var, deriv_futures_analytics
- **Positioning / events:** deriv_gex, deriv_expected_move, deriv_event_vol, deriv_iv_crush, deriv_bl_density
- **Rates / FX / credit:** price_bond, bond_yield_to_maturity, value_interest_rate_swap, value_cds, price_fx_option, deriv_fx_forward, deriv_rate_curve

Live chain fetching is intentionally UI-only (agents use stored/demo snapshots via `deriv_chain_snapshot`).

13 Quick reference & important notes

HOW DO I...

I want to…	Go to
Start with no setup	Just open the tab — the seeded **DEMO500** chain is already loaded.
Analyse a real chain	CHAIN ▸ Source = Cboe (US) or Deribit (crypto) ▸ FETCH.
Fit a vol surface	VOL SURFACE ▸ pick a model ▸ FIT SURFACE (unlocks rich/cheap & density elsewhere).
See if options are rich or cheap vs realized	VOL ANALYSIS — read VRP (IV30 – RV21) and the vol cone.
Build a spread / condor	STRATEGY LAB ▸ pick a template ▸ APPLY ▸ read the payoff.
Risk my position	STRATEGY LAB ▸ ANALYZE AS BOOK → RISK, or RISK ▸ Mode = BOOK/COMBINED.
Find dealer gamma levels	POSITIONING — net GEX, zero-gamma flip, support/resistance.
Size an earnings move	EVENTS (try DEMOEV) — implied move, event vol, IV-crush.
Price a bond / swap / exotic	PRICERS ▸ pick asset class + pricer ▸ PRICE.

Important notes for institutional users

- **Local & private.** The analytics engine is native C++ and runs entirely on your device; only live chain quotes (Cboe / Deribit / FRED) make a network call, and those are free public feeds — no API key, no credits.
- **Snapshot time, not wall clock.** Every price is repriced off the chain snapshot's as-of; there is no live tick. Monte-Carlo (VaR) uses a fixed seed, so results reproduce exactly.
- **Surface-gated features.** Chain rich/cheap, the event density, and market-implied strategy probabilities need a surface fitted on VOL SURFACE; switching the underlying clears the fit.
- **Known not-yet-wired items.** On VOL ANALYSIS, the ^VIX/^VIX3M/^SKEW/^INDIAVIX rows are placeholders and IV-rank needs a dated IV history; realized-vol analytics need daily OHLC, which currently exists only for the demo underlyings.
- **Dealer GEX is inferred.** POSITIONING uses the naive +call / −put sign convention — read it as a flow map, not a measurement of real dealer books.
- **Estimates are labelled.** Strategy margin, probability of profit and EV are approximations; broker/exchange margin will differ.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Alternative Investments

A complete, screen-by-screen walkthrough of the Alt Investments tab — a native-C++ LP-program workbench covering commitments, performance & PME, pacing, liquidity, exposure, stress, deal & instrument labs, and a live proxy market.

`>_ nine modules · one native engine`

Covers · **the 9-module workbench** · **a native C++ engine (55 models)** · **performance & PME** **pacing · liquidity · exposure · stress** · deal & instrument labs · 30 agent tools

Demonstrated live on the Fincept Terminal Alt Investments tab.

Contents

14 SECTIONS

GETTING ORIENTED

Overview — what the Alt Investments tab is 3

1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 4

THE MODULES

2 PROGRAM — set up & edit the LP program 6

3 PERFORMANCE — IRR / TVPI / PME & J-curves 8

4 PACING — Takahashi-Alexander projection & commitment solver 10

5 LIQUIDITY — calls vs distributions, coverage & redemption ladder 12

6 EXPOSURE — concentration, treemap, matrix & de-smoothing 14

7 SCENARIOS — historical & custom stress, denominator effect 16

8 DEAL LAB — waterfalls, fees, secondaries, RE/infra underwriting 18

9 INSTRUMENT LAB — 24 instrument calculators 20

10 MARKET — live proxies & embedded benchmarks 22

REFERENCE

11 The seeded demo program (reference numbers) 24

12 Analytics engine — capability catalog 25

13 AI agent tools (30 MCP tools) 25

14 Quick reference — “How do I…?” & important notes 26

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

What the Alt Investments tab is

OVERVIEW

The Alt Investments tab is a **workbench, not a data viewer**. Everything on it revolves around a single subject: your **limited-partner (LP) alternatives program** — a book of closed-end private-market fund commitments (buyout, venture, credit, real estate, infrastructure, secondaries) plus open-ended hedge-fund / liquid-alt / digital sleeves, framed by a top-level portfolio policy. You set the program up once on the PROGRAM desk; every other sub-tab reads that same program and answers a different institutional question about it.

All analytics run on a native C++ engine (the green NATIVE ENGINE badge) — 55 models spanning performance/PME, Takahashi-Alexander pacing, liquidity coverage, look-through exposure, historical stress, distribution waterfalls, secondary pricing, property underwriting and 24 instrument calculators. There is no Python and no server round-trip for the math: results are computed in-process and deterministically, the one exception being the MARKET tab, which pulls live listed-proxy quotes. The same program can also be driven by the terminal's AI agent through 30 dedicated tools.

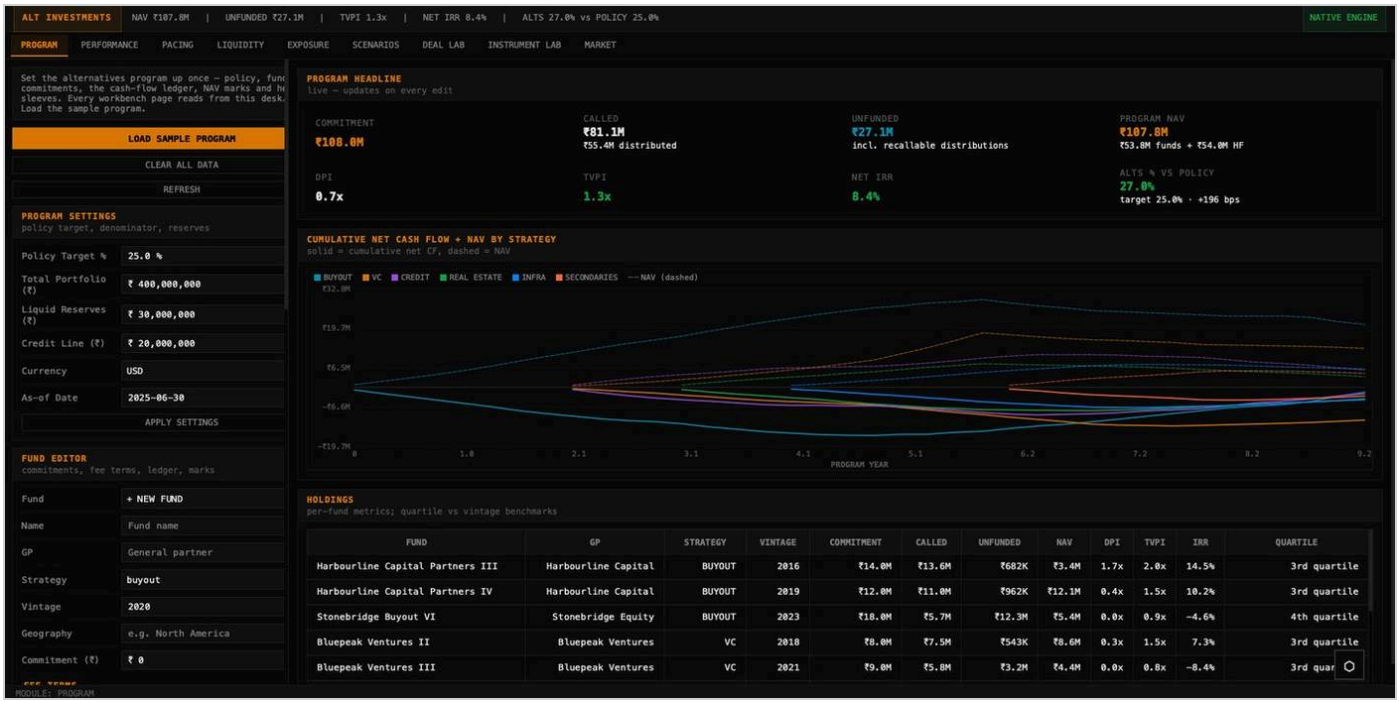
AT A GLANCE, IN ALT INVESTMENTS YOU CAN...

- **Build an LP program** — fund commitments, fee terms, the cash-flow ledger, NAV marks and hedge-fund sleeves — or load a ready-made ~$108M / 10-fund demo in one click.
- **Measure performance**: pooled net IRR, TVPI/DPI/RVPI, KS-PME & Direct Alpha vs a benchmark, horizon IRRs, per-fund J-curves and vintage-quartile placement.
- **Plan commitments** with the Takahashi-Alexander model — project the program forward and solve the steady annual commitment to hit a target allocation.
- **Manage liquidity** — projected calls vs distributions vs reserves, coverage ratios, HF redemption ladders, with an optional stress overlay.
- **Analyse exposure & concentration** — HHI / effective-N, top-GP share, a strategy×fund treemap and a full look-through matrix; de-smooth hedge-fund risk.
- **Stress the program** — 2001 / 2008 / 2022 shock sets or a custom per-sleeve editor — and see the “denominator effect” on your alts weight.
- **Model deals** — distribution waterfalls (European & American + clawback), fee drag, secondary pricing, and RE/infra underwriting with a promote.
- **Price 24 alternative instruments** (bonds, annuities, structured products, real assets, digital) — each a standalone what-if.
- **Watch a live alternatives proxy market** plus curated HFRI / secondary / NCREIF / PE-vintage benchmarks.

1 Anatomy of the workbench

ORIENTATION

Open the tab from the command bar (type `alt` → **Alt Investments**) or **Navigate ▸ Research & Intelligence ▸ Alt. Investments**. Every sub-tab shares the same three-part chrome and the same two-pane page layout.



The workbench on the PROGRAM desk. ① Program bar ② Sub-tab strip ③ Left "assumptions" rail ④ Right results canvas ⑤ Module status bar.

Region	What it shows / does
① **Program bar** (top)	The amber **ALT INVESTMENTS** brand, a live one-line program summary — `NAV · UNFUNDED · TVPI · NET IRR · ALTS% vs POLICY%` — that updates on every edit, and a green **NATIVE ENGINE** badge confirming the analytics run locally. Before any program exists it reads "No program — set up commitments on the PROGRAM desk".
② **Sub-tab strip**	Nine modules in a fixed order: **PROGRAM · PERFORMANCE · PACING · LIQUIDITY · EXPOSURE · SCENARIOS · DEAL LAB · INSTRUMENT LAB · MARKET**. Each has its own accent colour; the active one is highlighted.
③ **Assumptions rail** (left)	A scrollable stack of input cards ("SectionCard"s) — the knobs for that module. Width scales with your Appearance font size. This is where you type; nothing here leaves your device.
④ **Results canvas** (right)	KPI tiles, charts and tables. A thin status strip at the top shows ⏳ `Running…` (cyan) while a model runs, a green confirmation when done, or a red error line. Because the engine is synchronous, the busy state only flashes.
⑤ **Status bar** (bottom)	Shows `MODULE: <name>` for the active sub-tab.

The two-pane pattern & when results appear

Every module is a left **rail** (inputs) + right **canvas** (outputs). Three modules recompute automatically and are populated the moment you open them — **PROGRAM, PERFORMANCE, LIQUIDITY, EXPOSURE** and **MARKET**. The other four wait for you to press a run button — **PACING** (PROJECT / SOLVE), **SCENARIOS** (RUN SCENARIO), **DEAL LAB** (RUN / PRICE / UNDERWRITE) and **INSTRUMENT LAB** (ANALYZE) — and show a "Run…" prompt until you do.

Two behaviours worth knowing. (1) On the analytic modules, editing a rail input does **not** auto-refresh the results — press the module's run button (or change the program) to recompute. (2) Money is shown in your terminal's display currency (₹ in these screenshots); the program's own currency field (USD in the demo) governs the underlying figures.

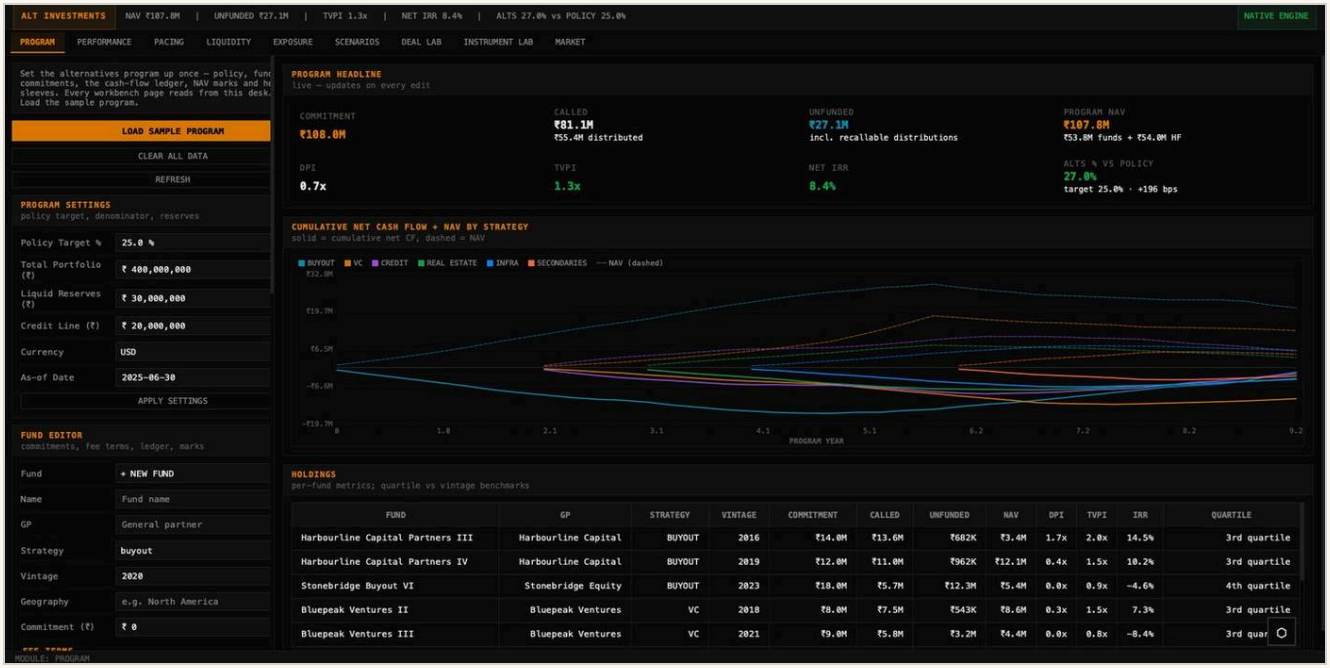
Shared building blocks

- **KPI tiles** — caption + large value + sub-line, colour-toned (green good / red bad / amber accent / cyan info).
- **Section cards** — titled input groups on the rail, each with a one-line hint.
- **Money / percent / number spin-boxes** — money shows a currency prefix + thousands separators; percent shows a "%" suffix.
- **Read-only result tables** — sortable, alternating rows; never editable.
- **Charts** — J-curve, cash-flow fan, stacked-flow, allocation treemap, liquidity ladder, tier-split, heatmap, waterfall, histogram, football-field.
- **Run buttons** — solid accent-filled; ghost buttons are secondary actions.

2 PROGRAM — set up & edit the program

DESK · TAB 1

The set-up-once desk. Enter the policy/denominator, fund commitments with fee terms, the cash-flow ledger, NAV marks and hedge-fund sleeves. Everything you save here flows to all eight other modules and to the AI agent.



PROGRAM desk — rail editor (left), eight live headline KPIs, program J-curve by strategy, and the per-fund holdings table.

Rail — program-level actions

Control	Type	What it does
LOAD SAMPLE PROGRAM	button	Installs the bundled deterministic demo (~$108M, 10 funds + 3 HF sleeves). **Destructive, no confirmation** — it replaces the current program.
CLEAR ALL DATA	button	Removes every fund, flow, mark and HF position. **Confirmation dialog**; the emptied program is persisted (the sample will not return on restart).
REFRESH	button	Re-runs the roll-up and per-fund analytics without changing any data.

PROGRAM SETTINGS card — policy & denominator

Field	Type	Meaning
Policy Target %	percent	Target alts allocation (demo 25%). Drives the ALTS%-vs-policy check.
Total Portfolio ($)	money	Whole-portfolio denominator (demo $400M). Alts% = program NAV ÷ this.
Liquid Reserves ($)	money	Cash earmarked for calls (demo $30M) — used by the liquidity forecast.
Credit Line ($)	money	Available credit facility (demo $20M).
Currency	combo	USD/EUR/GBP/CHF/JPY/INR/AUD/CAD/SGD — sets money formatting.
As-of Date	text	Program valuation date (YYYY-MM-DD; demo 2025-06-30). Flows after this date are ignored by the roll-up. Invalid dates are silently skipped.
APPLY SETTINGS	button	Writes all six fields and recomputes.

FUND EDITOR card — commitments, fee terms, ledger, marks

Control	Type	What it does
Fund	combo	`+ NEW FUND` or an existing fund. Selecting one loads its fields for editing.
Name / GP / Geography	text	Identifiers. Name is required to save.
Strategy	combo	buyout / vc / growth / credit / real_estate / infra / secondaries — used for chart grouping and quartile mapping.
Vintage	int	Vintage year.
Commitment ($)	money	Committed capital → drives COMMITMENT / unfunded.
Mgmt Fee / Carry / Hurdle	percent	Fee terms.
Waterfall	combo	european / american.
Step-down Year	int	Year the fee basis steps from committed to invested.
SAVE FUND / REMOVE FUND	button	Create/merge-update, or delete the selected fund (with its ledger & marks).
ADD CASH FLOW	date + combo + money	Append a **call / distribution / recallable_distribution** to a saved fund (amount positive; the type carries direction). Add-only — flows can't be edited/deleted in the UI.
ADD NAV MARK	date + money	Record a NAV at a date (one per date; a same-date mark is replaced).

HEDGE FUND / LIQUID ALTS card

Control	Type	What it does
Position	combo	`+ NEW POSITION` or an existing sleeve.
Name / Strategy / Invested ($) / Inception	mixed	Strategy ∈ macro / equity_ls / event / relative_value / credit / digital / other. Value = invested × Π(1+monthly return).
Monthly Returns	multi-line	Paste month-end returns, one per line (`YYYY-MM-DD, 0.0042` or `0.42%`). Bad lines are skipped.
Redemption / Notice / Gate / Lockup	mixed	Liquidity terms — daily…annual, notice days, gate %/window, lockup date. Feed the liquidity ladder.
SAVE / REMOVE POSITION	button	Create/merge-update or delete the sleeve.

Results canvas

- **PROGRAM HEADLINE** — eight live tiles: COMMITMENT, CALLED, UNFUNDED, PROGRAM NAV, DPI, TVPI, NET IRR, and **ALTS% vs POLICY** (flags red when the deviation exceeds ±300 bps).
- **Cumulative Net Cash Flow + NAV by Strategy** — a program J-curve: solid = cumulative net CF, dashed = NAV, one series per strategy.
- **HOLDINGS table** — per fund: GP, strategy, vintage, commitment, called, unfunded, NAV, DPI, TVPI, IRR and vintage-quartile placement.

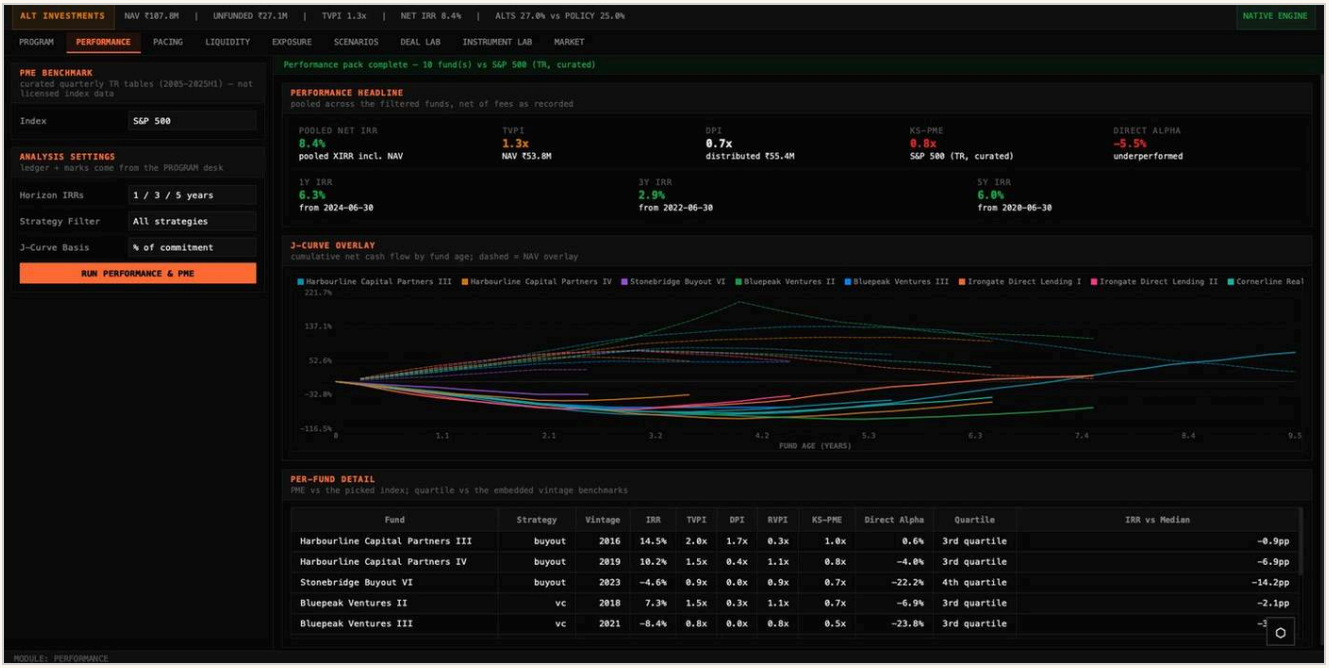
At a glance, in PROGRAM you can…

- Load a demo program or wipe everything; add/edit/remove funds with full fee terms; append calls, distributions and NAV marks; add hedge-fund sleeves from a pasted return series.
- Set policy target, portfolio denominator, reserves, credit line, currency and as-of date; watch eight KPIs and the program J-curve recompute on every edit.

3 PERFORMANCE — IRR / TVPI / PME

AUTO-RUNS · TAB 2

Runs the performance-and-PME engine over the program's fund ledger — pooled headline metrics, a multi-fund J-curve overlay, and a per-fund table with public-market-equivalent and quartile placement. Populated automatically on open and on any program change.



PERFORMANCE — pooled headline tiles (IRR/TVPI/DPI/KS-PME/Direct Alpha + horizon IRRs), the per-fund J-curve overlay with dashed NAV, and the per-fund detail table.

Rail controls

Control	Type	What it does
Index (PME Benchmark)	combo	S&P 500 / Russell 2000 / MSCI World — the public series the KS-PME & Direct Alpha are computed against. *Curated quarterly tables, not licensed index data.*
Horizon IRRs	combo	1/3/5, 3/5/10 or 1/5/10 years — sets the three horizon-IRR tiles.
Strategy Filter	combo	All, or a single strategy — scopes every output (tiles, J-curve, table).
J-Curve Basis	combo	% of commitment / net cash flow ($) / TVPI — sets the J-curve y-axis.
RUN PERFORMANCE & PME	button	Recomputes the full pack.

Results

- **Performance Headline** — POOLED NET IRR, TVPI, DPI, **KS-PME** (green ≥1.0), **Direct Alpha** (out/under-performed), plus three horizon-IRR tiles.
- **J-Curve Overlay** — one cumulative-net-CF curve per fund vs fund age, with a dashed NAV overlay; the emphasised zero line is the classic J-curve trough.
- **Per-Fund Detail** — Fund · Strategy · Vintage · IRR · TVPI · DPI · RVPI · KS-PME · Direct Alpha · Quartile · IRR vs Median.

Definitions. IRR is a dated-cash-flow XIRR with terminal NAV as a pseudo-distribution. **KS-PME** (Kaplan-Schoar) > 1.0 means the fund beat the public index dollar-for-dollar; **Direct Alpha** annualises that out-/under-performance. Quartiles use embedded PE/VC vintage tables (vintages clamped to 2005–2023).

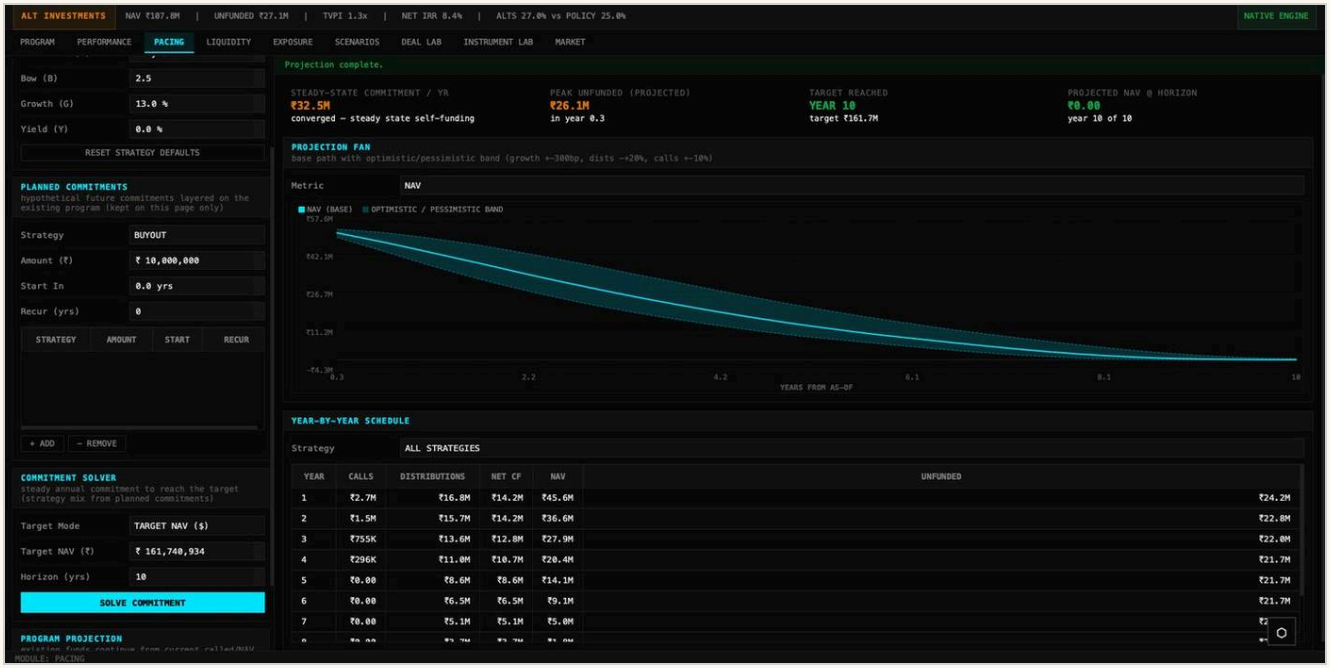
At a glance, in PERFORMANCE you can…

- Read pooled net IRR, TVPI, DPI, KS-PME and Direct Alpha for the whole (or strategy-filtered) program in one row, plus three horizon IRRs.
- Benchmark PME against S&P 500 / Russell 2000 / MSCI World; compare per-fund J-curves; drill into a per-fund table with quartile and IRR-vs-median.

4 PACING — projection & commitment solver

RUN TO POPULATE · TAB 3

Projects the whole program forward with the Takahashi-Alexander (Yale) model — existing funds continue mid-life plus hypothetical planned commitments — and solves for the steady annual commitment needed to hit a target NAV or a target % of portfolio.



PACING after PROJECT + SOLVE — steady-state commitment, peak unfunded, target-reached and horizon-NAV tiles; the optimistic/pessimistic projection fan; and the year-by-year schedule.

Rail cards

Card / control	What it does
TA Parameters — Strategy, Fund Life (L), Bow (B), Growth (G), Yield (Y), RESET	Per-strategy TA assumptions; each strategy keeps its own edits. Only strategies you change are sent as overrides; the engine fills the rest. Page-local (not saved).
Planned Commitments — Strategy, Amount, Start In, Recur, + ADD / – REMOVE	Layer hypothetical future commitments (one-shot or recurring) on the live program. Kept on this page only.
Commitment Solver — Target Mode (NAV$ / % of portfolio), Target NAV / Target% / Portfolio / Growth, Horizon, SOLVE COMMITMENT	Bisection solver for the steady annual commitment to reach the target. Fills the STEADY-STATE and TARGET-REACHED tiles.
Program Projection — Horizon, PROJECT PROGRAM	Projects existing funds (from current called/NAV) + planned commitments. Fills the PEAK-UNFUNDED and HORIZON-NAV tiles, the fan and the schedule.

Results

- **Four tiles** — STEADY-STATE COMMITMENT/YR & TARGET REACHED (from SOLVE); PEAK UNFUNDED (PROJECTED) & PROJECTED NAV @ HORIZON (from PROJECT).

- **Projection Fan** — base path + optimistic/pessimistic band (engine-fixed: growth ±300 bp, dists ∓20%, calls ±10%). Metric selector: NAV / calls / distributions / net CF.
- **Year-by-Year Schedule** — YEAR · CALLS · DISTRIBUTIONS · NET CF · NAV · UNFUNDED, filterable by strategy.

> **Note.** SOLVE fills tiles 1&3; PROJECT fills tiles 2&4 — running only one leaves the other two blank. Planned commitments and TA-parameter edits are page-local and are rebuilt each session by design.

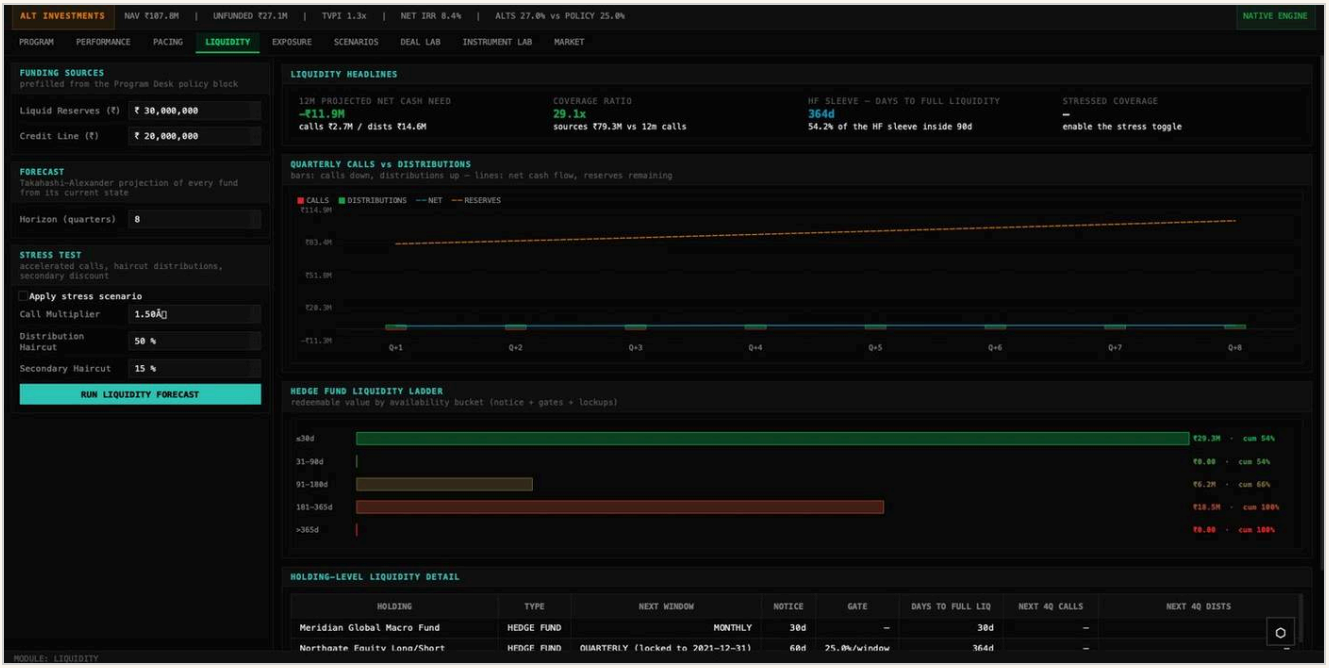
At a glance, in PACING you can…

- Edit TA assumptions per strategy; add hypothetical commitments; project the program 1–30 years and read peak unfunded and horizon NAV with an optimistic/pessimistic band.
- Solve the steady annual commitment to reach a target NAV or % of a growing portfolio, and see whether/when the target is reached.

5 LIQUIDITY — coverage & redemption ladder

AUTO-RUNS · TAB 4

Projects the program's near-term capital calls vs distributions against its funding sources (reserves + credit line + 90-day hedge-fund liquidity), with an optional stress overlay, plus the HF redemption-ladder timing and a holding-level detail.



LIQUIDITY — four headline tiles, the quarterly calls-vs-distributions timeline (calls down / dists up, with net & reserves lines), the HF liquidity ladder by availability bucket, and the holding-level table.

Rail controls

Control	Type	What it does
Liquid Reserves / Credit Line ($)	money	Funding sources (pre-filled from PROGRAM policy; your edits stick).
Horizon (quarters)	int	Length of the projected timeline (default 8).
Apply stress scenario	checkbox	Enables the three stress spins & a stressed timeline / coverage.
Call Multiplier / Distribution Haircut / Secondary Haircut	num / percent	Accelerate calls, haircut distributions, discount a secondary sale.
RUN LIQUIDITY FORECAST	button	Recomputes the full analysis.

Results

- **Headlines** — 12-month net cash need; coverage ratio (sources ÷ 12m calls); HF days-to-full-liquidity; stressed coverage (when the toggle is on).
- **Quarterly Calls vs Distributions** — calls as downward bars, distributions upward, with net-CF and reserves-remaining lines; a shaded "stressed" overlay when stressed.
- **HF Liquidity Ladder** — redeemable value by ≤30d / 31–90d / 91–180d / 181–365d / >365d bucket, with cumulative %.
- **Holding-Level Detail** — per holding: redemption window, notice, gate, days-to-full-liquidity, and each fund's next-4-quarter calls/distributions.

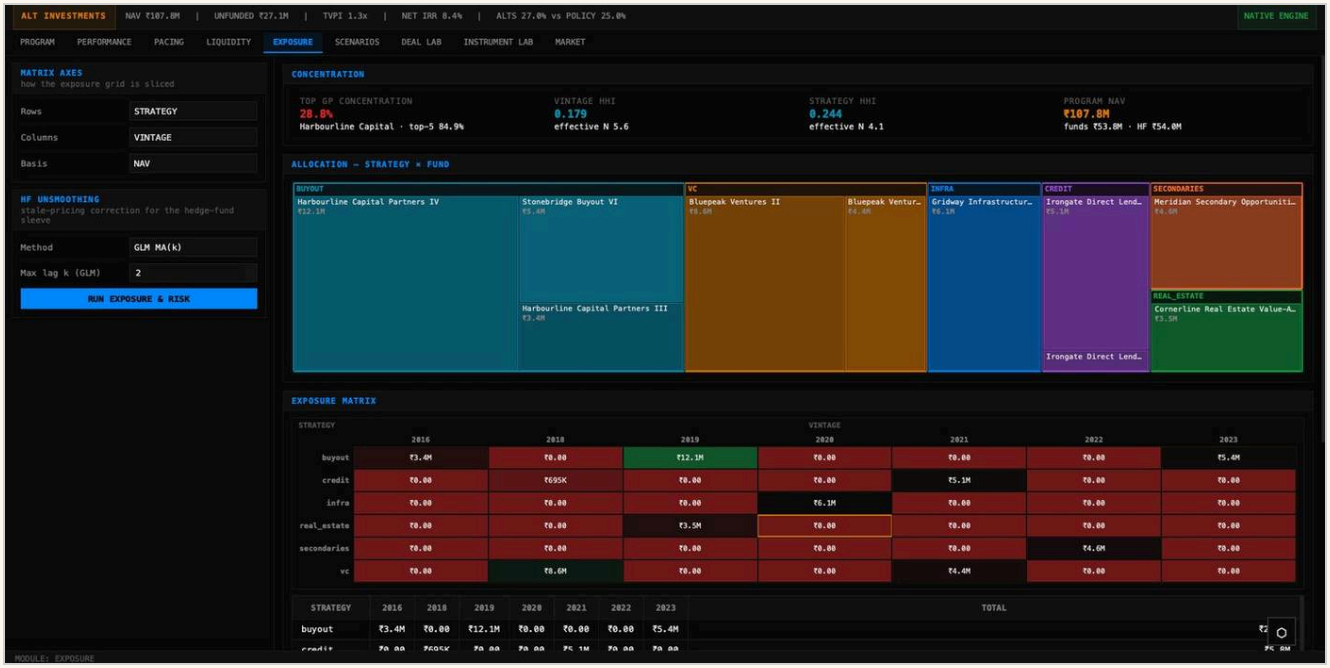
At a glance, in LIQUIDITY you can…

- See the 12-month net cash need, coverage ratio and HF days-to-liquidity; edit reserves, credit line and horizon.
- Toggle a stress scenario and read stressed coverage; review the quarter-by-quarter timeline, the redemption ladder and a holding-level liquidity table.

6 EXPOSURE — concentration & matrix

AUTO-RUNS · TAB 5

Slices program exposure across configurable axes and value bases, headlines the concentration (top-GP share, vintage/strategy HHI & effective-N, program NAV), and compares reported vs de-smoothed hedge-fund risk.



EXPOSURE — concentration tiles, the strategy→fund allocation treemap, and the full exposure heatmap + totals table.

Rail controls

Control	Type	What it does
Rows	combo	Grid row axis: strategy / GP / geography.
Columns	combo	Grid column axis: vintage / strategy.
Basis	combo	Value each fund contributes: NAV / NAV+unfunded / commitment.
HF Unsmoothing — Method / Max lag k	combo / int	GLM MA(k) or Geltner AR(1) de-smoothing algorithm for the HF risk table.
RUN EXPOSURE & RISK	button	Recomputes the page.

Results

- **Concentration** — top-GP share (flagged red ≥25%), vintage & strategy HHI with effective-N, and program NAV split funds vs HF.
- **Allocation Treemap** — two-level strategy→fund map, cell area ∝ NAV, with hover tooltips.
- **Exposure Matrix** — the rows×columns grid as a heatmap plus a totals table (row totals, column totals, grand total).
- **HF Smoothed vs Unsmoothed** — per position: observed vs de-smoothed volatility, Sharpe, and lag-1 autocorrelation, with a theta / smoothing-index note.

Honest caveat. The optional "nowcast Δ (neutral proxies)" line under PROGRAM NAV is **not functional** — it is invoked without proxy returns and errors internally, so that adjustment never renders. Concentration, treemap, matrix and the de-smoothing table are all fully live.

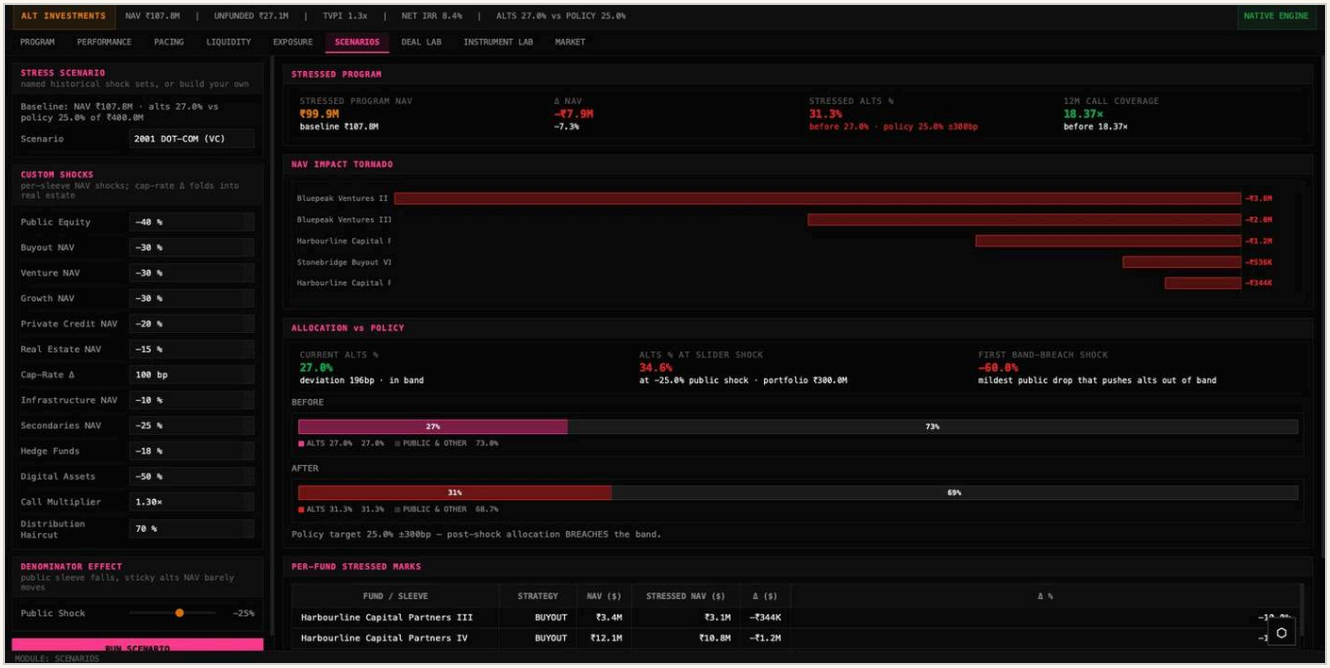
At a glance, in EXPOSURE you can…

- Choose the grid's row/column axes and value basis; read top-GP concentration, HHI/effective-N and program NAV.
- See a strategy→fund treemap and the full exposure matrix; compare each hedge fund's reported vs de-smoothed volatility & Sharpe (GLM or Geltner).

7 SCENARIOS — stress & denominator effect

RUN TO POPULATE · TAB 6

Apply a named historical shock set or a custom per-sleeve editor to the program, then read stressed NAV, the policy-band impact, a per-fund tornado & table, and a "denominator-effect" grid showing how a falling public sleeve inflates the alts weight.



SCENARIOS after running the 2001 Dot-Com shock — stressed-program tiles, the NAV-impact tornado, the before/after allocation-vs-policy bars, and the per-fund stressed-marks table.

Rail controls

Control	Type	What it does
Scenario	combo	**2001 Dot-Com (VC)** / **2008 GFC** / **2022 Rate Shock** / **Custom**. The three named sets carry fixed per-sleeve shocks.
Custom Shocks card	percent × 12	Enabled only for "Custom": per-sleeve NAV shocks (public equity, buyout, venture, growth, credit, real estate, infra, secondaries, HF, digital), a cap-rate Δ (bp, folds into real estate), call multiplier and distribution haircut.
Denominator Effect — Public Shock	slider	The public drawdown used to build the denominator-effect grid (–60%…0%).
RUN SCENARIO	button	Runs the stress (always) + the denominator effect (when a portfolio value & policy are set).

Results

- **Stressed Program** — stressed NAV, ΔNAV, stressed alts% vs the ±300 bp band, and stressed 12-month call coverage.
- **NAV Impact Tornado** — funds/sleeves ranked by dollar NAV impact.
- **Allocation vs Policy** — current alts%, alts% at the slider shock, the first band-breach shock, and before/after alts-vs-public bars.
- **Per-Fund Stressed Marks** — NAV → stressed NAV, Δ$ and Δ%, plus auto-generated pacing recommendations.

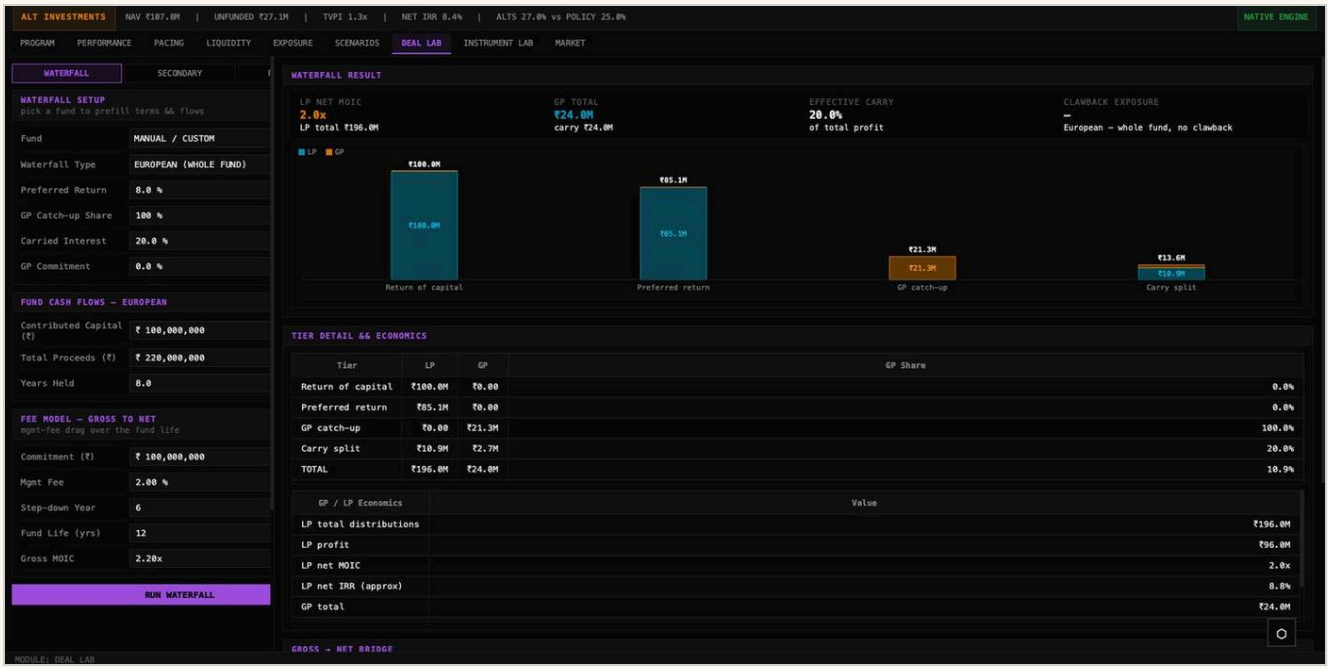
At a glance, in SCENARIOS you can…

- Stress the program with 2001 / 2008 / 2022 shock sets or a fully custom per-sleeve editor; read stressed NAV, alts-band breach and call coverage.
- See a NAV-impact tornado and per-fund stressed marks; model how a falling public sleeve inflates the alts weight across a 0…−60% grid.

8 DEAL LAB — waterfalls, fees, secondaries, RE

RUN TO POPULATE · TAB 7

A deal-level workbench behind one segmented mode selector — a distribution-waterfall + fee-bridge modeler, a fund-secondary pricing solver, and a real-estate/infra underwriter with promote. Each mode can pre-fill from the shared program or run fully manually.



DEAL LAB · Waterfall mode — LP/GP result tiles, the tier-split chart (return of capital → preferred → catch-up → carry), and the tier-detail & GP/LP economics tables.

Mode selector

WATERFALL · SECONDARY · RE / INFRA — switches both the rail form and the results canvas. Inputs entered in one mode are preserved when you switch.

Mode 1 · Waterfall modeler

Control	What it does
Fund picker	`MANUAL/CUSTOM` or a program fund — pre-fills carry, hurdle, contributed capital, proceeds, years & commitment.
Waterfall Type	European (whole fund) or American (deal-by-deal + clawback).
Preferred Return · Carry · GP Catch-up · GP Commitment	The carry economics.
Cash flows / Deals table	Contributed capital & total proceeds (European), or an editable per-deal table (American).
Fee Model (gross→net)	Commitment, mgmt fee, step-down year, fund life, gross MOIC — the fee-drag bridge.
RUN WATERFALL	Computes LP net MOIC, GP total, effective carry, clawback, the tier-split chart, tier & economics tables, and the gross→net bridge.

Mode 2 · Secondary pricing

Describe an LP stake (NAV, unfunded, strategy, age) and either solve **bid → buyer IRR** or **target IRR → bid**, with optional deferral. Results: bid % of NAV, buyer net IRR & MOIC, a comparison to curated market anchors, projected buyer cash flows and a projection table.

Mode 3 · RE / Infra underwriter

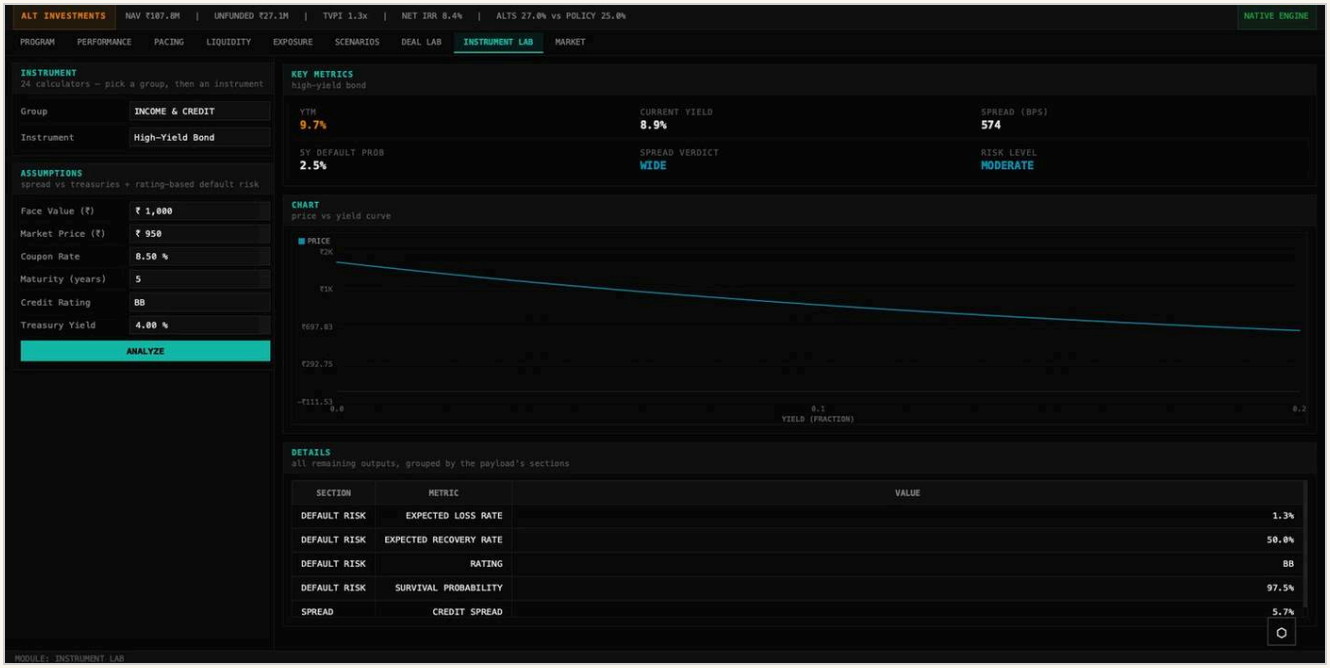
Property income (gross rent, vacancy, opex, price, market cap rate), DCF assumptions, financing (LTV, rate, amortisation), and an LP/GP promote over IRR hurdles. Results: cap rate, DSCR, LTV/debt-yield, unlevered/levered DCF IRR, an exit-cap × rent-growth value heatmap, and a promote waterfall.

> **At a glance, in DEAL LAB you can…**
>
> - Model a European or American distribution waterfall (with clawback) and the gross→net fee bridge; pre-fill from any program fund.
> - Price a fund secondary in either direction with deferral and market anchors; underwrite an RE/infra deal with cap rate, DSCR, levered IRR, a sensitivity heatmap and an LP/GP promote.

9 INSTRUMENT LAB — 24 calculators

RUN TO POPULATE · TAB 8

A spec-table-driven workbench that consolidates 24 alternative-asset calculators. Pick a group and instrument, adjust the auto-adapting assumptions form, press ANALYZE — get headline metric tiles, one decision chart and a full detail table. Every run is a standalone what-if; nothing here touches your saved program.



INSTRUMENT LAB · High-Yield Bond analysed — key-metric tiles (YTM, current yield, spread bps, default prob, verdict), the price-vs-yield chart, and the grouped detail table.

How it works

1. **Group** then **Instrument** dropdowns select one calculator; the assumptions form rebuilds for it.
2. Fields are pre-filled with sensible defaults — percent inputs, money spin-boxes, choice/yes-no dropdowns, and comma-separated or `months:price` series fields.
3. **ANALYZE** → 3–6 headline tiles, one chart (line / histogram / bar) and a `SECTION · METRIC · VALUE` detail table.

The 24 instruments (by group)

Group	Instruments
Income & Credit (7)	High-Yield Bond · Emerging-Market Bond · Convertible Bond · Preferred Stock · TIPS · Series I Savings Bond · Stable Value Fund
Annuities & Structured (5)	Fixed Annuity · Inflation-Indexed Annuity · Variable Annuity · Equity-Indexed Annuity · Structured Note
Equity Strategies (6)	Covered Call · Leveraged ETF Decay · SRI / ESG Fund · Managed Futures / CTA · Market-Neutral Fund · Asset Location
Real Assets & Digital (4)	International REIT · Commodity Futures · Precious-Metals Equity · Digital Asset
Analytics (2)	Time-Series Performance (TWR/Sharpe/Sortino) · VaR & Stress (historical / parametric / Cornish-Fisher + Monte Carlo)

Deterministic. Monte-Carlo instruments (Leveraged ETF Decay, VaR & Stress) expose a **Seed** field, so results are reproducible. Series fields left empty fall back to the engine's demo series.

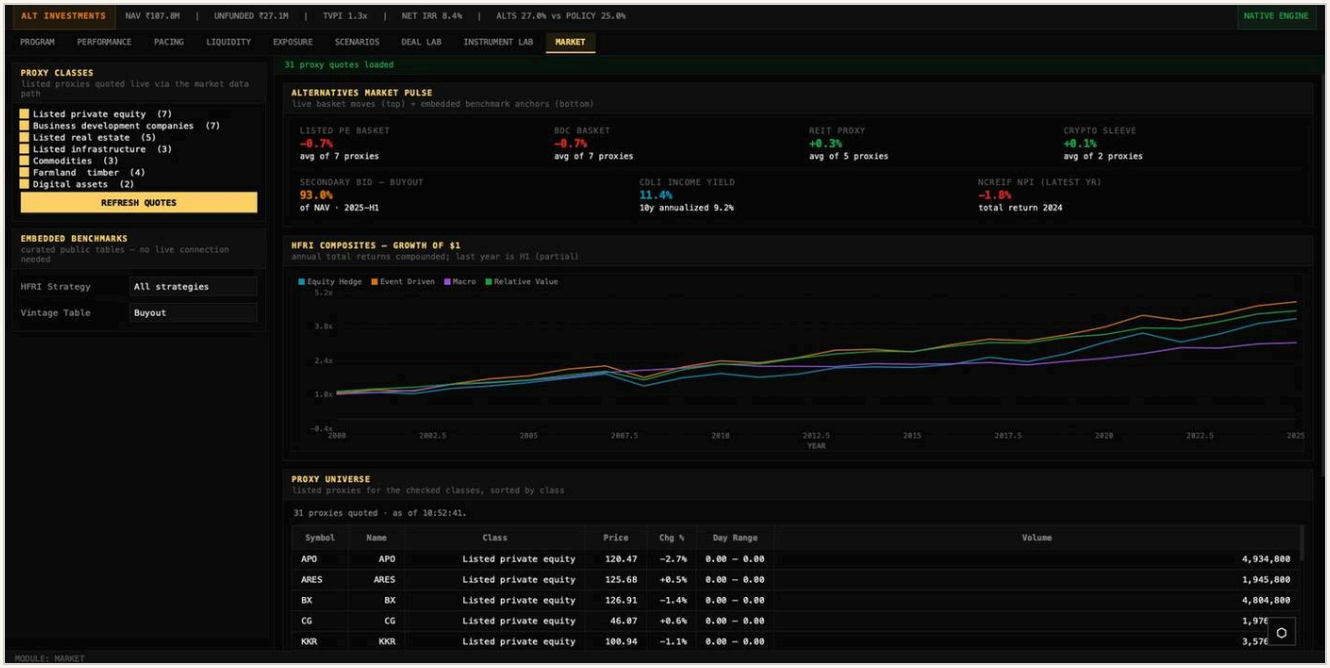
At a glance, in INSTRUMENT LAB you can…

- Price 24 alternative instruments — credit/income, annuities & structured products, equity strategies, real assets & digital — plus generic TWR and VaR analytics on any pasted return series.
- Get headline tiles, a decision chart and a full detail table for each, as a standalone what-if that never writes to your program.

10 MARKET — live proxies & benchmarks

LIVE QUOTES · TAB 9

A live listed-proxy dashboard for the alternatives program — quote baskets for seven proxy classes, alongside embedded curated benchmarks (HFRI composites, secondary anchors, CDLI/NCREIF headlines, PE vintage quartiles) that render even offline.



MARKET — live basket-move & benchmark tiles, the HFRI growth-of-$1 chart (2000–2025), and the live proxy-universe quote table.

Rail controls

Control	What it does
Proxy Classes (7 checkboxes)	Listed PE · BDCs · Listed real estate · Listed infrastructure · Commodities · Farmland & timber · Digital assets — toggle which tickers are quoted and shown.
REFRESH QUOTES	Fetches live quotes for the checked classes (auto-fires once on first open).
HFRI Strategy / Vintage Table	Filter the HFRI growth chart and the PE vintage-benchmark table.

Results

- **Alternatives Market Pulse** — live basket day-moves (listed PE, BDC, REIT, crypto) on top; embedded anchors (secondary bid ~93% of NAV, CDLI income yield, latest NCREIF NPI) below.
- **HFRI Composites** — compounded growth-of-$1 for four hedge-fund composites, 2000–2025.
- **Proxy Universe** — a live quote table (symbol, name, class, price, chg%, day range, volume) for the checked classes.
- **PE Vintage Benchmarks** — net-IRR quartile breakpoints by vintage for buyout / VC / credit / real estate.

> **Data provenance.** Live quotes route through the terminal's market-data path (local yfinance by default). Every *benchmark* number is a curated estimate transcribed from public headline figures (Cambridge Associates, HFRI, NCREIF, Cliffwater CDLI, Jefferies) — clearly labelled, refreshed quarterly, **not licensed index data**.

At a glance, in MARKET you can…

- Pull live quotes for seven listed-proxy baskets and read each basket's average day move; browse a sortable proxy-universe table.
- Reference HFRI growth curves, secondary/CDLI/NCREIF anchors and PE vintage quartiles — all available even offline.

11 The seeded demo program

REFERENCE

The numbers throughout this guide come from the bundled demo program (LOAD SAMPLE PROGRAM). It is deterministic — a fixed seed pinned to as-of **2025-06-30**, a $400M portfolio with a 25% alts policy target, $30M reserves and a $20M credit line. Actual alts share lands ~27%, so the policy-band flag fires out of the box.

10 private funds (~$108M committed)

Fund	GP	Strategy	Vintage	Commit
Harbourline Capital Partners III	Harbourline Capital	buyout	2016	$14M
Harbourline Capital Partners IV	Harbourline Capital	buyout (American)	2019	$12M
Stonebridge Buyout VI	Stonebridge Equity	buyout	2023	$18M
Bluepeak Ventures II	Bluepeak Ventures	vc	2018	$8M
Bluepeak Ventures III	Bluepeak Ventures	vc	2021	$9M
Irongate Direct Lending I	Irongate Credit	credit	2018	$9M
Irongate Direct Lending II	Irongate Credit	credit	2021	$10M
Cornerline Real Estate Value-Add II	Cornerline RE	real_estate	2019	$10M
Gridway Infrastructure Partners I	Gridway Infra	infra	2020	$9M
Meridian Secondary Opportunities V	Meridian Alternatives	secondaries	2022	$9M

3 hedge-fund / liquid-alt sleeves

Position	Strategy	Invested	Liquidity terms
Meridian Global Macro Fund	macro	$16M	monthly redemption, 30-day notice, no gate/lockup
Northgate Equity Long/Short	equity_ls	$17M	quarterly, 60-day notice, 25% gate, lockup to 2021-12-31 (appraisal-smoothed)
Digital Assets Sleeve (BTC)	digital	$3M	daily redemption, 1-day notice, no gate/lockup

Resulting headline (as of 2025-06-30): commitment ≈ $108M · called ≈ $81M · unfunded ≈ $27M · program NAV ≈ $108M (≈$54M funds + ≈$54M HF) · TVPI 1.3× · net IRR ≈ 8.4% · alts 27.0% vs 25% policy (+196 bps).

12 Analytics engine — capability catalog

NATIVE C++

A single in-process C++ engine powers every module — no Python, no server, deterministic. The highlights, by area:

Area	Key models & what they compute
Program	**program_rollup** (totals, pooled IRR/DPI/TVPI, alts% vs policy, grouped tables), **exposure_matrix** (HHI, effective-N, top-GP, treemap), **denominator_effect**, **liquidity_forecast** (coverage, shortfall, stress), **nav_nowcast**, **stress_scenario**.
Performance	**xirr**, **fund_metrics** (DPI/RVPI/TVPI/MOIC/XIRR), **pooled_metrics**, **horizon_irr**, **pme_analysis** (KS-PME, Direct Alpha, PME+, Long-Nickels), **quartile_placement**, **j_curve_series**.
Pacing	**ta_projection** & **program_pacing** (Takahashi-Alexander with base/optimistic/pessimistic bands), **commitment_solver** (steady-state bisection).
Deal lab	**distribution_waterfall** (4-tier EU/US + clawback), **fee_model** (gross→net), **secondary_pricing** (bid↔IRR + anchors).
Credit & RE	**loan_economics**, **unitranche_split**, **credit_portfolio**, **bdc_metrics**, **cap_rate_analysis**, **property_dcf**, **promote_waterfall**, **reit_nav**.
Hedge / series	**return_stats** (Sharpe/Sortino/Omega/Calmar, max DD, VaR/CVaR), **unsmooth_returns** (Getmansky-Lo-Makarov / Geltner), **style_regression**, **liquidity_ladder**.
Instruments	24 per-instrument calculators (see §9) plus generic time-series & VaR analytics.

13 AI agent tools (30 MCP tools)

AGENTIC

The terminal's AI agent can drive this exact program through 30 tools in the `alt-investments` category — the same store the UI shows. Program-scoped tools default to your saved program, so the agent analyses what's on your screen.

- **Program & views:** alt_program_snapshot, alt_program_rollup, alt_exposure_matrix, alt_denominator_effect, alt_liquidity_forecast, alt_stress_scenario, alt_nav_nowcast
- **Edit (store CRUD):** alt_upsert_fund, alt_add_cashflow, alt_add_nav_mark, alt_remove_fund*, alt_load_sample_program*
- **Fund analytics:** alt_fund_metrics, alt_pme_analysis, alt_quartile_placement, alt_j_curve
- **Planning:** alt_ta_projection, alt_program_pacing, alt_commitment_solver
- **Deal lab:** alt_distribution_waterfall, alt_fee_model, alt_secondary_pricing, alt_property_snapshot, alt_property_dcf, alt_promote_waterfall
- **Hedge / series:** alt_return_stats, alt_unsmooth_returns, alt_liquidity_ladder
- **Instruments & benchmarks:** alt_instrument_analyze (32 calculators), alt_benchmarks

* **alt_remove_fund** and **alt_load_sample_program** are marked destructive and require confirmation.

14 Quick reference & important notes

HOW DO I...

I want to...	Go to
Start from scratch quickly	PROGRAM ▸ **LOAD SAMPLE PROGRAM**.
Add my own fund + cash flows	PROGRAM ▸ Fund Editor ▸ SAVE FUND, then ADD CASH FLOW / ADD NAV MARK.
See net IRR, TVPI & PME	PERFORMANCE (auto-runs) — pick an index & horizon set.
Know how much to commit each year	PACING ▸ Commitment Solver ▸ set a target ▸ SOLVE COMMITMENT.
Check I can fund the calls	LIQUIDITY (auto-runs) — read coverage ratio & 12-month net need.
Find my concentration risk	EXPOSURE (auto-runs) — top-GP share, HHI, treemap.
Stress-test a crash	SCENARIOS ▸ pick 2008 GFC (or Custom) ▸ RUN SCENARIO.
Model a carry waterfall	DEAL LAB ▸ Waterfall ▸ pick a fund ▸ RUN WATERFALL.
Price a secondary stake	DEAL LAB ▸ Secondary ▸ choose solve direction ▸ PRICE SECONDARY.
Value a bond / annuity / crypto sleeve	INSTRUMENT LAB ▸ pick group + instrument ▸ ANALYZE.
Watch listed-alt proxies	MARKET (auto-fetches) — REFRESH QUOTES to update.

Important notes for institutional users

- **Local & private.** The entire program lives on your device (`alt_program.json`); the analytics engine is native C++ and never sends your holdings anywhere. Only the MARKET tab makes a network call (for public proxy quotes).
- **Deterministic, dated math.** Every model honours the program's as-of date and uses a fixed seed for any simulation — the same inputs always give the same output.
- **Benchmarks are estimates.** PME index tables, HFRI/CDLI/NCREIF anchors and PE vintage quartiles are curated approximations of public headline figures, not licensed index data — refreshed quarterly.
- **Two destructive actions.** PROGRAM ▸ LOAD SAMPLE (no confirm) and CLEAR ALL DATA (with confirm) replace your program; the cleared state persists across restarts.
- **Known limitation.** On EXPOSURE, the "nowcast Δ" adjustment line under Program NAV is inactive; all other analytics on that page are live.
- **Where results are live vs on-demand.** PROGRAM / PERFORMANCE / LIQUIDITY / EXPOSURE / MARKET populate automatically; PACING / SCENARIOS / DEAL LAB / INSTRUMENT LAB compute when you press their run button.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

M&A Analytics

A complete, screen-by-screen walkthrough of the M&A Analytics tab — a native-C++ deal workbench for running a live merger: deal set-up, four-way valuation, the full merger model, deal structure, market intelligence with a live SEC EDGAR scan, specialized valuation, and risk & fairness.

`>_ seven modules · one native engine`

Covers · **the 7-module workbench** · **a native C++ engine (42 models)** · **four-way valuation merger model · structure · risk & fairness** · live SEC EDGAR scan · 47 agent tools

Demonstrated live on the Fincept Terminal M&A Analytics tab.

Contents

12 SECTIONS

GETTING ORIENTED

Overview — what the M&A Analytics tab is 3

1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 4

THE MODULES

2 DEAL DESK — set up the deal 6

3 VALUATION — DCF, comps, precedents, LBO & the football field 8

4 MERGER MODEL — accretion/dilution, sources & uses, synergies 10

5 STRUCTURE — payment mix, exchange ratio, collar, earnout, CVR 12

6 MARKET INTEL — deal database, live EDGAR scan, cross-deal analytics 13

7 SPECIALIZED — startup valuation & industry lens 15

8 RISK & FAIRNESS — Monte Carlo, fairness opinion, premium, process 16

REFERENCE

9 The seeded sample deal & deal database (reference) 17

10 Analytics engine — capability catalog 17

11 AI agent tools (47 MCP tools) 17

12 Quick reference — “How do I…?” & important notes 18

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Numbers, charts and tables are live output, not mock-ups. Control labels are quoted exactly as they appear on screen.

What the M&A Analytics tab is

OVERVIEW

The M&A Analytics tab is a **deal workbench**. Everything on it revolves around a single subject: **one live transaction** — an acquirer buying a target on a set of offer terms. You set the deal up once on the DEAL DESK; every other sub-tab reads that same deal from a shared context and answers a different banker's question about it — what is the target worth, is it accretive, how is it financed, how does it compare to the market, and is the offer fair?

All analytics run on a native C++ engine (the green NATIVE ENGINE badge) — 42 models spanning DCF/WACC, trading comps, precedent transactions, LBO, the full merger model with accretion/dilution, sources & uses, synergies, deal structure (payment mix, exchange ratio, collar, earnout, CVR), startup valuation, fairness & Monte-Carlo risk. There is no Python and no server round-trip for the math. The one network call in the whole subsystem is a live SEC EDGAR filing scan on the Market Intel tab (keyless). The same deal can also be driven by the terminal's AI agent through 47 dedicated tools.

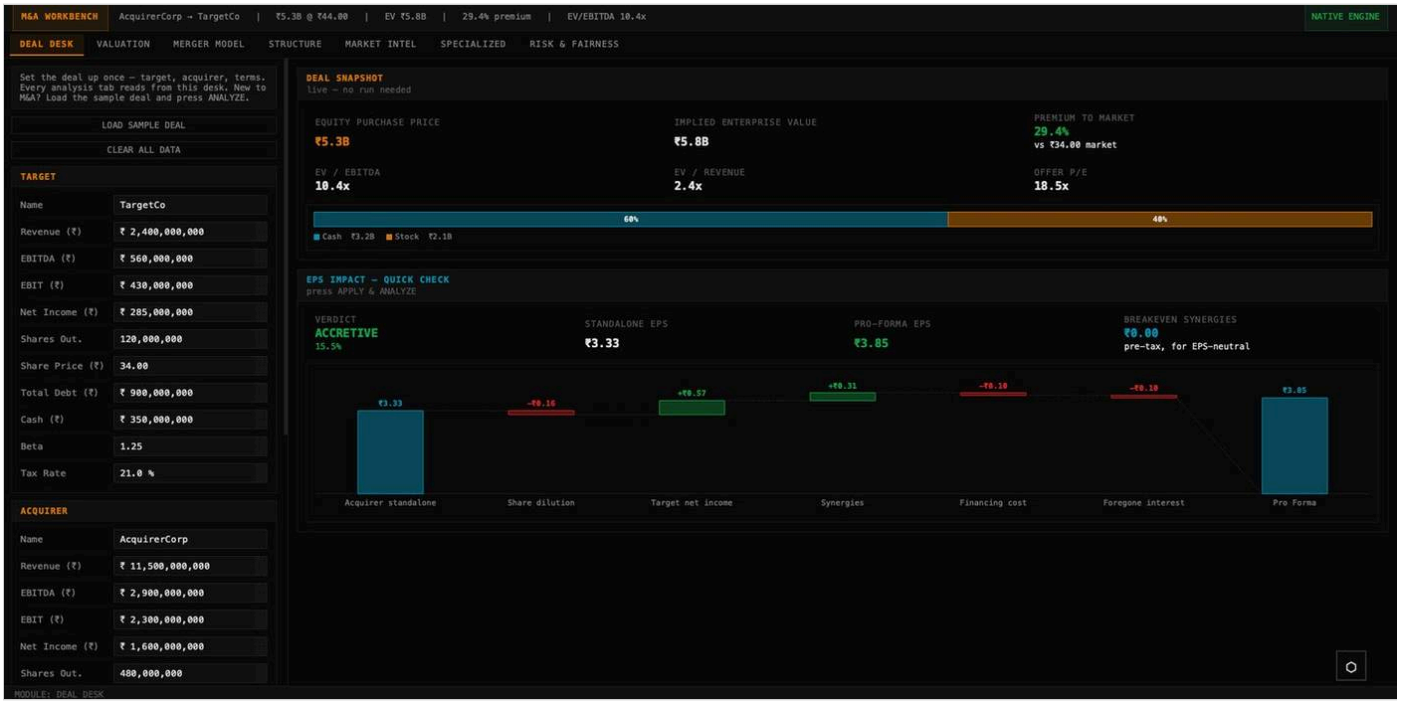
AT A GLANCE, IN M&A ANALYTICS YOU CAN…

- **Set up a deal** — target & acquirer financials plus offer terms — or load a worked sample in one click, and read live deal multiples + an instant accretion/dilution verdict.
- **Value the target four ways** — DCF (with a WACC × terminal-growth sensitivity grid), trading comps, precedent transactions, and an ability-to-pay LBO — converging on one football field.
- **Build the full merger model** — EPS bridge, sources & uses, contribution vs ownership, multi-year pro-forma, a price × synergy grid, and a synergy DCF.
- **Structure the consideration** — cash/stock mix feasibility & dilution, exchange ratio, collar protection, earnout tranches, and a regulatory CVR.
- **Work a market-intel deal database** (~20 real precedents), run a live SEC EDGAR merger-filing scan, and benchmark your premium, rankings, payment mix and industry mix.
- **Value a private / startup target** with five blended methods, and apply a SaaS / Pharma / Bank KPI lens.
- **Stress-test the deal** — Monte-Carlo valuation distribution, a fairness opinion, premium-vs-history, and a sale-process scorecard.

1 Anatomy of the workbench

ORIENTATION

Open the tab from the command bar (type `ma` → **M&A Analytics**) or **Navigate ‣ Research & Intelligence ‣ M&A Analytics**. Every sub-tab shares the same three-part chrome and the same two-pane page layout.



The workbench on the DEAL DESK. ① Deal bar ② Sub-tab strip ③ Left "assumptions" rail ④ Right results canvas ⑤ Module status bar.

Region	What it shows / does
① **Deal bar** (top)	The amber **M&A WORKBENCH** brand, a live one-line deal summary — `Acquirer → Target · offer @ /share · EV · premium · EV/EBITDA` — and a green **NATIVE ENGINE** badge. Before a deal exists it reads "No active deal — set one up on the DEAL DESK".
② **Sub-tab strip**	Seven modules in a fixed order: **DEAL DESK · VALUATION · MERGER MODEL · STRUCTURE · MARKET INTEL · SPECIALIZED · RISK & FAIRNESS**. Each has its own accent colour; the active one is highlighted.
③ **Assumptions rail** (left)	A scrollable stack of input cards — the knobs for that module. This is where you type; nothing here leaves your device.
④ **Results canvas** (right)	KPI tiles, charts and tables. A thin status strip at the top shows ⏳ `Running…` (cyan) while a model runs, a green confirmation, or a red error. Because the engine is synchronous, the busy state only flashes.
⑤ **Status bar** (bottom)	Shows `MODULE: <name>` for the active sub-tab.

The two-pane pattern & when results appear

Every module is a left **rail** (inputs) + right **canvas** (outputs). The **DEAL DESK** snapshot and the **MARKET INTEL** deal database are live the moment you open them. The other outputs wait for a run button — **VALUATION** (RUN DCF / TRADING COMPS / PRECEDENTS / LBO), **MERGER MODEL** (BUILD MERGER MODEL), **STRUCTURE** (per-tool run buttons), **SPECIALIZED** (VALUE STARTUP / SECTOR), **RISK** (RUN SIMULATION / ASSESS FAIRNESS / …) — and show a "Run…" prompt until pressed.

Two behaviours worth knowing. (1) The deal is entered **once** on the Deal Desk and shared everywhere; most Merger/Structure/Valuation runs error out with a "Set up the deal on the Deal Desk first" message until a deal exists (or you press **LOAD SAMPLE DEAL**). (2) Money is shown in your terminal's display currency (₹ in these screenshots); the underlying figures are the deal's own numbers.

Shared building blocks

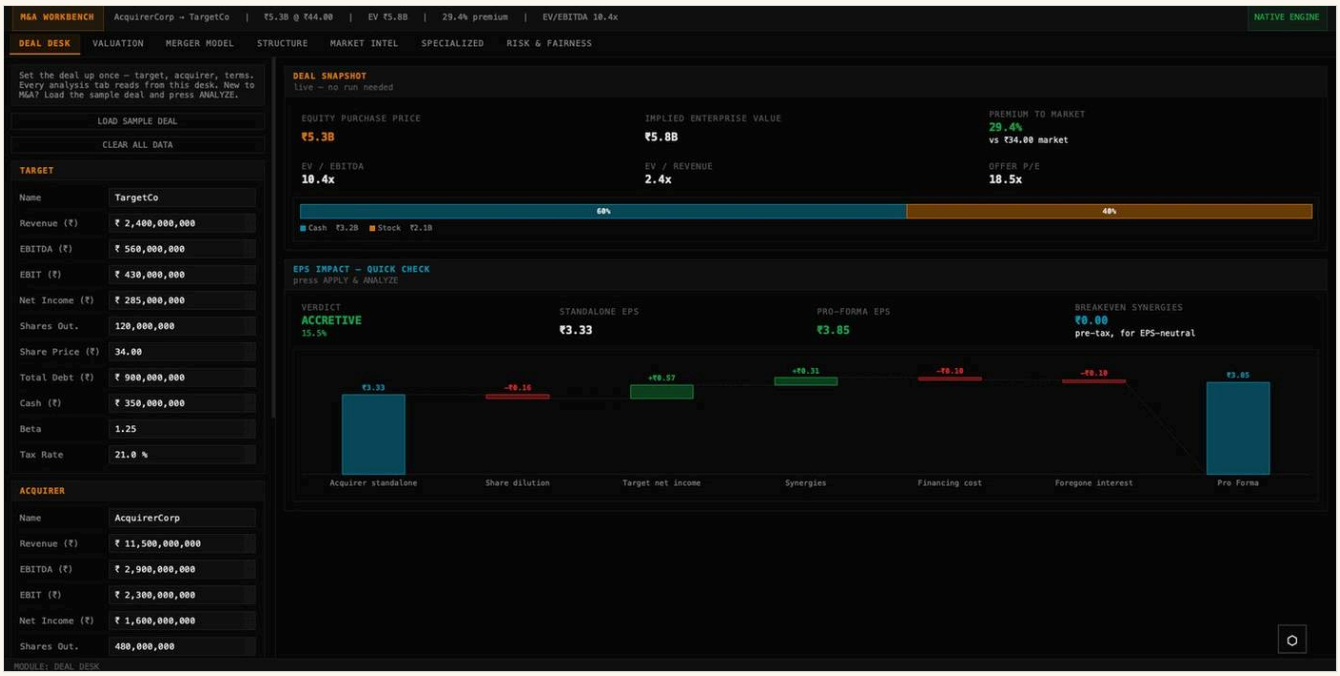
- **KPI tiles** — caption + large value + sub-line, colour-toned (green good / red bad / amber accent / cyan info).
- **Section cards** — titled input groups on the rail, each with a one-line hint.
- **Money / percent / number spin-boxes; editable input tables** (peers, tranches).
- **Charts** — football field, waterfall/EPS bridge, sensitivity heatmap, sources/uses segment bars, contribution bars, Monte-Carlo histogram, ranking bars.
- **Read-only result tables** — never editable.
- **Run buttons** (solid accent) and ghost buttons (secondary).

This widget library (*MAWidgets*) is the origin set the Alt Investments tab re-exports — the two workbenches look and behave alike.

2 DEAL DESK — set up the deal

LIVE · TAB 1

The set-up-once desk. Enter the target & acquirer standalone financials and the offer terms; every other tab reads them. Shows a live deal snapshot and an instant accretion/dilution quick check.



DEAL DESK — rail (target/acquirer/terms), the live deal snapshot with a consideration-mix bar, and the EPS accretion/dilution quick check with its bridge.

Rail — actions & company cards

Control	Type	What it does
LOAD SAMPLE DEAL	button	Loads a fully worked demo (TargetCo / AcquirerCorp, $44 offer ≈29% premium, 60% cash) and runs the quick check. Replaces the current deal (no confirm).
CLEAR ALL DATA	button	Confirm-gated. Wipes the desk deal **and** the local precedent-deal database (all 20 seeds) permanently.
TARGET / ACQUIRER cards	fields	For each company: Name, Revenue, EBITDA, EBIT, Net Income, Shares Out., Share Price, Total Debt, Cash, Beta, Tax Rate — the standalone profile reused by every downstream model.

DEAL TERMS card

Field	Type	Meaning
Offer / Share ($)	number	Offer price per target share — drives premium, all multiples, and equity purchase price.
Cash %	percent	Cash portion of consideration (remainder in acquirer stock).
Revenue / Cost Synergies ($)	money	Expected synergies fed to the accretion model & synergy DCF.
Financing Rate	percent	Interest on new acquisition debt.
Deal Fees	percent	Advisory + financing fees (% of deal value).
APPLY & ANALYZE DEAL	button	Commits the rail into shared state and runs the accretion/dilution quick check.

Results — Deal Snapshot & EPS quick check

- **Deal Snapshot** (live, no run) — Equity Purchase Price, Implied Enterprise Value, Premium to Market, EV/EBITDA, EV/Revenue, Offer P/E, plus a cash-vs-stock consideration-mix bar.

- **EPS Impact — Quick Check** — an ACCRETIVE/DILUTIVE verdict, standalone vs pro-forma EPS, breakeven synergies, and an EPS bridge (standalone → share dilution → target NI → synergies → financing cost → foregone interest → pro forma).

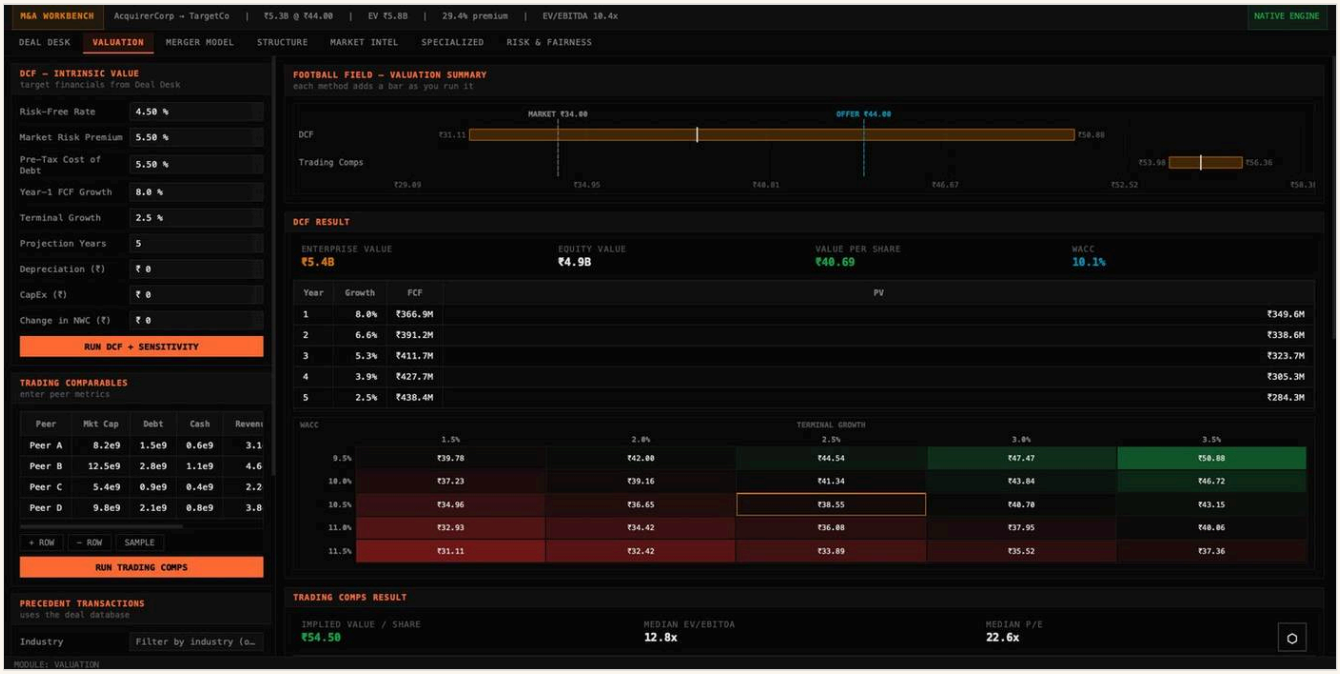
At a glance, in DEAL DESK you can…

- Enter target & acquirer financials and the offer terms once; load a worked sample in one click.
- Read a live deal snapshot (purchase price, EV, premium, EV/EBITDA, EV/Revenue, P/E) and an instant accretion/dilution verdict with a full EPS bridge.

3 VALUATION — what is the target worth

RUN TO POPULATE · TAB 2

Run the four core valuation methods for the target — DCF (with an automatic WACC × terminal-growth sensitivity grid), trading comparables, precedent transactions, and an ability-to-pay LBO — and roll each method's per-share range onto one football field.



VALUATION — the football field (DCF & Trading Comps ranges vs offer/market markers), the DCF result with FCF projection and sensitivity heatmap, and the trading-comps result.

Rail cards & run buttons

Card	Key inputs & run button
DCF — Intrinsic Value	Risk-free rate, market risk premium, pre-tax cost of debt, year-1 FCF growth, terminal growth, projection years, depreciation/capex/ΔNWC (auto-derived if 0). **RUN DCF + SENSITIVITY** → EV, equity value, value/share, WACC, a year-by-year FCF/PV table, and an automatic 5×5 sensitivity heatmap.
Trading Comparables	An editable peer table (Mkt Cap / Debt / Cash / Revenue / EBITDA / Net Inc), **+ ROW / – ROW / SAMPLE**, and **RUN TRADING COMPS** → median EV/EBITDA, median P/E, implied value/share.
Precedent Transactions	An optional industry filter and **RUN PRECEDENT ANALYSIS** — values the target off the Market-Intel deal universe (or 20 seeded real deals).
LBO — Ability to Pay	Entry/exit multiple, leverage, revenue growth, EBITDA margin, hold period. **BUILD LBO MODEL** → sponsor IRR/MOIC, an exit-multiple IRR spread, and a debt-paydown schedule.

The football field

Each method that runs adds a per-share range bar (DCF, Trading Comps, Precedent Txns) to the **Football Field** at the top, with dashed **OFFER** and **MARKET** markers — the one-glance "is the offer inside the value range?" view. **LBO contributes no bar** (it outputs IRR/MOIC, not a per-share value).

> **Definitions.** The DCF is a CAPM-WACC discounted-FCF model with a Gordon terminal value (WACC must exceed terminal growth). Comps & precedents value the target off peer/deal EV multiples (q1/median/q3). The sensitivity heatmap's centre cell is the base case (outlined amber); its centring WACC is an approximation, so it can differ slightly from the DCF tile.

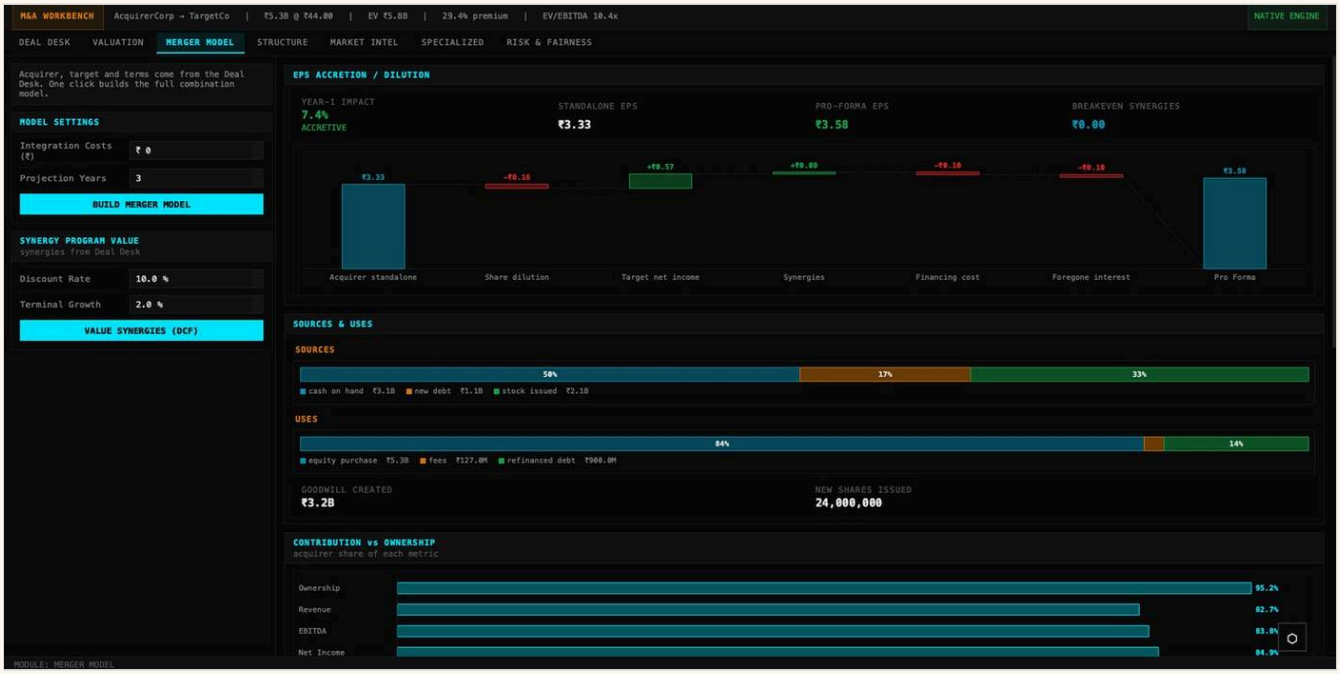
At a glance, in VALUATION you can…

- Run a full DCF with a sensitivity grid; value off trading comps and precedent transactions; run an ability-to-pay LBO.
- See DCF, comps and precedent ranges converge on one football field, benchmarked against your offer and the market price.

4 MERGER MODEL — the combination

RUN TO POPULATE · TAB 3

Build the full acquirer + target combination from the Deal Desk terms — EPS accretion/dilution with a bridge, sources & uses, contribution vs ownership, a multi-year pro-forma, a price × synergy grid, and a standalone DCF of the synergy program.



MERGER MODEL — accretion/dilution tiles + EPS bridge, sources & uses segment bars with goodwill/new-shares, and the contribution-vs-ownership bars.

Rail controls

Control	What it does
Integration Costs · Projection Years	The two overrides the build needs beyond the Deal Desk (year-1 full / year-2 half integration drag; number of pro-forma years).
BUILD MERGER MODEL	Runs the full model → accretion/dilution + EPS bridge, sources & uses, goodwill, new shares, contribution, multi-year pro-forma, and a price × synergy accretion heatmap.
Discount Rate · Terminal Growth · VALUE SYNERGIES (DCF)	Values the synergy program as a standalone DCF (explicit PV + terminal value). Errors if discount ≤ terminal growth.

Results

- **EPS Accretion/Dilution** — year-1 impact (ACCRETIVE/DILUTIVE), standalone vs pro-forma EPS, breakeven synergies, and the EPS bridge.
- **Sources & Uses** — two stacked bars (cash on hand / new debt / stock issued vs equity purchase / fees / refinanced debt), plus goodwill created and new shares issued.
- **Contribution vs Ownership** — the acquirer's share of revenue / EBITDA / net income vs its post-deal ownership.
- **Multi-year Pro-forma** table + a 5×5 **price × synergy** accretion heatmap; and (separately) the **Synergy DCF** tiles + schedule.

> **Two labelling notes.** The sensitivity heatmap's Y-axis reads "OFFER / SHARE" but its values are total purchase-price levels (±20%). Fees include a hardcoded 2% financing fee on new debt; if sources < uses the engine auto-balances into cash.

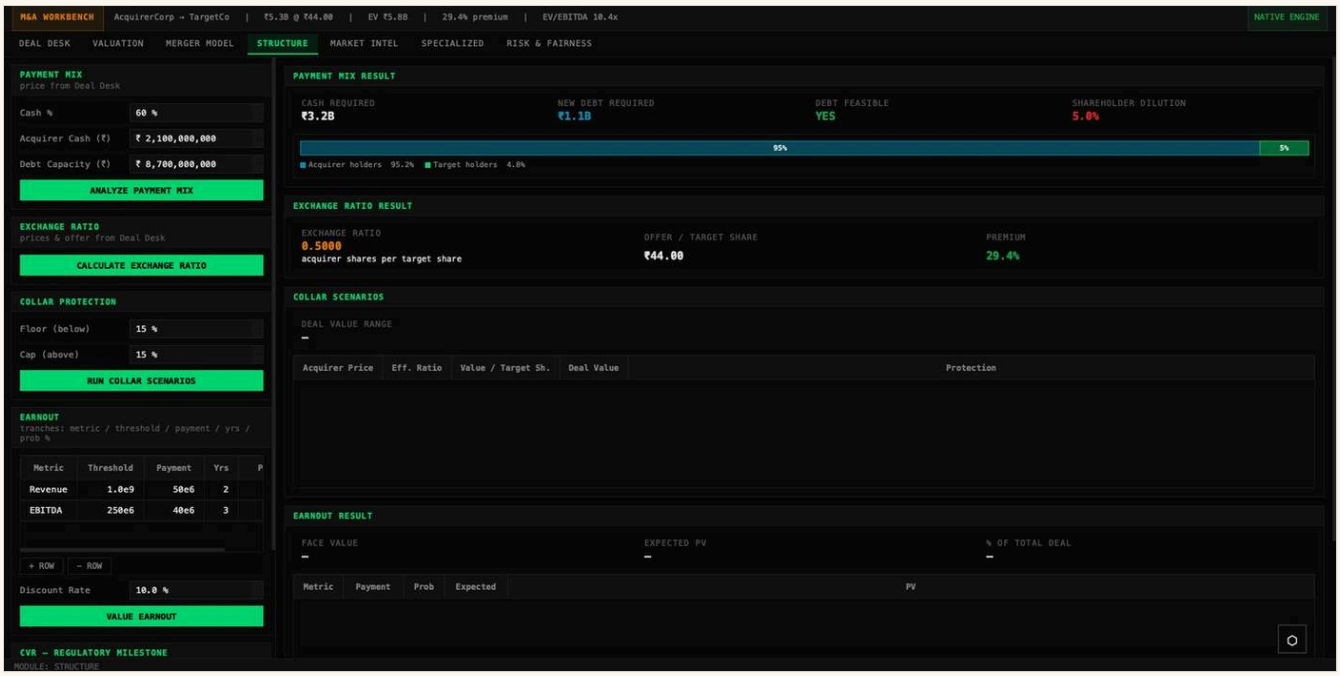
At a glance, in MERGER MODEL you can…

- Build a full combination model in one click; read the accretion/dilution verdict, EPS bridge, sources & uses, goodwill and new shares.
- Compare contribution vs ownership; scan a multi-year pro-forma and a price × synergy grid; value the synergy program by DCF.

5 STRUCTURE — how it's paid for

RUN TO POPULATE · TAB 4

Analyse how the deal is structured — cash/stock mix feasibility & dilution, a fixed exchange ratio, collar protection scenarios, earnout tranche valuation, and a regulatory CVR. Each tool has its own run button.



STRUCTURE — payment-mix result (cash/debt required, feasibility, dilution, ownership bar) and exchange-ratio result; collar & earnout tools sit below.

Tool	Inputs & what it returns
Payment Mix	Cash %, acquirer cash, debt capacity (auto ≈ 3× EBITDA) → ANALYZE PAYMENT MIX → cash required, new debt required, debt feasible (Y/N), shareholder dilution, post-deal ownership bar.
Exchange Ratio	CALCULATE EXCHANGE RATIO → acquirer shares per target share, implied offer/target share, premium.
Collar Protection	Floor / cap %, RUN COLLAR SCENARIOS → a value-protection scenario table across a ±30% acquirer-price band (deal value range, effective ratio, protection active/—).
Earnout	An editable tranche table (metric / threshold / payment / yrs / prob %) + discount rate, VALUE EARNOUT → face value, expected PV, % of deal.
CVR — Regulatory Milestone	Payment if approved, approval probability, years to decision, VALUE CVR → expected CVR value + success probability.

Honest caveats. The earnout table is pre-seeded with two rows whose probabilities sum to **105%**; the model requires them to total 100%, so the default errors until you edit them. The earnout "EXPECTED PV" tile shows the un-probability-weighted PV. The CVR tool exposes only the **regulatory** type (the engine also supports milestone / earnout / litigation CVRs).

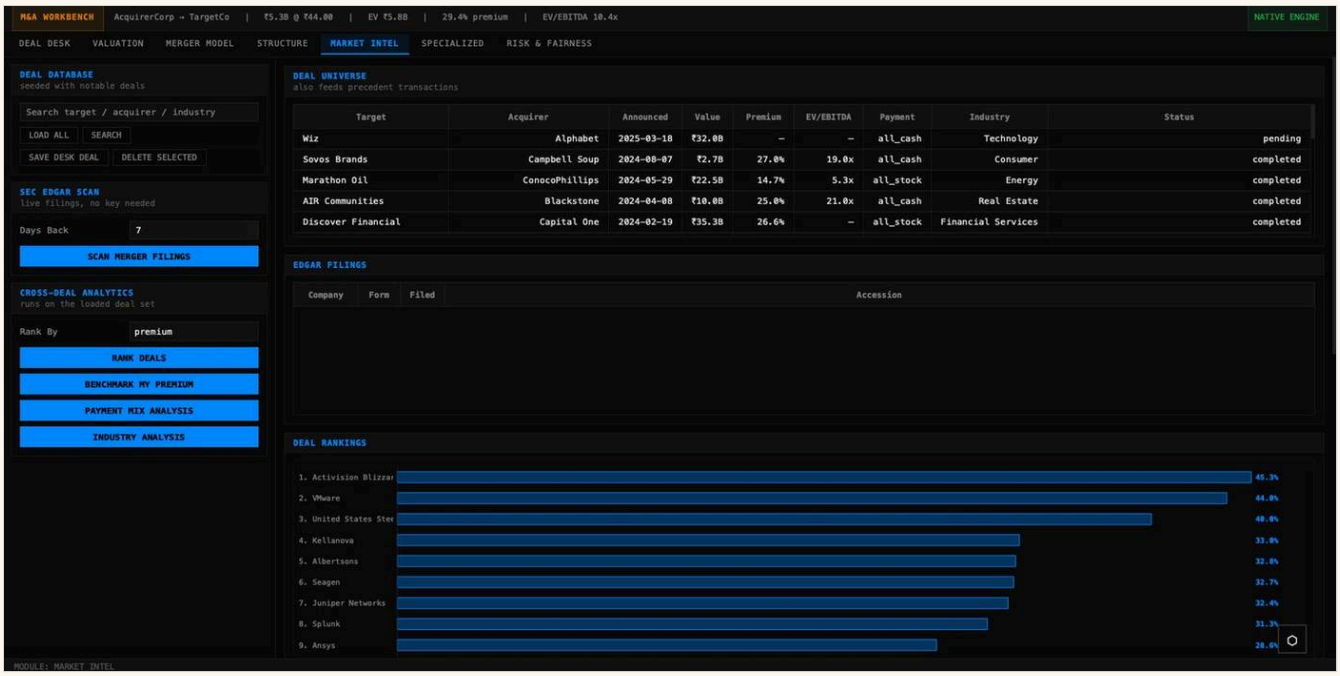
At a glance, in STRUCTURE you can…

- Test a cash/stock mix for fundability and dilution; compute a fixed exchange ratio and premium; model a collar's value protection.
- Value probability-weighted earnout tranches and a regulatory-approval CVR.

6 MARKET INTEL — deals & filings

LIVE DB + EDGAR · TAB 5

A local precedent-transaction database (~20 notable real deals), a live SEC EDGAR merger-filing scanner, and five cross-deal analytics. The loaded universe also feeds the Valuation tab's precedent transactions.



MARKET INTEL — the seeded deal universe table, the cross-deal analytics rail, and the premium-ranked deal-rankings chart.

Rail controls

Control	What it does
Deal Database — Search / LOAD ALL / SEARCH / SAVE DESK DEAL / DELETE SELECTED	Browse, search, save the Deal-Desk deal into, or delete from the local deal store (a device-local JSON file, seeded on first run).
SEC EDGAR Scan — Days Back + SCAN MERGER FILINGS	**Live network call** to SEC EDGAR full-text search (keyless; identifying User-Agent only) for recent 8-K / S-4 / SC TO-T / DEFM14A merger filings.
Cross-Deal Analytics — Rank By + RANK DEALS / BENCHMARK MY PREMIUM / PAYMENT MIX ANALYSIS / INDUSTRY ANALYSIS	Five in-process analytics over the loaded universe: rank deals, benchmark your premium vs the universe, payment-mix split, and per-industry aggregation.

Results

- **Deal Universe** — the seeded deals (target, acquirer, announced, value, premium, EV/EBITDA, payment, industry, status).
- **EDGAR Filings** — company / form / filed / accession from the live scan.
- **Deal Rankings** bar chart · **Premium Benchmark** (your premium vs median + percentile) · **Payment Mix** bar · **Industry Breakdown** table.

> **Honest caveats.** (1) The **EDGAR scan is live** — it hits SEC directly; it shows only the first ~10 results even when hundreds match, and its query is fixed to "merger agreement". (2) **Payment Mix Analysis** currently mis-buckets every seeded/saved deal as "Mixed / 0% cash" (a field-name mismatch: the analyzer reads `cash_pct` while deals store `cash_percentage`) — the deal table's own Payment column is correct. (3) Analytics always run on the full universe (the "run on selected rows" path is inert). (4) DELETE / CLEAR removes seeded deals permanently.

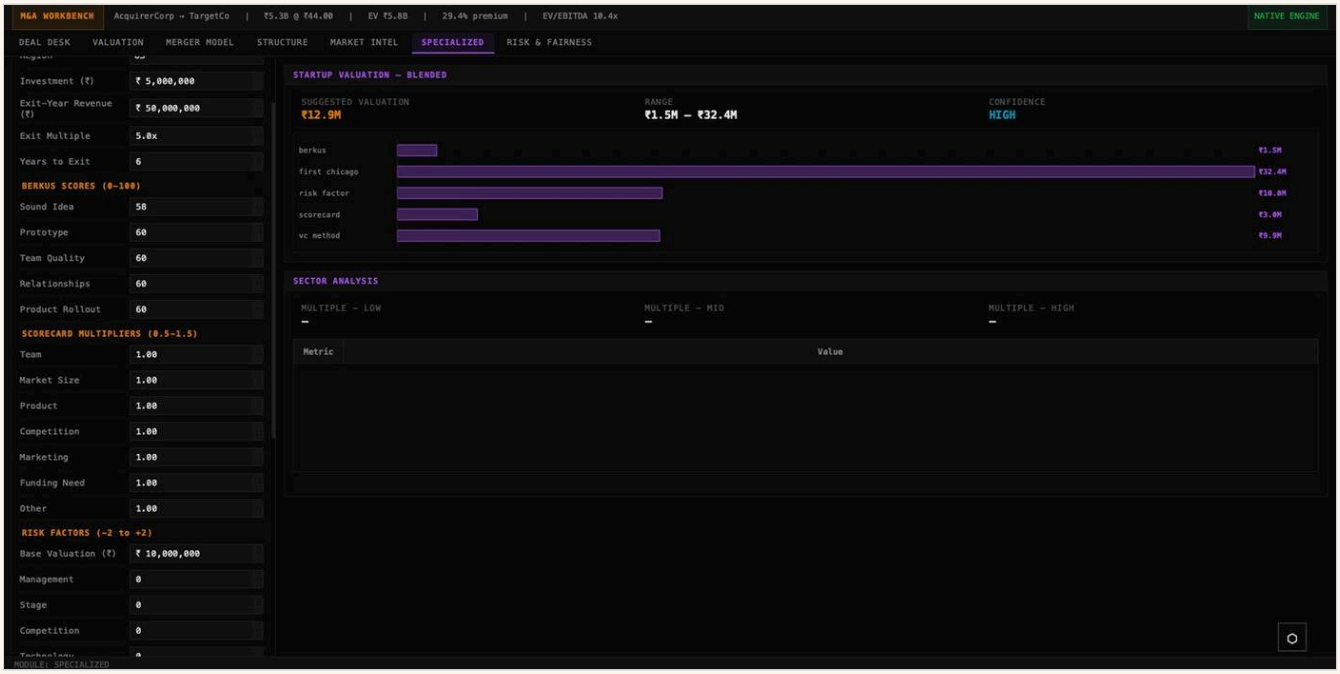
At a glance, in MARKET INTEL you can…

- Browse & search ~20 real M&A precedents; save your deal into or delete from the local database.
- Run a live keyless SEC EDGAR merger-filing scan; rank deals, benchmark your premium, and break the universe down by payment mix and industry.

7 SPECIALIZED — startups & sectors

RUN TO POPULATE · TAB 6

Value a private / startup target by blending five startup-valuation methods, and apply a sector KPI lens (SaaS / Pharma / Bank) that translates operating metrics into a suggested acquisition-multiple range. This page is self-contained — it does not read the shared deal.



SPECIALIZED — blended startup valuation (suggested value, range, confidence) with a per-method comparison bar chart; the sector-analysis lens sits below.

Rail cards

Card	Inputs & run button
Startup / Private Target	Stage, region, investment, exit-year revenue, exit multiple, years to exit; **Berkus scores** (5), **Scorecard multipliers** (7), **Risk factors** (base valuation + 6). **VALUE STARTUP (ALL METHODS)** → a blended suggested valuation, a low–high range, a confidence flag, and a per-method bar chart (Berkus, Scorecard, VC, First Chicago, Risk Factor).
Industry Lens	Sector selector (SaaS / Pharma / Bank) with sector-specific KPI inputs; **RUN SECTOR ANALYSIS** → a low/mid/high suggested multiple range, a KPI table, and benchmark insight bullets.

Note. The five methods are weighted Berkus 0.15 / Scorecard 0.20 / VC 0.30 / First Chicago 0.25 / Risk Factor 0.10. The First Chicago scenarios are hardcoded, and the industry lens exposes only the SaaS / Pharma / Bank branches (the engine also has marketplace / semiconductor / biotech / device / insurance / asset-management models, not reachable from this tab).

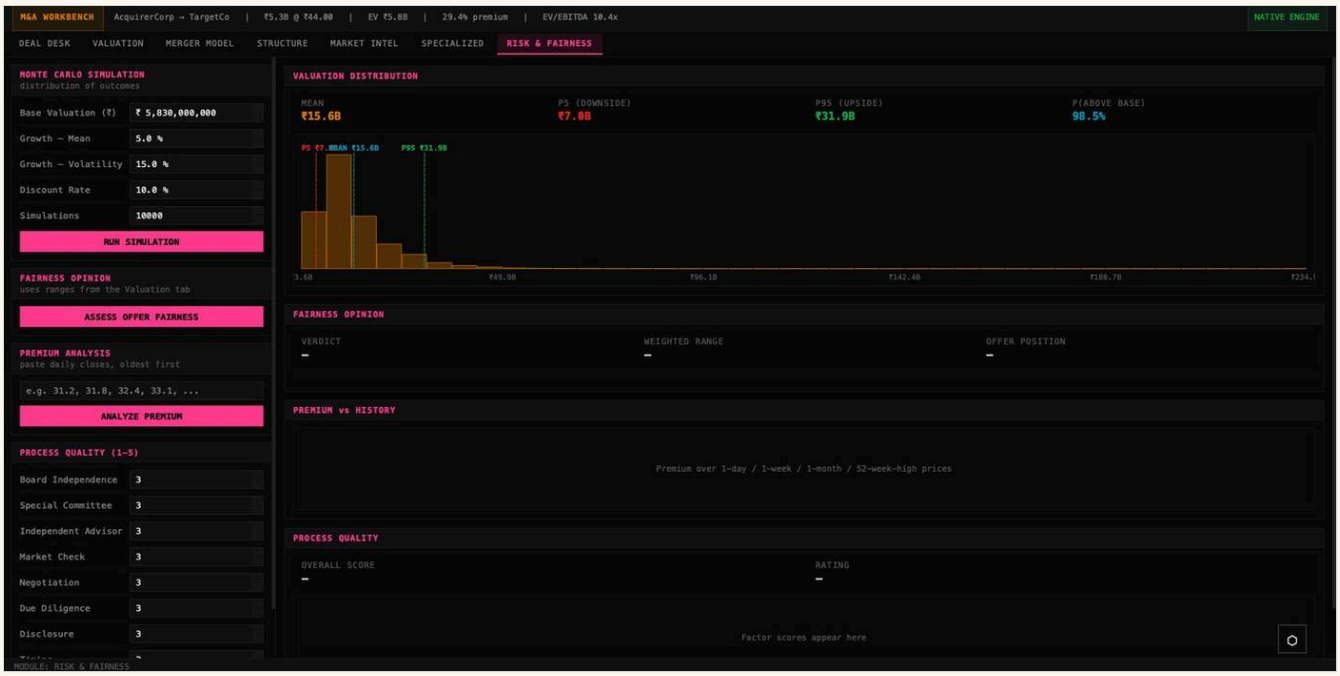
At a glance, in SPECIALIZED you can…

- Blend five startup methods into one private-company valuation with a range and confidence, and compare each method on a bar chart.
- Run a SaaS / Pharma / Bank KPI lens for a suggested acquisition-multiple range with benchmark insights.

8 RISK & FAIRNESS

RUN TO POPULATE · TAB 7

Stress-test the deal: a Monte-Carlo valuation distribution, a fairness opinion from the valuation ranges, the offer premium over a pasted price history, and a sale-process quality scorecard.



RISK & FAIRNESS — the Monte-Carlo valuation distribution (mean, P5/P95, probability above base) with its histogram; fairness, premium and process panels below.

Tool	Inputs & what it returns
Monte Carlo	Base valuation (auto-seeded from the deal's implied EV), growth mean/vol, discount rate, simulations. RUN SIMULATION → mean, P5/P95, probability above base, and an outcome histogram.
Fairness Opinion	ASSESS OFFER FAIRNESS → a FAIR/OUTSIDE-RANGE verdict, weighted valuation range, and offer position, using the ranges the Valuation tab produced + the Deal Desk offer.
Premium Analysis	Paste daily closes, ANALYZE PREMIUM → premium over 1-day / 1-week / 1-month / 52-week-high prices.
Process Quality	Eight governance scores (1–5), SCORE PROCESS → overall score, rating, factor bars, strengths/weaknesses, and a fairness-support flag.

Notes. The Monte-Carlo uses a fixed random seed (deterministic) and a hidden 15% margin assumption. The fairness opinion and premium analysis depend on cross-tab state — they need the Deal Desk offer and, for fairness, valuation ranges run on the Valuation tab first.

At a glance, in RISK & FAIRNESS you can…

- Run a Monte-Carlo valuation distribution and read downside/upside and probability of beating the base.
- Generate a fairness opinion, compute the offer premium over historical prices, and score the sale process across eight governance factors.

9 The seeded sample deal & database

REFERENCE

The numbers in this guide come from the bundled sample deal (Deal Desk ▸ LOAD SAMPLE DEAL) and the seeded deal database (Market Intel).

Sample deal

Target — TargetCo	rev $2.4B · EBITDA $560M · EBIT $430M · NI $285M · 120M sh @ $34 · debt $900M · cash $350M · β 1.25 · tax 21%
Acquirer — AcquirerCorp	rev $11.5B · EBITDA $2.9B · larger, listed
Terms	$44 offer (≈29% premium) · 60% cash / 40% stock · $60M rev + $140M cost synergies
Implied	equity ≈ $5.3B · EV ≈ $5.8B · EV/EBITDA 10.4× · ACCRETIVE

Seeded deal database (20 real precedents)

MSFT–Activision ($68.7B, 45.3%), Broadcom–VMware ($61B, 44%), Exxon–Pioneer ($59.5B), Chevron–Hess ($53B), Pfizer–Seagen ($43B), Cisco–Splunk ($28B), Synopsys–Ansys ($35B), Capital One–Discover ($35.3B), Mars–Kellanova ($35.9B), Nippon Steel–US Steel, Alphabet–Wiz (pending), plus terminated deals Adobe–Figma and Kroger–Albertsons — each with value, premium, EV/EBITDA, payment method, status. Feeds precedent transactions on the Valuation tab.

10 Analytics engine — capability catalog

A single in-process C++ engine (42 models) powers every module — no Python, deterministic. The one network call anywhere is the SEC EDGAR scan.

Area	Key models
Valuation	dcf, dcf_sensitivity, football_field, trading_comps, precedent_transactions
LBO	lbo_returns, lbo_model (tranches + cash sweep), lbo_debt_schedule, lbo_sensitivity
Merger	merger_model, accretion_dilution, pro_forma, sources_uses, contribution, revenue_synergies, cost_synergies, synergy_dcf, integration_costs
Structure	payment_structure, earnout, exchange_ratio, collar, cvr (milestone/regulatory/earnout/litigation)
Specialized	berkus, scorecard, vc_method, first_chicago, risk_factor, startup_comprehensive, fairness_opinion, premium_analysis, process_quality, tech/healthcare/finserv_metrics
Analytics	monte_carlo, regression, compare_deals, rank_deals, benchmark_premium, payment_structures_analysis, industry_deals_analysis

11 AI agent tools (47 MCP tools)

AGENTIC

The terminal's AI agent can drive this exact deal through 47 tools in the `ma-analytics` category — the same engine and deal store the UI uses.

- **Valuation (9):** ma_dcf, ma_dcf_sensitivity, ma_trading_comps, ma_precedent_transactions, ma_football_field, ma_lbo_returns/model/debt_schedule/sensitivity
- **Merger (9):** ma_merger_model, ma_accretion_dilution, ma_pro_forma, ma_sources_uses, ma_contribution_analysis, ma_revenue/cost_synergies, ma_synergies_dcf, ma_integration_costs
- **Structure (5):** ma_payment_structure, ma_earnout, ma_exchange_ratio, ma_collar_mechanism, ma_cvr
- **Deal database (5):** ma_get_deals, ma_search_deals, ma_create_deal, ma_update_deal, ma_delete_deal*
- **Startup (6) & Fairness (3):** ma_startup_berkus/vc/scorecard/first_chicago/risk_factor/comprehensive; ma_fairness_opinion, ma_premium_analysis, ma_process_quality
- **Industry (3), Analytics (2), Comparison (5):** tech/healthcare/financial_metrics; ma_monte_carlo, ma_regression; compare/rank/benchmark/payment/industry deals

* **ma_delete_deal** is the only destructive/confirm-gated tool. Note: the live EDGAR scan is UI-only — there is no MCP tool for it.

12 Quick reference & important notes

HOW DO I...

I want to…	Go to
Start quickly with a worked deal	DEAL DESK ▸ **LOAD SAMPLE DEAL**.
Enter my own deal	DEAL DESK ▸ fill Target / Acquirer / Terms ▸ **APPLY & ANALYZE DEAL**.
Value the target	VALUATION ▸ RUN DCF, then SAMPLE + RUN TRADING COMPS, then RUN PRECEDENTS — watch the football field build.
Check accretion/dilution	DEAL DESK (quick check) or MERGER MODEL ▸ BUILD MERGER MODEL (full).
See if the cash/stock mix is fundable	STRUCTURE ▸ ANALYZE PAYMENT MIX.
Benchmark my premium	MARKET INTEL ▸ BENCHMARK MY PREMIUM.
Find recent merger filings	MARKET INTEL ▸ SCAN MERGER FILINGS (live SEC EDGAR).
Value a startup / private target	SPECIALIZED ▸ VALUE STARTUP (ALL METHODS).
Test if the offer is fair	RISK & FAIRNESS ▸ run DCF/comps on Valuation first, then ASSESS OFFER FAIRNESS; RUN SIMULATION for the distribution.

Important notes for institutional users

- **Local & private.** The deal and the deal database live on your device; the analytics engine is native C++ and never sends your deal anywhere. The *only* network call is the Market Intel SEC EDGAR scan (public filings, keyless).
- **Deterministic math.** The Monte-Carlo uses a fixed seed; identical inputs always give identical output.
- **Set the deal up first.** Merger / Structure / Valuation / Risk runs read the shared Deal Desk deal — most error out until a deal exists (or you load the sample).
- **Destructive actions.** DEAL DESK ▸ CLEAR ALL DATA and MARKET INTEL ▸ DELETE SELECTED permanently remove data, including the 20 seeded precedent deals.
- **Known limitations to be aware of.** Market Intel's Payment-Mix Analysis mis-buckets seeded deals as "Mixed/0% cash" (a field-name mismatch — the Payment column itself is correct); the merger sensitivity heatmap's Y-axis is labelled "OFFER/SHARE" but shows purchase-price levels; the Structure earnout defaults sum to 105% and error until edited to 100%; the CVR tool is regulatory-only; the Specialized industry lens exposes only SaaS/Pharma/Bank.
- **Where results are live vs on-demand.** DEAL DESK snapshot and MARKET INTEL database are live; everything else computes when you press its run button.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Relationship Map

A complete, screen-by-screen walkthrough of the Relationship Map — one company at the centre of everything filed, held, covered and reported about it.

`>_ 14 node types · SEC Form 4 + 8-K · free, no credits`

Covers · **the search** · **every branch of the graph** · **filters** · **the canvas**
SEC EDGAR provenance · the data-quality score · where the numbers come from · what is not wired up

Demonstrated live on the Fincept Terminal Relationship Map tab.

Contents

17 SECTIONS

GETTING ORIENTED

01 What the Relationship Map is 3
02 Anatomy of the screen 4
03 The header bar 5
04 Finding a company 6

READING THE GRAPH

05 How the graph is arranged 7
06 The company card 8
07 Board and officers 9
08 Insiders — SEC Form 4 10
09 Metrics, governance and short interest 11
10 Corporate events — SEC 8-K 12
11 Peers — and what they really are 13
12 Holders and funds 14
13 Analysts 15

WORKING WITH IT

14 Filters 16
15 Moving around the canvas 17

REFERENCE

16 Where the data comes from, and what it costs 18
17 Quick reference and honest notes 19

About this guide. Every screenshot is a live capture of the Relationship Map in the shipping build. Three controls on this screen do not do what they appear to do; all three are named, with evidence, rather than described as though they work.

01 What the Relationship Map is

ORIENTATION · 01

The Relationship Map puts **one company at the centre of a single picture** and arranges around it everyone connected to it — the board, the insiders who filed with the SEC, the institutions and funds that hold it, the analysts who cover it, its peers, and the material events it has reported. It answers "who is around this company, and what have they been doing" in one screen rather than eight.

It is a **reading instrument, not a workbench**. Nothing here is edited, saved or traded; you type a ticker, the graph is built, and you look at it. That also makes it one of the few screens with nothing to break — and it is entirely free to use.

Fig 1 — The whole control surface: a search box, ANALYZE, and three buttons.

Where to find it

Route	How
Navigate menu	**Navigate ‣ Research & Intelligence ‣ Relationship Map**
Command bar	`relmap`, `relationships` or `map`
Main tab bar	**Not present.** This is not one of the fifteen top-level tabs

At a glance, in the Relationship Map you can…

- Map any listed company by ticker or name
- See real SEC Form 4 insider transactions
- See real SEC 8-K material events
- Read institutional and fund ownership
- Compare against peer companies
- See analyst ratings and price targets
- Check governance and short-interest flags
- Hide whole categories with filters
- Zoom, pan and rearrange the canvas
- Do all of it for free, with no credits

> **This screen spends nothing.** There is no metered action anywhere in it — no credit guard, no charge on ANALYZE, no charge on re-analysing. §16 explains where the data actually comes from and why that is the case.

02 Anatomy of the screen

A header, a legend, a canvas and a status bar. The filter panel slides in from the left when you ask for it.

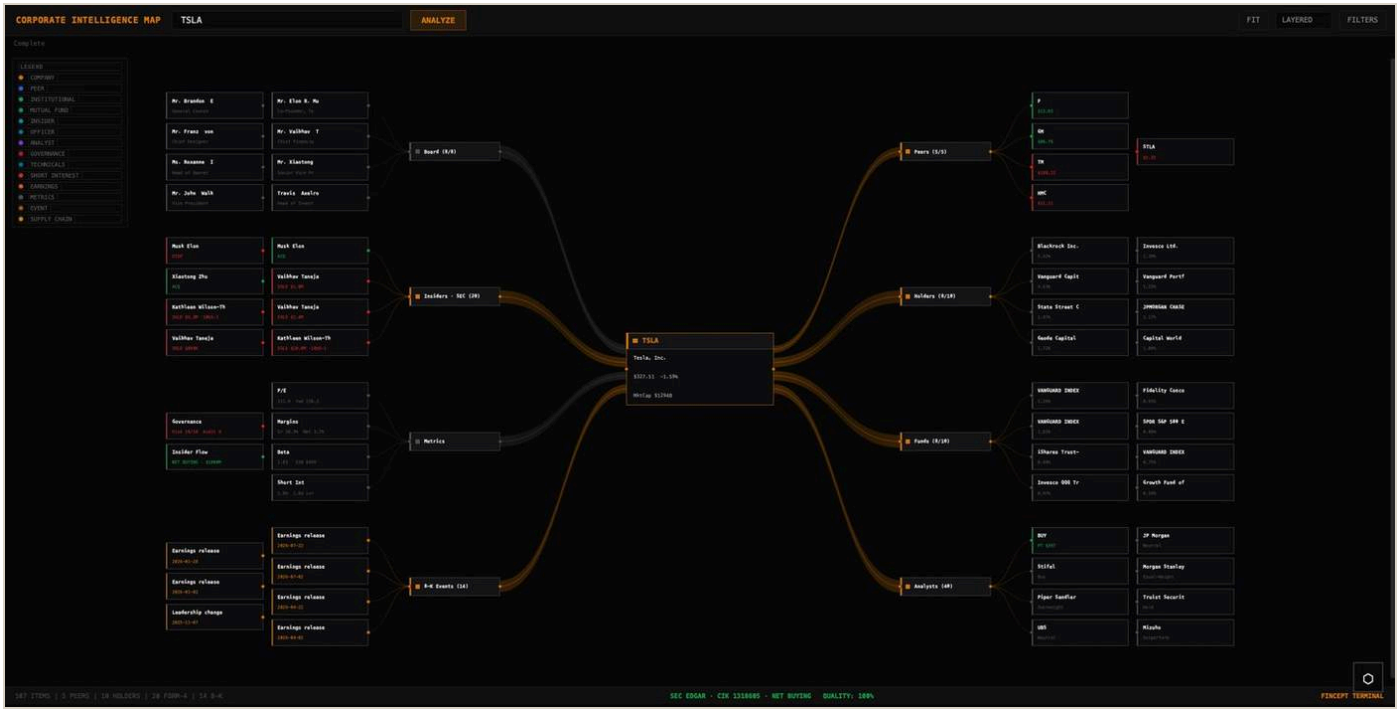


Fig 2 — Tesla mapped: 107 items, 5 peers, 10 holders, 20 Form 4 filings and 14 8-K events, with SEC EDGAR provenance confirmed in green along the status bar.

#	Region	What it is for
1	Header bar	**CORPORATE INTELLIGENCE MAP**, the search box, **ANALYZE**, and — at the right — **FIT**, the layout selector and **FILTERS**.
2	Progress line	Under the header. Shows a percentage and a message while a company is being assembled, then settles on *Complete*.
3	Legend	Top left, always visible. All fourteen node categories and their colours.
4	Canvas	The graph. The company card sits in the middle, hub nodes fan out left and right, and each hub carries its own leaves.
5	Status bar	Counts on the left; SEC provenance and the data-quality score in the centre, green when EDGAR answered.

> The **filter panel** (§14) and the **detail panel** are both built as part of this screen but neither is on view above: the filter panel opens from the FILTERS button, and the detail panel — see §17 — cannot be opened at all.

03 The header bar

Five controls, and one of them does nothing.

Fig 3 — The header with TSLA loaded.

Control	What it does
`Search box`	Type a ticker or a company name. A dropdown of matches appears as you type (§04). **Enter** runs the analysis.
`ANALYZE`	Builds the graph for whatever is in the box. Re-running the same ticker within ten minutes is served from cache and is instant.
`FIT`	Zooms and centres the whole graph in the window. The **Home** key does the same thing.
`Layout selector`	**LAYERED / RADIAL / FORCE. This control has no effect** — see the warning below.
`FILTERS`	Toggles the filter panel on the left (§14).

Fig 4 — Three layouts offered.

Fig 5 — The company card, which is what a click on the centre actually acts on.

> **The layout selector is not wired up.** Choosing RADIAL or FORCE changes the label on the combo and rebuilds the graph in exactly the same arrangement. In the shipping source the builder ignores the parameter outright — its signature reads `build_graph(data, filters, LayoutMode /*layout*/)` — and the routine that would position nodes differently, `apply_layout()`, has an empty body and is never called. Verified on screen as well as in code: the combo read FORCE and the picture did not move. Treat the graph as having exactly one layout.

> **Clicking the company card in the middle does something invisible but useful:** it loads that ticker into the **Equity Research** tab. Nothing happens here to tell you so — switch to Equity Research and the symbol will be waiting.

04 Finding a company

SEARCH · 04

Type at least a couple of characters and a dropdown of matching instruments appears, drawn from the terminal's own instrument search. It waits about a third of a second after your last keystroke before querying, and shows up to ten matches.

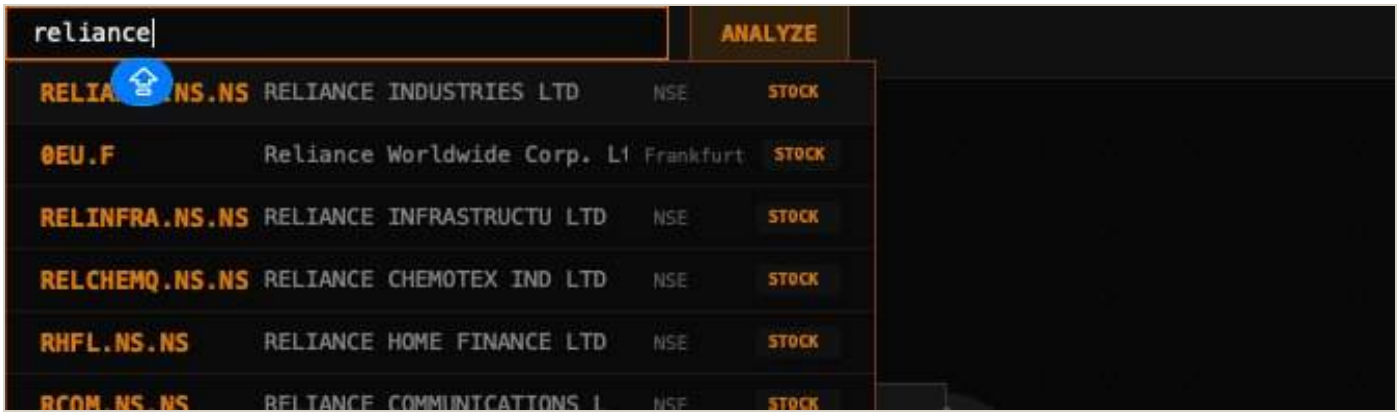


Fig 6 — Searching for *reliance*. Each row is symbol, company name, exchange and instrument type.

Column	What it is
`Symbol`	The ticker the analysis will actually be run on, in amber.
`Name`	The registered company name.
`Exchange`	NSE, Frankfurt, NASDAQ and so on.
`Type`	`STOCK`, `ETF`, and so on.

Picking a row fills the search box with the symbol in the first column and starts the analysis immediately — you do not need to press ANALYZE as well.

> **Non-US symbols get their exchange suffix added twice.** The screen converts a search hit into a data-provider symbol by appending a suffix for the exchange — `.NS` for the NSE, `.L` for London, `.T` for Tokyo and so on — without first checking whether the symbol already carries one. The instrument search already returns `RELIANCE.NS`, so the row reads — and submits — **`RELIANCE.NS.NS`**. You can see it in Fig 6, in the first column of every NSE row. §11 shows what that produces.

> **The workaround is to type the symbol yourself.** Typing `RELIANCE.NS` into the box and pressing Enter — without touching the dropdown — sends exactly what you typed. Only the dropdown adds the second suffix.

05 How the graph is arranged

One fixed arrangement, and it is worth learning because it never changes: the company in the centre, **hub** nodes fanning out left and right, and each hub's **leaves** beyond it.

Side	Hubs	What they are
`Left`	Board · Insiders · Metrics · 8-K Events	What the company and its people have done — governance, filings, fundamentals, disclosures.
`Right`	Peers · Holders · Funds · Analysts	How the outside world relates to it — comparison, ownership, coverage.

Reading a hub label

Most hubs are labelled `Name (shown/total)` — **Holders (8/10)** means ten institutional holders were found and the eight largest are drawn. Each branch has its own cap:

Hub	Drawn at most	Label form
`Peers`	12	shown / total
`Board`	10	shown / total
`Holders`	8	shown / total
`Funds`	8	shown / total
`Insiders · SEC`	8	**total only**
`Analysts`	8	**total only**
`8-K Events`	7	**total only**
`Supply Chain`	6	count drawn

> **Three hub labels are not node counts. Analysts (40)** means forty analysts cover the stock, not that forty are drawn — eight are. **Insiders · SEC (20)** is twenty Form 4 filings with eight shown, and **8-K Events (14)** is fourteen events with seven shown. The other hubs use the `shown/total` form and are honest about the gap; these three read as though the graph is complete when it is a sample of the largest or most recent.

> The line colours are categorical, not quantitative: green for ownership, blue for peers, yellow for supply chain, red for events, grey for internal structure. Thickness carries a weight but is not labelled anywhere — do not read size off the graph, read it off the leaves.

06 The company card

The centre of the picture, and the only node that cannot be dragged.

Fig 7 — Ticker, registered name, last price with the day's change, and market capitalisation.

Line	What it is
`Ticker`	The symbol the graph was built for — in amber, matching the COMPANY legend colour.
`Name`	The registered company name as the data provider has it.
`Price`	Last price and the day's percentage change, coloured up or down.
`MktCap`	Market capitalisation, abbreviated.

The card is a **snapshot at the moment you pressed ANALYZE**, not a live quote. It does not tick. Re-running the analysis refreshes it — though within ten minutes you will get the cached copy back unchanged, which is exactly what you want while you are reading the rest of the graph.

> **Clicking the card sends the ticker to Equity Research.** This is the one piece of cross-tab wiring on the screen, and it gives no feedback here — no highlight, no message, no tab change. If you want the full research workbench for the company you are looking at, click the middle and then switch to Equity Research yourself.

> **An unresolved ticker still draws a card.** If the data provider returns nothing for the symbol, the centre node is drawn anyway with the raw symbol as its name, `0.00%` as its change and a zero market cap. It is not obviously an error state — §11 shows a live example, and the status bar's quality score is the reliable tell.

07 Board and officers

The company's named officers, up to ten of them, each with the title the provider carries.

Fig 8 — **Board (8/8)** — eight officers found, all eight drawn.

Element	What it is
`Name`	The officer's name as filed. Long names are clipped to fit the node.
`Title`	Their role — Chief Financial Officer, General Counsel, and so on.
`Colour`	Dark cyan, the OFFICER category.

This branch answers "who runs it" and little more — there is no compensation, no tenure and no biography on the node, even though the underlying record carries pay and year of birth for some officers. What is drawn is the name and the title.

> **Officers are not the same people as insiders.** The Board branch is the company's own list of its officers. The Insiders branch (§08) is whoever actually filed a Form 4 with the SEC in the recent window — usually overlapping, but neither is a subset of the other. A director who has not traded appears here and not there; a ten-percent shareholder who traded appears there and not here.

> **Titles are clipped, not wrapped.** A long title such as *Chief Financial Officer & Head of Investor Relations* is cut at the node's width with no ellipsis, so it can read as a shorter, different title. Where a distinction matters, confirm it from the filings rather than from the node.

08 Insiders — SEC Form 4

This is the branch that makes the screen worth opening: **real Form 4 filings pulled from SEC EDGAR**, not a summary statistic. Each leaf is one insider's filing.

Fig 9 — **Insiders · SEC (20)**: twenty filings found, the eight most recent drawn. Note `10b5-1` on two of them.

On the leaf	Meaning
`Name`	The reporting person.
`Action`	`SOLD` / `ACQ` / `DISP` — what the filing did, coloured green for acquisition and red for disposal.
`Value`	The net dollar value of the transaction, abbreviated.
`10b5-1`	The trade was made under a pre-arranged plan.

> **Why `10b5-1` matters.** A 10b5-1 plan is scheduled in advance, so the insider was not deciding to sell on the day. A large flagged sale carries far less signal than the same sale unflagged — which is why the flag is on the node.

The signal line

The status bar and the **Insider Flow** node (§09) both carry a one-word verdict computed from **open-market transactions only** — code P purchases against code S sales. Grants, option exercises and tax withholding are excluded, because they are not decisions to buy or sell:

Verdict	Condition
`INSIDER BUYING`	Open-market buys, and no open-market sales at all
`INSIDER SELLING`	Open-market sales, and no open-market buys at all
`NET BUYING`	Both happened; buys were larger by value
`NET SELLING`	Both happened; sales were larger by value
`ROUTINE (grants/tax)`	Filings exist, but none of them were open-market

> **The flow node shows one side of the trade, never the net.** A **NET BUYING** verdict displays the open-market *buy* value; **NET SELLING** displays the *sell* value. So *NET BUYING · $1000M* tells you buys totalled a billion — it does not tell you sales were $900M of that. To size the net you must read the leaves.

09 Metrics, governance and short interest

A compact panel of the numbers you would otherwise open three screens to gather.

Fig 10 — The Metrics branch: valuation, margins, governance, insider flow, beta and short interest.

Leaf	What it carries
`P/E`	Trailing and forward price/earnings.
`Margins`	Gross and net margin.
`Beta`	Beta, and the 52-week range.
`Short Int`	Short interest as a percentage of float, and days to cover.
`Governance`	The governance risk scores — see below.
`Insider Flow`	The open-market verdict from §08.

Governance risk runs the wrong way round from most scores. The numbers are ISS-style risk grades from **1 to 10 where 10 is the worst**, not a rating out of ten. A node reading *Risk 10/10 Audit 4* is reporting maximum overall governance risk with a middling audit score — not a top mark. The screen does not label the direction anywhere, and it is the single easiest thing on this tab to misread.

Short interest is as of the exchange's last settlement, which for most US names means it is up to two weeks old. Days-to-cover is short interest divided by average daily volume, so it moves when volume moves even if the short position has not changed.

10 Corporate events — SEC 8-K

Material events the company has actually reported to the SEC on Form 8-K, categorised and dated. Up to seven are drawn, most recent first.

Fig 11 — **8-K Events (14)**: earnings releases and a leadership change, each with its filing date.

Element	What it is
`Label`	A plain-English reading of the 8-K's item number — *Earnings release*, *Leadership change*, *Material agreement*.
`Date`	The filing date, `YYYY-MM-DD`.
`Colour`	Varies by category, so a run of the same kind of event is visible at a glance.

The value here is that these are **filings, not news**. An 8-K is a legal disclosure with an item number attached; the terminal maps that number to a readable label rather than paraphrasing a headline. Nothing in this branch came from a news feed, so nothing in it is speculation.

> **The hub says 8-K Events only when EDGAR answered.** If the SEC lookup did not run — a non-US listing, or a company with no CIK — the hub is labelled plainly **Events** with a `shown/total` count, and whatever it contains came from the general provider rather than from filings. The word *8-K* in the label is the guarantee of provenance.

> **Only the seven most recent are drawn**, and the hub label counts all of them, so a company that files often shows seven leaves under a much larger number. There is no way to page through the rest from this screen.

11 Peers — and what they really are

The most useful branch when it works, and the most misleading when it does not — so it is worth knowing exactly how the list is chosen.

Fig 12 — Tesla's peers: Ford, GM, Toyota, Honda and Stellantis. Correct, and chosen from a fixed *Auto Manufacturers* list.

Peers are **not** computed from the company's actual competitors, its filings or its revenue mix. The terminal carries a hard-coded map of **nineteen industry names to lists of US tickers** — Technology, Semiconductors, Banks, Oil & Gas, Airlines, and so on. The company's industry string is matched against those nineteen keys, and the first one that matches supplies the peer list.

> **If nothing matches, the peers become Apple, Microsoft, Google, Amazon and Meta.** That is a literal fallback in the shipping code, and it is applied silently — the hub still reads **Peers (5/5)** as though five genuine peers were found.

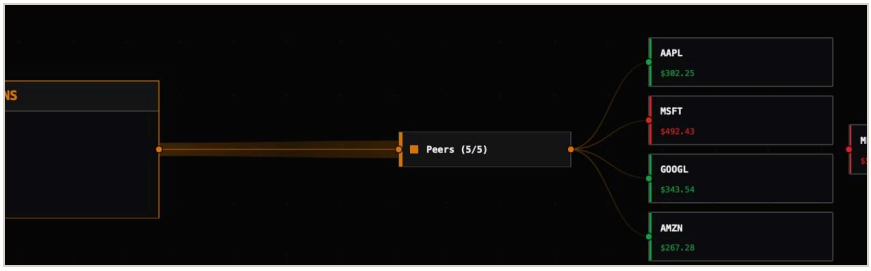


Fig 13 — Reliance Industries, submitted through the dropdown as `RELIANCE.NS.NS` (§04). The symbol did not resolve, so the company card is blank — and the five US mega-caps are presented as its peers.

The status bar is the tell

Status bar reads	Tesla (Fig 12)	Reliance (Fig 13)
`Items / peers / holders`	107 · 5 · 10	12 · 5 · **0**
`SEC provenance`	**SEC EDGAR · CIK 1318605**, in green	**absent**
`Quality`	**100%**	**55%**

> **The practical rule.** Peers are dependable for large US companies in one of the nineteen mapped industries, and unreliable everywhere else. Before reading anything into a peer set, check two things in the status bar: that the company card resolved to a real name and price, and that the quality score is high. A graph can look complete and be a default.

12 Holders and funds

Two ownership branches, split by holder type. Both are drawn largest-first and both cap at eight leaves.

Fig 15 — **Holders (8/10)**: the managers.

Fig 16 — **Funds (8/10)**: the vehicles.

Element	What it is
`Name`	The holder or fund, clipped to the node width.
`Percentage`	Percentage of shares outstanding held.
`Colour`	Green for INSTITUTIONAL, emerald for MUTUAL FUND — close enough that the two branches read as one family, which is the intent.

The distinction matters: an **institutional** holder is the manager (BlackRock, Vanguard, State Street), while a **fund** is a specific vehicle (Vanguard Index Fund, SPDR S&P 500 ETF). The same shares are frequently counted in both branches — a Vanguard index fund's holding is also part of Vanguard's institutional position. **Do not add the two columns together.**

> **Ownership percentages are as of the last quarterly filing**, which for 13F filers means they can be up to about four and a half months stale. They are a picture of the register at a past date, not of who holds the stock today, and the screen does not show the as-of date anywhere.

> These two branches share a single filter checkbox between them — see §14, which is a defect rather than a design.

13 Analysts

Sell-side coverage: the consensus, then named firms with their current ratings.

Fig 17 — **Analysts (40)**. The first leaf is the consensus and price target; the rest are individual firms.

Leaf	What it carries
`First leaf`	The consensus recommendation — `BUY`, `HOLD` — and the mean price target, e.g. `PT $397`. Coloured by direction.
`Firm leaves`	The firm's name and its current rating word — Neutral, Overweight, Buy, Equal-Weight, Outperform, Hold.

> **The number in the hub is the coverage count, not the leaf count. Analysts (40)** means forty analysts contribute to the consensus; **eight** firms are drawn. The eight are not necessarily the most influential — they are simply the first eight in the provider's order.

Ratings words are **not normalised across firms**. One house's *Overweight* is another's *Buy* and a third's *Outperform*, and the graph reproduces each firm's own vocabulary without translating it. The consensus leaf is the only place where a single scale is applied.

> **Price targets are carried only on the consensus leaf**, not on the individual firms, even though the underlying record holds a per-firm target and the firm's previous target. If you need the spread of targets rather than the mean, this screen will not give it to you — Equity Research will, and clicking the company card sends the ticker straight there.

14 Filters

FILTERS opens a panel of nine checkboxes down the left, each colour-keyed to the category it controls. Unticking one removes that whole branch and redraws the graph.

Fig 18 — All nine on.

Fig 19 — **Mutual Funds** alone unticked. **Institutional** is still ticked.

> **Institutional and Mutual Funds are two checkboxes wired to one setting.** Unticking either hides **both** branches, and the other checkbox goes on showing a tick it is not honouring. In the shipping source both rows write to the same `show_institutional` flag. Verified live: unticking Mutual Funds alone removed the Holders branch as well.

Fig 20 — The result of Fig 19: Holders and Funds have both gone, leaving only Peers and Analysts on the right — while the Institutional box still reads as enabled.

> **The other seven filters behave correctly**, and filtering is purely visual — it hides branches, it does not re-fetch anything, so toggling is instant and costs nothing. Filter settings are remembered for next time you open the tab.

15 Moving around the canvas

Standard graph-canvas handling, with one pleasant extra: you can rearrange the picture by hand.

Action	How	Notes
`Zoom`	Mouse wheel / trackpad scroll	Clamped at both ends so the graph cannot be lost off-scale.
`Pan`	Click and drag the background	The canvas is in hand-drag mode by default — no modifier key needed.
`Fit everything`	**FIT** button, or the **Home** key	Rescales and recentres the whole graph with a margin.
`Move a node`	Drag any hub or leaf	Nodes are individually movable, and their edges follow.
`Move the company card`	**Not possible**	The centre node is deliberately fixed — dragging it does nothing.

Hand-arranging is genuinely useful on a dense company: drag two branches you want to compare next to each other, then read them side by side. It is the practical answer to the layout selector not working (§03).

> **Nothing you rearrange is kept.** Node positions are not part of the saved state, so re-running the analysis, toggling a filter, or leaving and returning to the tab rebuilds the graph in its default arrangement and discards your layout. Finish reading before you touch a filter.

> **What is remembered** between sessions: the ticker that was loaded, the filter settings, and the layout-selector position. The graph is rebuilt from the ticker on return — from cache if it is under ten minutes old, otherwise fetched again.

16 Where the data comes from, and what it costs

Unusually for this terminal, the Relationship Map does not go through the platform at all.

Stage	What happens
`1. You press ANALYZE`	The screen asks its service for the ticker.
`2. Cache check`	If the same ticker was built in the last **ten minutes**, the stored result is reused and the graph appears instantly.
`3. Local script`	Otherwise a Python script bundled with the application runs **on your machine**, calling the market-data provider for company, ownership, peer and analyst data.
`4. SEC EDGAR`	For US-listed companies the script then queries EDGAR directly for Form 4 insider transactions and 8-K events.
`5. Graph`	The combined result is scored for completeness and drawn.

> **This is why it is free.** No part of the fetch touches a metered platform endpoint, so there is no charge for an analysis, no allowance to exhaust, and no difference between plans in what this screen will show you. A `relmap_hops` limit does exist in your plan's entitlement — 1, 3 or unlimited by tier — but nothing on this screen consults it: the graph is built in a single pass rather than expanded outward hop by hop, so there are no hops to count.

The quality score

The percentage in the status bar is the proportion of the expected data that actually arrived. **100%** means every branch was populated; the 55% in Fig 14 meant no company details, no holders and no filings. Treat anything under roughly 80% as a partial picture and check which branches are missing before drawing conclusions from what is there.

> **The item count is approximate by construction.** The *N ITEMS* figure on the left of the status bar counts every drawable object on the canvas — nodes, connecting lines and text labels alike — not entities. Tesla's *107 ITEMS* is not 107 people and institutions. The counts that follow it (peers, holders, Form 4, 8-K) are real entity counts and are the ones to quote.

17 Quick reference and honest notes

How do I...

Goal	Do this
`Map a company`	Type the ticker, press Enter — or pick a dropdown row
`Map a non-US company`	Type the symbol **with its suffix** yourself; avoid the dropdown (§04)
`See the whole graph`	**FIT**, or the **Home** key
`Compare two branches`	Drag them next to each other — but do not touch a filter afterwards
`Hide a category`	**FILTERS**, untick it
`Get the full research view`	Click the company card, then switch to Equity Research
`Refresh the data`	Press ANALYZE again — after ten minutes, or it serves the cache

Honest notes on this build

- **The detail panel cannot be opened.** The screen builds a node-detail panel — title, category, a properties area and a close button — but graph nodes are never made selectable and the handler that would populate it is never connected to anything. There is no click, anywhere, that shows it. Node information is limited to what is printed on the node.
- **The layout selector does nothing** — the builder ignores the parameter and the positioning routine is an empty stub (§03).
- **Institutional and Mutual Funds share one filter flag**, so either checkbox hides both branches while the other still shows a tick (§14).
- **The dropdown doubles the exchange suffix** on symbols that already carry one, which silently degrades the result rather than failing (§04, §11).
- **Unmatched industries fall back to AAPL/MSFT/GOOGL/AMZN/META as "peers"**, with no indication that a default was used (§11).
- **Governance risk is 1–10 where 10 is worst**, and the screen never says so (§09).
- **Three hub labels show totals, not what is drawn** — Analysts, Insiders · SEC and 8-K Events (§05).
- **N ITEMS counts drawable objects**, not entities (§16).
- **Ownership is quarterly and undated on screen**; short interest is up to a fortnight old (§09, §12).
- **Hand-arranged node positions are discarded** on any rebuild (§15).
- **No export.** There is no CSV, no image save and no print — the graph cannot leave the screen except as a screenshot.
- **No agent tools.** Nothing in this tab is exposed to the AI assistant or to workflows.

> Everything above was read from the shipping source and then confirmed on screen on the v5.0.0 build. The three wiring defects are reported because a guide that described the detail panel and the layout selector as working would send you looking for behaviour that is not there.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Surface Analytics

A complete, screen-by-screen walkthrough of the Surface Analytics tab — a 3D visual-analytics workbench spanning 35 quantitative surfaces across seven asset classes, driven by live Databento data or built-in synthetic demonstrations.

`>_ seven asset classes · one 3D engine`

Covers · **the workbench anatomy** · **35 surfaces** · **7 categories** **the control panel** · **3D / Table / Line views** · the data inspector · Databento & CSV · 15 agent tools

Demonstrated live on the Fincept Terminal Surface Analytics tab.

Contents

8 SECTIONS

GETTING ORIENTED

Overview — what the Surface Analytics tab is 3

1 Anatomy of the workbench 4

THE SURFACES & CONTROLS

2 The 7 categories & 35 surfaces — the catalog 5

3 The Control Panel — inputs & parameters 12

4 The three views — 3D, Table, Line 13

5 The Data Inspector — raw data, lineage, status 15

DATA & REFERENCE

6 Data & connectivity — the four tiers, Databento, CSV 15

7 AI agent tools (15 MCP tools) 17

8 Quick reference & important notes 17

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Where Databento is not connected the surfaces render from built-in synthetic data; control labels are quoted exactly as they appear on screen.

What the Surface Analytics tab is

OVERVIEW

Surface Analytics is a **3D visual-analytics workbench**. It renders **35 quantitative surfaces** — volatility surfaces, greek surfaces, yield curves, credit-spread grids, futures term structures, correlation matrices, VaR grids and more — organised into 7 asset-class categories. Each surface can be viewed as a rotatable 3D mesh, a colour-coded heatmap table, or (for term-structure surfaces) a line chart.

Data comes from one of two places. Most equity-derivative, commodity and risk surfaces are backed by Databento institutional market data (options, futures and equities), computed in a bundled Python provider. All fixed-income, FX, credit and macro surfaces — and every surface before you connect Databento — render from built-in synthetic demonstration data, so the whole tab is fully explorable offline. The same surfaces can also be driven by the terminal's AI agent through 15 dedicated tools.

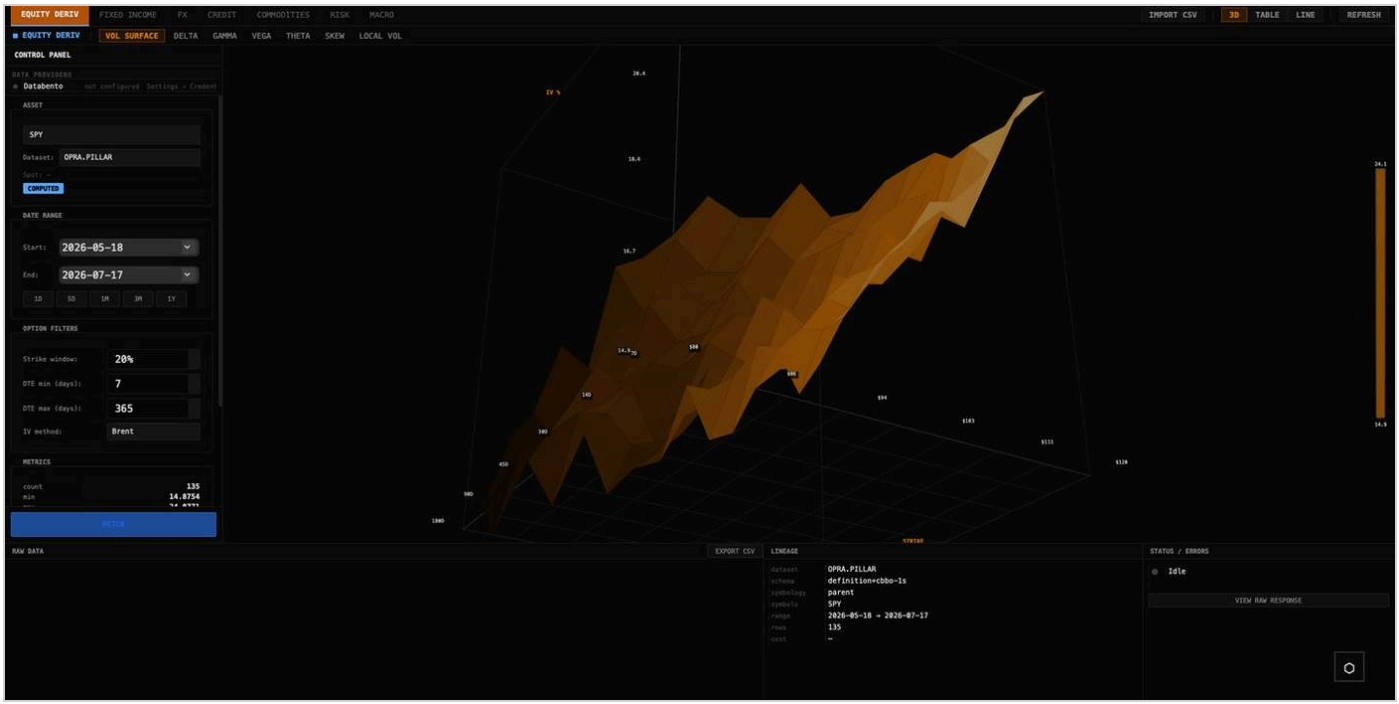
AT A GLANCE, IN SURFACE ANALYTICS YOU CAN...

- Browse **35 surfaces** across 7 categories from a two-row picker (category tabs → surface chips).
- View any surface in **3D** (drag to rotate, scroll to zoom), as a **heatmap table**, or — for term structures — as a **line chart**.
- Set the underlying symbol, date range, option strike/expiry filters and IV method, and build a correlation basket — with the control panel showing only the inputs each surface needs.
- Fetch live **Databento** data (options, futures, equities) for the market-sourced surfaces, or explore synthetic versions of all 35 offline.
- Read live per-surface statistics (count, min/max, mean, median, std, skew, kurtosis) and full data lineage (dataset, schema, symbology, date range, row count, cost).
- Import your own surface from CSV, export any raw dataset to CSV, and inspect the raw provider response.
- Link the tab to an equity symbol group so the surfaces re-centre on a shared symbol automatically.

1 Anatomy of the workbench

ORIENTATION

Open the tab from the command bar (type `surface`) or **Navigate ‣ Research & Intelligence ‣ Surface Analytics**. The screen is a single view with a two-row picker on top, a control-panel rail on the left, the main view area, and a data inspector strip along the bottom.



Surface Analytics on the Equity-Deriv vol surface. ① Category bar ② Surface bar ③ View toggles + Import/Refresh ④ Control panel (rail) ⑤ 3D view ⑥ Data inspector (raw data · lineage · status).

Region	What it shows / does
① **Category bar**	Seven asset-class tabs — **EQUITY DERIV · FIXED INCOME · FX · CREDIT · COMMODITIES · RISK · MACRO** — each with its own accent colour. Clicking one rebuilds the surface bar and jumps to that category's first surface.
② **Surface bar**	The surfaces available in the active category, shown as clickable chips (e.g. `VOL SURFACE · DELTA · GAMMA · VEGA · THETA · SKEW · LOCAL VOL`), prefixed with a colour-coded category label.
③ **View toggles & tools** (top-right)	**IMPORT CSV** · a **3D / TABLE / LINE** view switch · **REFRESH** (re-seeds the synthetic surfaces).
④ **Control panel** (rail)	The inputs for the active surface — symbol, dataset, date range, option filters, correlation basket — plus a live provider-status pill, a metrics readout, a lineage line, and the **FETCH** button. It shows only the sections the selected surface needs.
⑤ **Main view**	The surface rendered as a 3D mesh, heatmap table, or line chart — whichever view is selected (and supported).
⑥ **Data inspector** (bottom)	Three columns: **RAW DATA** (tabbed dataset snapshots + Export CSV), **LINEAGE** (dataset/schema/symbology/symbols/range/rows/cost), and **STATUS / ERRORS** (with a "View raw response" modal).

On first open the tab lands on Equity Deriv → **Vol Surface** → 3D, already rendered from synthetic data — it is never empty. **REFRESH** reshuffles the synthetic surfaces; it does not fetch live data (that is the control panel's FETCH).

2 The 7 categories & 35 surfaces

CATALOG

The heart of the tab. Every surface renders three grid quantities: two axes (X, Y) and a value (Z — the height and colour). The **data tier** tells you where each surface's numbers come from:

LIVE direct Databento fetch · COMPUTED derived from a Databento feed (Black-Scholes, realised vol, …) · EQUITIES built from Databento equities OHLCV · DEMO synthetic — no live source, always available offline.

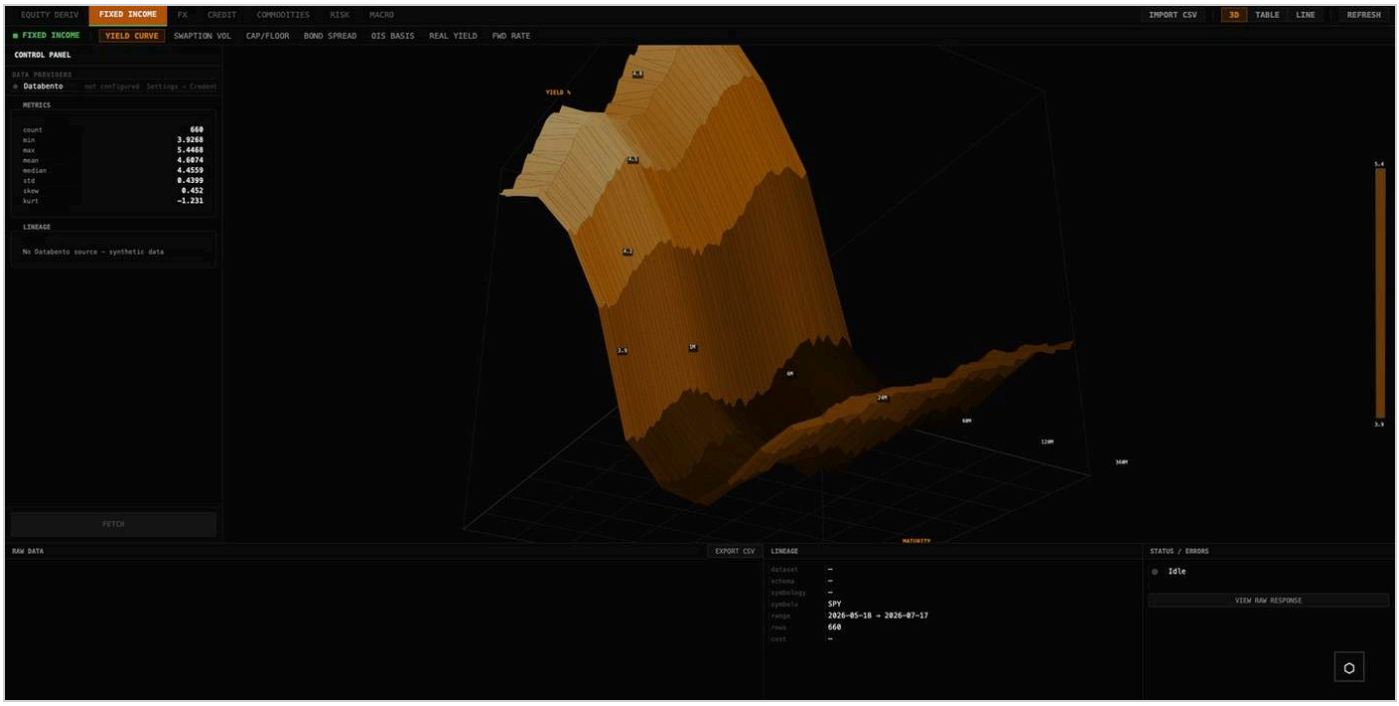
Of the 35 surfaces: **2 LIVE, 11 COMPUTED, 7 EQUITIES, 15 DEMO**. All support 3D and Table; only seven support the Line view (marked **+L**).

Equity Derivatives — option-implied surfaces, Black-Scholes from OPRA options + CBBO quotes

Surface	What it shows	Axes (X · Y · value)	Tier
VOL SURFACE	Implied volatility across strike & maturity	Strike · DTE · IV %	COMPUTED
DELTA / GAMMA / VEGA / THETA	The option greeks across strike & tenor (Theta diverging)	Strike · DTE · greek	COMPUTED
SKEW	IV skew across delta buckets & tenors	Delta · DTE · skew %	COMPUTED
LOCAL VOL	Dupire local vol from the implied-vol surface	Strike · DTE · LV %	COMPUTED

Fixed Income — rates & curve surfaces (synthetic)

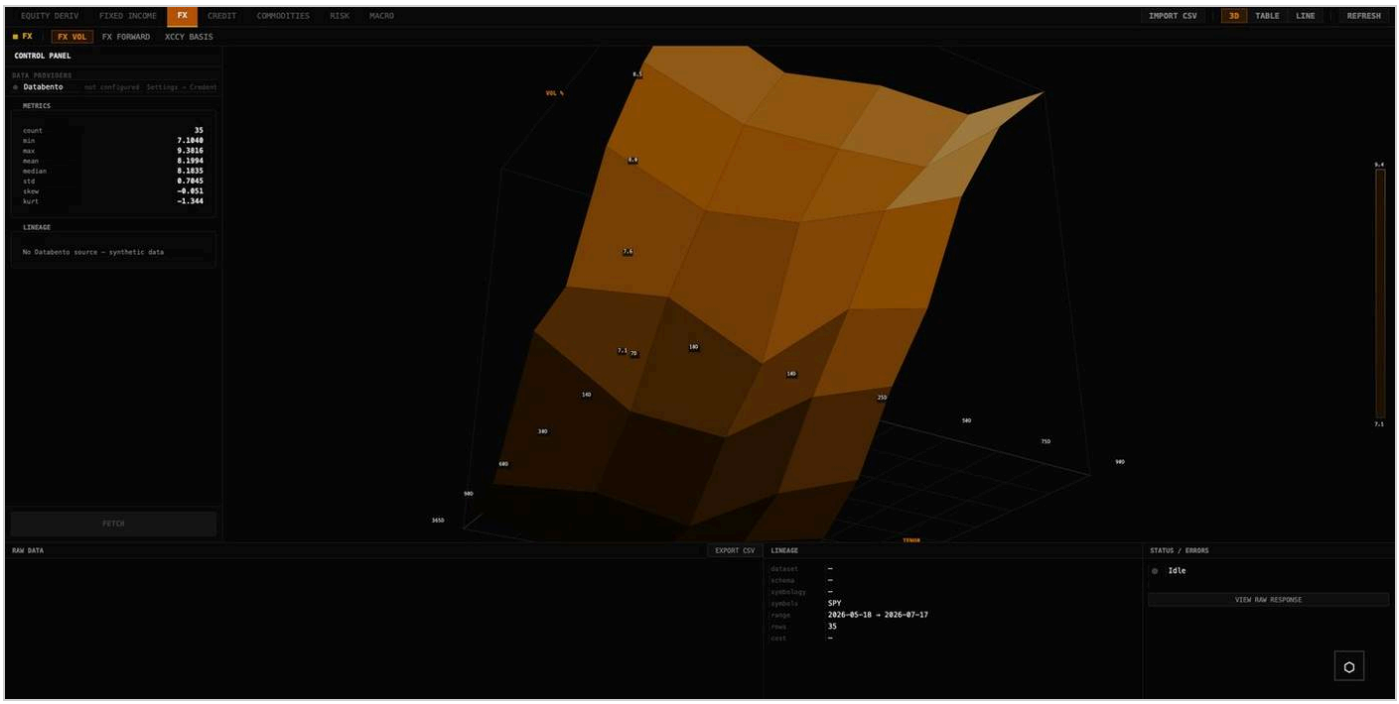
Surface	What it shows	Axes (X · Y · value)	Tier
YIELD CURVE **+L**	Treasury curve evolving over time	Maturity · Time · yield %	DEMO
SWAPTION VOL	Normal (bp) swaption vol cube	Swap tenor · Option expiry · vol bp	DEMO
CAP/FLOOR	Cap/floor vol by maturity & strike rate	Maturity · Strike · vol bp	DEMO
BOND SPREAD	Credit spread over benchmark by rating & maturity	Maturity · Rating · spread bp	DEMO
OIS BASIS	OIS basis by tenor over time (diverging)	Tenor · Time · basis bp	DEMO
REAL YIELD	Inflation-adjusted yield by maturity over time	Maturity · Time · real yld %	DEMO
FWD RATE	Forward-rate grid, start tenor × forward period	Start tenor · Fwd period · fwd %	DEMO



Fixed Income → Yield Curve in 3D. All fixed-income surfaces are synthetic (DEMO): the control panel collapses to just metrics + lineage, and FETCH is disabled.

FX — vol & forward/basis surfaces (synthetic)

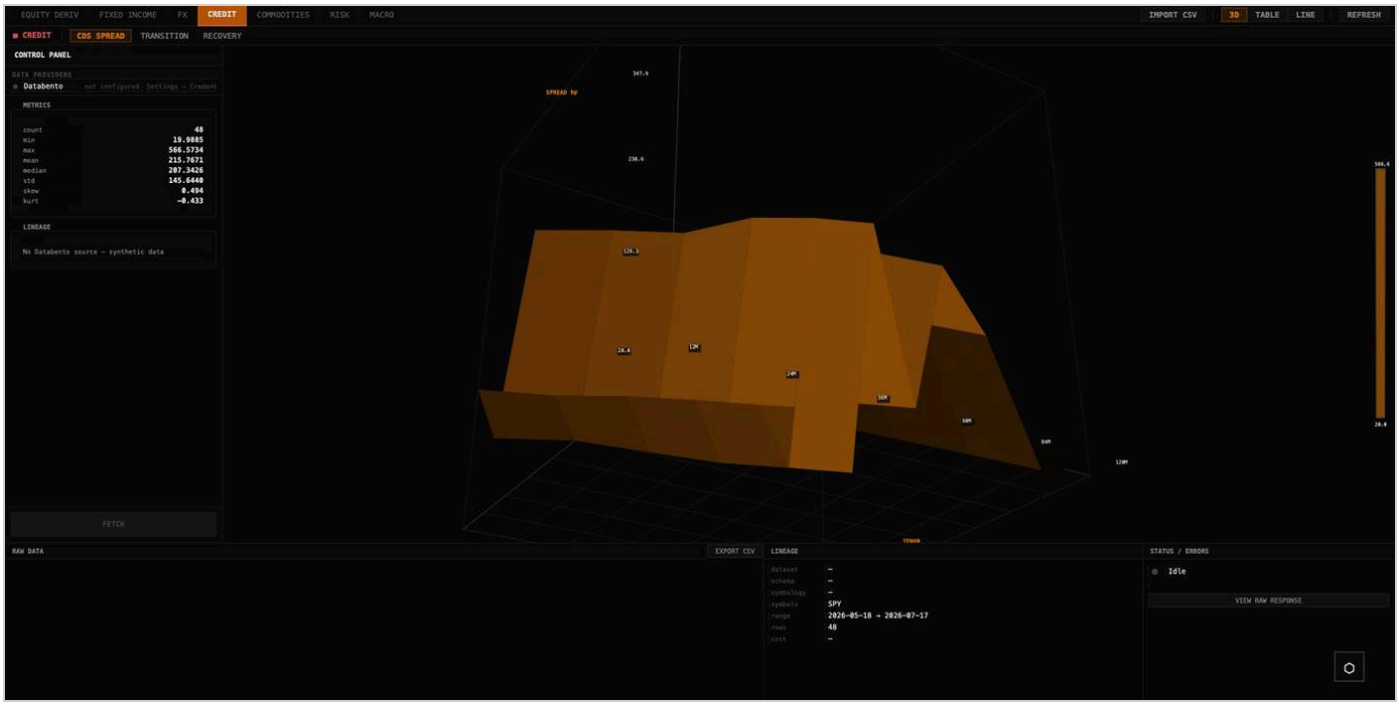
Surface	What it shows	Axes (X · Y · value)	Tier
FX VOL	FX implied-vol smile across delta & tenor	Tenor · Delta · vol %	DEMO
FX FORWARD **+L**	FX forward points across currency pairs & tenor	Tenor · Pair · fwd pts	DEMO
XCCY BASIS	Cross-currency basis by pair & tenor (diverging)	Tenor · Pair · basis bp	DEMO



FX → FX Vol surface (synthetic).

Credit — credit-risk surfaces (synthetic)

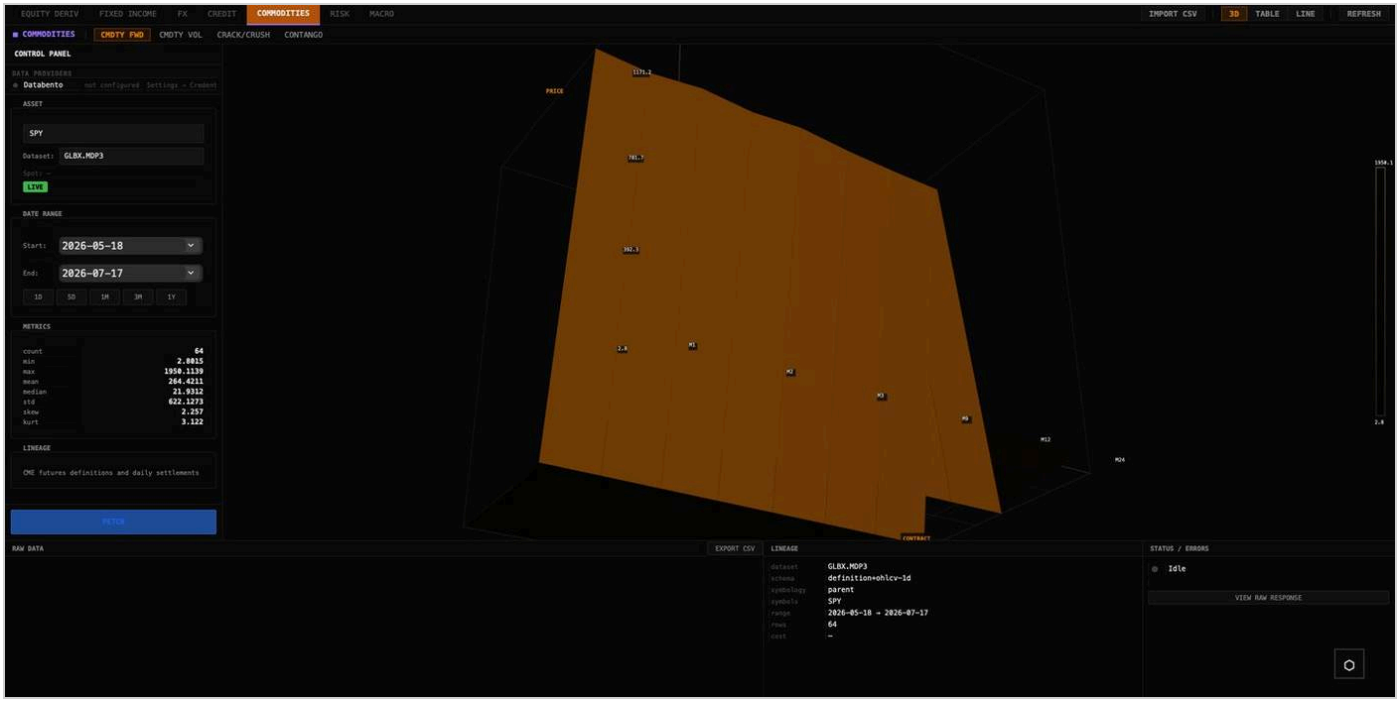
Surface	What it shows	Axes (X · Y · value)	Tier
CDS SPREAD	CDS term structure by reference entity	Tenor · Entity · spread bp	DEMO
TRANSITION	Annual rating-migration probability matrix	To rating · From rating · prob %	DEMO
RECOVERY	Recovery rates by sector & debt seniority	Sector · Seniority · recovery %	DEMO



Credit → CDS Spread surface (synthetic).

Commodities — futures term-structure & vol off CME (GLBX.MDP3)

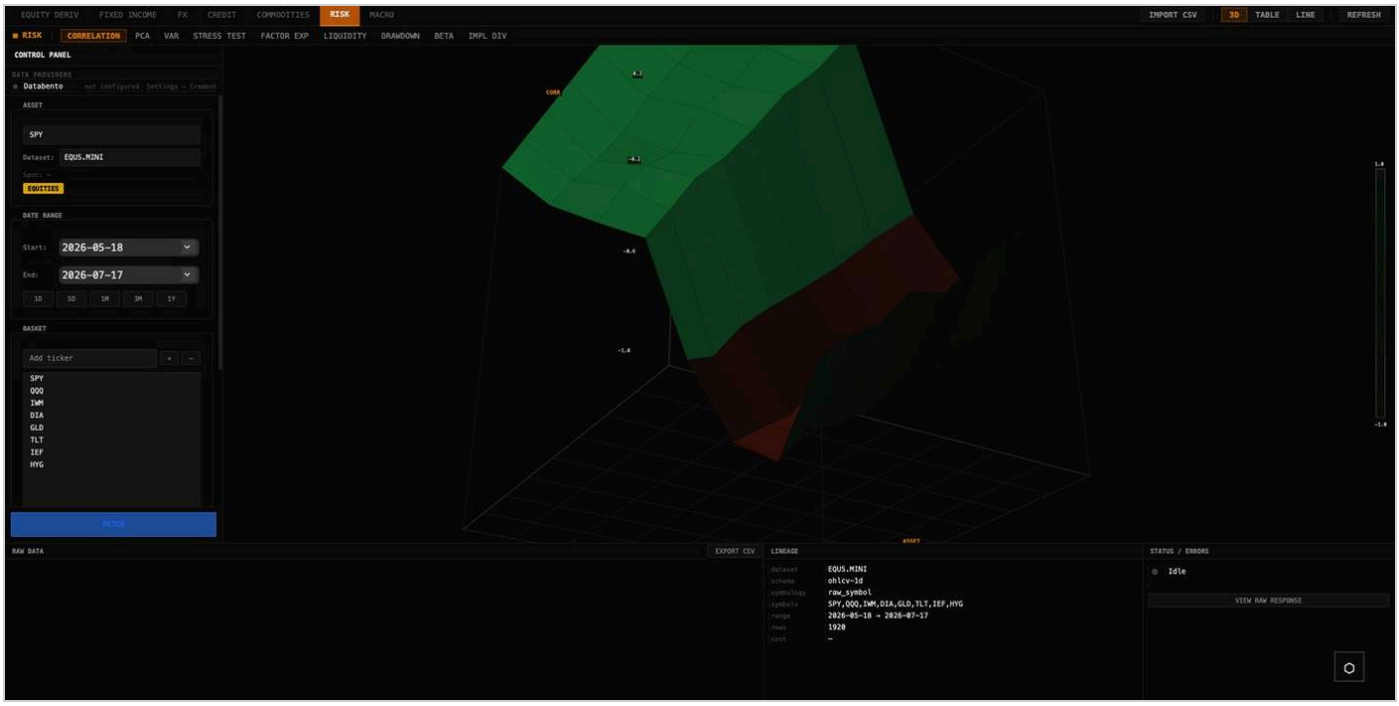
Surface	What it shows	Axes (X · Y · value)	Tier
CMDTY FWD **+L**	Futures forward curve by commodity (settlements)	Contract · Commodity · price	LIVE
CMDTY VOL	Daily realised vol from futures settlements	Strike · Expiry · vol %	COMPUTED
CRACK/CRUSH **+L**	Crack/crush & spark/dark spreads from related futures	Contract · Product · spread $ (diverging)	COMPUTED
CONTANGO **+L**	Term-structure shape / roll (contango vs backwardation)	Contract · Commodity · roll % (diverging)	LIVE



Commodities → Commodity Forward. LIVE/COMPUTED surfaces show synthetic data until Databento is connected.

Risk & Portfolio — equities-OHLCV analytics (EQUS.MINI) + two option-derived

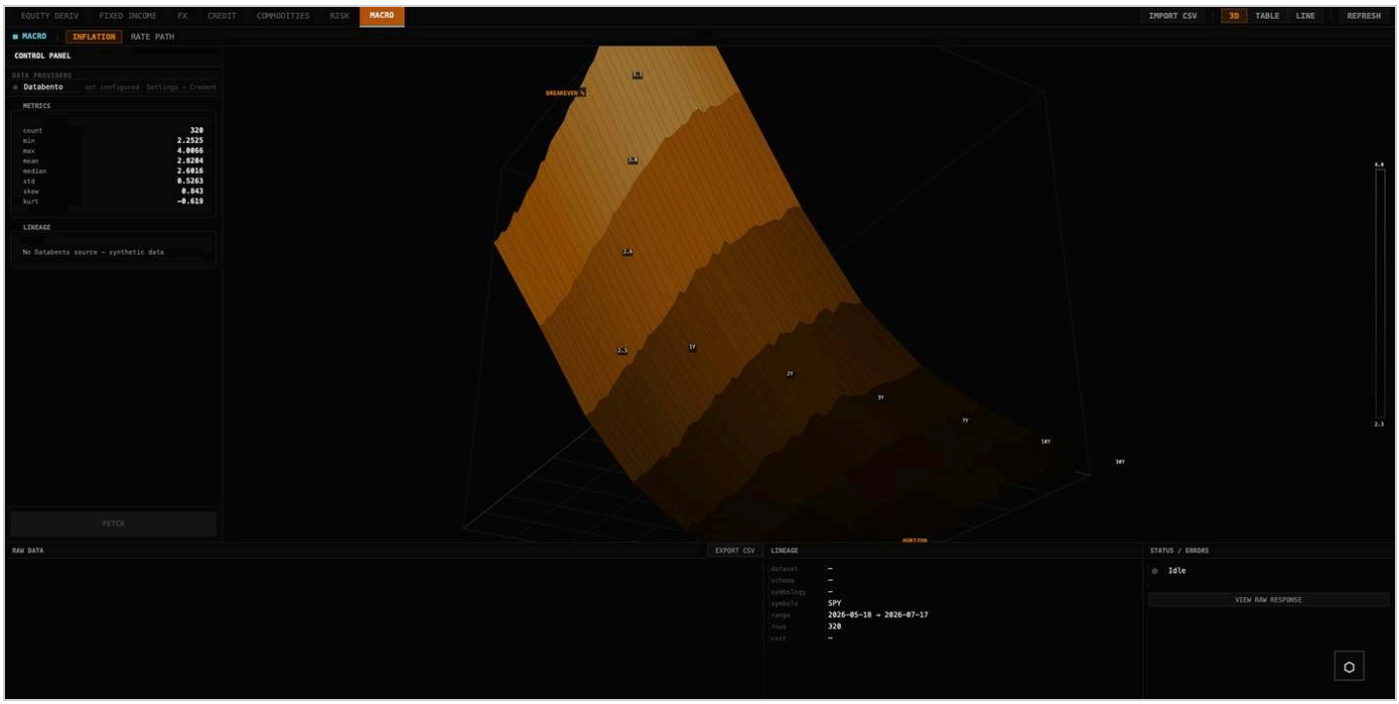
Surface	What it shows	Axes (X · Y · value)	Tier
CORRELATION	Pairwise return correlation matrix	Asset · Asset · corr (–1…1)	EQUITIES
PCA	Principal-component loadings of basket returns	Asset · PC1–PC5 · loading	EQUITIES
VAR	Value-at-Risk by confidence & horizon	Horizon · Confidence · VaR %	EQUITIES
STRESS TEST	Scenario P&L from shock vectors	Portfolio · Scenario · P&L % (diverging)	EQUITIES
FACTOR EXP	Factor loadings via OLS on basket returns	Factor · Asset · exposure (diverging)	EQUITIES
LIQUIDITY	Option liquidity (OI/volume, bid-ask) across strike × expiry	Expiry · Strike · bid-ask bp	COMPUTED
DRAWDOWN	Rolling drawdown across lookback windows	Window · Asset · drawdown %	EQUITIES
BETA	Rolling beta vs SPY across horizons	Horizon · Asset · beta (diverging)	EQUITIES
IMPL DIV	Implied dividend yield from put-call parity	Expiry · Strike series · div $	COMPUTED



Risk → Correlation matrix in 3D. Risk surfaces use an editable ticker basket (see §3).

Macro — macro-expectation surfaces (synthetic)

Surface	What it shows	Axes (X · Y · value)	Tier
INFLATION **+L**	Inflation-breakeven term structure over time	Horizon · Time · breakeven %	DEMO
RATE PATH **+L**	Expected policy-rate path by central bank	Meeting · Central bank · rate %	DEMO



Macro → Inflation-expectations surface (synthetic).

> **Coverage caveat for institutional users.** All Fixed-Income, FX, Credit and Macro surfaces (15 of 35) are **synthetic demonstrations** — the terminal has no live data source for them; their shapes are illustrative, not market-sourced. The market-sourced surfaces are the Equity-Derivative, Commodity and Risk families, and only once a Databento subscription is connected (§6).

3 The Control Panel

INPUTS · THE RAIL

The left rail holds every parameter for a surface fetch. It is **capability-driven**: each surface declares which inputs it needs, and the panel shows only those sections. A synthetic (DEMO) surface shows just metrics + lineage; an options surface adds asset, dates and option filters; a risk surface adds the basket editor.

Sections & controls

Section / control	Type	What it does · default
DATA PROVIDERS	status	A Databento status pill — dot + state + detail. Read-only (managed in Settings).
ASSET — Symbol	lineedit	Underlying / parent symbol; live Databento symbol autocomplete (300 ms debounce). Default **SPY**.
ASSET — Dataset	combo	Databento dataset (OPRA.PILLAR, GLBX.MDP3, EQUS.MINI, XNAS.ITCH, XNYS.PILLAR, IFEU/IFUS.IMPACT). Defaults to the surface's own dataset.
ASSET — Tier badge	label	Shows the surface's data tier (LIVE / COMPUTED / EQUITIES / DEMO).
DATE RANGE — Start / End	date	Fetch window (yyyy-MM-dd). Default = last 60 days ending yesterday.
DATE RANGE — 1D / 5D / 1M / 3M / 1Y	buttons	One-click lookback presets.
OPTION FILTERS — Strike window	spin	± band around spot; 1–100%, default **20%**. (Options surfaces only.)
OPTION FILTERS — DTE min / max	spin	Days-to-expiry range; default **7 / 365** (min ≤ max enforced).
OPTION FILTERS — IV method	combo	Implied-vol solver: **Brent** / Bisection / Newton-Raphson.
BASKET — Add / Remove / list	list	Correlation/risk basket (risk surfaces only). Add via input/+/Enter, remove via –. Seeded with 8 ETFs: SPY, QQQ, IWM, DIA, GLD, TLT, IEF, HYG.
METRICS	readout	Live stats of the active surface: count, min, max, mean, median, std, skew, kurtosis.
LINEAGE	readout	A one-line description of the surface's data provenance.
FETCH	button	Pulls live Databento data for the surface. Enabled for LIVE/COMPUTED/EQUITIES; **disabled** (greyed) for DEMO surfaces with a "viewing synthetic data" tooltip.

Good to know. The "Spot: —" readout in the ASSET card is a fixed placeholder and does not update. Editing rail inputs stages the parameters; the surface only pulls live data when you press **FETCH** (and only for non-synthetic surfaces).

At a glance, from the control panel you can…

- Set the underlying (with live symbol search), dataset, and date range (or one-click presets).
- Filter options by strike window, DTE range and IV method; build a correlation/risk basket.
- Read live surface statistics and lineage, see Databento status, and trigger a live fetch.

4 The three views

3D · TABLE · LINE

The **3D / TABLE / LINE** toggle (top-right) chooses how the active surface is drawn. All 35 surfaces support 3D and Table; only the seven term-structure surfaces support Line (Yield Curve, FX Forward, Commodity Forward, Crack/Crush, Contango, Inflation, Rate Path).

3D surface

A shaded, rotatable 3D mesh inside a wireframe cube, with axis labels, tick labels and a colour-bar legend. **Drag** with the left mouse to rotate; **scroll** to zoom. The colour scale is sequential (dark → amber) for level surfaces, or diverging (red → neutral → green) for signed surfaces (theta, skew, OIS basis, correlation, beta, stress, …). There is no per-cell hover readout, and very large grids are down-sampled to ~40×40 for the software renderer.

Table (heatmap)

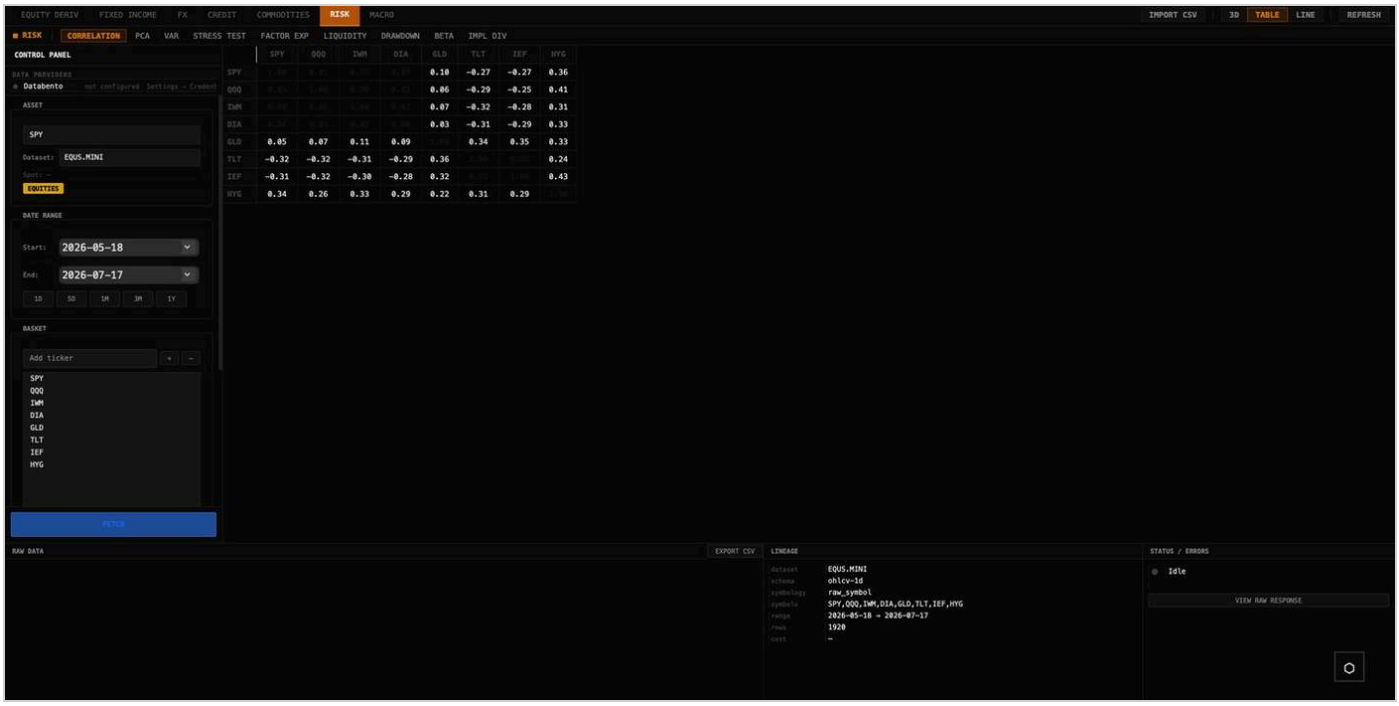
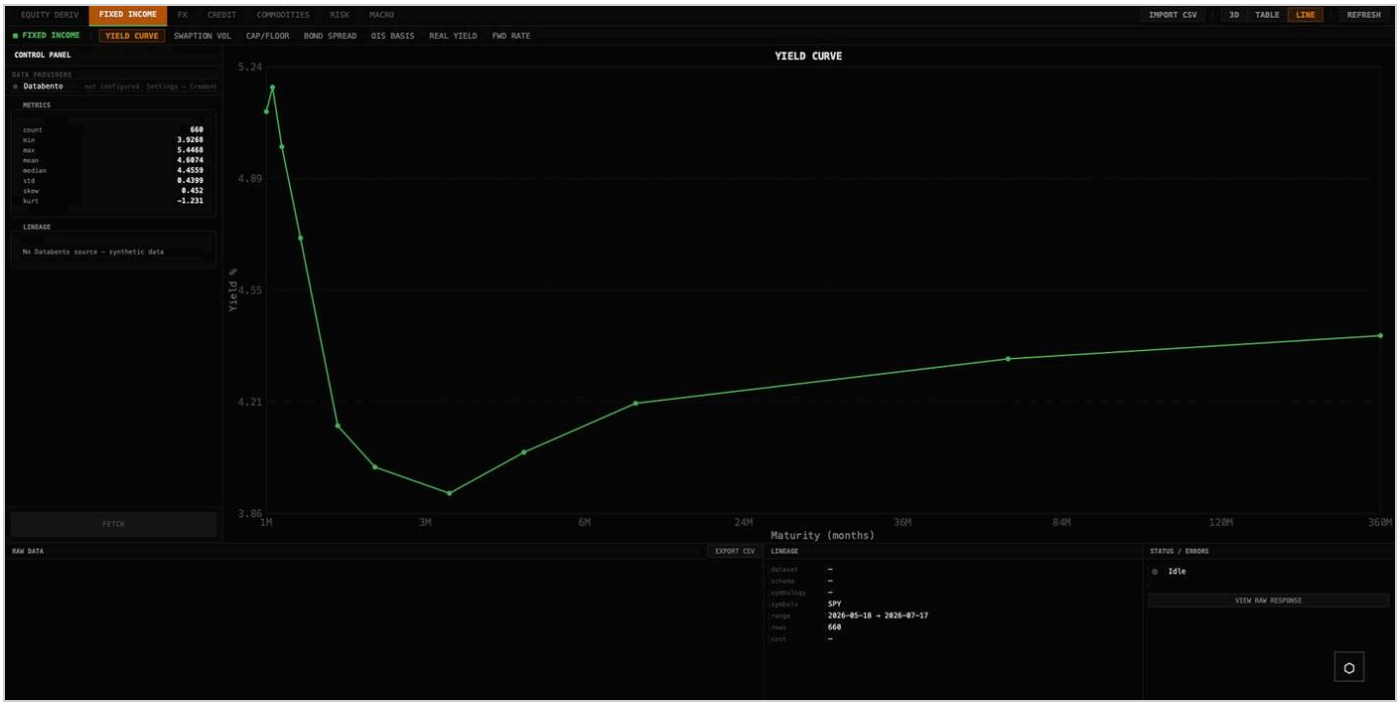


Table view of the Risk → Correlation surface — a colour-coded heatmap matrix; note the 8-ticker basket editor in the rail and the EQUS.MINI lineage at the bottom.

The same grid as a colour-coded numeric table — headers are the two axes, each cell the value (2 decimals), background heat-mapped on the same ramps as 3D. Read-only (no sorting/selection in the main table). Ideal for correlation, transition and stress matrices where exact numbers matter.

Line



Line view of Fixed Income → Yield Curve — a term-structure slice as a polyline (Yield % vs maturity). Multi-series term structures (commodities, FX forwards, rate paths) draw one line per curve with a legend.

> **One behaviour to know.** The LINE button is clickable on all 35 surfaces, but for the 28 that are not term structures it silently falls back to the Table view (the button stays highlighted). Use Line for the seven term-structure surfaces; use 3D or Table for everything else.

5 The Data Inspector

PROVENANCE

The bottom strip is the audit trail for whatever is on screen — three columns that answer "what is this data, and where did it come from?"

Column	Contents
RAW DATA	Tabbed snapshots of the raw provider records (e.g. a `vol_surface` tab with strike/expiration/iv, or an `ohlcv-1d` tab with OHLCV bars). The table is sortable. **EXPORT CSV** writes the current tab's raw rows to a file.
LINEAGE	Seven fields: **dataset** (e.g. OPRA.PILLAR), **schema** (definition+cbbo-1s), **symbology** (parent / raw_symbol), **symbols**, **range** (start→end), **rows** (cell count), and **cost** (the estimated Databento query cost in USD, looked up in the background). All read "—" for synthetic surfaces.
STATUS / ERRORS	A timestamped status line with a green/red dot (e.g. "Vol surface loaded", "Fetch failed", "No Databento API key configured"). **VIEW RAW RESPONSE** opens a modal with the full raw provider output of the last fetch.

Together the metrics (control panel) and the lineage + cost (inspector) let an institutional user validate every surface: its statistical shape, its exact data source and schema, how many records it drew, and what the query cost.

6 Data & connectivity

DATABENTO · CSV

Surface Analytics has a single live provider — **Databento** — plus built-in synthetic data and CSV import.

The four data tiers

Tier	Meaning & source	Surfaces
LIVE	Direct Databento fetch — CME futures (GLBX.MDP3).	Commodity Forward, Contango (2)
COMPUTED	Derived in a Python provider from a Databento feed — Black-Scholes greeks/vol from OPRA options, realised vol & spreads from CME futures.	All equity greeks, Local Vol, Liquidity, Implied Div, Commodity Vol, Crack (11)
EQUITIES	Built from Databento equities daily bars (EQUS.MINI) for the basket.	Correlation, PCA, VaR, Stress, Factor, Drawdown, Beta (7)
DEMO	Synthetic — no market source; generated locally, re-seeded on Refresh. Always available offline.	All Fixed Income, FX, Credit, Macro (15)

Connecting Databento

- Add your Databento API key in **Settings ▸ Credentials ▸ Databento**. The provider pill then reads `configured` (amber); until then it reads `not configured` and FETCH on live surfaces reports "No Databento API key configured".
- Databento is a **paid market-data service**. OPRA options and CME futures each require a paid subscription; the lineage panel shows a per-query **cost** estimate before you commit. Without an entitled key, only the synthetic (DEMO) surfaces are usable.
- A failed live fetch does **not** wipe the view — the previous (or synthetic) surface stays on screen with an error line in the inspector.

CSV import

IMPORT CSV loads your own grid from a comma-separated file (first row treated as a header). It currently populates the **five equity option surfaces** — Vol, Delta, Gamma, Vega, Theta — expecting three columns `strike, dte, value`. Importing while another surface is active is silently ignored, so switch to one of those five first. Any raw dataset can be exported back out via the inspector's **Export CSV**.

7 AI agent tools (15 MCP tools)

AGENTIC

The terminal's AI agent can drive Surface Analytics through 15 tools in the `surface-analytics` category. The fetch tools drive the same on-screen surface and share its cache; the discovery/cache tools are read-only.

Tool	What it does	Live / billable
`list_surface_categories`	List the 7 categories and their surfaces.	no
`list_surface_types`	List all 35 surfaces with capability metadata.	no
`get_surface_capability`	Capability of one surface (tier, dataset, required inputs, views).	no
`test_databento_connection`	Verify the configured Databento key works.	key check
`fetch_databento_options_surface`	Full options surface (vol + greeks + skew) for an underlying.	**yes**
`fetch_databento_ohlcv`	Historical OHLCV bars (feeds correlation/PCA/drawdown/beta).	**yes**
`fetch_databento_futures_term_structure`	Futures forward curve + contango for commodities.	**yes**
`fetch_databento_local_vol`	Local-volatility surface.	**yes**
`fetch_databento_implied_dividend`	Implied-dividend surface.	**yes**
`fetch_databento_liquidity`	Options liquidity heatmap.	**yes**
`fetch_databento_commodity_vol`	Commodity volatility surface for a root.	**yes**
`fetch_databento_crack_spread`	Crack/crush spread surface.	**yes**
`fetch_databento_stress_test`	Stress-test P&L surface for a basket.	**yes**
`fetch_databento_yield_curve`	Yield-curve surface.	**yes**
`get_databento_cached_data`	Read the last-fetched data without a network call.	no (cache)

> **Spend control.** The ten `fetch_databento_*` tools each make a **live, billable** Databento call and are flagged as requiring confirmation — the agent proposes the fetch and you approve it before it runs. The three discovery tools, the connection test and the cache read make no billable call.

8 Quick reference & important notes

HOW DO I...

I want to…	Do this
Pick a surface	Click a category tab, then a surface chip.
Rotate / zoom the 3D view	Drag to rotate; scroll to zoom.
Read exact numbers	Switch to **TABLE** — a heatmap of the values.
See a term structure as a curve	Switch to **LINE** (Yield, FX Fwd, Cmdty Fwd, Crack, Contango, Inflation, Rate Path).
Change the underlying	Type it in the ASSET box (with autocomplete), or link an equity symbol group.
Pull real market data	Connect Databento (Settings ▸ Credentials), then press **FETCH** on a LIVE/COMPUTED/EQUITIES surface.
Refresh the synthetic surfaces	Press **REFRESH** (re-seeds the demo data).
Load my own surface	Select Vol/Delta/Gamma/Vega/Theta, then **IMPORT CSV** (strike,dte,value).
Export / audit the data	Inspector ▸ **EXPORT CSV**; check LINEAGE for source, rows & cost; **VIEW RAW RESPONSE** for the raw feed.

Important notes for institutional users

- **15 of 35 surfaces are synthetic.** All Fixed-Income, FX, Credit and Macro surfaces are illustrative demonstrations with no live data source; treat their shapes as examples, not market data.
- **Live data needs a Databento subscription.** The Equity-Derivative, Commodity and Risk surfaces draw from paid Databento datasets (OPRA options, CME futures, US equities); each fetch shows a cost estimate and only runs with an entitled key.
- **Everything works offline.** On open — and whenever Databento is unconfigured — all 35 surfaces render from built-in demo data, so the tab is always explorable.
- **Views:** 3D and Table apply to all 35; Line applies to the 7 term-structure surfaces (on the others the Line button falls back to Table).
- **Session state:** the tab remembers your category, view mode and control inputs; it reopens on that category's first surface.
- **No hover readout in 3D:** use the Table view (or the metrics/lineage panels) when you need exact values rather than the shape.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

AI Quant Lab

A complete, screen-by-screen walkthrough of the AI Quant Lab — the terminal's 24-module quantitative-research workbench over an embedded Python stack.

`>_ 24 modules · Qlib + GS Quant + Python · 100 MCP tools`

Covers · **24 quant modules across 4 groups** — CORE · AI/ML · ADVANCED · ANALYTICS · **the Python engine & the 100-tool MCP suite**
On the FKeyBar as **QUANT LAB** · command bar **"quantlab"**

Demonstrated live — with real computed results — on the Fincept Terminal Quant Lab tab.

Fincept Research Desk · Professional Edition v5.0 · Prepared for institutional clients

Contents

8 SECTIONS

GETTING ORIENTED

— Overview — what the AI Quant Lab is 3

1 Anatomy — the 24-module workbench 4

THE FOUR MODULE GROUPS

2 CORE — factors, models, backtests, signals 6

3 AI / ML — agents, RL, online & meta learning 8

4 ADVANCED — HFT, optimization, feature engineering 10

5 ANALYTICS — GS Quant, statsmodels, CFA, forecasting 12

REFERENCE

6 The engine — Python, credits & dependencies 14

7 Driving Quant Lab by AI — the MCP tool suite 15

8 Quick reference & honest caveats 16

About this guide. Every screenshot is a real, live capture of the Quant Lab tab — including genuinely computed results (OLS regressions, risk metrics, a 518-factor library, normalizer catalogs). Module names, sub-tabs, field defaults and the run behaviour are transcribed from the shipping code.

What the AI Quant Lab is

OVERVIEW

The AI Quant Lab is a **24-module quantitative-research workbench** — a native front-end over the terminal's embedded Python analytics stack (Microsoft **Qlib**, a Goldman-style **GS Quant** toolkit, **statsmodels**, forecasting libraries, reinforcement learning, and autonomous research agents). Each module is a self-contained app: fill a form, press a run button, and the terminal shells out to a Python script and paints the result back as metric cards, tables and charts.

The modules are organised into four groups in the left rail — **CORE** (factor discovery, an ML model library, backtesting, live signals), **AI/ML** (a LangGraph deep agent, RL trading, online and meta learning), **ADVANCED** (HFT microstructure, portfolio optimization, feature engineering, rolling retraining), and **ANALYTICS** (GS Quant, statsmodels, CFA methods, and three forecasting benches). Reached from the FKeyBar tab **QUANT LAB** (or the command bar, `quantlab`), it is **not plan-gated**.

Local compute, no credits. With a single exception (the AI/ML *Deep Agent*, which drives a hosted model), **every module runs locally in Python and spends no credits**. Most operate entirely offline on the data you paste or on built-in sample data; a few fetch free market data (Yahoo Finance, crypto exchanges). This makes the Lab safe to explore freely.

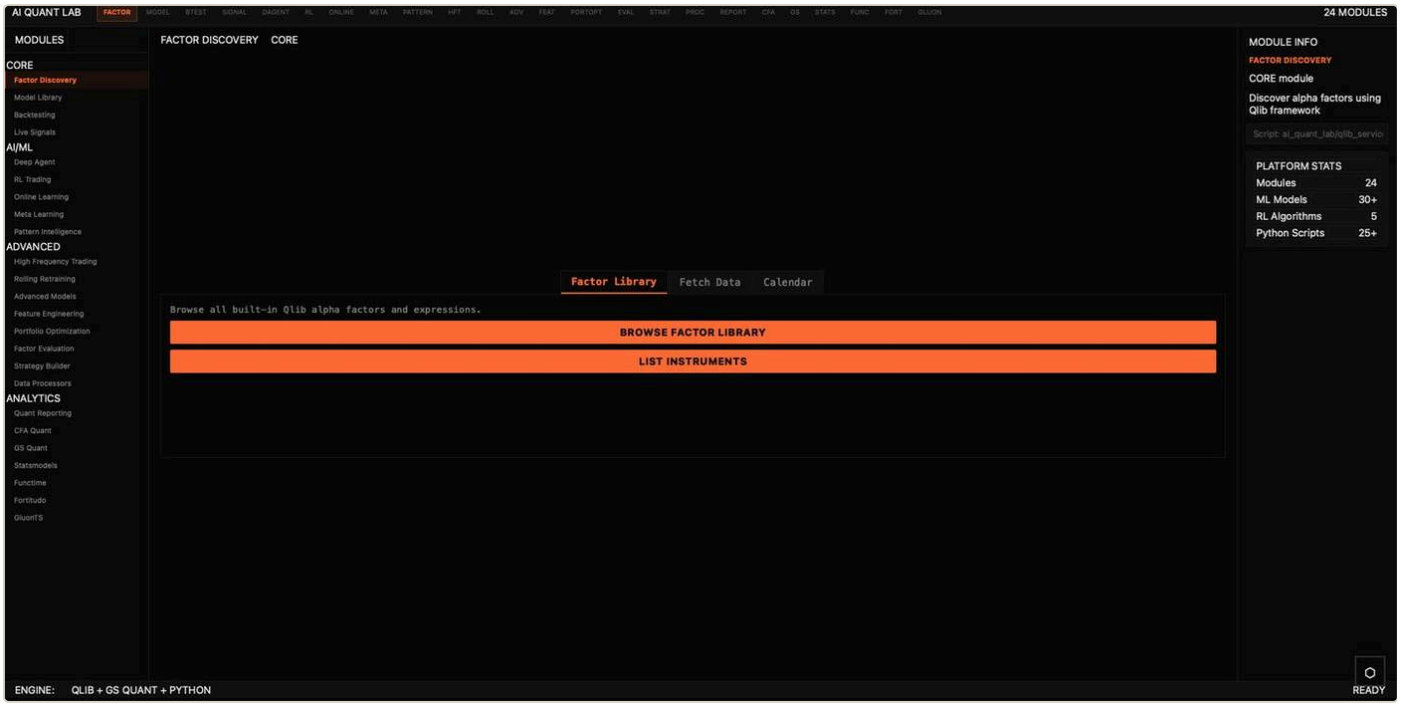
AT A GLANCE, IN THE AI QUANT LAB YOU CAN…

- Browse **500+ Qlib alpha factors** with their exact expressions.
- Train ML models and run momentum **backtests** on live prices.
- Fit **OLS / ARIMA** and run full econometric diagnostics.
- Compute **risk metrics, VaR, Greeks** and stress tests.
- Optimize portfolios — **HRP, min-var, max-Sharpe, CVaR, Black-Litterman**.
- Forecast probabilistically with quantile bands and interval calibration.
- Detect **anomalies**, decompose seasonality, test stationarity.
- Score factors by **IC / ICIR** and build strategy signals.
- Drive an **autonomous research agent** and RL trading loops.

1 Anatomy — the workbench

THE WORKBENCH

Every module shares one frame: a top strip of 24 short codes, a grouped left rail, a central module host, a right MODULE INFO / PLATFORM STATS panel, and a bottom ENGINE bar. You land on **Factor Discovery**.



The Quant Lab shell — 24-code top strip, the CORE / AI-ML / ADVANCED / ANALYTICS rail, the module host, and the MODULE INFO + PLATFORM STATS panel. Bottom: ENGINE — QLIB + GS QUANT + PYTHON.

1.1 How a module runs

Pick a module (rail or top code) → choose a sub-tab → fill the form (many inputs have a **LOAD SAMPLE** button that injects ready-made data) → press the run button. The terminal packages your inputs as JSON, runs the module's Python script as a subprocess, and renders the returned JSON as cards, tables and charts. A run is asynchronous — the UI stays responsive and shows a spinner (*"first invocation per session takes longer — cold start"*). Two modules stream live (RL training, Rolling Retraining) with a progress bar and log. Results can be exported to the File Manager with **EXPORT RESULTS**; only the selected module index is remembered between sessions.

1.2 The 24 modules

Code	Module	Group	What it does
FACTOR	**Factor Discovery**	CORE	Browse Qlib alpha factors, the instrument universe and calendar.
MODEL	**Model Library**	CORE	Catalog ~15 Qlib models and train one (LightGBM, LSTM, Transformer…).
BTEST	**Backtesting**	CORE	Momentum backtest on live Yahoo prices with an equity curve.
SIGNAL	**Live Signals**	CORE	Pull signal data; factor-analysis / feature-importance views.
DAGENT	**Deep Agent**	AI/ML	LangGraph multi-agent research + Microsoft RD-Agent loops.
RL	**RL Trading**	AI/ML	Train a PPO/DQN/A2C/SAC/TD3 agent with live progress.
ONLINE	**Online Learning**	AI/ML	Incremental models with concept-drift detection.
META	**Meta Learning**	AI/ML	AutoML: model selection, ensembles, hyperparameter search.
PATTERN	**Pattern Intelligence**	AI/ML	Technical patterns — a raw command runner (experimental).
HFT	**High Frequency Trading**	ADVANCED	Live order-book, market-making quotes, toxic-flow, slippage.
ROLL	**Rolling Retraining**	ADVANCED	Schedule & stream walk-forward model retraining.

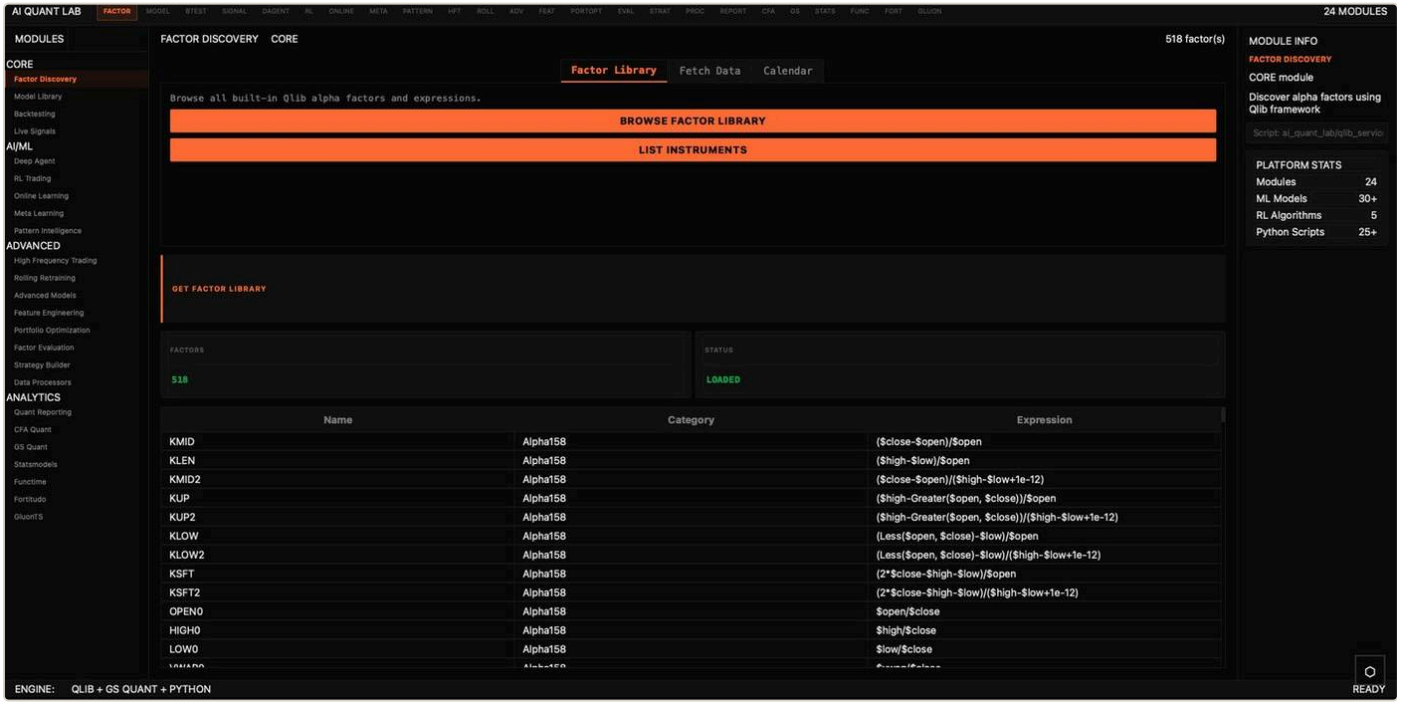
ADV	**Advanced Models**	ADVANCED	Create-and-train deep/GBDT models in one shot.
FEAT	**Feature Engineering**	ADVANCED	Indicators, IC feature-selection, expression engine.
PORTOPT	**Portfolio Optimization**	ADVANCED	HRP, min-variance, max-Sharpe, frontier, Black-Litterman.
EVAL	**Factor Evaluation**	ADVANCED	IC metrics, a scored factor report, return risk metrics.
STRAT	**Strategy Builder**	ADVANCED	TopK-Dropout / Risk-Parity strategies + portfolio metrics.
PROC	**Data Processors**	ADVANCED	10 normalizers and composable processing pipelines.
REPORT	**Quant Reporting**	ANALYTICS	IC, cumulative-return, risk and model-performance tearsheets.
CFA	**CFA Quant**	ANALYTICS	12 CFA-curriculum analyses (trend/ARIMA/ML/sampling…).
GS	**GS Quant**	ANALYTICS	Risk, portfolio, Greeks, VaR, stress test, backtest, stats.
STATS	**Statsmodels**	ANALYTICS	OLS, ARIMA, stationarity, ACF/PACF, Granger, descriptive.
FUNC	**Functime**	ANALYTICS	Forecasting, anomalies, seasonality, intervals, metrics.
FORT	**Fortitudo**	ANALYTICS	CVaR/MV optimization, covariance, frontier, decay weights.
GLUON	**GluonTS**	ANALYTICS	Probabilistic & quantile forecasts, distribution fits.

> **Chrome that doesn't introspect.** The right-panel PLATFORM STATS show a live **Modules 24** count, but *ML Models 30+*, *RL Algorithms 5* and *Python Scripts 25+* are fixed labels. The bottom `ENGINE / READY` bar is decorative — it does not reflect live Python health (that surfaces only when you run a module).

2 CORE

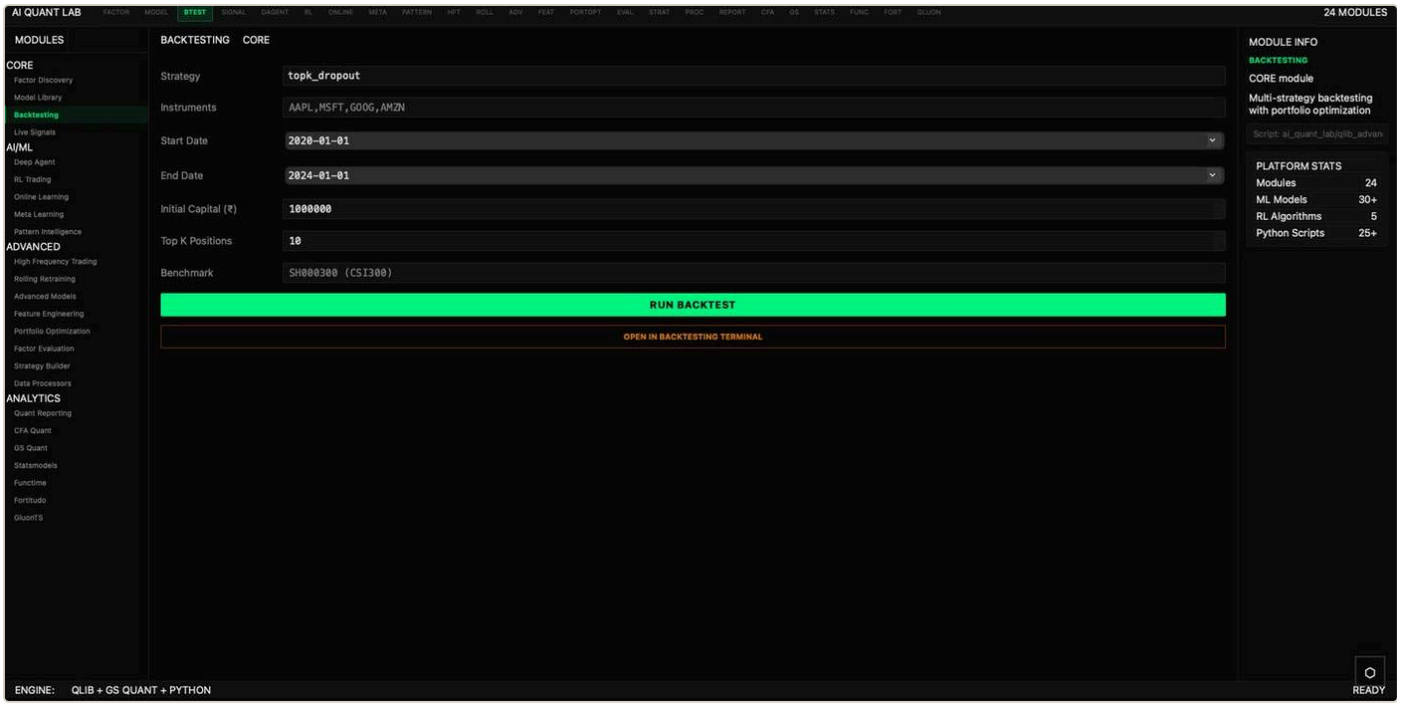
GROUP · CORE

The CORE group is the Qlib front door: discover factors, browse and train the model zoo, run a self-contained backtest, and inspect signals. Here **Factor Discovery** has returned its full library — **518 Qlib alpha factors** with expressions.



Factor Discovery → BROWSE FACTOR LIBRARY — 518 Alpha158/360 factors as Name / Category / Expression (KMID = `($close-$open)/$open`, …). A real, computed result.

Module · sub-tabs	What you do & what a run produces
Factor Discovery **Factor Library · Fetch Data · Calendar**	**BROWSE FACTOR LIBRARY** lists every built-in alpha factor with its Qlib expression; **LIST INSTRUMENTS** enumerates the universe; **FETCH DATA / GET TRADING CALENDAR** pull OHLCV metadata and the calendar for a date range.
Model Library **Browse · Train Model · Backtest**	**LIST ALL MODELS** shows ~15 models with an availability flag; **CHECK QLIB STATUS** reports live Qlib health; **TRAIN MODEL** runs a real Qlib training job (LightGBM/XGBoost/LSTM/…).
Backtesting **single form**	A momentum backtest: pick a strategy, tickers, dates, capital, Top-K and benchmark, then **RUN BACKTEST** for a KPI board (return, Sharpe, drawdown, Calmar, win-rate) + a portfolio-vs-benchmark equity curve. **OPEN IN BACKTESTING TERMINAL** hands off to the full backtester.
Live Signals **Signal Data · Factor Analysis · Feature Importance**	**FETCH SIGNALS** pulls OHLCV signal data; the Factor-Analysis and Feature-Importance tabs show IC / feature-importance layouts.



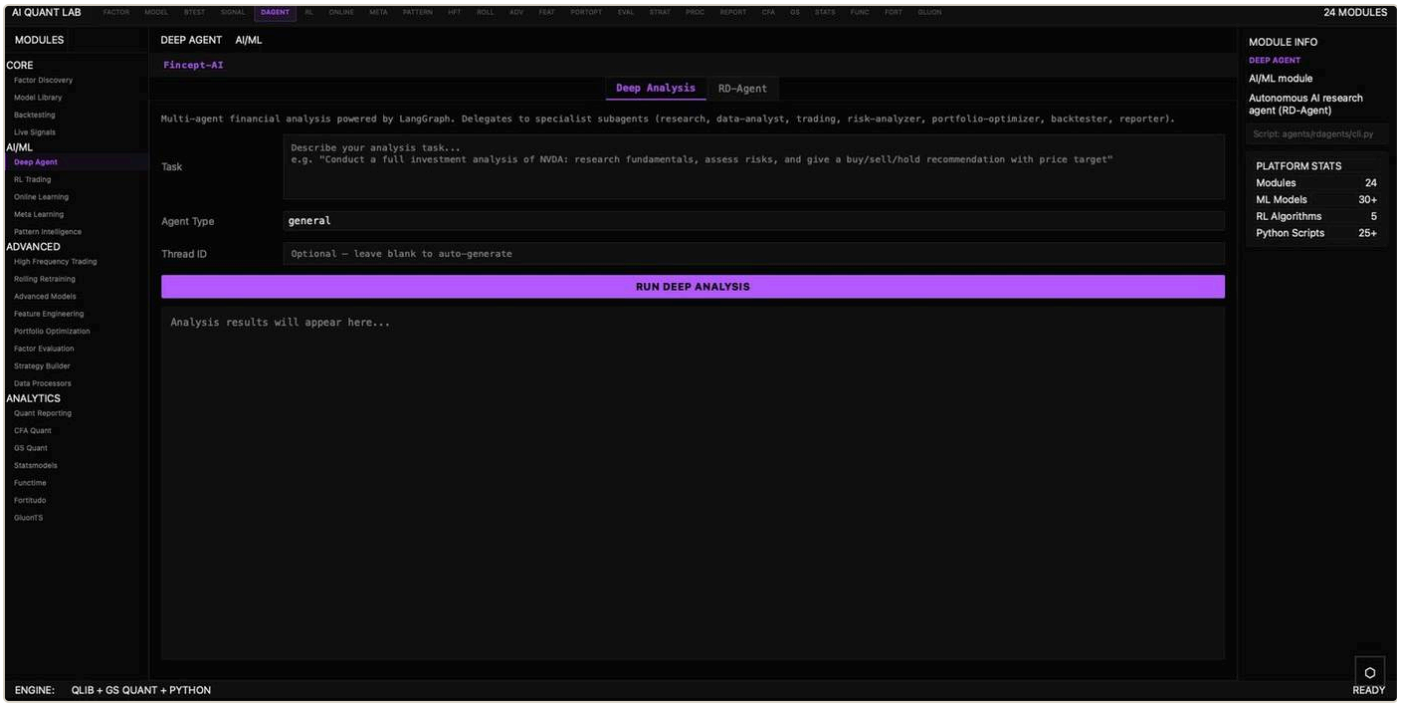
The Backtesting module — a pre-filled Qlib momentum backtest (strategy `topk_dropout`, capital, Top-K, benchmark) with RUN BACKTEST and a hand-off to the standalone Backtesting terminal.

> **Honest CORE notes.** The **Backtesting** module is the fully end-to-end one (real Yahoo prices; its “Commission” figure is actually market-impact bps). **Model Library → Backtest** and **Live Signals → Factor Analysis / Feature Importance** depend on a trained model persisting across calls — which it doesn't (each run is a fresh subprocess) — so they error. Factor Discovery's data tabs need a locally-downloaded Qlib dataset; the *factor-library list* always works.

3 AI / ML

GROUP · AI / ML

The AI/ML group spans an autonomous research agent, reinforcement-learning trading, and adaptive learning. **Deep Agent** is the one place in the Lab that drives a hosted model.



Deep Agent — a LangGraph multi-agent analyst (Deep Analysis) plus Microsoft RD-Agent research loops. The `Fincept-AI` badge marks this as the one credit-bearing module.

Module · sub-tabs	What you do & what a run produces
Deep Agent **Deep Analysis · RD-Agent**	Describe a task in natural language; **RUN DEEP ANALYSIS** delegates to specialist sub-agents (research, risk, portfolio…) and returns a written analysis. RD-Agent launches autonomous factor-mining / model-optimization / quant-research loops you monitor in a task table. **Runs on Fincept-AI — spends credits.**
RL Trading **single form**	Pick an algorithm (PPO/DQN/A2C/SAC/TD3), a ticker, episodes and learning rate; **TRAIN RL AGENT** streams a live progress bar + log, then final training cards. Trains on Yahoo prices — no credits.
Online Learning **Models · Create Model · Incremental Train · Predict**	Create an incremental model, train it one sample at a time, predict, and watch MAE + concept-drift health.
Meta Learning **Model Selection · Ensemble · Hyperparameter Tuning**	Auto-train & rank candidate models into a leaderboard, build voting/stacking ensembles, and grid/random-search hyperparameters.
Pattern Intelligence **generic runner**	A free-text command + JSON runner over the technical-indicators script — experimental, no curated UI yet.

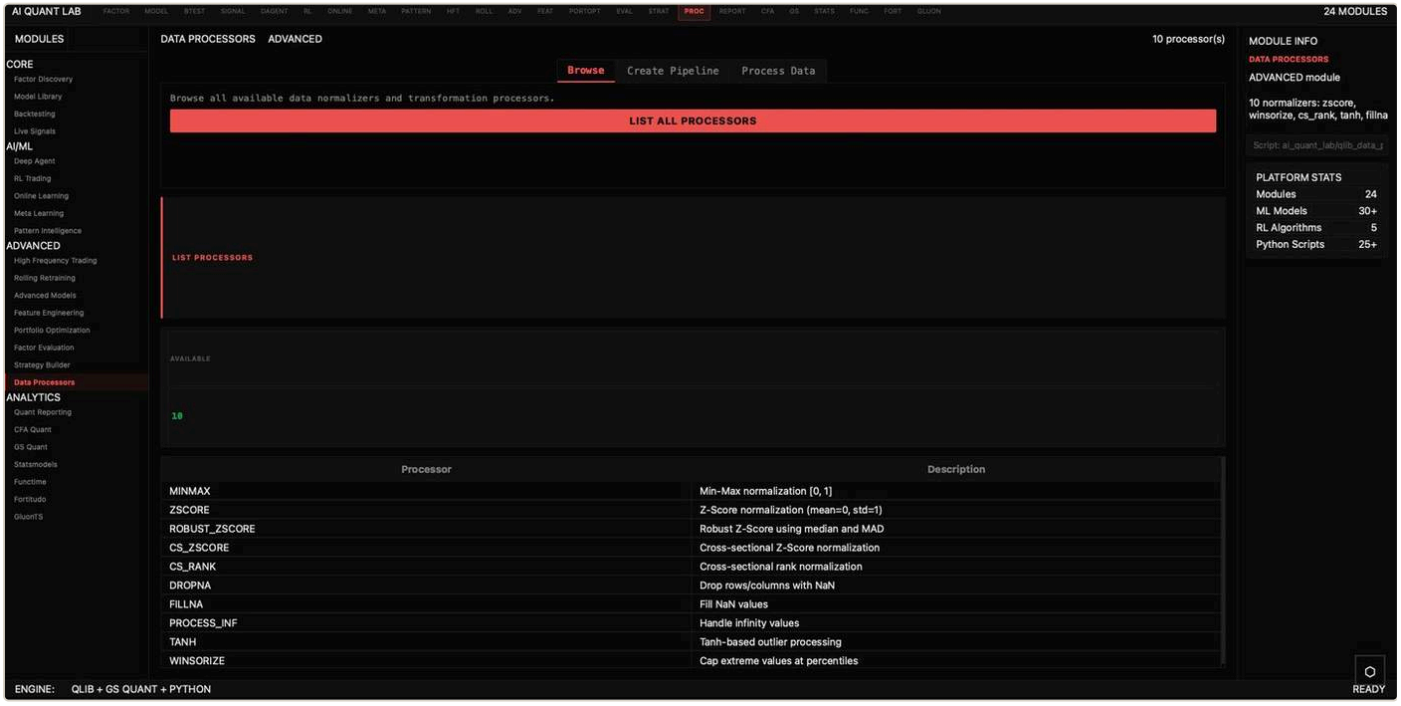
> **The one credit-bearing module.** Deep Agent routes to `Fincept-AI`, the managed model — a multi-step run makes several LLM calls. Its RD-Agent loops additionally require the Microsoft `rdagent` package; without it they return an install hint rather than running. Every other AI/ML module is local ML.

ML modules need their native libraries. The learning modules import heavyweight Python packages (LightGBM, PyTorch, River). Where a required native library is missing on the host, the module returns a clean red error card rather than crashing — see §6 for the exact behaviour and a real example.

4 ADVANCED

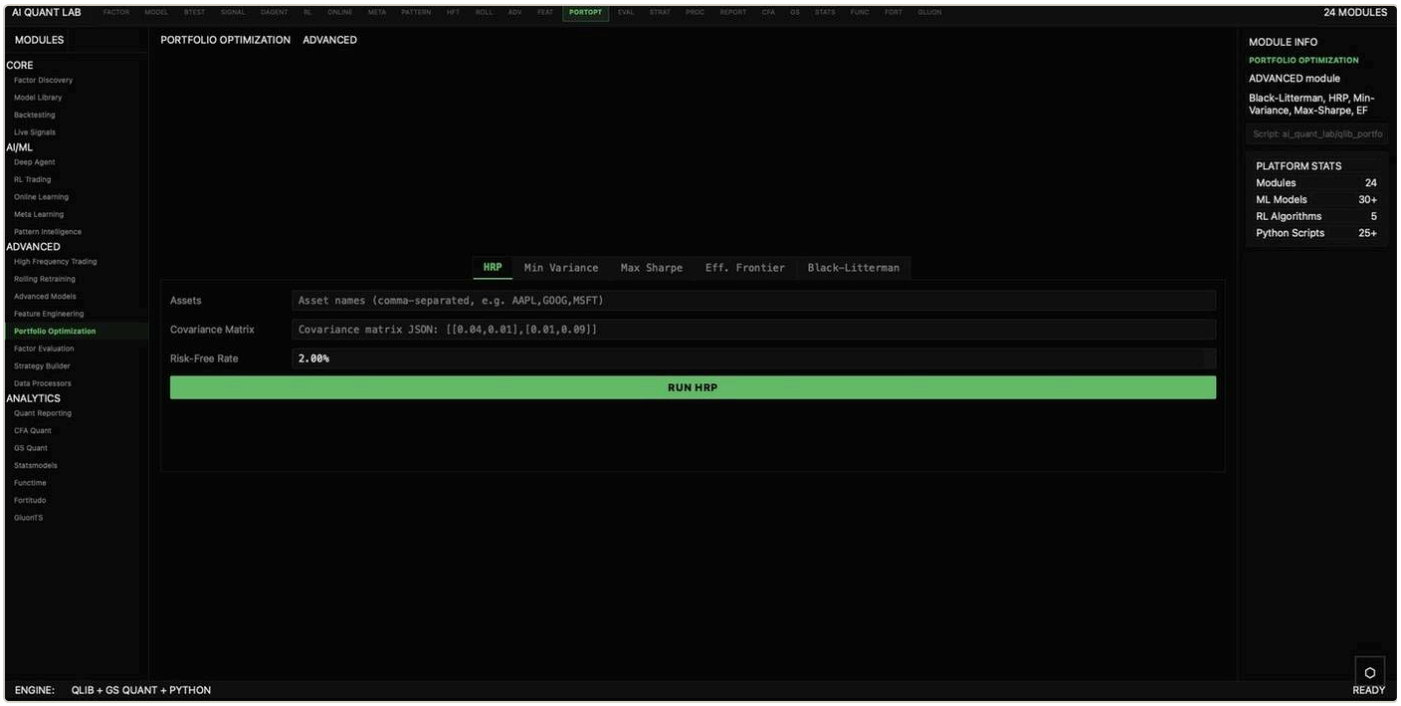
GROUP · ADVANCED

The ADVANCED group is the Qlib-suite toolbox: market microstructure, portfolio optimization, feature engineering, factor evaluation, strategy construction, and data processing. Here **Data Processors** has listed its **10 built-in normalizers**.



Data Processors → LIST ALL PROCESSORS — the 10 normalizers (MINMAX, ZSCORE, ROBUST_ZSCORE, CS_RANK, WINSORIZE, TANH, …) with descriptions. A real, computed result.

Module	What a run produces
High Frequency Trading	Live L2 order-book metrics (mid/spread/imbalance), Avellaneda-Stoikov market-making quotes, PIN toxic-flow score, and order-book-walk slippage. Fetches live data from 6 crypto exchanges (CCXT).
Rolling Retraining	Create walk-forward retraining schedules, preview the train/test windows, and stream a live retrain with per-window progress.
Advanced Models	Configure a deep/GBDT architecture (hidden/layers/dropout/epochs) and **CREATE & TRAIN** in one shot; final loss + a sampled loss curve.
Feature Engineering	Compute 9 indicators on pasted prices, rank features by IC, or evaluate a Qlib expression over pasted OHLCV.
Portfolio Optimization	Five optimizers over a supplied covariance matrix — HRP, min-variance, max-Sharpe, efficient frontier, Black-Litterman — with sorted target weights.
Factor Evaluation	IC statistics (mean/ICIR/p-value), a 0–100 scored factor report, and return-based risk metrics (VaR/CVaR, capture ratios).
Strategy Builder	Build a TopK-Dropout or Risk-Parity strategy, or compute a full performance panel (Sharpe/Sortino/Calmar/alpha/beta) for any return series.
Data Processors	List 10 normalizers, compose an ordered pipeline, and run data through it (shape/null/mean/std deltas).



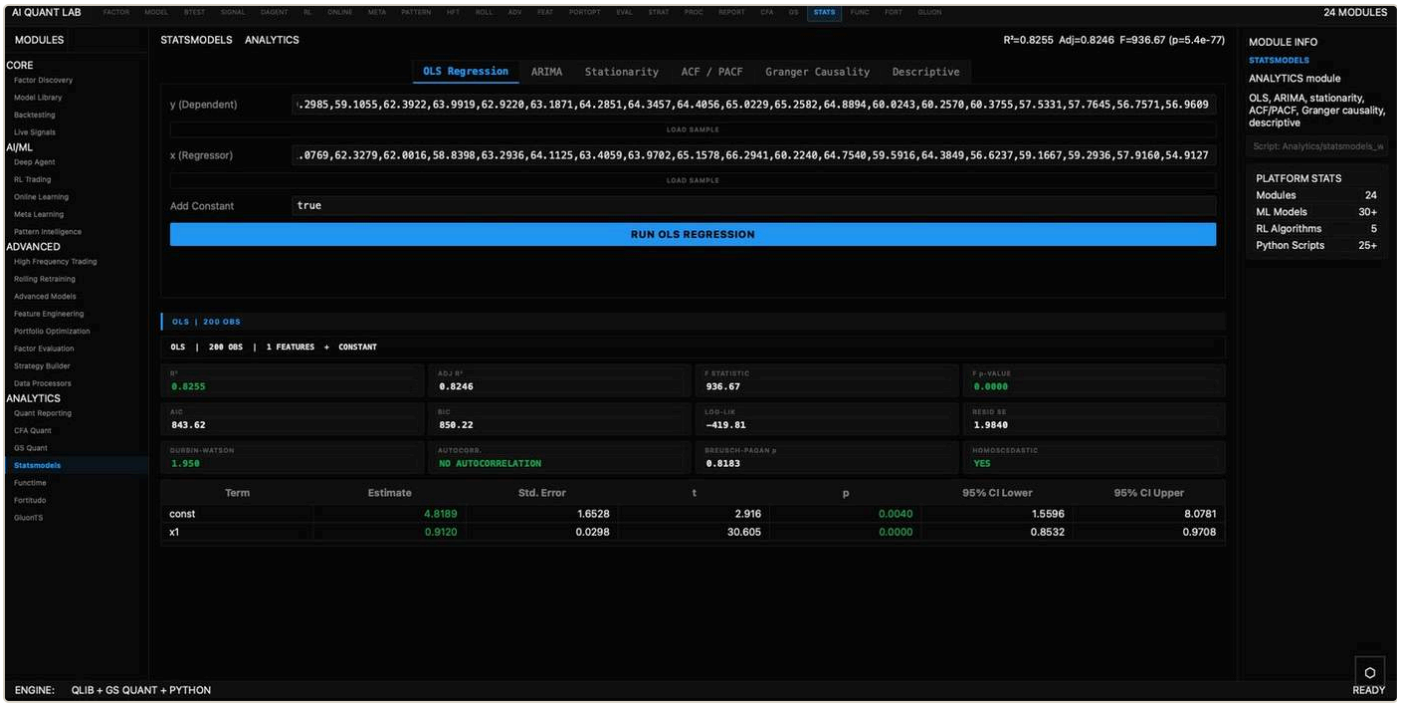
Portfolio Optimization — five optimizer sub-tabs (HRP · Min Variance · Max Sharpe · Eff. Frontier · Black-Litterman); enter a covariance matrix and expected returns, then optimize.

> **Honest ADVANCED notes.** These modules take data as pasted CSV/JSON — Portfolio Optimization, Feature Engineering and Strategy Builder do not fetch prices themselves (you supply the covariance/returns). HFT is the one that reaches the network (public crypto exchanges). Advanced Models, Factor Evaluation and Strategy Builder each ship renderers for extra commands (predict, model-info, Enhanced-Indexing…) that have no button in their panel.

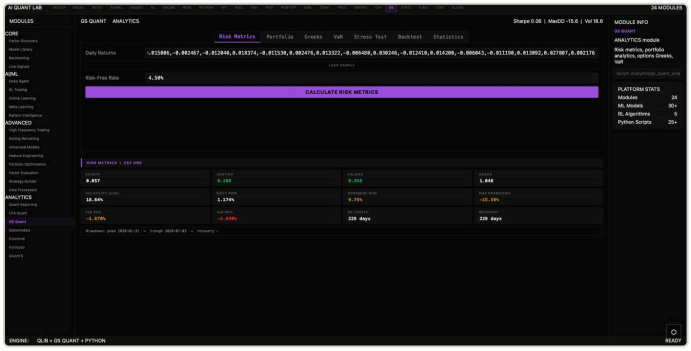
5 ANALYTICS

GROUP · ANALYTICS

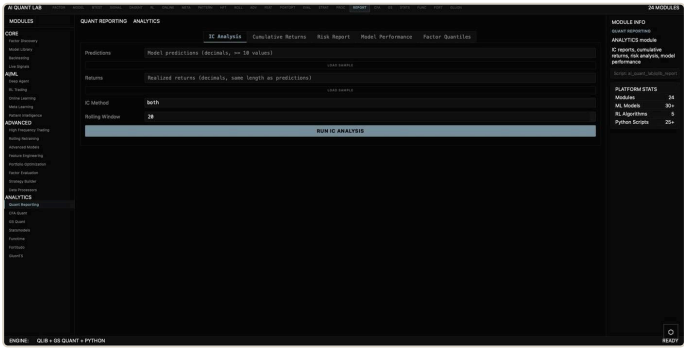
The ANALYTICS group wraps mature quant libraries into point-and-click benches. All are local compute on pasted or built-in sample data. Two live results below: a full **OLS regression** and a **GS Quant risk** panel.



Statsmodels → OLS Regression — R²=0.83, F=936.7, with Durbin-Watson / Breusch-Pagan diagnostics and a full coefficient table (estimate, std-error, t, p, 95% CI). Real statsmodels output.



GS Quant → Risk Metrics — Sharpe/Sortino/Calmar/Omega, annualized vol 18.6%, VaR 95/99%, drawdown length & recovery.



CFA Quant — a single form with a 12-analysis dropdown (trend, ARIMA, ML, PCA, resampling, CLT, sampling error, validation).

Module · sub-tabs	What a run produces
Quant Reporting **IC Analysis · Cumulative Returns · Risk Report · Model Performance · Factor Quantiles**	Institutional tearsheets from pasted prediction/return arrays — IC/ICIR with rolling stability, cumulative-return attribution vs a benchmark, a drawdown+VaR/CVaR risk report, a model scorecard, and factor-quantile spreads.
CFA Quant **12-analysis dropdown**	CFA-curriculum quant: trend/stationarity/ARIMA/forecasting, supervised & unsupervised ML, model

	evaluation, resampling, sampling techniques, CLT, sampling-error, data validation. Ticker inputs fetch via Yahoo.
GS Quant **Risk Metrics · Portfolio · Greeks · VaR · Stress Test · Backtest · Statistics**	Risk-ratio suite, benchmark attribution, Black-Scholes Greeks (to 3rd order), four-method VaR, 9-scenario historical stress test, a 4-strategy Yahoo backtest, and descriptive stats.
Statsmodels **OLS · ARIMA · Stationarity · ACF/PACF · Granger · Descriptive**	Real econometrics: OLS with full inference & residual diagnostics, ARIMA with forecast bands, ADF+KPSS, ACF/PACF order-ID, Granger causality, descriptives + normality tests.
Functime **Forecast · Anomalies · Seasonality · Metrics · Intervals · Stationarity**	Forecast with 9 models, detect anomalies four ways, STL-decompose, score accuracy, build bootstrap/residual intervals, auto-recommend a differencing order.
Fortitudo **Portfolio Metrics · Covariance · MV/CVaR Optimize · Frontier · Decay Weights**	From a ticker list (Yahoo): annualized risk/return, VaR/CVaR, covariance & correlation, mean-variance and mean-CVaR optimization, the efficient frontier, and exp-decay scenario weights.
GluonTS **Probabilistic · Quantile · Distribution Fit · Evaluate · Seasonal Naive**	Full predictive distributions via bootstrap ensembles: probabilistic & custom-quantile forecasts, candidate distribution fitting (AIC/BIC/KS), forecast evaluation with interval calibration, and a seasonal-naive baseline.

Honest ANALYTICS notes. “GS Quant” runs an **in-house GS-style wrapper**, not Goldman's `gs-quant` SDK (no Marquee/network except a Yahoo backtest). **Functime** and **GluonTS** are lighter self-contained reimplementations, not the upstream libraries. **Fortitudo** fetches from Yahoo Finance. All four are zero-credit local compute.

6 The engine

ENGINE

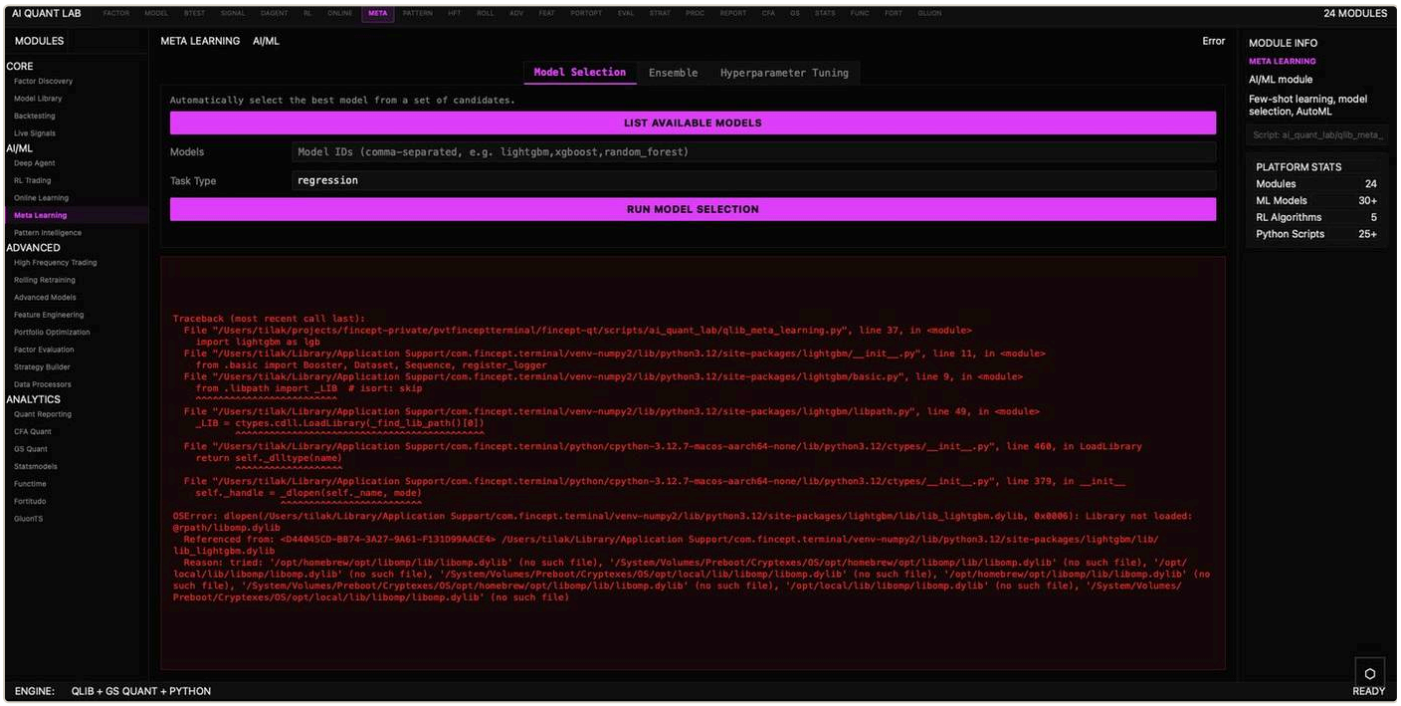
One service backs all 24 modules: it maps each module to a Python script, runs it as a subprocess, and streams the JSON result back. Understanding credits, network and dependencies explains every module's behaviour.

6.1 Credits & network — the decisive facts

Behaviour	Which modules
Spends credits (hosted AI)	**Deep Agent only** — routes to the managed Fincept-AI model.
Local compute, offline	Statsmodels, Functime, GluonTS, CFA (non-ticker), Quant Reporting, and the ADVANCED Qlib-suite compute modules (Portfolio Opt, Factor Eval, Strategy Builder, Data Processors, Feature Engineering) — **run on pasted / sample data, no network.**
Free but need internet	**HFT** (crypto exchanges), **Backtesting** & **Fortitudo** & GS-Backtest & CFA-ticker (Yahoo Finance).
Need a local Qlib dataset	Model Library training, Factor Discovery data tabs, Live Signals — download once to `~/.qlib`.

6.2 When a native library is missing

Each module imports its own Python packages. If one isn't installed (or a native library can't load), the module returns a clean, red error card with the traceback — the terminal never crashes. Below, Meta Learning surfaces a real example: LightGBM can't load because the host is missing the OpenMP runtime.



An honest error state — Meta Learning → LIST AVAILABLE MODELS returns a red card: `OSError … lib_lightgbm.dylib: Library not loaded: libomp.dylib`. Install the missing runtime and it works.

> **Cold start, concurrency & persistence.** The first run of a given script per session is slower (the Python interpreter + heavy imports load once). At most three Python subprocesses run at a time; extra runs queue. Results are not saved across sessions — use **EXPORT RESULTS** to write a run's JSON into the File Manager; only the selected module index is remembered.

7 Driving Quant Lab by AI

MCP · REFERENCE

The whole Lab is exposed to the terminal's AI as **100 MCP tools** in the `quant-lab` category, so an analyst can run any module by asking in natural language.

Tool group	Count	What it does
Generic dispatch	3	`list_quant_modules` (the 24 modules), `list_quant_module_commands` (commands per module), `run_quant_module` (run any module command with JSON params).
Per-command tools	97	One named, async tool per module command — e.g. run an OLS regression, price Greeks, optimize a portfolio, browse the factor library — each mapping directly onto a module's Python command.

Because the MCP tools reuse the exact same Python dispatch as the buttons, they inherit the same credit and network characteristics: only the Deep-Agent path spends credits; the rest are local compute.

> **What this means for an institution.** A quant can drive the Lab from the chat bar — “fit an ARIMA(2,1,2) to this series and give me the forecast bands”, or “run a max-Sharpe optimization on this covariance matrix” — and the answer comes back as the same structured result the module renders.

8 Quick reference & caveats

REFERENCE

8.1 How do I…

I want to…	Do this
Open the Quant Lab	FKeyBar → **QUANT LAB**, or command bar → `quantlab`.
Switch module	Click a left-rail module (grouped) or a top-strip code.
Get data into a form fast	Press **LOAD SAMPLE** next to an input.
Browse Qlib's alpha factors	CORE → Factor Discovery → **BROWSE FACTOR LIBRARY**.
Backtest a momentum basket	CORE → Backtesting → tickers + dates → **RUN BACKTEST**.
Fit a regression	ANALYTICS → Statsmodels → OLS → LOAD SAMPLE → **RUN OLS REGRESSION**.
Compute risk metrics / VaR	ANALYTICS → GS Quant → Risk Metrics / VaR → LOAD SAMPLE → run.
Optimize a portfolio	ADVANCED → Portfolio Optimization → pick a method → enter covariance → run.
Forecast with uncertainty	ANALYTICS → GluonTS / Functime → LOAD SAMPLE → run.
Run an autonomous analysis	AI/ML → Deep Agent → describe the task → **RUN DEEP ANALYSIS** (spends credits).
Keep a result	Press **EXPORT RESULTS** — writes JSON to the File Manager.

8.2 Honest caveats worth knowing

- **Credits.** Only **Deep Agent** spends credits (hosted model). Everything else is local Python — free.
- **Dependencies.** ML-heavy modules need their native libraries (LightGBM/PyTorch/River); a missing one shows a red error card, not a crash.
- **Qlib data.** Model training and Factor-Discovery data tabs need a one-time `~/.qlib` dataset download; the factor-library list works without it.
- **Dead paths.** Model Library → Backtest and Live Signals → Factor Analysis / Feature Importance error because model state doesn't persist across the subprocess boundary.
- **Naming.** “GS Quant” is an in-house GS-style wrapper (not Goldman's SDK); Functime & GluonTS are lighter reimplementations of those libraries.
- **Backtesting.** The “Commission” cost figure is actually market-impact bps; the benchmark defaults to SPY.
- **Persistence.** Results aren't saved between sessions (export to keep them); only the selected module index is remembered.
- **Platform stats.** Only “Modules 24” is live; the other stat tiles and the ENGINE/READY bar are fixed labels.
- **Pattern Intelligence** is a raw command runner with no curated UI — treat as experimental.

> **Fincept Research Desk.** This guide documents the AI Quant Lab as shipped in Professional Edition v5.0. Screenshots are live captures, including genuinely computed results; module names and behaviour are transcribed from the shipping build.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Alpha Arena

A complete, screen-by-screen walkthrough of Alpha Arena — a live competition where AI models trade crypto perpetuals against each other on real market data, tick by tick.

`>_ AI vs AI · real Hyperliquid marks · paper & live venues · full audit trail`

Covers · **the live dashboard** · **the competition wizard** · **running & controls**
the trading engine · **live mode & its gate** · cost, data & AI tools

Demonstrated live on the Fincept Terminal **Alpha Arena** tab.

Contents

8 SECTIONS

GETTING ORIENTED

01 What Alpha Arena is 3

02 Anatomy — the live dashboard 4

RUNNING A COMPETITION

03 Setting up — the competition wizard 5

04 Running & controlling a competition 6

05 Reading the arena — the six panels 7

REFERENCE

06 Under the hood — the trading engine 9

07 Live mode & its beta status 10

08 Cost, AI tools & honest notes 12

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal. A live paper competition was created and run for several rounds while capturing — the model reasoning, positions, trades and equity curve shown are genuine engine output. Control labels are quoted exactly as they appear on screen.

01 What Alpha Arena is

OVERVIEW

Alpha Arena is a **trading competition for AI models**. You set up a contest, and each competing model is handed a paper (or live) account and asked, every round, to trade a universe of crypto perpetual futures on **real Hyperliquid market data**. You watch their reasoning, positions and equity in real time and see which model comes out ahead.

It is a research and benchmarking tool: a controlled, fully-audited environment for asking "how does this model actually trade?" Every prompt, decision, order and fill is stored, so any round can be replayed. Paper mode is a high-fidelity simulator (real marks, real funding, real fees); a beta live mode can route to Hyperliquid.

AT A GLANCE, IN ALPHA ARENA YOU CAN…

- Run a **competition** of AI models trading crypto perps.
- Trade on **real Hyperliquid** prices, fees and funding.
- Read each model's **reasoning** for every trade.
- Watch **positions, trades & equity** update live.
- Set the **universe, capital, cadence & leverage**.
- **Force a round**, pause, or kill the whole contest.
- Run on **paper** — or gate into a beta **live** venue.
- Replay any round from a full **audit trail**.

How to get here

Open the **Navigate** menu (or the command bar, top-left) and type `alpha`, then pick `Alpha Arena`. If no competition is active you'll see the empty state with a single **+ NEW COMPETITION** button.

> **RESEARCH TOOL — NOT ADVICE**
>
> Alpha Arena is for research and benchmarking. Past performance is not indicative of future results, and models can produce poor or harmful trades. Live mode can forward real orders to a real venue and may incur real losses (see §7).

02 Anatomy — the live dashboard

LAYOUT

Once a competition is running, the screen is a live dashboard: a two-strip **header** on top, a large **equity curve** and the **leaderboard** in the upper half, and a grid of five **panels** across the bottom.

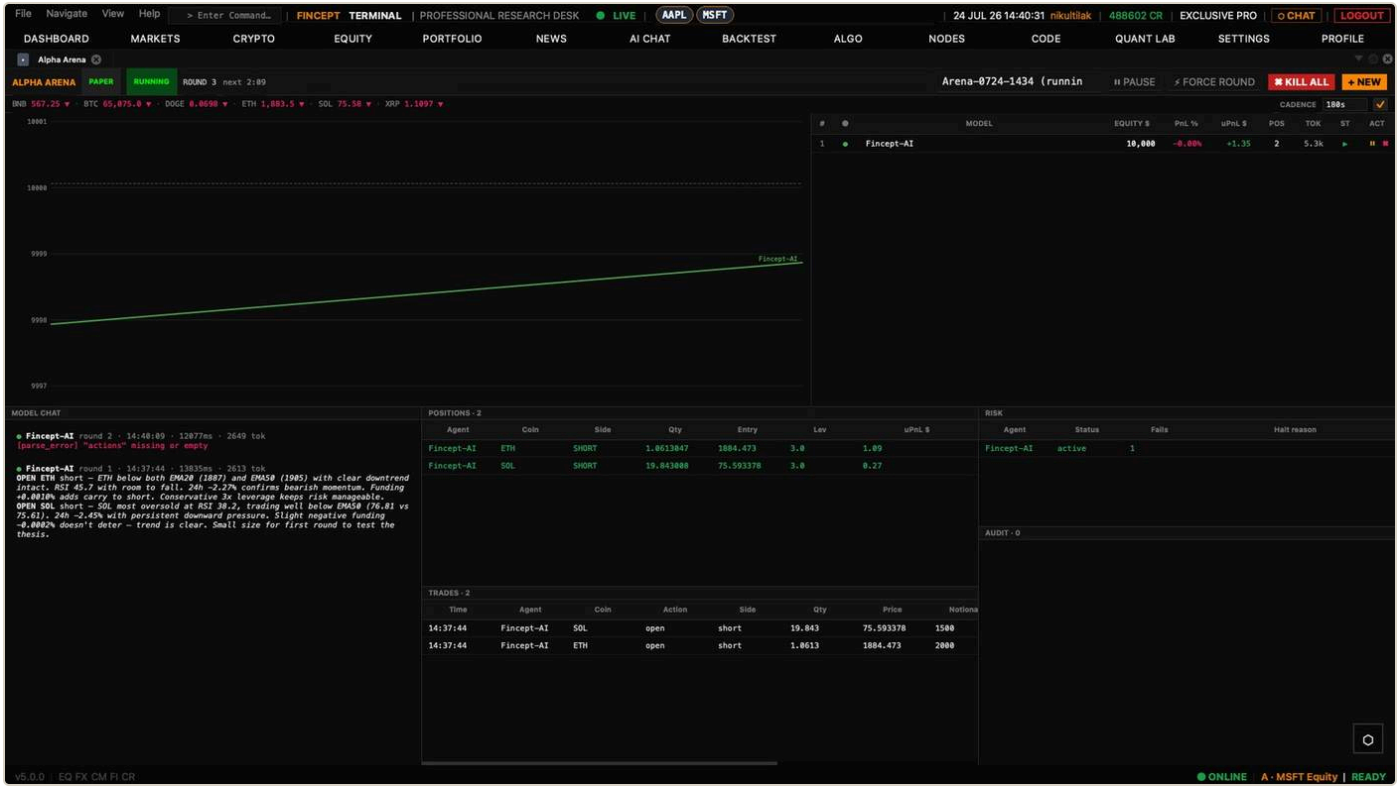


Fig 1. A live paper competition mid-run. Top: header (PAPER · RUNNING · ROUND 3) + the mids ticker. Upper: the equity curve (left) and leaderboard (right). Bottom: MODEL CHAT · POSITIONS · TRADES · RISK · AUDIT.

Region	What it shows
Header (row 1)	Title · PAPER/LIVE badge · IDLE/RUNNING state · ROUND N · countdown · the competition selector · PAUSE · FORCE ROUND · KILL ALL · + NEW (§4).
Header (row 2)	A live **mids ticker** (BTC/ETH/SOL…) and the **CADENCE** control.
Equity curve	One line per model, equity over time, with a dashed starting-capital baseline (§5).
Leaderboard	Every model ranked by equity, with PnL, positions, tokens and status (§5).
Panel grid	MODEL CHAT · POSITIONS · TRADES · RISK · AUDIT — always visible, no tabs (§5).

Fig 2. The two header strips — controls on row 1, the live mids ticker and CADENCE on row 1, the live mids ticker and CADENCE on row 2.

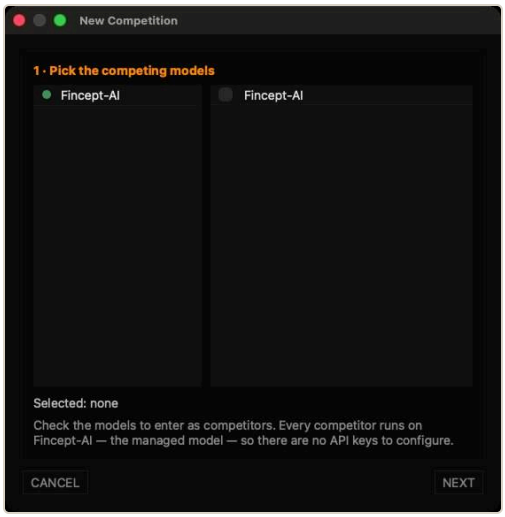
> **THE SCREEN IS A VIEWER**
>
> The competition runs in a background engine, not in the screen — so it keeps ticking even if you switch tabs, and you can re-open Alpha Arena to find it exactly where it is. The competition selector lets you view past competitions while a different one runs.

03 Setting up — the competition wizard

NEW

Click + **NEW COMPETITION** to open the two-step wizard.

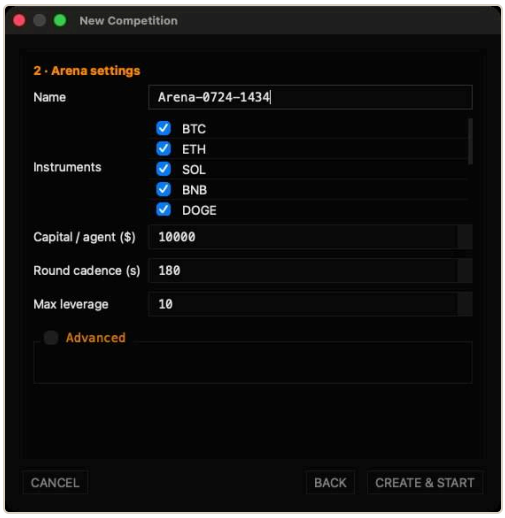
Step 1 · Pick the competing models



Check the models that will compete — each checked model becomes one competing agent. In this build the available model is **Fincept-AI**, the managed model, so there are no API keys to configure. As the dialog itself says: "Every competitor runs on Fincept-AI — the managed model."

Press **NEXT** once at least one model is checked.

Step 2 · Arena settings



Setting	Default
Name	Auto (Arena-MMdd-HHmm)
Instruments	BTC, ETH, SOL, BNB, DOGE, XRP (+ AVAX, LINK, ARB, OP available)
Capital / agent	$10,000 (100–10M)
Round cadence	180 s (60–3600)
Max leverage	10× (1–50)
Advanced	collapsed → Paper

The **Advanced** section (collapsed by default) holds an optional **custom system prompt** and the **venue** choice — **Paper** (the default) or **Hyperliquid LIVE** (§7). Press **CREATE & START** to launch; the competition begins immediately and the first round can be triggered at once.

> **THE PERP UNIVERSE**
>
> Ten crypto perpetuals are offered — **BTC, ETH, SOL, BNB, DOGE, XRP** (checked by default) plus **AVAX, LINK, ARB, OP**. Models may hold one position per coin and flip sides between rounds.

04 Running & controlling a competition

CONTROLS

Everything you need to drive a live competition sits in the header.

Control	What it does
⚡ **FORCE ROUND**	Runs a decision round **immediately** — every active model is prompted, trades, and the board updates. No confirmation. (Each round uses AI credits — see §8.)
⏸ **PAUSE** / ▶ **RESUME**	Pauses or resumes the automatic round timer.
CADENCE (row 2)	Seconds between automatic rounds (60–3600, default 180). Press ✓ to apply.
✖ **KILL ALL**	Closes every model's positions and **ends** the competition, after a confirmation. It's the master stop.
Competition selector	Switches which competition you're *viewing*. Past/ended ones are labeled `(ended)`.
+ NEW	Opens the wizard (blocked while a competition is running — KILL ALL first).

Per-model controls & kill switches

Each leaderboard row has its own **halt/resume** and **kill** buttons (in the **ACT** column) — halt pauses one model (keeping its positions), kill closes its positions and stops it. On top of that, the engine **auto-halts** a model after **3 consecutive decision failures** (a "circuit breaker"), flagging it with a ⚡ in the leaderboard.

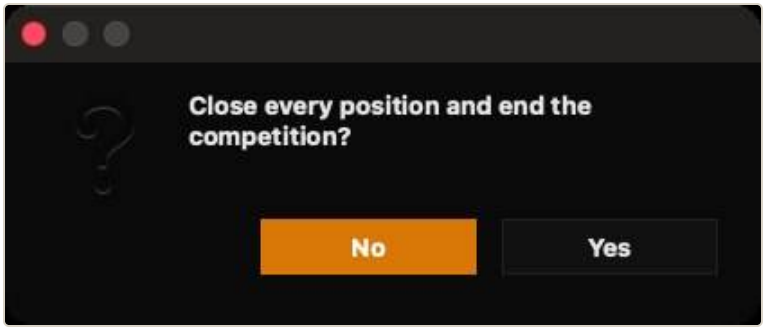


Fig 3. KILL ALL asks before it ends the competition and closes positions.

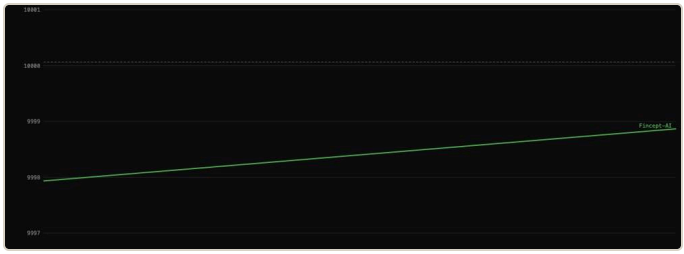
> **CRASH RECOVERY**
>
> If the app was closed while a competition was running, on next open Alpha Arena offers to **Resume** it (or dismiss it) — nothing is silently lost.

05 Reading the arena — the six panels

PANELS

Leaderboard & equity curve

The **leaderboard** ranks every model by equity (rank 1 = highest). Columns: rank · color · **MODEL** · **EQUITY $** · **PnL %** · **uPnL $** · **POS** (open positions) · **TOK** (tokens used) · **ST** (status: ▶ active, ⚡ circuit-halted, ⏸ user-halted) · **ACT** (halt/kill). The **equity curve** plots one line per model over time against a dashed starting-capital baseline; hover for a value tooltip.



#	●	MODEL	EQUITY $	PnL %	uPnL $	POS	TOK	ST	ACT
1	●	Fincept-AI	10,000	-0.00%	+1.35	2	5.3k	▶	⏸ ✖

Fig 4. The leaderboard (top) and the equity curve (left), from a live run — Fincept-AI's equity ticking up over three rounds.

MODEL CHAT — the reasoning

This is the heart of the arena: each model's own explanation for what it did, per round, with timing and token counts. A failed/unparseable decision is flagged in red.

```
MODEL CHAT

● Fincept-AI round 2 · 14:40:09 · 12077ms · 2649 tok
[parse_error] "actions" missing or empty

● Fincept-AI round 1 · 14:37:44 · 13835ms · 2613 tok
OPEN ETH short — ETH below both EMA20 (1887) and EMA50 (1905) with clear downtrend
intact. RSI 45.7 with room to fall. 24h -2.27% confirms bearish momentum. Funding
+0.0010% adds carry to short. Conservative 3x leverage keeps risk manageable.
OPEN SOL short — SOL most oversold at RSI 38.2, trading well below EMA50 (76.81 vs
75.61). 24h -2.45% with persistent downward pressure. Slight negative funding
-0.0002% doesn't deter — trend is clear. Small size for first round to test the
thesis.
```

Fig 5. Real model output — "OPEN ETH short — ETH below both EMA20/50… RSI 45.7 with room to fall… Conservative 3× leverage." A later round shows a red `[parse_error]` that the engine counts toward the circuit breaker.

POSITIONS & TRADES

POSITIONS · 2

Agent	Coin	Side	Qty	Entry	Lev	uPnL $
Fincept-AI	ETH	SHORT	1.0613047	1884.473	3.0	0.98
Fincept-AI	SOL	SHORT	19.843008	75.593378	3.0	-0.58

TRADES · 2

Time	Agent	Coin	Action	Side	Qty	Price	Notiona
14:37:44	Fincept-AI	SOL	open	short	19.843	75.593378	1500
14:37:44	Fincept-AI	ETH	open	short	1.0613	1884.473	2000

Fig 6. POSITIONS (open perps with side, qty, entry, leverage, live uPnL) and TRADES (the fill log — time, coin, action, side, qty, price, notional).

RISK & AUDIT

RISK

Agent	Status	Fails	Halt reason
Fincept-AI	active	1	

AUDIT · 0

RISK shows each model's live status and its consecutive-failure count — the number that trips the auto-halt circuit breaker at 3.

AUDIT tails the append-only event stream (round starts, kills, market-data errors, live-mode events). It's the human-readable face of the full record that makes every competition replayable round by round.

06 Under the hood — the trading engine

ENGINE

How a round works

On each round the engine pulls a fresh market snapshot (real Hyperliquid prices, plus computed indicators — EMA, RSI, ATR), builds a prompt for each model, and calls them concurrently. Each reply is parsed into trade actions, checked by the risk engine, and — if approved — executed on the venue. Equity is then snapshotted. Between rounds, prices are re-marked every ~10 s so equity stays live.

The four actions

Action	What it does
`open` · long	Open or add to a long. Same coin, opposite side → flips the position.
`open` · short	Open or add to a short.
`close`	Close a position — fully, or partially by dollar size.
`hold`	Do nothing this round.

A reply that can't be parsed gets **one repair retry**; if it still fails it's recorded as a parse error and counts toward the model's circuit breaker.

The risk engine

Rule	Limit
Max leverage	Per competition (1–50×, default 10×)
Max size per position	50% of equity
Max new notional per round	100% of equity
Auto-halt (circuit breaker)	3 consecutive decision failures

Each proposed trade is **accepted or rejected** against these limits; rejected trades are logged (no partial "amend"). Closes and holds never add risk, so they're always allowed.

Paper venue economics (high-fidelity)

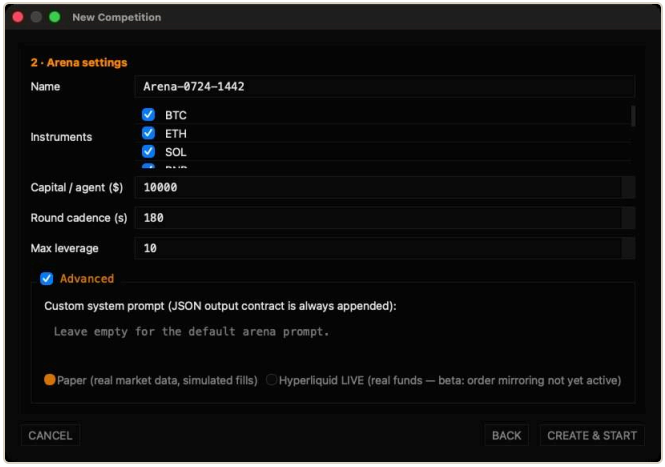
Paper mode is a faithful simulator, not a toy: prices are the **real Hyperliquid marks**, with Hyperliquid-style **taker fees (~4.5 bps)**, real **funding**, and isolated-margin **liquidation** at a 2.5% maintenance margin. It adds a small fixed **2 bps slippage** per side and settles instantly.

> **REPLAYABLE BY DESIGN**
>
> Every round stores its full prompt, raw model reply, parsed actions, orders and equity snapshot in a local database, and the event log is append-only — so any competition can be reconstructed round by round for audit.

07 Live mode & its beta status

LIVE

By default a competition runs on **paper**. In the wizard's **Advanced** section you can instead choose **Hyperliquid LIVE** — and the label itself flags it as **beta**.



Choosing Live and pressing **CREATE & START** triggers a hard gate before anything runs: the build's signer must be available, your locale must pass a geofence, and you must clear the **Live Mode** dialog.

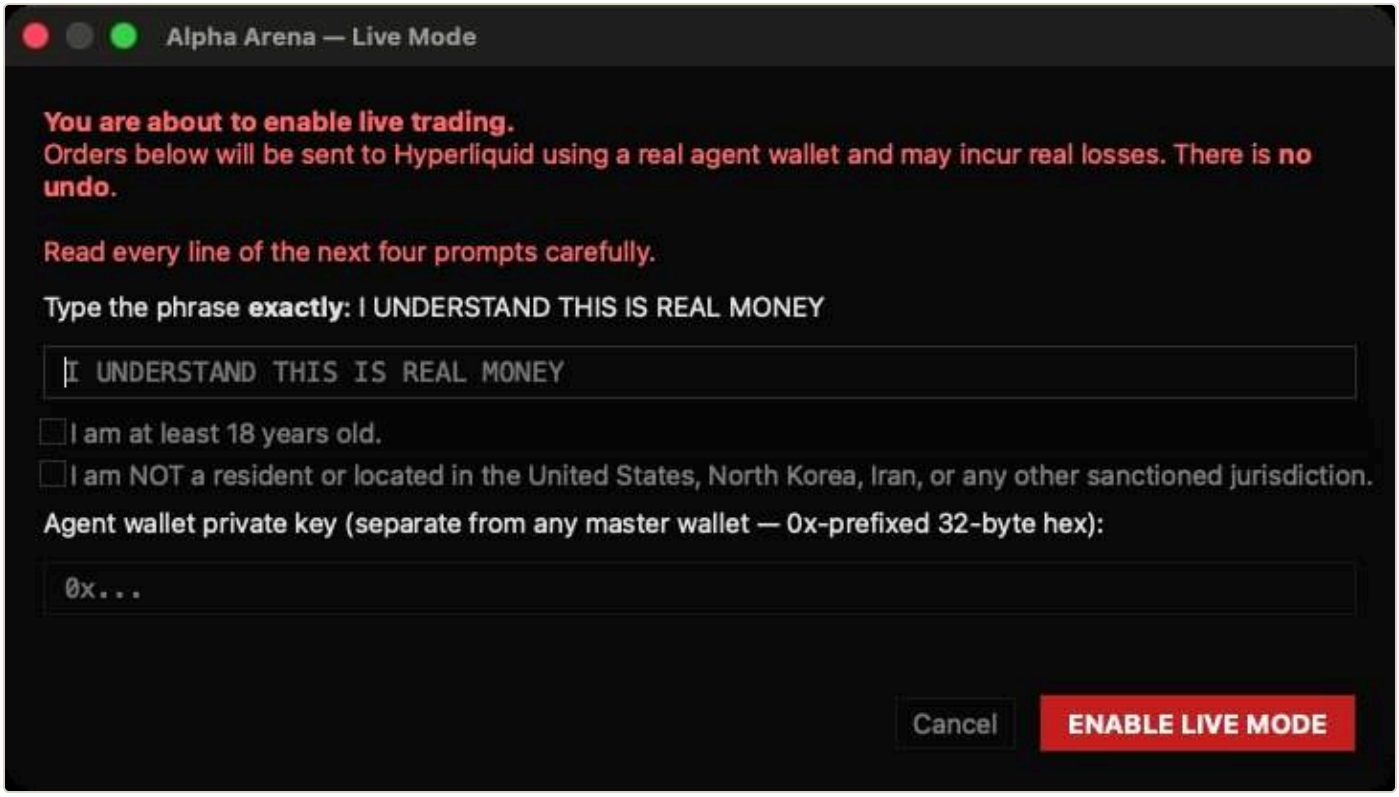


Fig 7. The Live Mode gate — a typed acknowledgement ("I UNDERSTAND THIS IS REAL MONEY"), two mandatory checkboxes (age; not in a sanctioned jurisdiction), and a per-competition agent-wallet private key stored in the OS keychain. ENABLE stays disabled until all four are satisfied.

To enable live mode

1. In the wizard's **Advanced** section, select **Hyperliquid LIVE** as the venue.
2. Press **CREATE & START** — the Live Mode gate opens (the build's signer and your locale are checked first).

3. Type `I UNDERSTAND THIS IS REAL MONEY` exactly, tick both checkboxes, and paste an **agent-wallet private key** (0x + 64 hex — use a dedicated wallet, never your main one).
4. **ENABLE LIVE MODE** unlocks only when all four are satisfied; the key is saved to your OS keychain and the competition starts in live mode.

> **IMPORTANT — LIVE IS BETA; ORDERS ARE NOT YET FORWARDED**
>
> In this build, live mode is **not yet wired to place real orders** — the venue label reads "order mirroring not yet active," and live orders are recorded but rejected internally rather than sent to Hyperliquid. **The gate does, however, store a real private key** in your keychain. Treat live mode as a preview: don't assume orders execute, and don't provide a key you're not comfortable storing.
>
> The **geofence** (blocking sanctioned jurisdictions) is an **advisory development guard** keyed off your OS region — not a compliance control. Final responsibility for legality and compliance is yours.

08 Cost, AI tools & honest notes

REFERENCE

What it costs

Every decision a model makes is an AI call billed to your Fincept **credit balance** — the same managed model that powers AI Chat. So the cost of a competition is roughly **(number of models) × (number of rounds)** AI calls. A forced round bills on demand; the automatic timer bills every cadence interval. Between-round price marks are free. The arena screen doesn't show a running cost, so watch your balance on long runs with a tight cadence.

AI tools (read-only)

Five tools (category `alpha-arena`) let the assistant inspect competitions — they read only; all control stays in the UI.

Tool	What it returns
`arena_list_competitions`	All competitions, newest first, with their settings.
`arena_leaderboard`	The ranked board for one competition.
`arena_recent_decisions`	Recent model decisions (actions, tokens, latency).
`arena_list_orders`	Executed / rejected orders for a competition.
`arena_status`	Live engine status — active competition, round, countdown.

Honest notes & current limitations

- **Every round spends credits.** All models run on the managed Fincept-AI model; there is no free or bring-your-own-provider path in this build (despite the empty-state tagline).
- **One model competes today.** The picker is built for many, but the current build ships a single managed model, so a competition is effectively one contestant until more are added.
- **Live mode is beta.** It gates and stores a real key but does not yet forward orders to Hyperliquid (§7).
- **The geofence is advisory**, based on OS region — not a compliance guarantee.
- **Paper realism is high but bounded** — real marks/fees/funding, but instant fills and a fixed 2 bps slippage regardless of order size.
- **Replay ≠ reproducibility.** The record is complete and auditable, but the model itself is non-deterministic, so re-running won't reproduce identical trades.
- **All arena data is stored locally** on this machine.

> **FINCEPT RESEARCH DESK**
>
> Prepared for institutional clients · Professional Edition v5.0 · Captured live on the Alpha Arena tab, running a real paper competition. Alpha Arena is a research and benchmarking tool — not financial advice, and not a regulated brokerage.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Global Data

A complete, screen-by-screen walkthrough of the Global Data tab — one explorer over the platform's own indexed store, in place of the five per-provider screens it replaced.

`>_ 28 sources · 12 topics · 2 charts · ask in plain English`

Covers · **search and the topic tree** · **Ask AI** · **the two-chart stage** · **AI analysis** windows, styles and transforms · the data table · CSV export · what each action costs

Demonstrated live on the Fincept Terminal Global Data tab.

Contents

18 SECTIONS

GETTING ORIENTED

01 What the Global Data tab is 3
02 Anatomy of the screen 4
03 The command strip 5
04 The twenty-eight sources 6

FINDING SOMETHING

05 Browsing by topic 7
06 Reading the results list 8
07 Ask AI — questions instead of keywords 9

PUTTING IT ON A CHART

08 Loading a series 10
09 The series rail 11
10 Window, style and transform 12
11 Comparing — two charts, one screen 13
12 Reading the chart 14

GETTING IT OUT

13 The data table 15
14 AI analysis of what you charted 16
15 Exporting to CSV 17

REFERENCE

16 What it costs 18
17 What carries over between sessions 19
18 Quick reference and honest notes 20

About this guide. Every screenshot is a live capture of the Global Data tab in the shipping build, and every behaviour described here was read from the shipping source and then confirmed on screen — including the ones that are inconvenient.

01 What the Global Data tab is

ORIENTATION · 01

Global Data is **one explorer over the whole indexed store** — twenty-eight statistical agencies, central banks and international bodies, searched together, charted together and exported together. You find a series by topic, by keyword or by asking a question in plain English; you put it on one of two charts; and you read it, compare it, tabulate it or take it away as CSV.

It replaces five screens. **Economics**, **Government Data**, **DBnomics**, **AkShare** and **Asia Markets** were each a window onto one provider's own API, with its own search box, its own quirks and its own idea of what a series was. Everything now arrives through a single catalogue with common fields — title, source, region, cadence, coverage, units — so a German series and a Brazilian one can sit on the same chart without you translating between two products.

Fig 1 — **GLOBAL DATA** is a top-level tab, third along the main bar.

Where to find it

Route	How
Main tab bar	**GLOBAL DATA**, third from the left — it is a top-level tab, not a menu entry
Navigate menu	**Navigate ‣ Markets & Data ‣ Global Data**
Command bar	`gd`, `global` or `data` — and the retired tabs' own names still work: `econ`, `economics`, `govt`, `gov`, `dbnomics`, `akshare`, `asia`, `apac` all land here

At a glance, in Global Data you can…

- Search 28 sources at once
- Browse a twelve-branch subject tree
- Ask for data in plain English
- Chart up to six series per pane
- Compare on two independent charts
- Rebase to 100 or switch to % change
- Read exact values with a crosshair
- Read the observations as a table
- Export everything on stage as CSV
- Have the AI write up what you charted

> **Muscle memory still works.** The five retired tabs left their names behind as command-bar aliases. Typing `akshare` or `govt` does not report an unknown screen — it opens Global Data, which is where that data now lives.

02 Anatomy of the screen

Two halves, split horizontally and draggable: the **stage** on top, where the charts live, and the **discovery dock** below, where you find things to put on them.

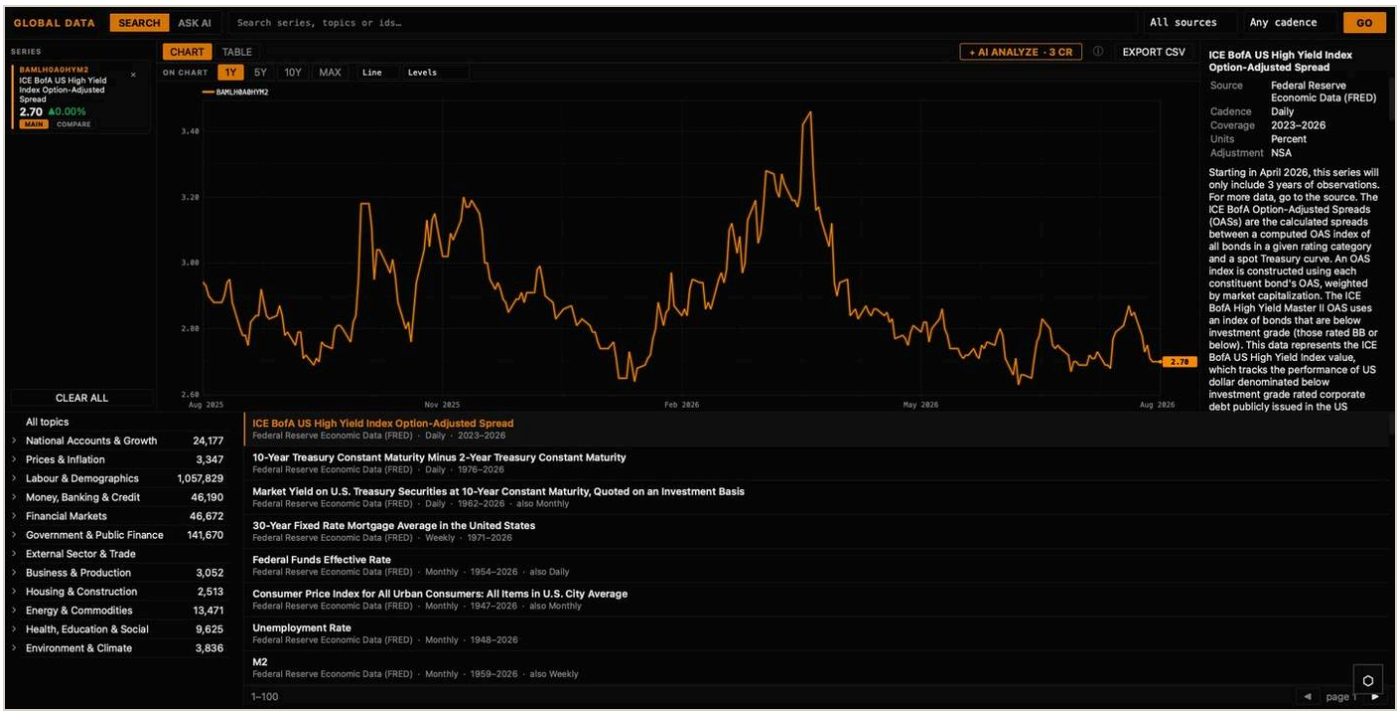


Fig 2 — The tab as it opens. The stage holds the series rail, the chart panes and the About panel; the dock holds the topic tree, the results list and the status bar. One series is already charted because a search landed on an empty stage.

#	Region	What it is for
1	Command strip	**SEARCH** / **ASK AI**, the query box, the source and cadence filters, and **GO**.
2	Series rail	One card per loaded series: colour key, short id, full name, latest value and change, which chart it is on, and a ✕ to drop it. **CLEAR ALL** sits at the bottom.
3	Stage actions	**CHART** / **TABLE**, then ✦ **AI ANALYZE** and **EXPORT CSV** on the right.
4	Chart panes	**ON CHART** always; **COMPARE** appears beside it only once you put something on it. Each pane owns its window, style and transform.
5	About panel	The selected series' source, region, cadence, coverage, units and adjustment, then its description. After an analysis it holds the written read instead.
6	Topic tree	Twelve subject branches with per-node counts, expandable to sub-topics. Clicking one filters the results.
7	Results list	The matches — title on top, then source · region · cadence · coverage.
8	Status bar	The range shown, and ◀ **page N** ▶. Errors appear here in red.

03 The command strip

Everything that starts a search is on one row.

Fig 3 — The strip in **SEARCH** mode: a plain keyword box, two filters, and a **GO** button that carries no price.

Control	What it does
`SEARCH / ASK AI`	The two ways in. **SEARCH** matches your words against the catalogue. **ASK AI** sends the sentence to an interpreter that turns it into topic, region, cadence and provider filters.
`Query box`	Free text. **Enter** runs the search — you do not have to reach for the button. The × at the right clears it.
`All sources`	Restrict to one of the 28 providers. **Disabled in ASK AI mode**, because the interpreter picks the provider itself.
`Any cadence`	Daily, Weekly, Monthly, Quarterly or Annual. Also disabled in ASK AI mode.
`GO / ASK`	Runs it. In ASK AI mode the button becomes **ASK** and shows the price before you press it.

Fig 4 — The same strip in **ASK AI** mode. The placeholder changes to an example question and the button states its cost — the price is on the control, before the click.

> **A sentence is charged even in SEARCH mode.** The tab watches what you type: a query of six or more words, or three or more beginning with *give, show, find, get, tell, what, how, why, which, where, compare, can, could, do, does, is, are, I want, I need*, is routed to the AI interpreter whatever mode you are in — because literal matching on a sentence returns whichever titles happen to contain those words. That routing is metered exactly as **ASK** is. §07 explains how to tell, after the fact, which path answered you and whether you were charged.

04 The twenty-eight sources

The **All sources** filter is the full inventory. Everything in the list is indexed into the same catalogue; picking one here narrows a search to that publisher.

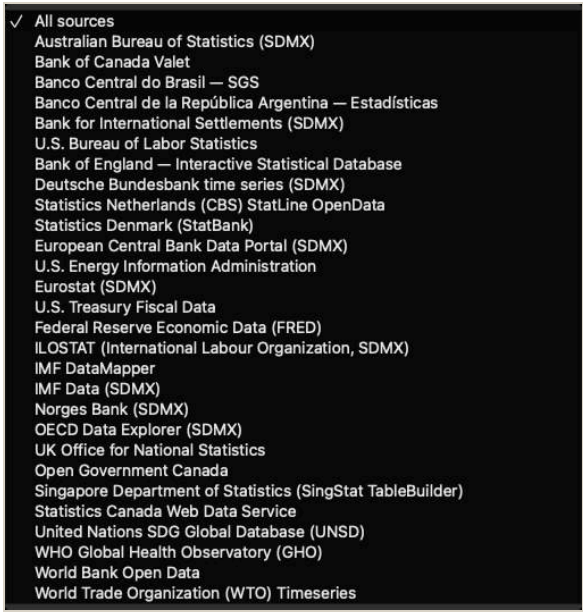


Fig 5 — Every provider, listed in the tab's own order (by internal id, which is why Australia leads and the WTO closes).

Kind	Who
`National statistics`	Australian Bureau of Statistics · Statistics Netherlands (CBS) · Statistics Denmark · UK Office for National Statistics · Statistics Canada · Open Government Canada · Singapore Department of Statistics
`Central banks`	Bank of Canada · Banco Central do Brasil · Banco Central de la República Argentina · Bank of England · Deutsche Bundesbank · European Central Bank · Norges Bank
`US federal`	Federal Reserve Economic Data (FRED) · Bureau of Labor Statistics · Energy Information Administration · Treasury Fiscal Data
`International`	BIS · Eurostat · IMF DataMapper · IMF Data · OECD · World Bank · WTO · ILOSTAT · UN SDG Global Database · WHO Global Health Observatory

Ten of the twenty-eight are reached over **SDMX**, the statistical exchange standard, which is what makes a single catalogue possible: the field names arrive already agreed. The rest are read through each publisher's own API and mapped into the same shape on the way in.

> **There is deliberately no catalogue-size counter in the header.** An earlier build totalled the inventory across the top of the tab; it was removed because it advertised the tab by how much it held rather than by whether it held what you asked for. The counts that remain are the per-topic ones in the tree, which are actionable.

05 Browsing by topic

When you do not have a search term, start from the subject tree. Twelve branches, each with a count and expandable sub-topics; clicking any node filters the results to it and its children.

Fig 6 — The tree with **Prices & Inflation** expanded. Selecting a node also raises a dismissable chip above the stage so the filter is never invisible.

Branch	Count
National Accounts & Growth	24,177
Prices & Inflation	3,347
Labour & Demographics	1,057,829
Money, Banking & Credit	46,190
Financial Markets	46,672
Government & Public Finance	141,670
External Sector & Trade	—
Business & Production	3,052
Housing & Construction	2,513
Energy & Commodities	13,471
Health, Education & Social	9,625
Environment & Climate	3,836

A branch's number is not its subtree's total. *Prices & Inflation* reads 3,347, but expanding it shows Consumer Prices 65,400, Producer & Import Prices 41,403 and Deflators & Indices 372 — over a hundred thousand between them. The figure beside a node counts what is labelled at **that node itself**, while clicking it searches the node **and everything under it**. Read the counts as a rough sense of where the depth is, not as a promise of result volume. A branch showing no number at all, like External Sector & Trade, still has children worth opening.

Clicking **All topics** clears the filter, as does the × on the topic chip above the stage.

06 Reading the results list

Each row is one series: the title, wrapped to at most three lines, over a metadata line that tells you whether it is the one you want before you spend anything loading it.

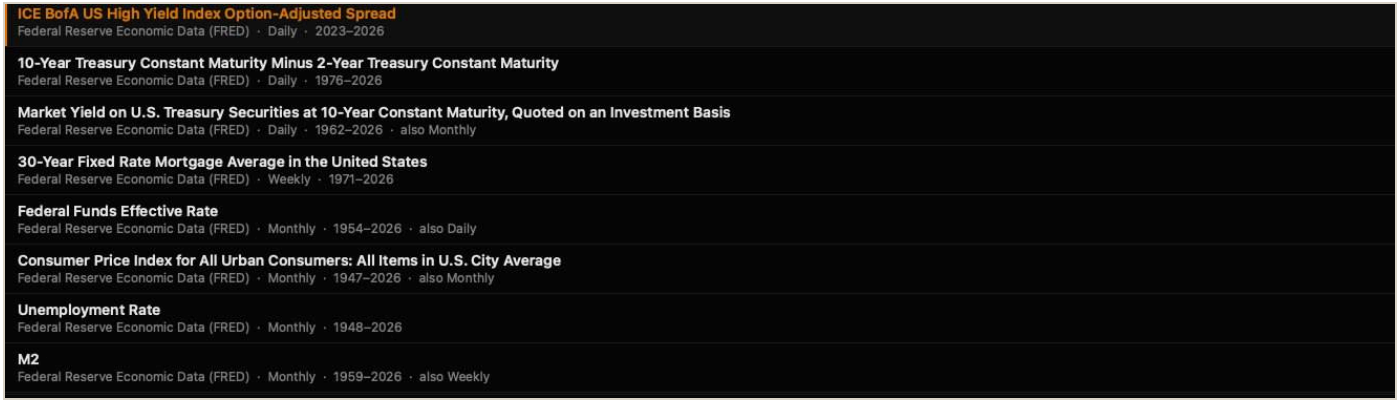


Fig 7 — Results for the default landing search. The selected row is amber with a bar down its left edge. Note the **also Monthly** and **also Daily** suffixes.

Field	Meaning
`Title`	The publisher's own name for the series — never shortened, only wrapped.
`Source`	Which of the 28 providers it comes from.
`Region`	Resolved to a country name where the code allows it; `EA` and `EU` become Euro Area and European Union.
`Cadence`	Daily, Weekly, Monthly, Quarterly or Annual.
`Coverage`	First and last year, e.g. `1947–2026`.
`also …`	Sibling cadences — see below.

Why near-duplicates are folded together

Publishers issue the same indicator at several cadences: FRED carries the CPI monthly and quarterly, the Federal Funds rate monthly and daily. Ranked literally, those sit next to each other and read as accidental duplicates. Rows sharing a title, provider and region are therefore **collapsed into the best-ranked one**, and the cadences that were folded away are named at the end of the metadata line as **also Monthly**, **also Daily** and so on.

> **The folded siblings are still reachable** — set the **cadence** filter to the one you want and search again, and that variant becomes the row that survives.

Fig 8 — The status bar: the range on show, then the pager. Pages are 100 rows; ▶ is live only while a full page came back, so it greys out on the last one.

07 Ask AI — questions instead of keywords

Switch to **ASK AI** and write the question you actually have. The sentence goes to an interpreter that knows the taxonomy and the provider list, and comes back as a set of filters — topic, region, cadence, provider, and exact series ids where it recognises them — which then run as an ordinary catalogue query.

Fig 9 — After asking *"how fast are US wages growing compared with inflation"*. The amber chip is what the question was **understood** to mean; the grey chips are the filters it chose — the **Wages & Compensation** topic, **United States**.

Fig 10 — and the answer: average hourly earnings and the CPI family, which no literal match on that sentence would have found.

The chips tell you which path answered — and whether you paid

Chip	What happened	Charged?
Understood: …	The interpreter read the question and set filters.	**Yes** — one charge
Searched for: …	The interpreter was unavailable or returned nothing usable, so your own words were run as a plain search.	**No** — refunded

The fallback is deliberate: a model that fails should leave you with a working keyword search, not an error page. The refund is automatic on every path where no interpretation was produced.

> **The chips outlive the search that made them.** Clearing the box and running a plain search afterwards leaves the previous *Understood:* chips sitting above the stage, describing an interpretation that no longer has anything to do with the rows below. Trust the results list, not a stale chip; the chips refresh only on the next AI search or topic change.

08 Loading a series

Click a row. What happens next depends on whether anything is already charted.

Fig 11 — An empty stage says so plainly. The first series you click goes straight onto **ON CHART** without asking.

Fig 12 — With something already charted, the click asks which chart to use.

Situation	What the click does
`Stage empty`	Loads onto **ON CHART** immediately.
`Stage occupied`	Offers **Show on MAIN chart** / **Add to COMPARE chart**.
`Series already charted`	Nothing is moved — the click just shows its details in the About panel.
`Pane already holds six`	Refused, with *Each chart holds up to 6 series — remove one first*.

This is why a search never disturbs what you are looking at. The auto-load on an empty stage is the one exception, and it exists so that a fresh tab shows the best match instead of a blank plane — an earlier build reloaded the top hit on *every* search, which is how every query ended up dragging the user back to GDP.

While it loads

A pulsing overlay names the series being fetched. Observations are read from the platform's own store rather than the publisher's API, but a series that has never been pulled has to be fetched once: the request comes back marked *fetching* and the tab re-asks on the interval the server suggests, up to **eight** times. Past that it gives up with *that series is taking longer than usual — try again in a few minutes* and removes the card. Try it again later; the second attempt is normally instant.

> Because observations come from our own store and not from the publisher live, a source being slow or down does not empty your chart — and re-charting something you have already loaded this month never costs anything.

09 The series rail

Every loaded series gets a card. The rail is the legend, the read-out and the routing control in one column.

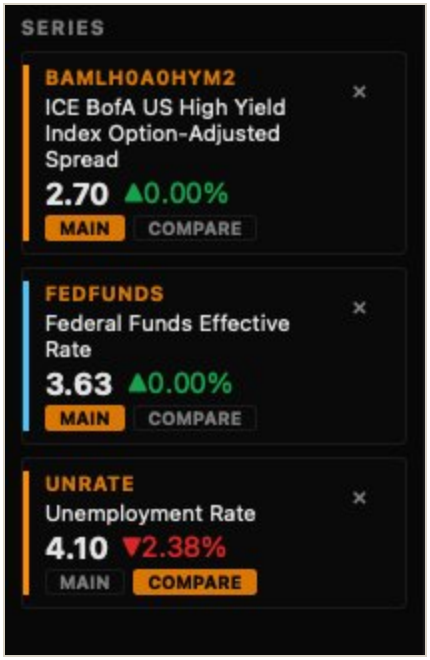


Fig 13 — Three cards: two on MAIN, one on COMPARE. The assignment chips show where each one currently sits.

Element	What it is
`Colour key`	The stripe down the left edge matches the line on the chart.
`Short id`	The publisher's own code — `UNRATE`, `FEDFUNDS` — in amber. This is what the chart legend, the table header and the crosshair use.
`Full name`	Wrapped over as many lines as it needs. Names are never cut mid-word.
`Value`	The latest observation, abbreviated (`K`, `M`, `B`, `T`) once it passes a thousand.
`Change`	▲ or ▼ against the **immediately preceding observation** — so for a monthly series that is month-on-month, and for a daily one it is one day.
`MAIN / COMPARE`	Move the series between charts. A series lives on exactly one.
`×`	Remove it from the stage.

Colours repeat across the two charts. A pane assigns its colours by slot, and both panes start at slot one — so the first series on MAIN and the first on COMPARE are both amber. Within one chart the six colours are always distinct; between charts they are not, and the card's own stripe is the reliable key.

CLEAR ALL, at the foot of the rail, empties both charts at once. It appears only when something is loaded, and it does not ask for confirmation — but nothing is destroyed: the series are still in the catalogue and re-loading them is free for the rest of the month.

10 Window, style and transform

Each chart carries its own three controls, on its own title bar. Nothing here is global — the two charts can sit on entirely different settings, which is the point of having two.

Fig 14 — A pane's controls: the window chips, the style list and the transform list.

Control	Options	Notes
`Window`	**1Y** · 5Y · 10Y · MAX	Defaults to 1Y. The cutoff is measured back from the series' **own last observation**, not from today — so a series a publisher stopped updating in 2020 still shows a meaningful final year.
`Style`	**Line** · Area · Step · Bars	Step suits policy rates and anything that holds a level between changes. Bars force zero into view.
`Transform`	**Levels** · Index=100 · % change	Levels is as published. Index=100 rebases every series to 100 at the start of the visible window. % change is period-over-period.

Fig 15 — The transform list.

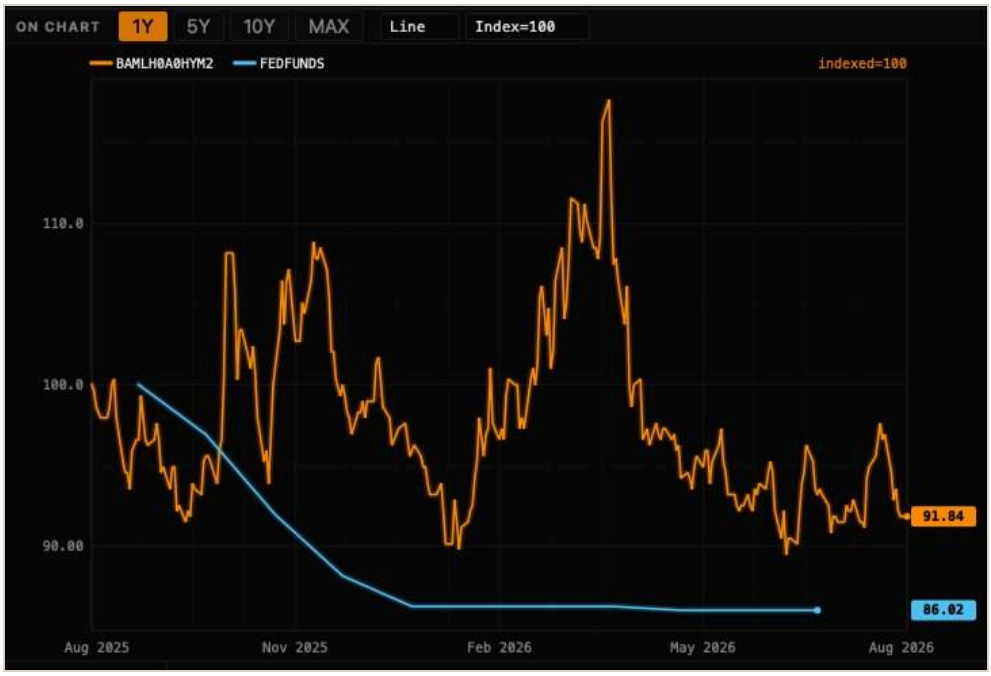


Fig 16 — The same two series rebased. An amber **indexed=100** badge marks the chart whenever a transform is active.

> **One chart, one y-axis.** Every series in a pane shares a single scale, and there is no second axis. Put a percentage next to a level in billions and the smaller one flattens to a line along the bottom. That is what **Index=100** is for — and if two series genuinely need their own axes, that is what the second chart is for.

11 Comparing — two charts, one screen

The **COMPARE** chart is hidden until you put something on it, then splits the stage side by side with **ON CHART**. Read left against right.

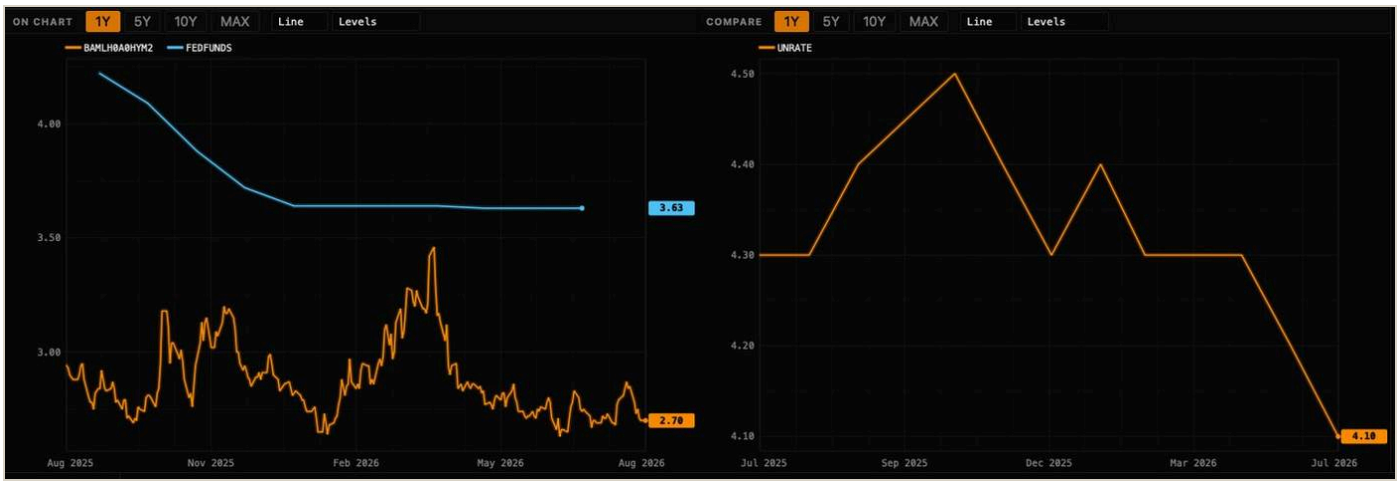


Fig 17 — A credit spread and the policy rate on the left, unemployment on the right. Each chart keeps its own window, style, transform and y-axis — the left is showing a daily series over a year, the right a monthly one, and neither distorts the other.

Two charts of up to six series each is **twelve on stage at once**. The reason for the split rather than one crowded plane is scale: a spread measured in tenths and a level in millions cannot share an axis legibly, and stacking the charts vertically would halve the height both of them need.

You want to…	Do this
`Compare like with like`	Put everything on **ON CHART** and set Index=100.
`Compare across units`	Split them — one family per chart — and leave both on Levels.
`Compare across eras`	Same series on both, different **window** on each.
`Compare level against rate of change`	Same series on both, **Levels** on one and **% change** on the other.

> Moving a series between charts costs nothing and re-fetches nothing — press **MAIN** or **COMPARE** on its card and it moves. Emptying the COMPARE chart hides it again and gives the full width back.

12 Reading the chart

The charts are drawn by the terminal rather than by a charting library, which is why they read the same as the rest of the desk and stay legible at six series.

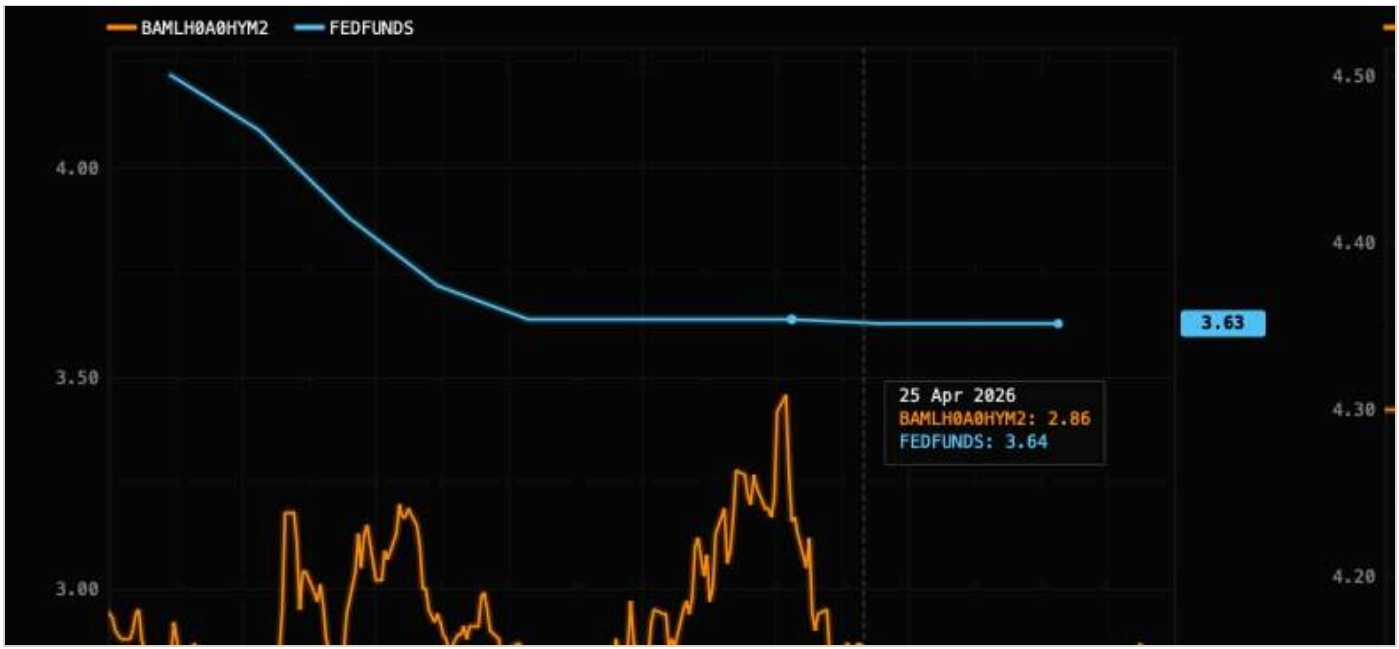


Fig 18 — Hovering drops a crosshair and reads every series at that date at once, each in its own colour. The dot marks the actual observation nearest the cursor.

Element	What it tells you
`Legend`	Top left — the short id of each series in its colour.
`Y-axis`	Left. Gridlines at five major steps with a finer half-step behind them; the scale adapts to what is in the window, with a 4% margin.
`X-axis`	Real dates along the bottom, spaced to the window.
`Value tag`	A coloured tag in the right gutter carries each series' latest value, so the current reading is legible without hovering.
`Zero line`	Drawn whenever the picture straddles zero under **Bars** or **% change**.
`Transform badge`	Top right — `indexed=100` or `% change`. Absent on Levels.
`Crosshair box`	Date in `dd MMM yyyy`, then one line per series.

> **There is no zoom, no pan and no click-drag range selection.** The window chips are the whole of the time control — and because they anchor to the series' own last point, they do the job a drag-zoom is usually needed for. To inspect a specific date, hover it; to read a run of dates, switch to **TABLE**.

13 The data table

TABLE · 13

TABLE swaps the charts for the numbers behind them — every series on stage, from both charts, merged into one date grid.

CHART TABLE + AI ANALYZE · 3 CR EXPORT CSV

DATE	BAMLH0A0HYM2	FEDFUNDS	UNRATE
2026-08-10	2.70		
2026-08-07	2.70		
2026-08-06	2.71		
2026-08-05	2.75		
2026-08-04	2.73		
2026-08-03	2.78		
2026-07-31	2.85		
2026-07-30	2.84		
2026-07-29	2.87		
2026-07-28	2.84		
2026-07-27	2.81		
2026-07-24	2.79		

Fig 19 — Three series in one grid, newest first. The blanks are real: `FEDFUNDS` and `UNRATE` are monthly, so they have nothing to say on a daily row.

Property	Behaviour
`Rows`	One per distinct date across all staged series, **newest first**.
`Columns`	`DATE`, then one per series, headed by its short id in stage order — ON CHART first, then COMPARE.
`Gaps`	Left blank. A date only carries a value for the series that actually published on it.
`Values`	Two decimal places, right-aligned, in your locale's number format.
`Sorting`	Fixed. The grid is read-only and the columns do not re-sort.
`Window`	**Ignored.** The table shows every observation held for each series, not just the chart's 1Y/5Y/10Y window.

> **Mixing cadences makes a sparse grid.** One daily series alongside two monthly ones gives roughly twenty blank rows for every populated one, because each trading day earns a row of its own. It is accurate, but it is not readable — for a table, stage series that share a cadence, or use the **cadence** filter to pick the monthly variant of a daily series before loading it.

> The table is free, and so are the statistics underneath it. Only the written analysis in §14 is metered.

14 AI analysis of what you charted

✦ **AI ANALYZE** writes up the chart in front of you. It is enabled as soon as anything is on stage, and the button carries its own price.

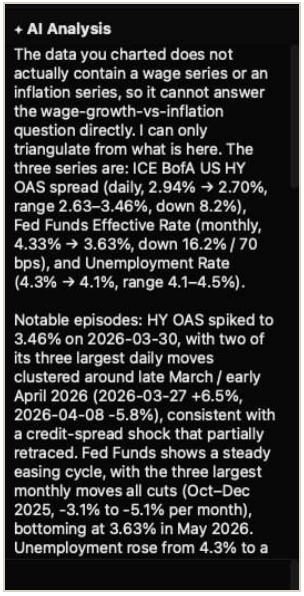


Fig 20 — The read: what each series did, with the numbers.

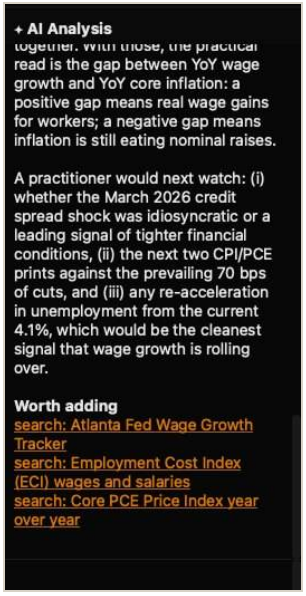


Fig 21 — and it closes with clickable suggestions.

What is actually sent

Not your data. The terminal reduces each staged series to summary statistics **on this machine** and sends only those — so the request stays small and the raw observations never leave:

- Point count in the window
- First and last value, with dates
- Minimum and maximum, with dates
- Mean and standard deviation
- The three largest single-period moves
- Total change across the window
- Pairwise correlations
- Units and cadence

Each series is summarised over **its own chart's window**, so narrowing ON CHART to 1Y narrows what the analysis is reasoning about. Correlations are computed on dates the two series actually share and are suppressed below ten overlapping points.

> **A question you typed in ASK AI is carried through** and answered first. In Fig 20 that produced the honest reply that the charted series could not answer the wage-versus-inflation question — the behaviour you want from a tool that only sees the numbers you gave it.

> **The suggestions are the model's own words, used verbatim.** Clicking one drops that exact text into the query box and runs a plain search — so when the model prefixes a suggestion with `search:`, as in Fig 21, that prefix is searched for too. Edit the box before pressing GO if the result looks thin.

15 Exporting to CSV

EXPORT CSV writes everything on stage — both charts — to a file you choose. It is enabled whenever the stage is not empty, and it is free.

Property	Behaviour
`Scope`	Every series on both charts, in stage order.
`Default name`	`global-data.csv`, into whichever folder you last used.
`Header`	`date`, then one column per series identified as `provider:native_id` — the fully-qualified reference, e.g. `fred:UNRATE`.
`Row order`	**Oldest first** — the opposite of the on-screen table.
`Gaps`	Empty fields, exactly as in the table.
`Precision`	Up to twelve significant figures — the stored value, not the two-decimal display.
`Window`	Ignored. The file holds every observation, not the charted window.

Two things differ from what you were just looking at. The file runs oldest-to-newest while the table runs newest-to-oldest, and its columns are headed `fred:UNRATE` where the table says `UNRATE`. Neither is a defect — the file is built for a spreadsheet's sort order and needs globally unique column names — but a chart rebuilt from the CSV will run backwards against the screen unless you check.

The export does not appear in File Manager. Unlike the Excel and Notes tabs, which register what they write, this is a plain save to the path you pick. Put it somewhere you will find it — the terminal will not remember it for you.

16 What it costs

CREDITS · 16

Three actions in this tab can spend credits. Everything else — searching, browsing all twelve branches, charting, comparing, the table, the statistics and the export — is free across all 1.7 million series. Reading the store is the product; only interpretation and writing are metered.

Action	Price	What you are buying
`Ask AI for data`	1 credit	Your question translated into the catalogue's own terms. Charged only when it actually interprets — a fallback to plain search is refunded automatically.
`AI chart analysis`	3 credits	The written read of the on-chart statistics, plus suggested companion series. The statistics and the chart themselves are free.
`Series load`	0.10 credit	Only past your plan's monthly allowance, and only for series that are **new to you this month**. Charged per load request, not per series.

The monthly series allowance

Plan	Distinct series per month
`Exclusive`	300
`Exclusive Plus`	1,500
`Exclusive Pro`	Unlimited

The unit is **distinct series**, counted once per calendar month. A series you have charted this month is free to re-chart, re-window, re-transform, move between charts, tabulate and export for the rest of it — the allowance meters discovery, not use. Loading five new series in one click is one charge, not five.

> **The prices are on the controls. ASK · 1 CR** and **✦ AI ANALYZE · 3 CR** carry their cost in the label, and a click you cannot afford is refused with an explanation rather than failing silently. The figures above are the current published rates; the label on the button is always the authority.

17 What carries over between sessions

SESSION · 17

Close the tab, close the terminal, come back tomorrow: the stage is rebuilt as you left it.

Remembered	Not remembered
Which series are on each chart, and in what order	The observations themselves
Each chart's window, style and transform	Which result row was selected
The query text and whether you were in ASK AI mode	The About panel's contents
The topic filter, and its name for the chip	Any AI analysis you ran
The source and cadence filters	Scroll position in the results
Whether you were on CHART or TABLE	How far the topic tree was expanded

On return the tab shows *Restoring your charts…* while it re-reads the observations for the staged series. They are deliberately not cached on disk: reading them back from the platform's store is fast, and it means a restored chart is never quietly stale — a revision the publisher made overnight is in the picture you come back to.

> **Restoring re-runs the saved search, and re-runs it in the mode you left.** If you close the tab in ASK AI mode with a question still in the box, that question is asked again on restore — and an interpreted answer is charged again. It costs one credit, but it is a credit you did not press anything to spend. Clear the box before you leave if that matters to you.

> The restore is per profile and travels with your terminal session, not with your account — this is window state, which by design stays on the machine.

18 Quick reference and honest notes

How do I...

Goal	Do this
`Find a series without knowing its name`	Click down the topic tree, or ask for it in ASK AI
`Get the older tabs' data`	It is all here — `econ`, `govt`, `dbnomics`, `akshare` and `asia` all open this tab
`Reach a folded cadence`	Set the cadence filter, then search again
`Compare different units`	Index=100 on one chart, or split across both charts
`See exact numbers`	Hover for one date; **TABLE** for a run of them
`Start over`	**CLEAR ALL** at the foot of the series rail
`Avoid spending anything`	Stay in SEARCH mode and keep queries to a few keywords

Honest notes on this build

- **A sentence typed in SEARCH mode is still charged** when it interprets — the routing is by shape of the query, not by which mode button is lit (§03, §07).
- **A topic's count is that node's own**, not its subtree's, while clicking it searches the whole subtree (§05).
- **Interpretation chips persist** after a later plain search, describing a query that is no longer on screen (§07).
- **Both charts start their colour palette at the same colour**, so MAIN's first series and COMPARE's first series are both amber (§09).
- **One y-axis per chart** — no secondary axis, no per-series scaling (§10).
- **No zoom or pan**; the four window chips are the whole time control (§12).
- **The table ignores the chart window** and shows every observation held (§13).
- **CSV runs oldest-first and the table newest-first**, and their column headings differ (§15).
- **The CSV is not registered in File Manager**, unlike Excel's and Notes' exports (§15).
- **Restoring a session can re-ask a saved AI question**, and re-charge for it (§17).
- **No agent tools.** Global Data is not exposed to the AI assistant or to workflows — there are no MCP tools in this category, so nothing can drive this tab for you. Everything here is done by hand.

> Everything in this guide was read from the shipping source and then confirmed on screen on the v5.0.0 build. Where the two disagreed, the screen won and the guide says what the screen does.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Global Economy

A complete, screen-by-screen walkthrough of the Global Economy tab — bilateral trade flows, 130 country indicators and the physical infrastructure of the world economy, on one map.

`>_ 3 trade views · 130 indicators · 17 infrastructure layers · 2 prices`

Covers · **Chord, Treemap & Products** · **the IMF world map** · **318,000 infrastructure features**

Every control, every status line, every credit price, and what the screen gets wrong

Demonstrated live on the Fincept Terminal Global Economy tab, v5.0.1.

Contents

23 SECTIONS

GETTING ORIENTED

01 What the Global Economy tab is ... 3
02 Anatomy of the screen ... 4
03 Getting here, and what the tab remembers ... 5
04 Asking a question ... 6
05 What it costs ... 7

TRADE FLOWS

06 Chord — one country against its partners ... 8
07 Treemap — the same answer, sized ... 9
08 The ranking table ... 10
09 Products — what the country actually trades ... 11

THE WORLD MAP

10 The map, and how to read it ... 12
11 The indicator picker ... 13
12 The year timeline and playback ... 15
13 Zoom, pan and the country readout ... 16

INFRASTRUCTURE

14 The Layers menu ... 17
15 The seventeen layers ... 18
16 Reading a layer ... 19
17 Infrastructure over an indicator ... 21

REFERENCE

18 The four dashboard tiles ... 22
19 Every control on the tab ... 23
20 Where the numbers come from ... 24
21 Every status message ... 25
22 Honest notes ... 26
23 Quick reference — how do I… ... 27

About this guide. Every screenshot is a live capture of the shipping v5.0.1 build, taken on a real account against the production data plane. It replaces the *Trade Viz* guide, which described a screen that no longer exists.

01 What the Global Economy tab is

WHAT IT IS · 01

One screen holding three different pictures of the world economy: **who trades with whom**, **what every country's economy looks like**, and **the physical plant that carries the trade** — pipelines, cables, power stations, ports, airports and the maritime chokepoints.

Pick a country and a year, press **Load**, and the answer is drawn three ways at once — as a flow wheel, as a treemap and as a ranked table. Switch the same screen to **World Map** and it becomes an IMF choropleth of any of 130 indicators, playable forward through fifty years. Switch on an infrastructure layer and the same map shows the hardware underneath. The trade query is the only thing on the tab that costs anything.

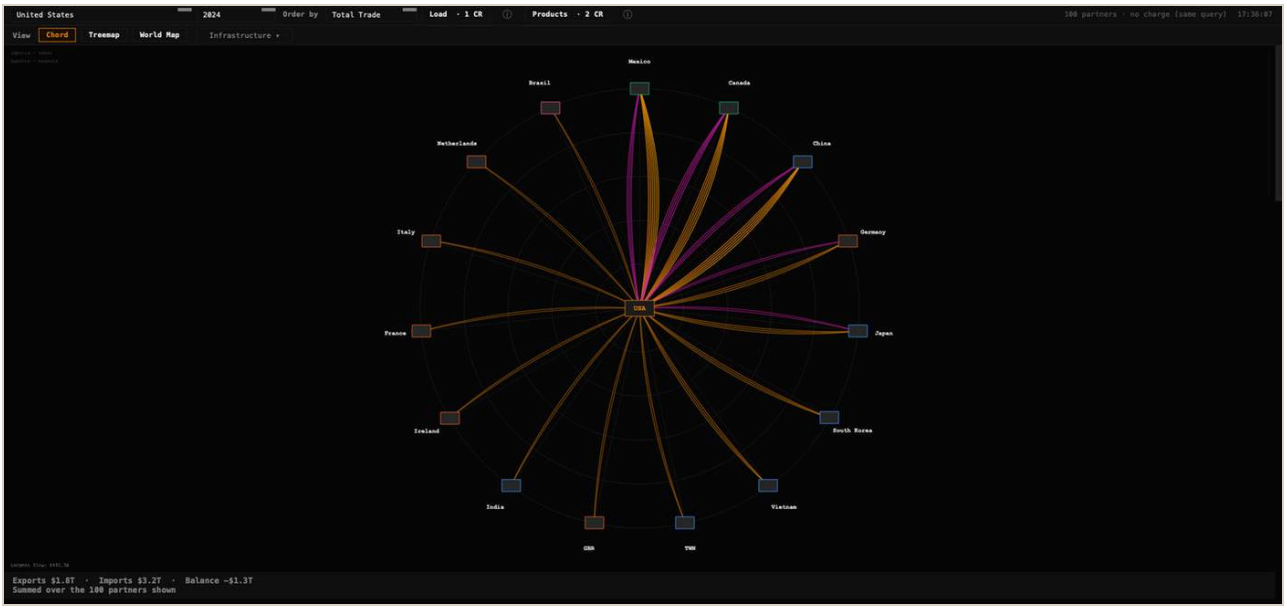


Fig 1 — The tab as it opens: the United States against its hundred largest trading partners in 2024, drawn as a flow wheel with the totals strip and ranking table below it.

At a glance, in Global Economy you can…

- Rank any country's trading partners for any year from 1995
- Read exports, imports and the balance for each one
- Order the answer by total trade, imports, exports or balance
- Break a country-year down by product to individual HS6 lines
- Shade every country in the world by one of 130 IMF indicators
- Play an indicator forward one year at a time
- See which countries have no figure at all, counted separately
- Draw 318,000 pieces of world infrastructure on the same map
- Hover any pipeline, cable, plant or port for its own detail
- Click a country on the map to load its trade

> **This guide replaces the Trade Viz guide.** The tab was called *Trade Viz* until v5.0.1 and was a fixed illustration: fifteen hard-coded United States partners, 2024 estimates, no network call and no way to change the country. Nothing in that guide describes this screen.

02 Anatomy of the screen

Sixteen regions in four bands. The top two bands are fixed — they stay put while the chart area and the table scroll beneath them. The third band, the map toolbar, exists only in the World Map view.

Fig 2 — The sixteen regions, numbered. Shown in the World Map view, which is the only one with a third toolbar row.

#	Region	What it is
1	Country	Which economy the trade query is about — roughly 200 reporters
2	Year	1995 to 2024 (§04)
3	Order by	Total Trade · Imports · Exports · Trade Balance
4	**Load · 1 CR**	Runs the trade query. Nothing loads until you press it
5	**Products · 2 CR**	Breaks the same country-year down by product (§09)
6	Read-out and clock	What came back, what it cost, and the time it landed
7	View switch	**Chord** · **Treemap** · **World Map** — free to flip between
8	**Infrastructure** ▾	The layer menu; its label becomes the active layer (§14)
9	Indicator	Which IMF series shades the map, or *None — infrastructure only*
10	▶ / ⏸	Plays the indicator forward through every year
11	Year timeline	Scrub the map's year; independent of the trade year in region 2
12	Coverage	How many countries the chosen indicator has a figure for this year
13	Chart area	The wheel, the treemap or the map. It grows with the window
14	Legend and attribution	Colour key, value domain, no-data count, and the source credit
15	Totals strip	Exports, imports and balance summed over the partners drawn
16	Results table	The ranked answer — partners, or products after a sweep (§08)

> **Two different years live on this screen.** Region 2 is the year of the *trade* query and only changes when you press Load. Region 11 is the year of the *map* and redraws instantly and free of charge. They are not linked, and they routinely disagree.

03 Getting here, and what the tab remembers

Global Economy is not one of the fifteen screens on the main tab bar. There are two routes.

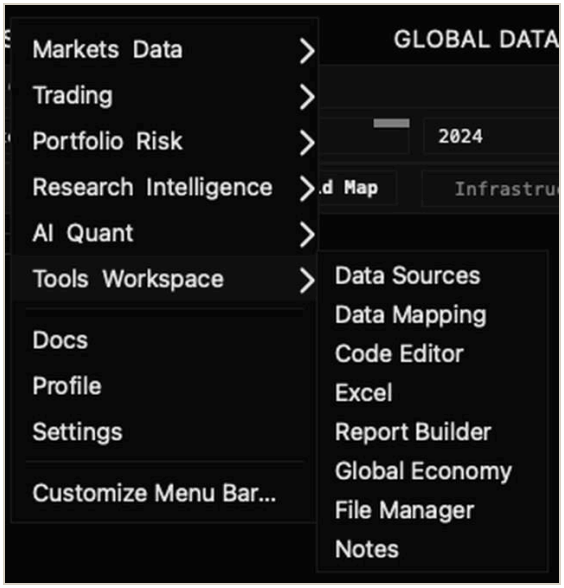


Fig 3 — **Navigate ‣ Tools & Workspace ‣ Global Economy**, sixth of eight.

By menu

Navigate ‣ Tools & Workspace ‣ Global Economy.

By command bar

Click the command box (or just start typing) and enter `global_economy` — or any of **41 aliases**, which is far more than any other screen carries. The alias list is written to be guessed rather than learned: the thing you are looking for is usually itself an alias.

Alias family	Type any of these
The tab	`global_economy` · `geco` · `economy` · `tradeviz` · `tradegraph` · `flow` · `visualization`
Indicators	`world` · `indicators` · `gdp`
Infrastructure	`infra` · `infrastructure`
Energy	`pipelines` · `gas` · `oil` · `lng` · `power` · `plant` · `plants` · `coal` · `mine` · `grid` · `transmission`
Telecom	`cable` · `cables` · `submarine` · `fibre` · `fiber` · `landing` · `data centre` · `data center` · `ixp` · `peering`
Transport	`ports` · `airport` · `airports`
Chokepoints	`chokepoint` · `chokepoints` · `strait` · `hormuz` · `malacca` · `suez`

What survives a restart

The tab saves four things under the key `GlobalEconomy` and restores them the next time it opens: the **country**, the **year**, the **map indicator** and the **infrastructure layers**. It does *not* save which of the three views you were on, the map's zoom, or the map's year — those reset. It also does not re-run the trade query for you, so a restored country and year sit in the bar unanswered until you press Load.

> **Two catalogues disagree about where this tab belongs.** The Navigate menu files it under *Tools & Workspace*; the Component Browser (`Ctrl+K`) files the same screen under *Research*. Search either one and you will find it — but only one of them matches the menu you will be told to use.

04 Asking a question

Everything about a trade query is set in the top bar, and nothing happens until **Load**.

Fig 4 — The query controls: country, year, ranking metric, and the two priced buttons with their info marks.

Fig 5 — The right-hand read-out: what came back, what it cost, and when.

Control	What it does	Notes
`Country`	The reporting economy	Alphabetical, roughly 200 entries, from Afghanistan to Zimbabwe. Territories and dependencies are listed separately
`Year`	The trade year	**2024 down to 1995** — thirty years, newest first
`Order by`	How partners are ranked	Total Trade, Imports, Exports or Trade Balance. Changes the wheel order, the treemap sizes and the table sort together
`Load · 1 CR`	Runs the query	The only button that spends on the partner side
ⓘ `(first)`	Explains the price	Hover for the sentence; click for the full price list (§05)
`Products · 2 CR`	Product breakdown	Needs a country and a year already chosen (§09)
ⓘ `(second)`	Explains that price	Same dialog
`Read-out`	Result and charge	e.g. `100 partners · 1 CR`
`Clock`	When the answer landed	Ticks once a second while the tab is visible

Fig 6 — Thirty years.

Fig 7 — Four ranking metrics.

Nothing is automatic

Change the country, the year or the metric and the read-out turns to **Selection changed — press Load.** The three panels keep showing the *previous* answer until you do. That is deliberate — every load is metered — but it means a screenshot of this tab can show one country in the bar and a different one in the chart.

The one exception is opening the tab: a restored country and year are put back in the bar, but still not queried.

Fig 8 — The dirty-selection state. The wheel below it is still the old answer.

05 What it costs

Two things on this tab cost credits. Everything else — the whole world map, all 130 indicators, every infrastructure layer, flipping between the three views, sorting the table, and opening the tab at all — is free.

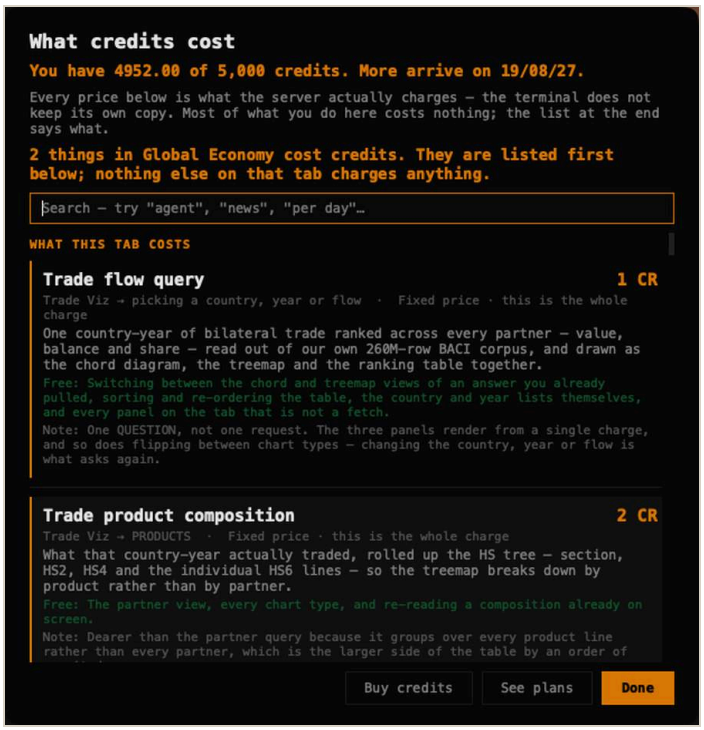


Fig 9 — The price list, reached from either ⓘ. It is served by the API, not compiled into the terminal, so it is the authority.

Priced action	Cost
Trade flow query — Load	**1 CR**
Trade product composition — Products	**2 CR**

What one credit buys

One *question*, not one request. A single Load charge pays for the wheel, the treemap *and* the ranking table, and for flipping between them as often as you like. You are charged again only when the question changes — a different country, a different year or a different ranking metric.

Measured

Over one session: one Load, one Products sweep and one further Load took the balance from **4953.00** to **4949.00** — exactly 1 + 2 + 1.

Fig 10 — The Load price, in full.

Fig 11 — Why the product sweep is dearer.

The repeat rule, exactly

Press Load twice in a row without touching anything and the second press reads **no charge (same query)** and the balance does not move — the server recognises the identical request. That dedup only survives an *immediately* repeated query. Change the country and change it back, or run a Products sweep in between, and the next Load charges again. Both behaviours were observed on the same balance within eight minutes.

> **The price list still calls this tab Trade Viz.** Both meters are captioned *Trade Viz → picking a country, year or flow* and *Trade Viz → PRODUCTS*, although the dialog's own summary line above them says *2 things in Global Economy cost credits*. The rename did not reach the pricing catalogue's captions. Nothing about the charge is affected. The Load tooltip also reads *Costs 1 credits*.

06 Chord — one country against its partners

The default view. The reporting economy sits at the hub; its partners ring the outside; the bundles between them are the trade.

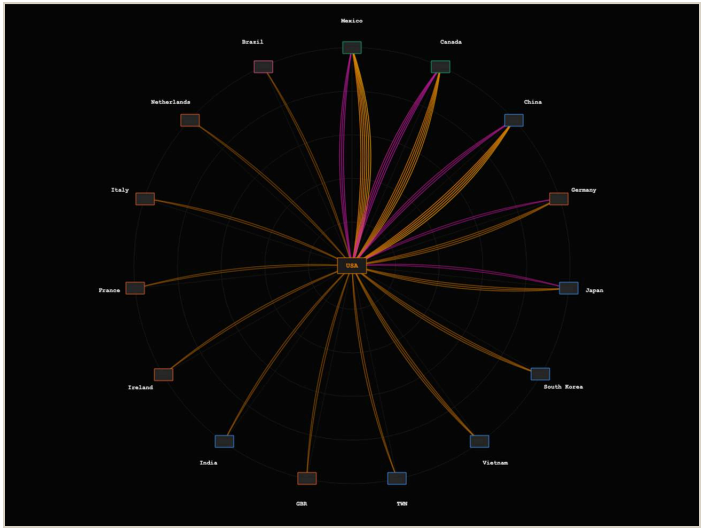


Fig 12 — The United States, 2024, ordered by total trade.

Element	How to read it
`Hub`	The reporting country, in amber
`Partner boxes`	Coloured by continent, matching the treemap key (§07)
`Position`	**Clockwise from twelve o'clock, in descending rank** — so the box at the top is number one
`Amber bundles`	Imports — what the hub buys
`Magenta bundles`	Exports — what the hub sells
`Strand count`	Scales with the size of the flow
`Rings`	A background scale; not labelled
`Foot`	**Largest flow:** the biggest single leg drawn

Fig 13 — The key, top-left.

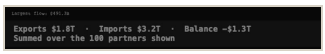


Fig 14 — The scale note and totals strip.

The wheel is a shortlist, the table is the answer. The read-out says 100 partners and the ranking table lists 100 rows, but the wheel draws only the **leading fifteen**. If a partner is not on the wheel it has not been dropped from the query — look further down the table.

A missing magenta bundle does not mean no exports. Strand counts are proportional, so small legs round down to nothing. In the 2024 United States answer only five partners are drawn with any magenta at all — Mexico, Canada, China, Germany and Japan — yet the table for the same answer shows $71.4 bn of exports to South Korea, $67.0 bn to the United Kingdom and $63.2 bn to the Netherlands, none of which the wheel shows. **Read export detail off the table, never off the wheel.**

Long partner names are shortened to a code so the ring stays legible — the United Kingdom is drawn as **GBR** and Chinese Taipei as **TWN**, while the table spells both out.

07 Treemap — the same answer, sized

The same query, laid out so that area is value. Nothing is re-fetched when you switch — this view is drawn from the answer you already bought.

Fig 15 — The 2024 United States answer as a treemap. Blocks are grouped by continent, then sized by the ranking metric.

Fig 16 — Hovering a block names it, gives its continent, its value and its share.

How to read it

- **Area** is whatever *Order by* is set to — total trade by default.
- **Colour** is the continent, not the value. Each continent gets its own hue and its blocks are packed together, so you can see a bloc without reading a single label.
- **Labels** are dropped as blocks get smaller; the smallest carry a truncated name and a value, and the very smallest carry nothing. Hover for the full detail.
- The **hover card** adds the one number the table does not print: the partner's **share of the total**.

Fig 17 — The continent key, bottom-left. The same six hues colour the partner boxes on the wheel and the country names in the table.

> The treemap is the best of the three views for spotting concentration: in the United States answer, Mexico, Canada and China together take a visibly larger area than all of Europe.

08 The ranking table

RANK TABLE · 08

Under both trade views, the full answer — every partner returned, not just the ones drawn.

Trading Partner	Exports	Imports	Balance
Mexico	$236.9B	$491.3B	−$254.4B
Canada	$258.1B	$400.0B	−$141.9B
China	$155.1B	$448.8B	−$293.7B
Germany	$90.4B	$156.5B	−$66.1B
Japan	$80.6B	$144.2B	−$63.7B
South Korea	$71.4B	$134.6B	−$63.2B
Vietnam	$12.6B	$140.5B	−$127.9B
Chinese Taipei	$38.3B	$112.6B	−$74.3B
United Kingdom	$67.0B	$64.3B	$2.7B
India	$38.7B	$88.8B	−$50.1B
Ireland	$24.0B	$88.8B	−$64.9B
France	$49.6B	$54.2B	−$4.6B
Italy	$26.8B	$70.5B	−$43.7B
Netherlands	$63.2B	$30.3B	$32.9B
Brazil	$43.7B	$41.7B	$2.0B
Thailand	$19.3B	$64.8B	−$45.6B
Switzerland	$29.9B	$51.2B	−$21.3B
Malaysia	$26.7B	$51.4B	−$24.7B
Singapore	$42.6B	$35.2B	$7.4B
Spain	$28.8B	$21.2B	$7.6B
Belgium	$23.2B	$26.6B	−$3.4B
Australia	$34.2B	$15.4B	$18.8B
Indonesia	$11.7B	$29.7B	−$18.1B
Turkey	$16.5B	$17.0B	−$566.0M
Colombia	$16.6B	$15.9B	$733.5M

Fig 18 — The ranking table. Partner names are coloured by continent; a negative balance is red, a surplus green.

Column	What it holds
`Trading Partner`	The counterpart economy, coloured by continent
`Exports`	What the reporting country sold to it
`Imports`	What the reporting country bought from it
`Balance`	Exports minus imports — **red** for a deficit, **green** for a surplus

Working with it

- **Row order follows *Order by***, not the column you are reading. With the default Total Trade, Vietnam sits seventh on $153.1 bn of turnover even though its exports column ($12.6 bn) is far down the list.
- **Re-ordering is free.** Changing *Order by* and pressing Load again re-ranks the same country-year; the price list states explicitly that re-ordering the table costs nothing.
- **The table is the whole answer.** Anything the wheel or the treemap leaves out is here.
- The table sizes itself to its contents rather than scrolling internally, so a hundred partners make a long page — scroll the tab, not the table.

> **Balance is the column the old screen never had.** The retired Trade Viz table printed a single *total* per partner — imports plus exports — and no direction at all. Every figure here is a real bilateral read, both legs and the difference.

09 Products — what the country actually trades

Products · 2 CR asks a different question about the same country and year: not *who*, but *what*. The answer replaces the treemap and the table; the flow wheel is not affected.

Fig 19 — The 2024 United States export composition. Blocks are HS product lines, grouped and coloured by HS section.

Fig 20 — The HS section key, first half.

…and the second. Seventeen sections in all.

Product	Section	Value	HS6 lines
Crude Petroleum	Mineral Products	$122.0B	72
Refined Petroleum	Mineral Products	$100.7B	101
Petroleum Gas	Mineral Products	$88.4B	488
Gas Turbines	Machines	$63.9B	744
Cars	Transportation	$63.0B	1296
Vaccines, blood, antisera, toxins and cultures	Chemical Products	$53.1B	584
Integrated Circuits	Machines	$43.9B	334
Packaged Medicaments	Chemical Products	$40.7B	954
Motor vehicles; parts and accessories (8701 to 8705)	Transportation	$39.8B	2015
Planes, helicopters, and/or Spacecraft	Transportation	$39.2B	371
Medical Instruments	Instruments	$35.2B	1507
Machinery Having Individual Functions	Machines	$28.2B	899
Aircraft parts (gliders, balloons, and powered aircraft)	Transportation	$28.0B	564
Computers	Machines	$25.0B	771
Broadcasting Equipment	Machines	$24.0B	377
Soybeans	Vegetable Products	$23.2B	98
Other Plastic Products	Plastics and Rubbers	$22.6B	796
Gold	Precious Metals	$22.3B	213
Delivery Trucks	Transportation	$21.3B	687
Coal, Briquettes and Similar Manufactured Solid Fuels	Mineral Products	$18.8B	196
Ethylene Polymers	Plastics and Rubbers	$16.5B	535
Orthopedic Appliances	Instruments	$14.9B	1828
Valves	Machines	$14.8B	
Office Machine Parts	Machines	$12.2B	

Fig 21 — The product table that replaces the partner table.

Column	What it holds
`Product`	The rolled-up product line, e.g. *Crude Petroleum*, *Integrated Circuits*
`Section`	Its HS section, coloured to match the treemap
`Value`	Trade in that line for the country-year
`HS6 lines`	How many individual six-digit tariff lines were rolled into that row

The **HS6 lines** column is the one to watch: *Crude Petroleum* at $122.0 bn rolls up 72 tariff lines, while *Motor vehicles; parts and accessories* at $39.8 bn rolls up 2,015. A large number there means the row is a broad category, not a single commodity.

> **Why it is dearer, and what you actually got.** The price list explains that the sweep groups over every product line rather than every partner — the larger side of the table by an order of magnitude. The 2024 United States sweep returned **150 product lines**, rolled up from the section, HS2, HS4 and HS6 levels of the tariff tree. Note the read-out's parenthesis: it came back as `150 product lines (exports)` — **one direction, not both**, with *Order by* left on Total Trade. Check it before quoting a product figure as a country's total trade in that line.

10 The map, and how to read it

The third view drops the trade query entirely and shades every country in the world by one economic indicator. It is free — picking an indicator, changing the year and redrawing cost nothing.

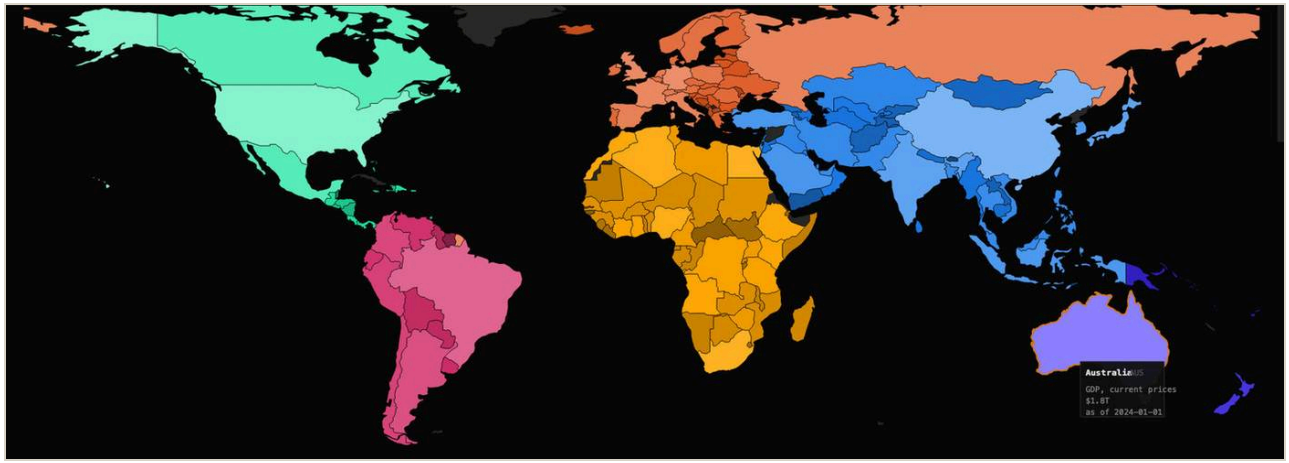


Fig 22 — GDP at current prices, 2024. Hue is the continent; brightness is the value.

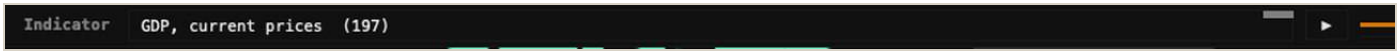


Fig 23 — The map toolbar, left: the indicator picker and the play button.

…and right: the year timeline, the year, and how many countries have a figure for it.

Fig 24 — The key: the six continent ramps, the value domain, and the no-data count.

The colour rule

Each continent has its own hue and its own ramp. **Brightness within that hue is the value**, on a **log scale** — stated on the key itself. So you compare *within* a continent by brightness and *between* continents by hue, and a bright African country is not claiming parity with a bright Asian one.

The key also prints the **domain** the ramp is stretched over — `$1.2B → $39.0T` for 2024 GDP — and, in its own swatch, the number of countries with **no data** for the chosen indicator and year.

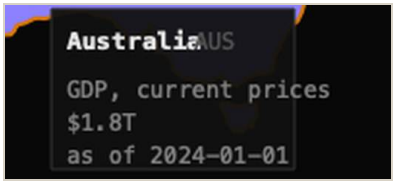


Fig 25 — The country readout.

The readout

Hovering a country highlights it in amber and names it, with its ISO3 code, the indicator, the value and — the important line — **the date the figure is actually for**. That last line is what tells you whether you are looking at a reported number or one the map has carried over.

Under the key sits the source credit, which for every IMF indicator reads *Source: IMF DataMapper terms of use. Attribution to the International Monetary Fund required*. If you republish a figure from this map, that sentence is the licence condition.

11 The indicator picker

130 indicators, each labelled with the number of countries it covers — the bracketed figure is the coverage, not an id. The list opens with **None — infrastructure only**, which clears the shading and leaves the basemap for infrastructure work.

```
None — infrastructure only
Broad Money (% of GDP)  (45)
Broad Money Growth  (45)
Claims on Nonfinancial Private Sector (% of GDP)  (45)
Claims on Nonfinancial Private Sector (%)  (45)
Consumer Prices, Average (Annual % Change)  (45)
Consumer Prices, End of Period (Annual % Change)  (45)
Exports of Goods and Services (% of GDP)  (45)
External Current Account, Incl.Grants (% of GDP)  (45)
External Debt, Official Debt, Debtor Based (% of GDP)  (45)
Government Debt (% of GDP)  (45)
Government Expenditure (% of GDP)  (45)
Government Revenue, Excluding Grants (% of GDP)  (45)
Gross National Savings (% of GDP)  (43)
Imports of Goods and Services (% of GDP)  (45)
Net Foreign Direct Investment (% of GDP)  (45)
Nominal Effective Exchange Rates (2010=100)  (43)
Overall Fiscal Balance, Excluding Grants (% of GDP)  (45)
Overall Fiscal Balance, Including Grants (% of GDP)  (45)
Real Effective Exchange Rates (2010=100)  (43)
Real GDP Growth  (45)
Real Non-Oil GDP Growth  (44)
Real Per Capita GDP Growth  (45)
Reserves (Months of Imports)  (31)
Terms of Trade (Index, 2010 = 100)  (44)
Total Investment (% of GDP)  (43)
```

Fig 26 — The head of the list: the IMF's regional-economic-outlook block, then the capability indices and the reserve-adequacy ratios.

```
Debt Forgiveness  (168)
Debt Securities Assets  (163)
Debt Securities Liabilities  (163)
Direct Investment Abroad  (178)
Direct Investment In Country  (180)
Equity Securities Assets  (166)
Equity Securities Liabilities  (160)
Financial Derivatives  (152)
Nominal GDP  (183)
Other Investment Assets  (180)
Other Investment Liabilities  (180)
Portfolio Investment Assets  (177)
Portfolio Investment Liabilities  (177)
Private Inflows excluding Direct Investment  (182)
Private Inflows excluding Direct Investment (% of GDP)  (182)
Private Outflows excluding Direct Investment  (180)
Private Outflows excluding Direct Investment (% of GDP)  (180)
Proxy for Official Other Investment Liabilities  (180)
Bond openness index (1=fully liberalized)  (166)
Collective investment openness index (1=fully liberalized)  (167)
Commercial credit openness index (1=fully liberalized)  (164)
Derivative investment openness index (1=fully liberalized)  (163)
Direct investment liquidation openness index (1=fully liberalized)  (167)
Direct investment openness index (1=fully liberalized)  (166)
Equity openness index (1=fully liberalized)  (167)
Financial Market Openness Index (equity, bond, money market, collective investment, derivates) 1=fully liberalized  (161
```

Fig 27 — The middle: external-position stocks, then the eighteen capital-account openness indices.

```
   Chemicals  (166)
   Commodity & transactions not classified accord to kind  (165)
   Crude materials, inedible, except fuels  (166)
   Export Diversification Index  (196)
   Export Quality Index  (166)
   Extensive Margin  (196)
   Food and live animals  (166)
   Intensive Margin  (196)
   Machinery and transport equipment  (166)
   Manufact goods classified chiefly by material  (166)
   Mineral fuels, lubricants and related materials  (164)
   Miscellaneous manufactured articles  (166)
   Current account balance U.S. dollars  (196)
   Current account balance, percent of GDP  (196)
   GDP based on PPP, share of world  (197)
   GDP per capita, current prices  (197)
   GDP per capita, current prices  (197)
✓  GDP, current prices  (197)
   GDP, current prices  (197)
   General government gross debt  (194)
   General government net lending/borrowing  (197)
   Implied PPP conversion rate  (197)
   Inflation rate, average consumer prices  (197)
   Inflation rate, end of period consumer prices  (196)
   Population  (197)
   Real GDP growth  (197)
   Unemployment rate  (118)
```

Fig 28 — The tail: fiscal aggregates, gender and debt indicators, the trade-quality block, and the World Economic Outlook headline series.

Family	Roughly	Typical coverage
`Regional outlook aggregates`	Broad money, consumer prices, fiscal balance, reserves, terms of trade	43–45 countries
`Capability indices`	AI Preparedness, Digital Infrastructure, Human Capital, Innovation	165–174
`Reserve adequacy`	Reserve/ARA, Reserve/(Import/12), Reserves/Broad Money, Reserves/STD	58–68
`External position`	Direct, portfolio, equity, debt-security and other investment, both sides	152–183
`Capital-account openness`	Eighteen 1 = fully-liberalised indices by instrument	160–167
`Fiscal`	Revenue, expenditure, gross and net debt, cyclically adjusted balances	81–197
`Gender`	Gender Development, Gender Inequality, Gender Budgeting	73–154
`Private and household debt`	All instruments, and loans plus debt securities	44–160
`Trade quality`	Export Diversification, Export Quality, Extensive/Intensive Margin, SITC groups	164–196
`World Economic Outlook`	GDP, GDP per capita, PPP share, inflation, population, unemployment	118–197

Three names appear twice. *GDP, current prices (197)*, *GDP per capita, current prices (197)* and *Human Capital and Labor Market Policies* each occur as two separate rows — the last with two different coverage counts, 173 and 174. They are distinct upstream series that arrive with the same display name, and the picker offers no way to tell them apart. Note which of the pair you chose before quoting a figure.

The picker is a fixed-width list with no search box, and the longest entry — the Financial Market Openness Index — runs off the right-hand edge rather than wrapping or eliding.

12 The year timeline and playback

The slider beside the indicator picker moves the map through time. Dragging it redraws instantly and costs nothing.

For GDP at current prices the range runs **1980 to 2031**. Watch the coverage figure at the right-hand end as you scrub: **144 countries** in 1980, **195** in 2024, **188** in 2030. Coverage is a property of the year, not of the map.

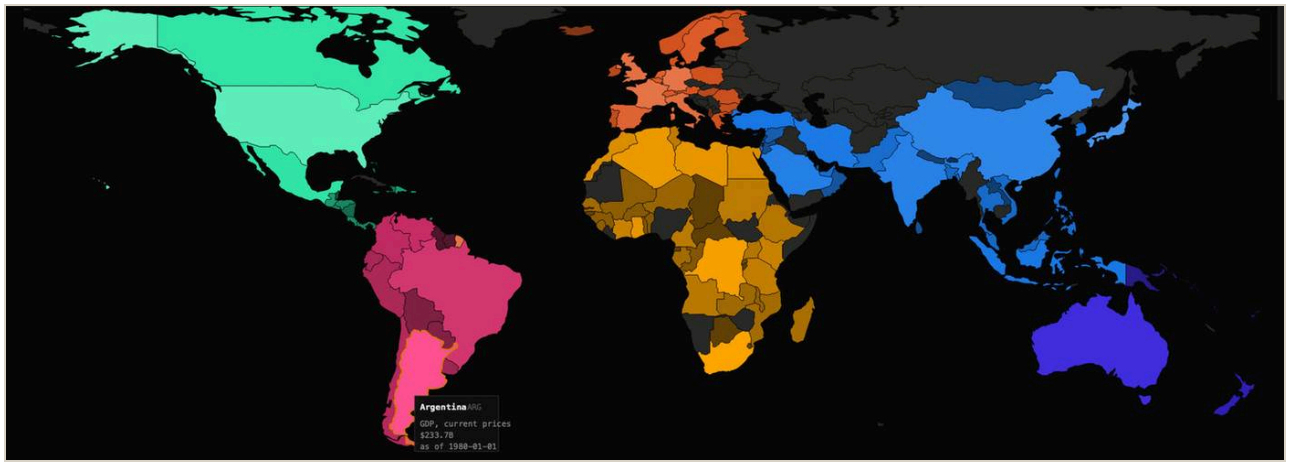


Fig 29 — The same indicator in 1980. The grey countries are the no-data bucket, and the key counts them: 51, against 13 in 2024.

Fig 30 — The readout carries the year it is for.

Playback

▶ plays the indicator forward one year at a time, about two years a second; the button becomes ⏸ while it runs and reverts on its own when it reaches the end of the range. The year label, the coverage count and the whole map update on every frame. Press it again at any point to stop.

Reading the grey

Countries with no figure are drawn in a distinct grey and counted separately in the key, so a gap in the data never reads as a value of zero. Sweeping 1980 to 2024 makes the widening of IMF coverage as visible as the economics.

> **The last seven years are forecasts, drawn exactly like history.** The range ends in 2031 because the World Economic Outlook publishes projections, and the map shades a projected year with the same ramp, the same readout and the same *as of* line as a reported one. Nothing on the screen marks the boundary. Cross-check the year against the WEO vintage before presenting anything past the last actual year.

> The map's year is independent of the trade year in the filter bar and does not persist across a restart — it reopens at the newest year with data.

13 Zoom, pan and the country readout

The map is a live canvas, not a picture.

Gesture	What it does
`Scroll`	Zooms toward the cursor, eased rather than stepped
`Drag`	Pans
`Double-click`	Resets to the whole world
`Hover a country`	Highlights it and shows its readout (§10)
`Hover a feature`	Highlights it and shows the feature card (§16)
`Click a country`	Puts that country into the trade filter bar

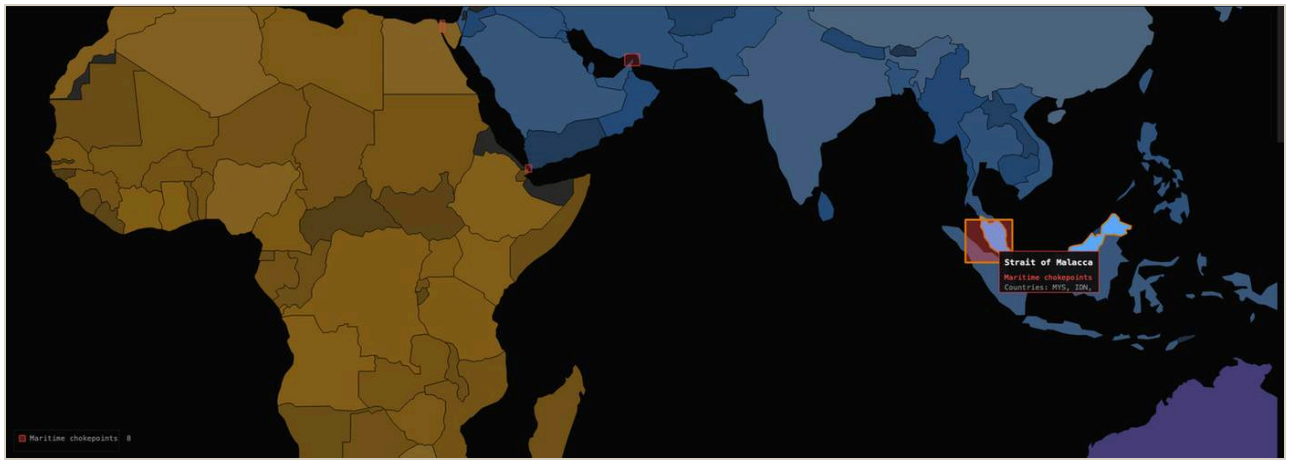


Fig 31 — Zoomed into the Indian Ocean with the chokepoint layer on. The hovered feature is picked out in amber.

Clicking a country is the bridge between the two halves of the tab

Click any country on the map and its name drops into the **Country** box, the read-out turns to **Selection changed — press Load**, and one press gives you that country's trade. It is the fastest route from *that one looks interesting* to a bilateral answer, and it is the reason the map and the trade views share a screen at all.

If the country you click is not a reporter in the trade corpus the tab says so plainly — **<country> is not in the trade corpus** — rather than loading an empty answer.

> **Double-click does two things.** It resets the view *and* selects whatever country is under the cursor. Resetting the zoom over land therefore changes your trade selection as a side effect — during capture, a double-click intended only to reset the view replaced the United States with Ethiopia in the filter bar. Double-click over open ocean to reset without touching the query.

14 The Layers menu

Infrastructure ▾ sits beside the view switch and is available in every view, though the layers only draw on the map. Its tooltip is the honest summary of the whole feature: *Pipelines, cables, ports, plants and chokepoints — free to read.*

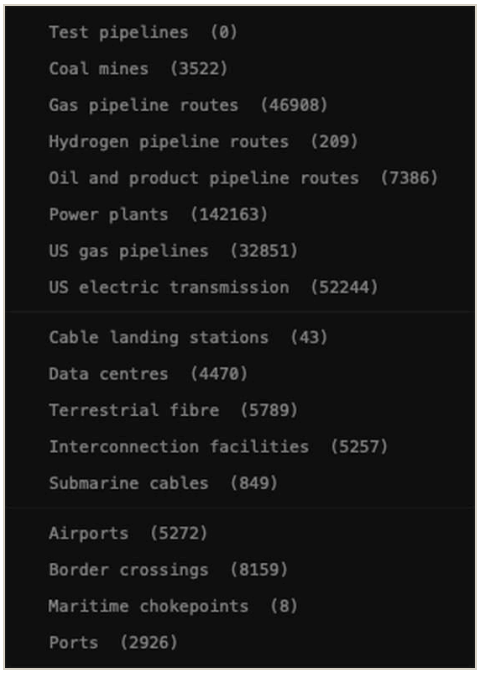


Fig 32 — Seventeen layers in three groups. The bracketed number is how many features that layer holds.

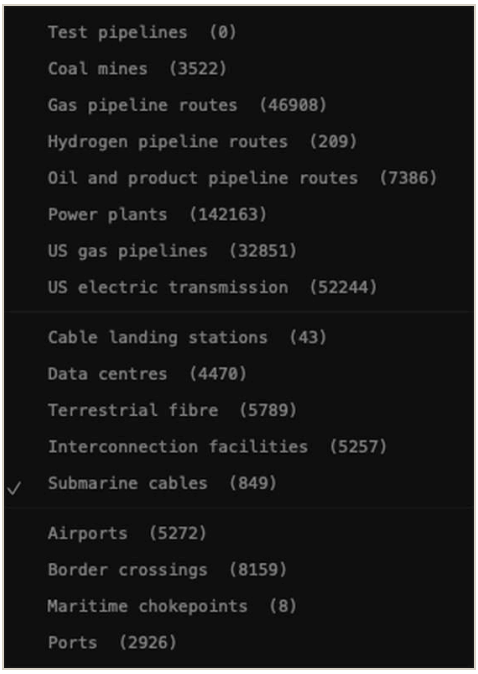


Fig 33 — A layer switched on. The tick is the only indication in the menu.

How it behaves

- The three groups — **energy**, **telecom**, **transport** — are separated by rules. The rules carry no printed heading.
- Choosing a layer closes the menu and starts the fetch; the button's own label becomes the layer name and its loaded feature count, e.g. **Submarine cables (849)** ▾.
- Choosing the same layer again switches it off and the button reverts to **Infrastructure** ▾.
- Every layer is free, however large.

> **You get one layer at a time.** The items are drawn as checkboxes and do toggle off individually, but switching one *on* silently switches off whichever was on before — confirmed in both directions (Submarine cables → Power plants cleared the cables; Power plants → Submarine cables cleared the plants). There is no way to put cables and ports on the map together, and nothing in the menu says so: after the swap the previous layer's tick has simply gone. The release notes describing the menu as switching layers *on and off individually* are describing the toggle, not multi-select.

> **Turning on your first layer also clears the choropleth.** The indicator picker flips to *None — infrastructure only* so the lines read against a plain basemap. To see both, re-pick an indicator afterwards (§17).

15 The seventeen layers

Counts are what the menu prints — the size of the layer on the server, before the per-layer draw cap (§16).

Group	Layer	Features	What it is
Energy	**Coal mines**	3,522	Producing and proposed coal mines
Energy	**Gas pipeline routes**	46,908	Gas transmission routes worldwide — the largest linear layer
Energy	**Hydrogen pipeline routes**	209	Built and announced hydrogen pipelines
Energy	**Oil and product pipeline routes**	7,386	Crude and refined-product pipelines
Energy	**Power plants**	142,163	Generating stations of every fuel — the largest layer on the tab
Energy	**US gas pipelines**	32,851	United States gas network, at higher resolution than the global layer
Energy	**US electric transmission**	52,244	United States transmission lines
Energy	**Test pipelines**	0	An empty layer — see the warning below
Telecom	**Submarine cables**	849	Subsea fibre routes between landing points
Telecom	**Cable landing stations**	43	Where those cables come ashore
Telecom	**Terrestrial fibre**	5,789	Long-haul land fibre routes
Telecom	**Data centres**	4,470	Named facilities
Telecom	**Interconnection facilities**	5,257	Exchange points and carrier hotels — where networks meet
Transport	**Ports**	2,926	Seaports
Transport	**Airports**	5,272	Airports with scheduled traffic
Transport	**Border crossings**	8,159	Land crossings
Transport	**Maritime chokepoints**	8	Hormuz, Malacca, Suez, Bab el-Mandeb and the rest

318,056 features in total, of which the energy group is 285,283 — nine parts in ten. The two United States layers are national by design, and the tab says so on the map rather than in a footnote.

> ***Test pipelines (0)* ships in the production menu.** It is a development leftover: selecting it draws nothing, changes nothing, and gives no message — the map simply stays as it was while the button reads *Test pipelines (0)*. It is harmless, and it is not a broken layer to report.

> If the menu comes up saying **No layers on this server — run the infrastructure loader**, the catalogue has not been populated on the environment you are pointed at. Nothing is wrong with the client.

16 Reading a layer

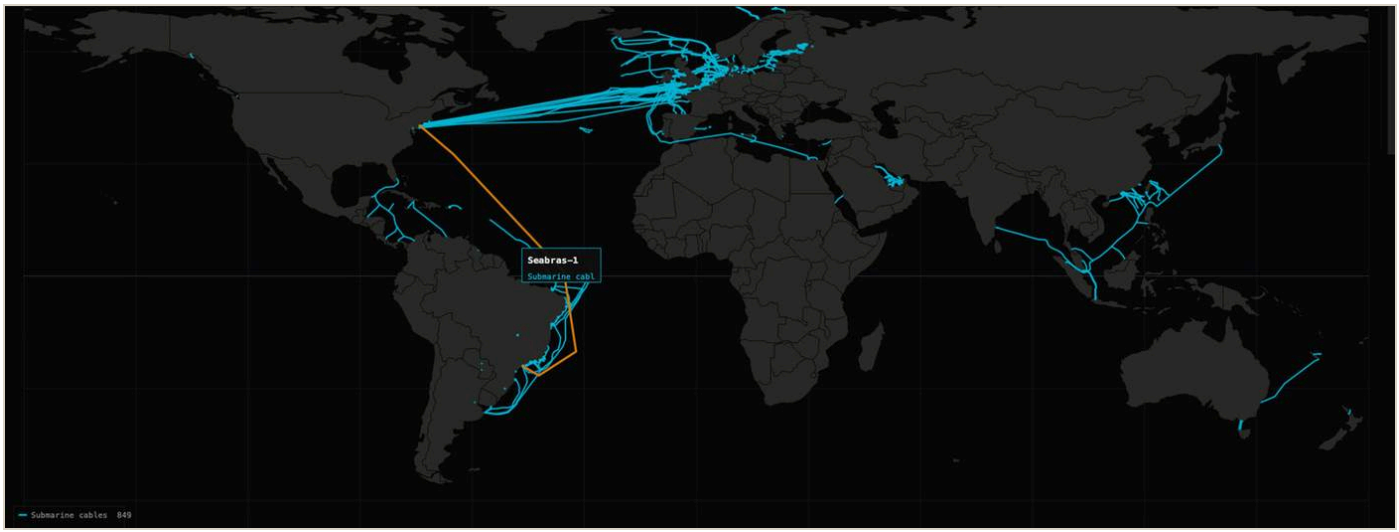


Fig 34 — Submarine cables, 849 routes. The hovered cable is picked out in amber and named.

Fig 35 — The feature card.

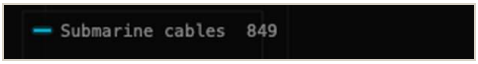


Fig 36 — The layer key.

Four things appear whenever a layer is on: the **features**, a **key** naming the layer and its colour bottom-left, the **source credit** bottom-right, and the **feature card** under the cursor. Hovering names the feature and adds what that layer knows about it — status, operator, capacity, or the countries it touches.

Each layer carries its own colour and its own credit line, so switching layers changes both.

Feature cards truncate mid-word. The card is a fixed width and its text is not wrapped or elided: the cable above reads *Submarine cabl*, and the Strait of Malacca reads *Countries: MYS, IDN,* — with the trailing comma and at least one missing country. Treat the card as an identifier, not as a complete record.

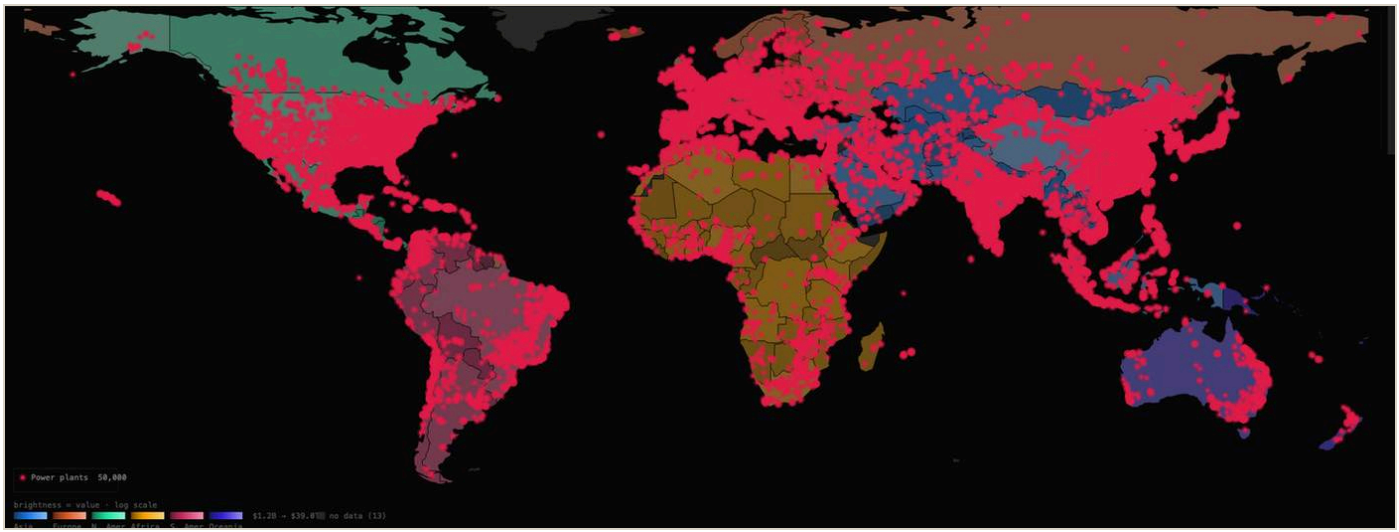


Fig 37 — Power plants. 142,163 exist; 50,000 are drawn.

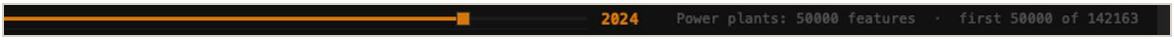


Fig 38 — The read-out says so, in full, rather than quietly drawing part of the layer.

Fig 39 — The source credit for that layer, bottom-right of the map.

> **50,000 features per layer is the ceiling.** Power plants is the only layer that reaches it. The read-out then prints `first 50000 of 142163` and the menu button reads **Power plants (50000)** rather than the true total — so **a sparse region on a capped layer may be a region the draw never reached**, not a region without power stations. Zoom does not re-query; the cap applies to the fetch, not the view.

17 Infrastructure over an indicator

The two halves of the map are meant to be read together — that is the point of putting them on one tab. Turning on a layer clears the indicator, so getting both takes one extra step: switch the layer on first, then re-pick the indicator.

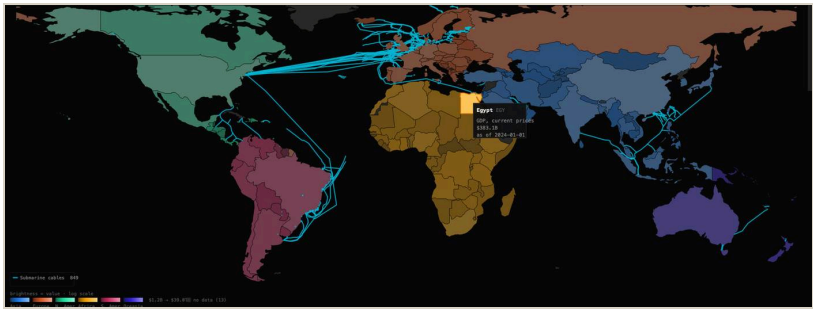


Fig 40 — GDP at current prices with submarine cables over it. Compare Fig 22: the same shading, deliberately held back.

With any layer on, the choropleth is **dimmed to about half brightness** so the lines and points read clearly over it — measured on the same country across the two captures, a fill of `(226,102,147)` becomes `(119,67,85)`. The ramp, the domain and the key do not change; only the drawn saturation does. Switch the layer off and full brightness returns immediately.

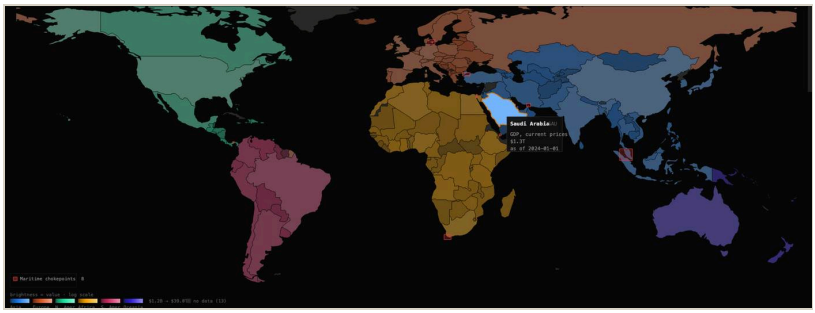


Fig 41 — The eight maritime chokepoints over the same indicator. Eight features are enough to change how a GDP map reads.

To get…	Do this
`Infrastructure only`	Switch on a layer. The indicator clears itself
`An indicator only`	Switch the layer off. Shading returns to full brightness
`Both`	Switch the layer on, **then** re-pick the indicator
`Neither`	Set the indicator to *None — infrastructure only* with no layer on — a plain basemap

> **The IMF credit stays on screen even with no indicator chosen.** Running a layer with the indicator set to *None* still prints *Source: IMF DataMapper…* under the key, alongside that layer's own credit. Nothing on the map is IMF data at that point.

18 The four dashboard tiles

Four of this tab's panels are also dashboard widgets, addable from **+ ADD** on the Dashboard tab. Each is configurable through its own gear.

Tile	What it shows	Configure
Trade Flows `trade_chord`	The flow wheel for one country and year	Country · Year (or *Latest available*) · Rank by · Partners drawn
Trade Treemap `trade_treemap`	The same answer as a treemap, grouped by continent	Same four fields — and the same fetch as the Trade Flows tile
World Map `world_choropleth`	One IMF indicator across every country	Country · Indicator · Colour: *Continent hue* or *Single ramp*
Infrastructure Map `infra_map`	One infrastructure layer	Layer · Status filter · Viewport as `minLon,minLat,maxLon,maxLat`

Things worth knowing before you pin them to a board

- **The two trade tiles are metered like the tab** — they run the same 1 CR query. They fetch **once per session** rather than on every dashboard refresh, precisely so that leaving a board open does not spend repeatedly.
- **The World Map and Infrastructure tiles are free**, like their counterparts on the tab.
- The World Map tile adds a **Single ramp** colour mode that the tab itself does not offer: one continuous scale across all countries instead of per-continent hues, which is the better choice for a ratio indicator.
- The Infrastructure tile adds a **status filter** — *Every status*, *Under construction*, *Proposed*, *Cancelled or shelved* — which the tab's own layer menu has no equivalent of.
- The Infrastructure tile takes a fixed **viewport**, so a board can hold one tile pinned to the Gulf and another to the North Sea. Leave it empty for the whole world.
- **One layer per tile** — the same single-layer rule as the tab (§14). Two layers means two tiles.

> Because the tiles are separate widgets they sidestep the tab's own single-layer limit: put four Infrastructure tiles on a board and you can watch four layers at once, each with its own viewport and status filter.

19 Every control on the tab

Where	Control	Type	What it does	Cost
`Filter bar`	**Country**	Dropdown	Reporting economy for the trade query; ~200 entries, alphabetical	Free
`Filter bar`	**Year**	Dropdown	Trade year, 2024–1995	Free
`Filter bar`	**Order by**	Dropdown	Total Trade / Imports / Exports / Trade Balance	Free
`Filter bar`	**Load**	Button	Runs the partner query; fills all three panels	**1 CR**
`Filter bar`	ⓘ	Button	Opens *What credits cost*; hovering gives the one-line price	Free
`Filter bar`	**Products**	Button	Product composition for the loaded country-year	**2 CR**
`Filter bar`	Read-out	Label	Rows returned, the charge, and the time	—
`View bar`	**Chord**	Toggle	The flow wheel (§06)	Free
`View bar`	**Treemap**	Toggle	Blocks sized by value (§07)	Free
`View bar`	**World Map**	Toggle	The indicator map (§10)	Free
`View bar`	**Infrastructure** ▾	Menu	Seventeen layers, one at a time (§14)	Free
`Map bar`	**Indicator**	Dropdown	131 entries: *None* plus 130 IMF series (§11)	Free
`Map bar`	▶ / ⏸	Button	Plays the indicator forward through every year	Free
`Map bar`	Year slider	Slider	Scrubs the map year; 1980–2031 for GDP	Free
`Map bar`	Coverage	Label	Countries with a figure, or the layer's feature count	—
`Map`	Scroll	Gesture	Zoom toward the cursor	Free
`Map`	Drag	Gesture	Pan	Free
`Map`	Double-click	Gesture	Reset the view — **and** select the country under the cursor	Free
`Map`	Click a country	Gesture	Loads it into the Country box; needs a Load to answer	Free
`Map`	Hover	Gesture	Country readout, or the feature card over a layer	Free
`Treemap`	Hover a block	Gesture	Name, continent or section, value, share of total	Free
`Table`	Column headers	Labels	Row order follows *Order by*, set in the filter bar	Free

> There is no export button, no copy-to-clipboard and no print on this tab. To get a table out, read it off the screen or rebuild the query in Report Builder.

20 Where the numbers come from

Half of the tab	Source	Extent
`Trade flows and products`	The Fincept bilateral trade corpus — **260 million rows**, built on BACI, named by the server's own price list	~200 reporters · 1995–2024 · up to 100 partners per answer · product detail to individual HS6 lines
`Country indicators`	**IMF DataMapper**. Attribution to the International Monetary Fund is required and the map prints the condition under its key	130 indicators · up to 197 countries · 1980–2031 for the headline WEO series, including published projections
`Infrastructure`	Per layer, credited on the map whenever that layer is visible — **OpenStreetMap contributors (ODbL)**, **Global Energy Monitor / Global Integrated Power**, and **Fincept**'s own chokepoint set	17 layers · **318,056 features** · up to 50,000 drawn per layer
`The basemap`	A country-outline file shipped inside the application, so the map draws with no network call at all	Land by continent over a graticule; borders and coastlines only

What the tab does about provenance

- **Every source is credited on the map itself**, bottom-right, and it changes as you change layers — not buried in an About box.
- **Every value carries its own as-of date** in the readout, so a 2024 map cell that is really a 2023 reading says so.
- **No-data is a category, not a zero.** The key counts it and the map draws it in its own grey (§12).
- **Coverage is printed per indicator and per year** — the bracket in the picker and the count beside the timeline.

> **The trade corpus is a corpus, not a feed.** Nothing on this tab is live or intraday. The newest trade year is 2024 and the newest indicator reading is annual. The one-second clock in the filter bar is a clock, not a refresh.

21 Every status message

The right-hand read-out is where the tab talks to you. Every message it can print, and what each one means.

Message	Means
`100 partners · 1 CR`	The query ran and was charged
`100 partners · no charge (same query)`	The identical request was repeated back-to-back; the server deduplicated it (§05)
`150 product lines (exports) · 2 CR`	A product sweep returned, and which leg it covers
`Selection changed — press Load.`	The bar and the panels disagree; nothing has been fetched
`Loading trade flows…`	The partner query is in flight
`Loading product composition…`	The product sweep is in flight
`Loading countries…`	The reporter list is being fetched, on first open
`Pick a country and a year first.`	Products was pressed with nothing selected
`<country> is not in the trade corpus.`	You clicked a country on the map that is not a reporter (§13)
`30 years loaded, 1995–2024. Press Load.`	The year list arrived; the tab is ready but idle
`The trade corpus holds no years yet.`	The corpus is empty on this environment
`Could not load the country list: …`	The reporter list failed; the error follows
`195 countries`	Map coverage for the chosen indicator and year
`195 countries · 12 still loading`	The map is drawing while more values arrive
`Loading indicators…`	The indicator list is being fetched
`No mappable indicators.`	The server published none that can be drawn
`Fetching from the IMF — reopen this view shortly.`	The indicator is being pulled upstream for the first time; come back to it
`Submarine cables: 849 features`	A layer loaded completely
`Power plants: 50000 features · first 50000 of 142163`	A layer hit the draw cap (§16)
`Drawing …`	A large layer is being rendered after the fetch
`No layers on this server — run the infrastructure loader.`	The layer catalogue is empty on this environment (§15)

> **Read the read-out before quoting a number.** It is the only place the tab tells you what the answer actually covers — which leg the product sweep returned, whether a layer was truncated, and how many countries a map year really has.

22 Honest notes

Everything below was reproduced on the shipping v5.0.1 build. None of it is fatal; all of it will change what you conclude if you do not know it.

Things that will mislead you if you do not know them

- **The wheel draws fifteen partners of a hundred**, and only the five largest export legs are drawn in magenta at all. The table is the answer (§06).
- **Only one infrastructure layer can be on at a time**, and switching layers silently clears the previous one (§14).
- **Power plants is truncated at 50,000 of 142,163.** An empty region on that layer may be an undrawn region (§16).
- **The map timeline runs to 2031** — the last seven years are IMF projections, drawn exactly like reported history with nothing to distinguish them (§12).
- **Three indicator names appear twice**, one pair with different coverage counts, and the picker gives you no way to tell which is which (§11).
- **Double-clicking the map to reset the view also changes the trade country** if you happen to be over land (§13).
- **The product read-out names one leg.** A product figure is not necessarily that country's total trade in the line (§09).

Cosmetic, but visible

- **Feature cards truncate mid-word** — *Submarine cabl*, *Countries: MYS, IDN,* (§16).
- **The map key's continent labels collide** — the six ramps print as *N. Amer…Africa* and *S. Amer…Oceania*, two labels running together with no gap.
- **The credit dialog still says Trade Viz** in both meter captions, and the Load tooltip reads *Costs 1 credits* (§05).
- **The IMF credit stays on screen with no IMF data on it**, when a layer is drawn with the indicator set to *None* (§17).
- **Country names lose their hyphens** in the picker — *TimorLeste*, *GuineaBissau*.
- **An empty *Test pipelines (0)* layer ships in the production menu** (§15).
- **After a product sweep, switching back to the wheel leaves the product table underneath it** — and the totals strip still reads *Summed over the 100 partners shown* over a table of products. Press Load to put the two back in step.

> **No AI or agent tools reach this tab.** The terminal registers twenty-two groups of assistant tools — maritime, geopolitics, equity research, dashboards, notebooks and the rest — and none of them covers trade flows, country indicators or infrastructure. Ask the assistant about a bilateral trade balance and it cannot query this corpus; it has no route to it. Everything on this tab is done by hand.

23 Quick reference — how do I…

How do I…	Do this	Cost
Open the tab	**Navigate ▸ Tools & Workspace ▸ Global Economy**, or type `geco` in the command bar	Free
Get a country's trading partners	Pick the country and year, press **Load**	1 CR
See exports separately from imports	Read the ranking table — the wheel under-draws exports	Free
Rank by surplus instead of turnover	**Order by** → **Trade Balance**, then Load	1 CR
Compare the same answer three ways	Flip between **Chord**, **Treemap** and the table	Free
See a partner's share of the total	Hover its block on the treemap	Free
Find what a country actually sells	Load it, then press **Products**	2 CR
Tell a broad product line from a narrow one	Read the **HS6 lines** column	Free
Shade the world by GDP	**World Map**, then pick the indicator	Free
Choose a different indicator	The indicator dropdown — 130 of them	Free
See how an indicator changed over fifty years	Drag the timeline, or press ▶	Free
Tell a gap in the data from a zero	Grey countries, counted as *no data (n)* on the key	Free
Check whether a figure is reported or projected	Read the year — anything past the last actual year is a WEO projection	Free
Zoom in on a region	Scroll over it; drag to pan; double-click over ocean to reset	Free
Jump from the map to a country's trade	Click the country, then press **Load**	1 CR
Put pipelines on the map	**Infrastructure ▾** → **Gas pipeline routes**	Free
See subsea cables and GDP together	Switch the layer on, **then** re-pick the indicator	Free
Identify a cable, plant or port	Hover it — the card names it and its operator or status	Free
Find who supplied a layer	The credit bottom-right of the map, which changes per layer	Free
Watch two layers at once	Not on this tab — use two **Infrastructure Map** dashboard tiles (§18)	Free
Keep a trade answer on a board	Add the **Trade Flows** or **Trade Treemap** tile	1 CR/session
Check what anything here costs	Click either ⓘ in the filter bar	Free
Avoid paying twice for the same question	Do not change and change back — only a back-to-back repeat is free	—

> **The short version.** Two buttons on this tab spend credits and nothing else does. Load buys a country-year, drawn three ways, and Products buys the same country-year broken down by commodity. The entire world map — 130 indicators, fifty years, 318,000 infrastructure features — is free to read for as long as you like.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Data Sources

A complete, screen-by-screen walkthrough of the Data Sources tab — the connector command-center where you browse a catalog of 118 built-in data connectors, configure named connections, and test and manage them.

`>_ 118 connectors · 11 categories · 2 views`

Covers · **the two views** · **BROWSE** · **CONNECTIONS** · **the connector Inspector** **the Connection Config window** · **118 connectors** across 11 categories · testing & live status · import / export

Demonstrated live on the Fincept Terminal Data Sources tab.

Contents

9 SECTIONS

GETTING ORIENTED

Overview — what the Data Sources tab is 3

1 Anatomy of the screen (header, stats, tabs, two views) 4

BROWSING & CONFIGURING

2 BROWSE — search & filter the catalog 5

3 The connector Inspector 6

4 The Connection Config window (add / edit / clone) 7

5 Testing connections & live status 9

MANAGING & REFERENCE

6 CONNECTIONS — manage saved connections 10

7 Import / Export / Template 10

8 The connector catalog — 118 connectors by category 12

9 Security & data handling · Quick reference · Notes 14

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal, including its live catalog, inspector and connection dialog. Control labels are quoted exactly as they appear on screen.

What the Data Sources tab is

OVERVIEW

Data Sources is a **connector command-center**. It ships with a catalog of **118 built-in connector definitions** across 11 categories; you pick a connector, fill in a schema-driven form to save a named connection, test it, enable it, and manage all your connections in one place. The connector catalog is a static in-app registry; your saved connections persist to the backend and sync per user.

The screen has two views (sub-tabs) toggled by a tab bar — BROWSE (the catalog gallery plus a detail inspector) and CONNECTIONS (a flat table of everything you've saved) — plus one pop-up window, the Connection Config dialog, used to add, edit or clone a connection. Categories span databases, market-data and crypto APIs, cloud storage, streaming, time-series, search, warehouses, alternative data and open banking.

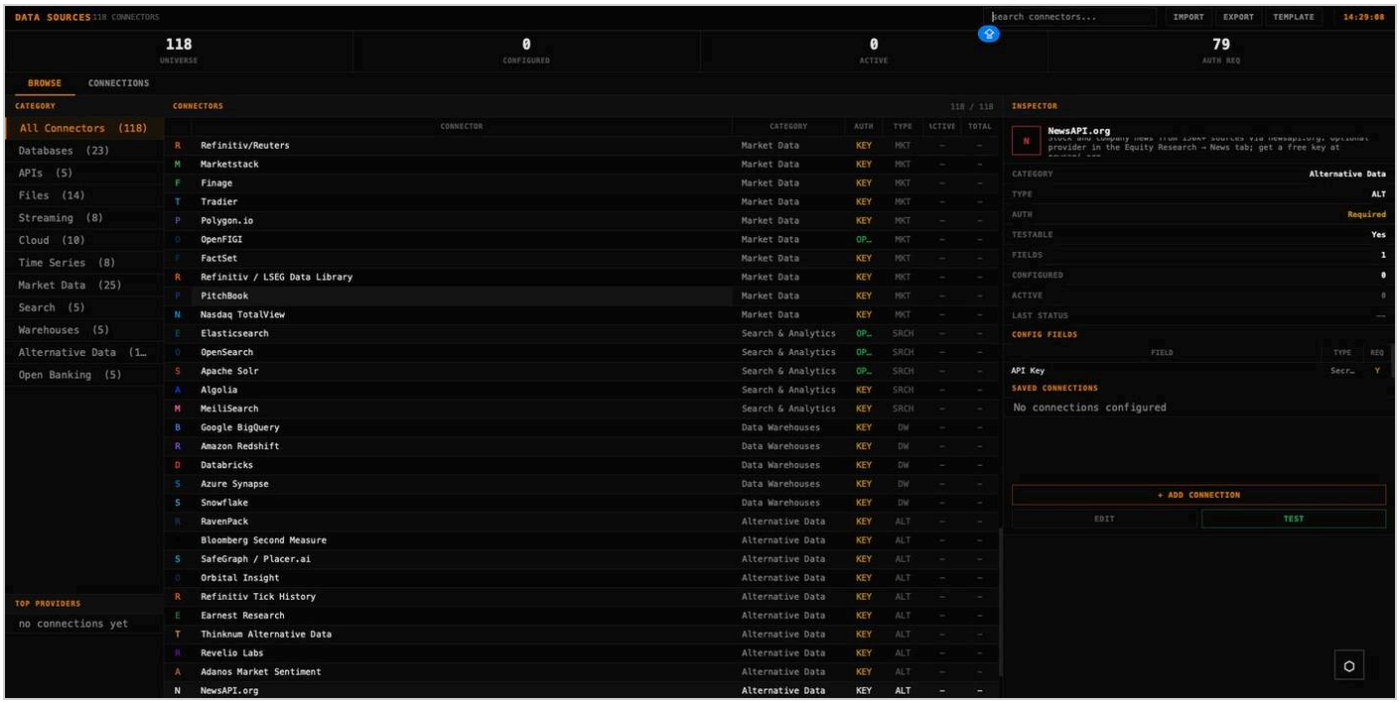
AT A GLANCE, IN DATA SOURCES YOU CAN…

- Browse **118 connectors**, search by name/type/description, and filter by category from the sidebar.
- Read headline counts (Universe / Configured / Active / Auth-required) and click them to filter the catalog.
- Inspect a connector's metadata and config-field schema, and see which of your connections belong to it.
- Add, edit or clone a named connection through a dynamic, validated form (masked secrets, tags, enable toggle).
- Test a connection (TCP reachability), with auto status refresh every 30 seconds.
- Manage all connections in one table — inline enable/disable, and bulk enable-all / disable-all / delete-selected.
- Import / export your connections as JSON, and download a pre-filled template covering every connector.

1 Anatomy of the screen

ORIENTATION

Open the tab from the command bar (type `data` → **Data Sources**) or **Navigate ▸ Tools ▸ Data Sources**. The BROWSE view below shows the whole workbench.



BROWSE view. ① Screen header (title · search · Import/Export/Template · clock) ② Stats strip (4 filter boxes) ③ View tabs (Browse / Connections) ④ Category sidebar + Top Providers ⑤ Connector table ⑥ Detail inspector.

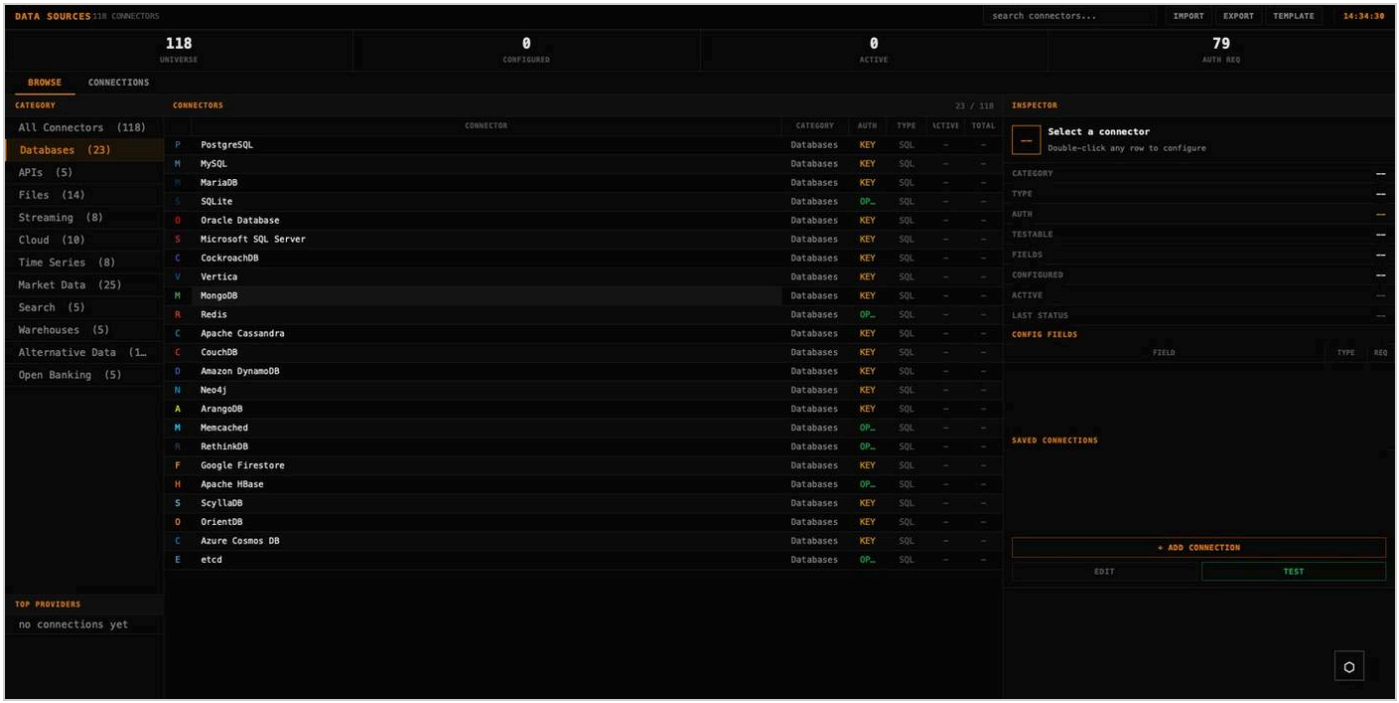
Region	What it shows / does
① **Screen header**	**DATA SOURCES** title + a live `N CONNECTORS` count; a **connector search** box; **IMPORT / EXPORT / TEMPLATE** buttons; and a live `HH:MM:SS` clock. The search filters the Browse catalog only (not the Connections table).
② **Stats strip**	Four boxes — **UNIVERSE** (total connectors), **CONFIGURED** (connectors with ≥1 saved connection), **ACTIVE** (enabled connections), **AUTH REQ** (connectors needing credentials). Clicking a box toggles it as a filter (amber highlight); clicking again clears it. *UNIVERSE only highlights — it applies no filter.*
③ **View tabs**	**BROWSE** (catalog gallery + inspector) and **CONNECTIONS** (saved-connections table).
④ **Category sidebar**	12 rows — **All Connectors** + the 11 categories, each with a live count. Selecting one filters the catalog. Below it, **TOP PROVIDERS** lists your most-used connectors (shows "no connections yet" when empty).
⑤ **Connector table**	The catalog. Columns: *badge* · **CONNECTOR** · **CATEGORY** · **AUTH** (KEY=needs credentials / OPEN) · **TYPE** (transport badge: SQL/REST/WS/FILE/CLOUD/TSDB/MKT/SRCH/DW/ALT/OB) · **ACTIVE** · **TOTAL** (your enabled / total connections for that connector). Single-click selects (fills the inspector); double-click opens the config window.
⑥ **Detail inspector**	Full metadata + config-field schema for the selected connector, its saved connections, and the **+ ADD CONNECTION / EDIT / TEST** buttons (see §3).

> **Where connections live.** The connector catalog is a static in-app registry (118 definitions). Your saved *connections* are persisted through the Exclusive API settings store and **sync per user**, so they follow you across machines. A 1-second clock and a 30-second connection-status poll run only while this tab is visible.

2 BROWSE — search & filter

VIEW 1

The Browse view is a scannable catalog. Narrow it three ways — the category sidebar, the header search box, and the stats-strip filter boxes — which combine.



Browse filtered to the **Databases** category (23 connectors). The count reads "23 / 118"; the AUTH column shows KEY vs OPEN connectors; the inspector shows its empty "Select a connector" state.

Control	Type	What it does
Category sidebar	list	Click a category to filter the table to it; "All Connectors" shows everything. Each row's count reflects the current search.
search connectors…	lineedit	Case-insensitive substring match over each connector's name / id / type / description / category. Re-filters the table and updates the sidebar counts live.
Stats-strip boxes	toggle	CONFIGURED / ACTIVE / AUTH REQ each narrow the catalog to matching connectors; click again to clear.
Top Providers row	list	Click a provider to jump to (select) that connector.
Connector row	row	**Single-click** = select & inspect. **Double-click** = open the config window (edit if a connection exists, else add).

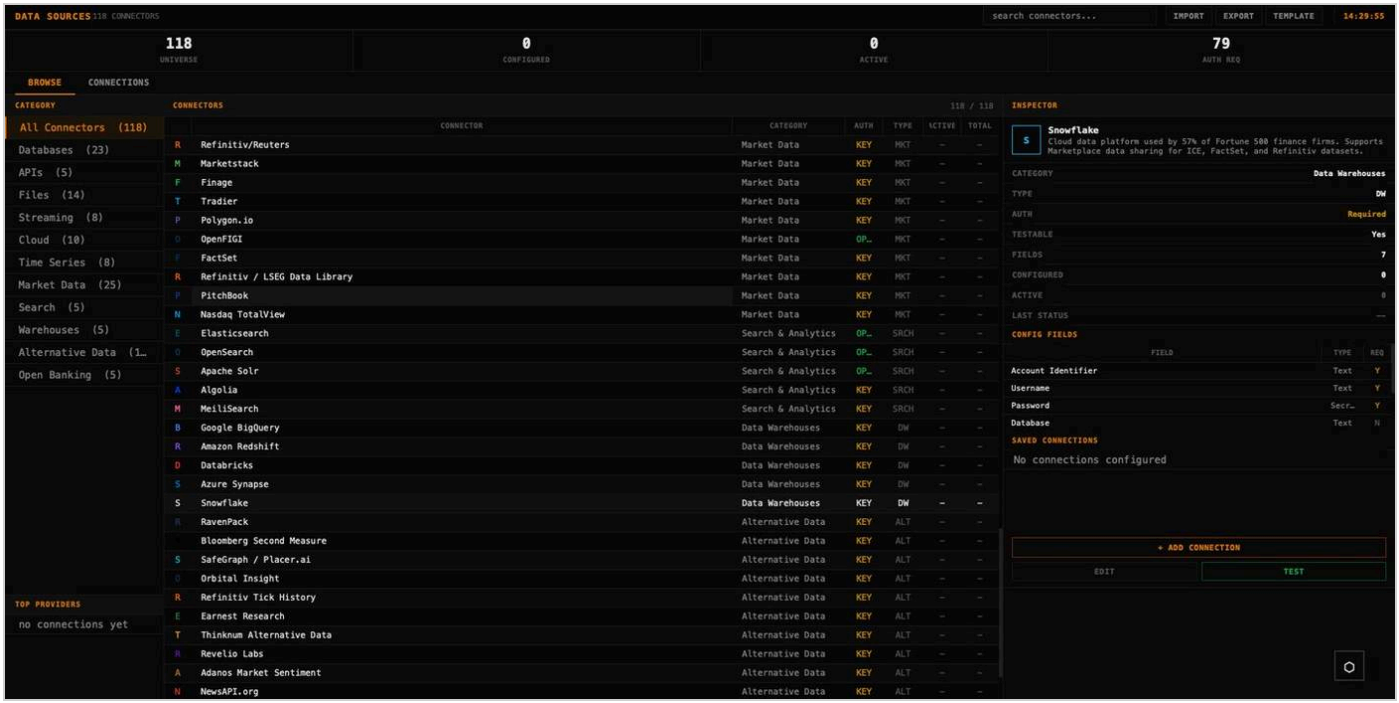
At a glance, in BROWSE you can…

- Find any of the 118 connectors by category, keyword, or configured/active/auth status.
- See at a glance which connectors need credentials (KEY) vs are open, and how many connections you already have for each.

3 The connector Inspector

RIGHT PANEL

Selecting a connector fills the right-hand inspector with everything about it — metadata, the exact configuration schema, your saved connections for it, and the three actions.



Inspector for **Snowflake** — category, type, auth, testable, field count, and the CONFIG FIELDS schema (Account Identifier · Username · Password[secret] · Database…), with **+ ADD CONNECTION / EDIT / TEST**.

Element	What it shows
Identity	Connector badge, name and description.
Metadata rows	**CATEGORY** · **TYPE** · **AUTH** (Required/None) · **TESTABLE** (Yes/No) · **FIELDS** (schema size) · **CONFIGURED** (your total connections) · **ACTIVE** (enabled) · **LAST STATUS** (OK / ERR / —).
CONFIG FIELDS	The schema table — **FIELD** · **TYPE** · **REQ** — one row per input the connector needs (Text / Secret / Number / URL / Select / Textarea / Bool). "Y" marks required fields.
SAVED CONNECTIONS	Your connections for this connector, each tagged `[ACTIVE]` / `[OFF]`. Click one to target EDIT / TEST at it. Empty state: "No connections configured".
+ ADD CONNECTION	Opens the config window to create a new connection (see §4).
EDIT	Opens the config window to edit the effective connection (enabled once one is selected/exists).
TEST	Runs a connectivity test on the effective connection (see §5).

> **Empty state.** With nothing selected, the inspector reads "Select a connector — Double-click any row to configure", all values show "—", and the three buttons are disabled.

4 The Connection Config window

POP-UP WINDOW

A modal pop-up used to add, edit or clone a connection. Its form is **generated from the connector's schema** — every connector shows a different set of fields. The title reads `Configure — <name>` (add), `Edit — <name>`, or `Clone — <name>`.

The **Configure — Snowflake** window: header (badge + name + description), Connection Name, Enable toggle, the dynamic per-connector fields (Account Identifier* · Username* · Password* · Database · Schema · Virtual Warehouse · Role), Tags, and Cancel / Save Connection. Required fields are marked *.

Field / button	Type	What it does
Connection Name	lineedit	A friendly label (defaults to "<connector> Connection"; "Copy of…" when cloning). Not required.
Enable Connection	checkbox	"Active" — whether the connection is enabled on save (default on).
Dynamic connector fields	text / password / number / URL / select / textarea / checkbox	One input per schema field. Passwords are masked; numbers are integer-validated; selects are dropdowns; textareas hold JSON (e.g. service-account keys). Fields marked * are required.
Tags	lineedit	Comma-separated tags (e.g. prod, live) for filtering later.
Cancel	button	Discards — nothing is saved.
Save / Update Connection	button	Validates required fields (blocks with "Missing required field: X"), then persists. Adding/cloning mints a new connection; editing updates in place. The screen

Field / button	Type	What it does
		refreshes and selects it.

> **Security — read this.** Every field value, **including API keys, tokens and passwords, is stored in plaintext** inside the connection record (it is *not* placed in the OS keychain, unlike broker credentials). Those secrets are therefore also written verbatim into any **Export** file. Treat exported connection files as sensitive credential dumps and store them accordingly.

5 Testing connections & live status

TEST

The **TEST** button (inspector) runs a connectivity check and pops a result dialog; a background poll keeps every enabled connection's status fresh.

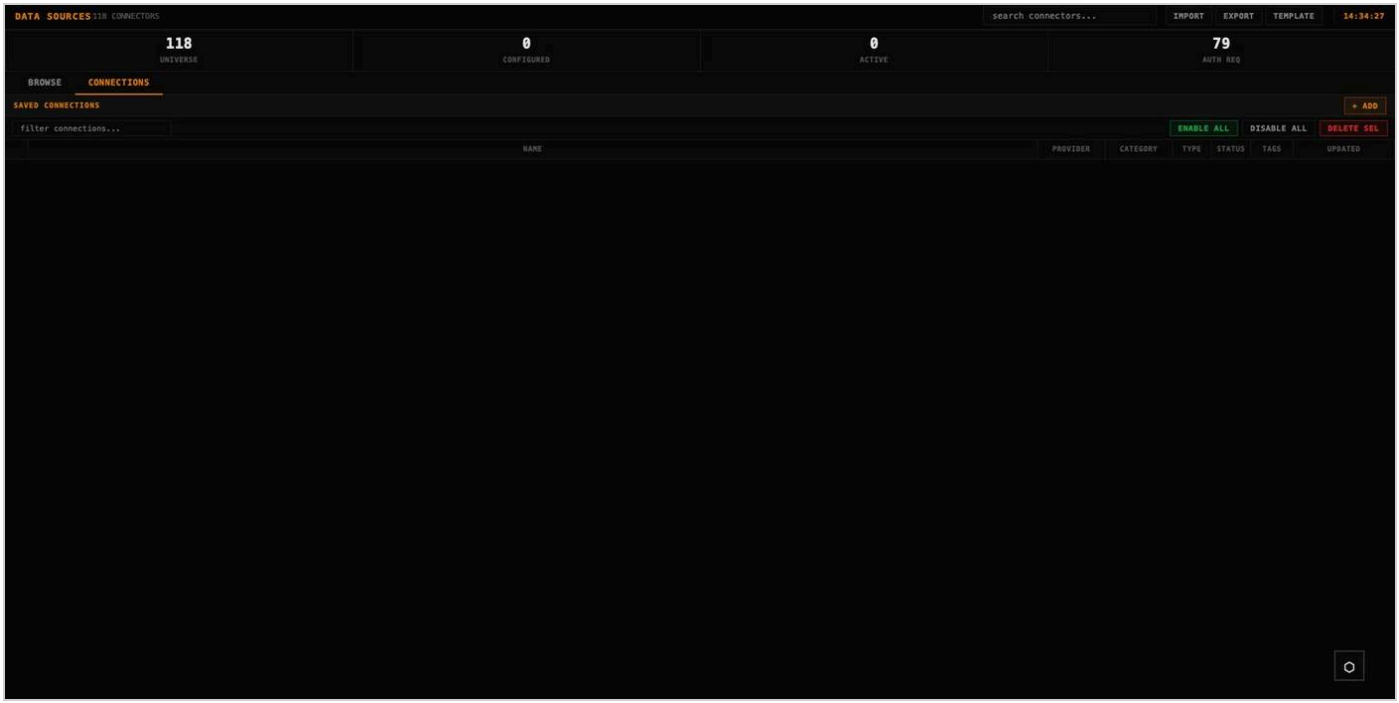
Aspect	Behaviour
What "Test" does	Resolves a provider-specific probe endpoint (a built-in table of each API's cheap health URL, e.g. Binance `/api/v3/ping`) or the saved host/port, then opens a **TCP socket** to it on a worker thread (5 s for a URL, 3 s for host:port).
Result dialog	"Connection successful" (green) or "Connection failed" (red) + the probe message. On success it adds: "*Note: TCP reachability confirmed. API key validity is not verified here.*"
Non-testable connectors	File/table formats (CSV, Parquet, Iceberg…) and SQLite show "This connector does not support connectivity testing." (13 of 118 are non-testable.)
Background poll	Every **30 s** (while the tab is visible) each enabled + testable connection is probed; the STATUS cell and LAST STATUS update to **OK** / **ERR** / **—**, with the message as a tooltip.

> **What a green result means.** Testing confirms only that a **TCP socket opened to the host** — it does **not** validate your API key, auth, or HTTP status. A wrong or expired key can still show "OK". Use it to confirm reachability/firewall, not credential validity.

6 CONNECTIONS — manage saved

VIEW 2

The second view is a flat table of everything you've saved, with inline enable toggles, live status, and bulk actions.



The CONNECTIONS view — **+ ADD**, a connections filter, the bulk actions **ENABLE ALL / DISABLE ALL / DELETE SEL**, and the table columns. (Shown empty — no connections are configured on this instance.)

Control / column	What it does / shows
+ ADD	Opens the config window to add a connection.
filter connections…	Substring filter over name / provider / category / tags.
ENABLE ALL / DISABLE ALL	Enable or disable every connection in the current filter (no confirmation).
DELETE SEL	Deletes the selected rows — **the only action with a confirm dialog** ("Delete N selected connection(s)? This cannot be undone.", default Cancel).
Table columns	*enable checkbox* · **NAME** · **PROVIDER** · **CATEGORY** · **TYPE** · **STATUS** (OK/ERR/—) · **TAGS** · **UPDATED**. Each row is one saved connection; the checkbox toggles enabled instantly.

> **Where actions live.** There are no per-row edit/test/duplicate buttons and no right-click menu in this table. **Edit** and **Test** a connection from the Browse inspector; **enable/disable** via the row checkbox or bulk buttons; **delete** via multi-select + DELETE SEL. (A populated table shows one row per connection with its live status badge.)

7 Import / Export / Template

JSON I/O

Button	What it does
EXPORT	Writes all your connections to `fincept_connections.json` (pretty-printed array). **Includes config verbatim — i.e. plaintext secrets.**
IMPORT	Reads a JSON array and adds each entry (requires a `provider`). **Import is additive** — every entry gets a fresh id, so re-importing the same file creates duplicates; imported connections default to disabled.

Button	What it does
TEMPLATE	Writes `fincept_connections_template.json` — a blank scaffold with one fill-in-the-blank entry per connector (fields pre-keyed with defaults, `enabled:false`, plus a `_help` note). Contains no secrets.

8 The connector catalog

118 CONNECTORS

118 connectors across 11 categories. **105** support a live TCP test; **79** require credentials. Below, every connector by category, with its auth style.

Category	Count	Nature
Market Data	25	Equity/crypto/FX price & reference APIs
Databases	23	8 relational + 15 NoSQL
Files	14	File & table formats + FTP/SFTP
Cloud	10	Object storage (S3 & compatibles)
Alternative Data	10	Sentiment, geospatial, web/transaction data
Streaming	8	Message buses & real-time feeds
Time Series	8	TSDBs & metrics stores
APIs	5	Generic protocol connectors
Search	5	Search & analytics engines
Warehouses	5	Cloud data warehouses
Open Banking	5	OAuth2 financial-data aggregators

Market Data (25)

Yahoo Finance ·open	Alpha Vantage ·key	Finnhub ·key	IEX Cloud ·token	Twelve Data ·key
Quandl / Nasdaq Data Link ·key	Binance ·key+secret	Coinbase ·key+secret	Kraken ·key+priv	CoinMarketCap ·key
CoinGecko ·open	EOD Historical ·token	Tiingo ·token	Intrinio ·key	Fincept Data API ·host
Refinitiv / Reuters ·appkey+user	Marketstack ·key	Finage ·key	Tradier ·token	Polygon.io ·key (+WS)
OpenFIGI ·open	FactSet ·user+key	Refinitiv LSEG ·appkey	PitchBook ·key	Nasdaq TotalView ·user/pass

Databases (23) — 8 relational + 15 NoSQL

PostgreSQL ·user/pass	MySQL ·user/pass	MariaDB ·user/pass	SQLite ·file	Oracle ·user/pass
SQL Server ·user/pass	CockroachDB ·user/pass	Vertica ·user/pass	MongoDB ·user/pass	Redis ·open
Cassandra ·user/pass	CouchDB ·user/pass	DynamoDB ·AWS keys	Neo4j ·user/pass	ArangoDB ·user/pass
Memcached ·open	RethinkDB ·open	Firestore ·SA JSON	Apache HBase ·open	ScyllaDB ·user/pass
OrientDB ·user/pass	Azure Cosmos DB ·key	etcd ·open		

Files (14) — only FTP & SFTP are testable

CSV	Excel (XLSX)	JSON	Parquet	XML	Avro	ORC
Feather	FTP ·user/pass	SFTP ·user/key	Apache Iceberg	Delta Lake	GeoParquet	Arrow IPC

Cloud (10) · Warehouses (5) · Search (5)

Cloud storage
AWS S3 · Google Cloud Storage · Azure Blob
DigitalOcean Spaces · MinIO · Backblaze B2
Wasabi · Cloudflare R2 · Oracle Cloud · IBM COS

Warehouses & Search
DW: BigQuery · Redshift · Databricks · Azure Synapse · Snowflake
Search: Elasticsearch · OpenSearch · Apache Solr · Algolia · MeiliSearch

Streaming (8) · Time Series (8) · APIs (5)

Streaming
WebSocket · MQTT · Apache Kafka · RabbitMQ
NATS · Redpanda · Apache Flink · FIX Protocol
APIs (generic)
REST API · GraphQL · gRPC · SOAP · OData

Time Series
InfluxDB · QuestDB · TimescaleDB · Prometheus
VictoriaMetrics · OpenTSDB · kdb+ · ClickHouse

Alternative Data (10) · Open Banking (5)

Alternative Data (all ·key)
RavenPack · Bloomberg Second Measure
SafeGraph/Placer.ai · Orbital Insight
Refinitiv Tick History · Earnest Research
Thinknum · Revelio Labs · Adanos · NewsAPI.org

Open Banking (OAuth2)
Plaid · MX Technologies
Finicity (Mastercard) · TrueLayer
FDX API

Legend: open = no credentials · key = API key · token = access token · user/pass = username+password · SA JSON = service-account key · AWS keys = access key ID + secret.

9 Security, quick reference & notes

REFERENCE

Quick reference — "How do I…?"

I want to…	Do this
Find a connector	BROWSE ▸ pick a category or type in the search box.
See what a connector needs	Single-click it → read CONFIG FIELDS in the inspector.
Add a connection	Select the connector ▸ **+ ADD CONNECTION** ▸ fill the form ▸ Save.
Edit a connection	BROWSE ▸ inspector ▸ pick the saved connection ▸ **EDIT**.
Check a connection works	Inspector ▸ **TEST** (TCP reachability).
Turn a connection on/off	CONNECTIONS ▸ row checkbox (or ENABLE ALL / DISABLE ALL).
Delete connections	CONNECTIONS ▸ select rows ▸ **DELETE SEL** (confirms).
Move connections to another machine	EXPORT here ▸ IMPORT there (remember: export holds plaintext secrets).
Bootstrap many connections	TEMPLATE ▸ fill the JSON ▸ IMPORT.

Notes for institutional users

- **Secrets are stored in plaintext** in the connection record (not the OS keychain) and are included in Export files. Protect exported `.json` files as credential material.
- **A successful Test is TCP-only** — it proves the host is reachable, not that your credentials are valid.
- **Connections sync per user** via the backend settings store; the connector catalog itself is a fixed in-app registry (118 definitions).
- **Import is additive** — re-importing the same file creates duplicate connections (each gets a new id); imported connections start disabled.
- **Only DELETE SEL confirms.** ENABLE ALL / DISABLE ALL apply immediately with no prompt.
- **Non-testable connectors (13):** the 12 file/table formats and SQLite — local/file-based, so there's no endpoint to probe (FTP & SFTP are the testable File members).
- **Editing & deleting live in specific places:** edit/test from the Browse inspector; delete via bulk DELETE SEL. There is no per-row menu on the Connections table.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Data Mapping

A complete, screen-by-screen walkthrough of the Data Mapping tab — the engine that turns any third-party market-data or broker API into the terminal's seven standard schemas through a guided 5-step wizard, 14 templates and a live test harness.

`>_ connect any API · normalise to a standard schema`

Covers · **the 3 views** · **the 5-step wizard** · **7 standard schemas**
14 templates · **the live test harness** · transforms & caching · network, storage & security

Demonstrated live on the Fincept Terminal Data Mapping tab.

Contents

11 SECTIONS

GETTING ORIENTED

Overview — what the Data Mapping tab is 3

1 Anatomy of the screen (chrome, panels, status) 4

VIEWS & THE 5-STEP WIZARD

2 View · MAPPINGS — the saved-mappings list 5

3 View · TEMPLATES — 14 ready-made APIs 6

4 Wizard Step 1 · API Config 8

5 Wizard Step 2 · Schema 9

6 Wizard Step 3 · Field Mapping 10

7 Wizard Step 4 · Cache & Security 11

8 Wizard Step 5 · Test & Save 12

REFERENCE

9 Reference — the 7 schemas, parsers, transforms, templates 13

10 Under the hood — network, storage, security 14

11 Quick reference & notes 15

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal, taken while building a live Binance-to-OHLCV mapping end to end. Control labels are quoted exactly as they appear on screen.

What the Data Mapping tab is

OVERVIEW

The Data Mapping tab (the **DATA MAPPING ENGINE**) lets you plug any HTTP/JSON data source — a crypto exchange, an equity data vendor, your broker's REST API, a macro-data service — into the terminal and normalise its response into one of seven standard schemas. You describe the request, pick a target schema, map each schema field to a path in the API's JSON response, test it against a live sample, and save. Saved mappings can then be run on demand to pull and normalise fresh data.

Three things make it approachable: a guided 5-step wizard (① API Config → ② Schema → ③ Field Mapping → ④ Cache → ⑤ Test & Save), 14 pre-built templates for popular APIs so common sources are one click, and a live test harness that fetches a real sample and shows you exactly what the normalised output looks like before you commit. The tab has three views: MAPPINGS (your saved mappings), TEMPLATES (the ready-made APIs), and CREATE (the wizard).

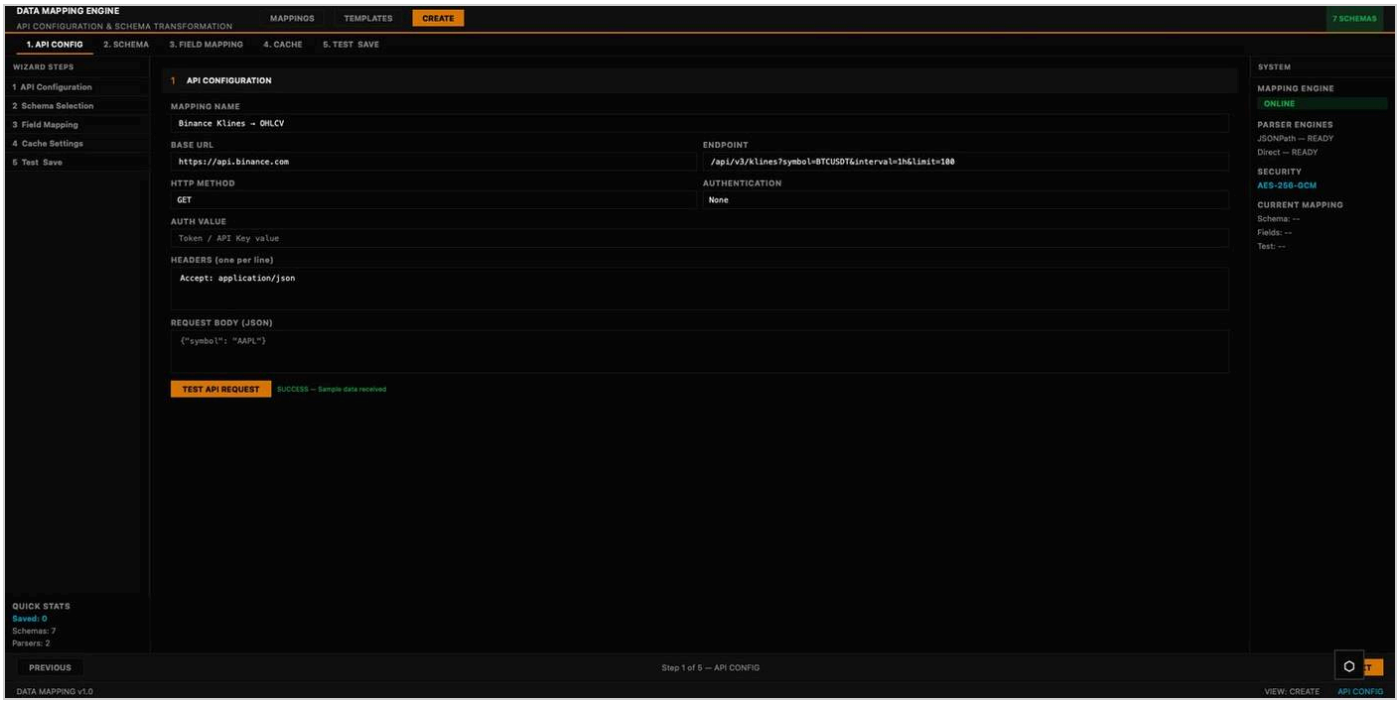
AT A GLANCE, IN DATA MAPPING YOU CAN…

- Connect any REST/JSON API with a request builder (URL, method, auth, headers, body) and test it live.
- Target one of **7 standard schemas** — OHLCV, QUOTE, TICK, ORDER, POSITION, PORTFOLIO, INSTRUMENT.
- Map fields visually with a JSON explorer beside a target-field table, using JSONPath (or plain "Direct" paths) plus per-field transforms and defaults.
- Start from **14 templates** (CoinGecko, Yahoo, Binance, Kraken, Alpha Vantage, Twelve Data, Finnhub, CoinMarketCap, Alpaca, Tradier, DBnomics).
- Dry-run the mapping against a live sample and preview the normalised rows before saving.
- Cache responses (TTL), and keep API credentials encrypted (AES-256-GCM) — never in plaintext.

1 Anatomy of the screen

ORIENTATION

Reach the tab from the command bar (type `data` → **Data Mapping**) or **Navigate ▸ Tools ▸ Data Mapping**. Every view shares the same frame:



The Data Mapping engine (Create ▸ Step 1). ① Header + view switcher ② Step bar ③ Left step-nav + Quick Stats ④ Center work area ⑤ Right SYSTEM panel ⑥ Nav footer ⑦ Status bar.

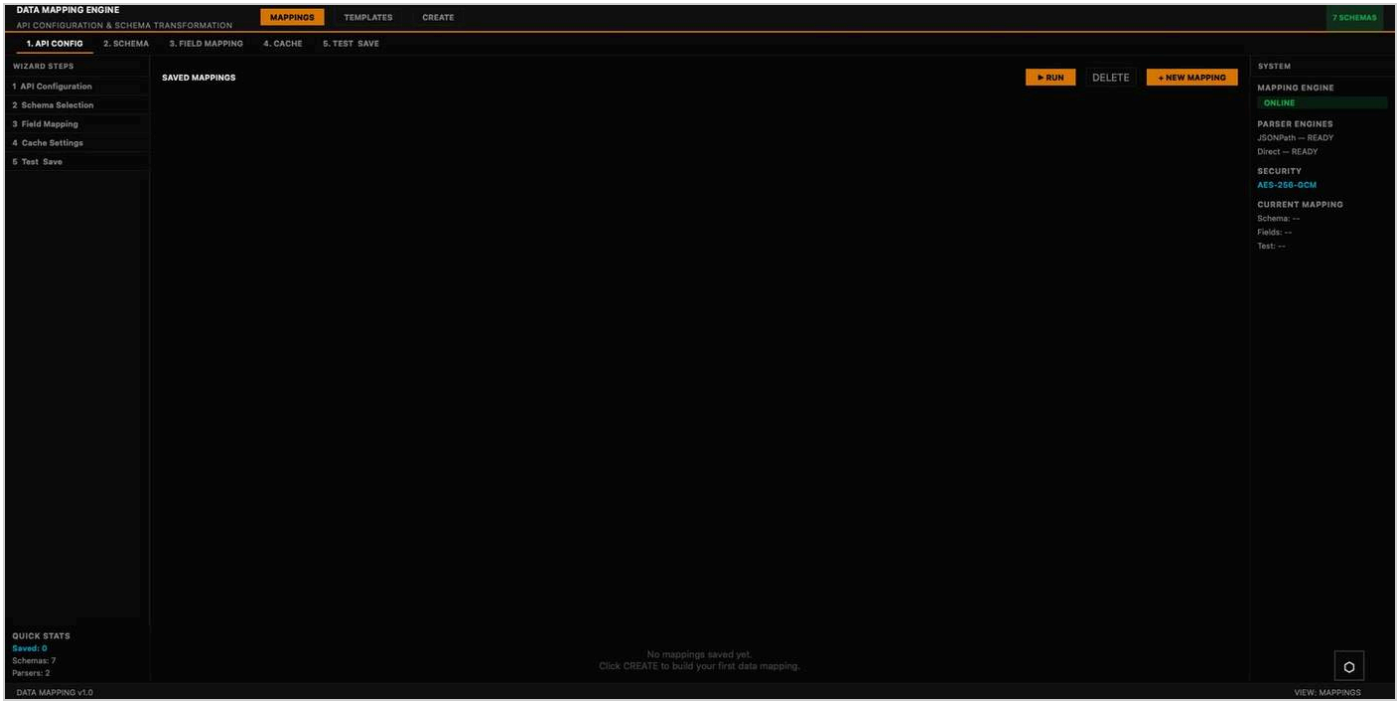
Region	What it shows / does
① **Header + view switcher**	Title **DATA MAPPING ENGINE**, three view buttons **MAPPINGS · TEMPLATES · CREATE** (the active one is amber), and a green **7 SCHEMAS** badge.
② **Step bar**	**1. API CONFIG · 2. SCHEMA · 3. FIELD MAPPING · 4. CACHE · 5. TEST & SAVE**. The active step is underlined amber, completed steps turn green. (These chips jump between wizard pages; the *left-panel* step buttons additionally switch into the Create view.)
③ **Left panel**	**WIZARD STEPS** — five buttons that jump straight to a step in the Create wizard — and **QUICK STATS**: Saved (live count), Schemas (7), Parsers (2).
④ **Work area**	The current view's content (list, templates, or a wizard step).
⑤ **Right SYSTEM panel**	Engine status (**MAPPING ENGINE · ONLINE**), the **PARSER ENGINES** (JSONPath, Direct), the **SECURITY** standard (AES-256-GCM), and **CURRENT MAPPING** — a live readout of the selected Schema, its field count, and the last Test result.
⑥ **Nav footer**	Shown only in the Create view: **PREVIOUS / NEXT** and a "Step N of 5 — NAME" label. (Prev is disabled on step 1, Next on step 5.)
⑦ **Status bar**	**DATA MAPPING v1.0**, the current **VIEW**, and the active step name.

> **Good to know.** Moving **Next/Prev** through the wizard does **no validation** — you can page through freely. The only gate is the final **Save**, which needs a successful test and a mapping name (§8).

2 View · MAPPINGS (the list)

HOME

Your saved mappings live here. It's the landing view and the launch point for running a mapping or creating a new one.



The MAPPINGS list (empty state shown). Toolbar: ▶ RUN · DELETE · + NEW MAPPING. Saved mappings appear as "*name — schema*" rows.

Control	Type	What it does
Saved-mapping list	list	One row per saved mapping, shown as `name — schema`. Select a row to target RUN / DELETE.
▶ RUN	button	Fetches the selected mapping's endpoint live, normalises it, and pops a summary — "Extracted N row(s) for schema X" — with a preview of the first 5 rows. Also persists the run's output locally.
DELETE	button	Confirms ("Delete '…'? This cannot be undone."), then removes the mapping **and** its stored credential.
+ NEW MAPPING	button	Clears the wizard and jumps to Create ▸ Step 1 for a fresh mapping.
Empty state	—	"No mappings saved yet. Click CREATE to build your first data mapping."

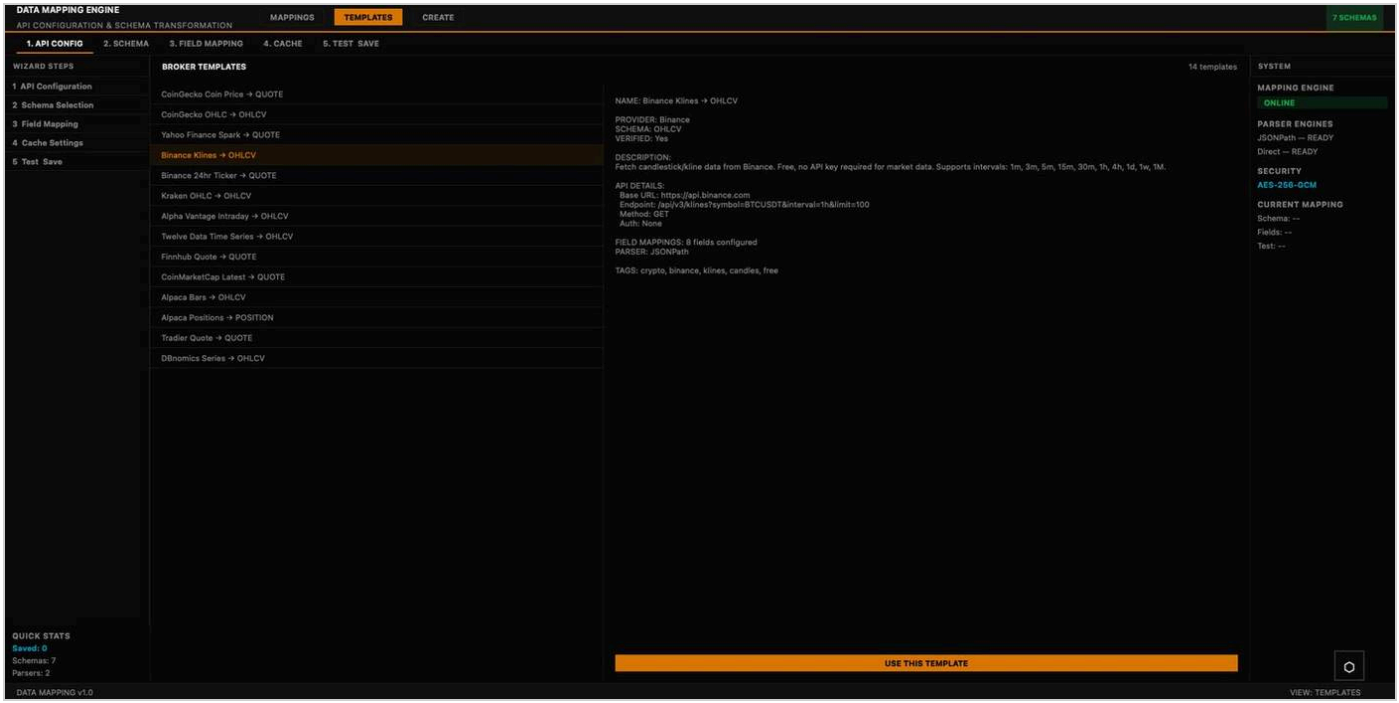
At a glance, in MAPPINGS you can…

- See every saved mapping and which schema it targets; run one to pull & normalise fresh data; delete one (with its credential); or start a new mapping.

3 View · TEMPLATES

FAST START

Fourteen ready-made API templates. Selecting one shows its full detail; **USE THIS TEMPLATE** pre-fills the entire wizard so you only need to add your API key (if any) and save.



TEMPLATES — the 14-template list (left), the selected template's detail (right: provider, schema, API details, field mappings, parser, tags), and USE THIS TEMPLATE.

Control	What it does
Template list	One row per template (name → schema). Click to preview.
Detail panel	NAME, PROVIDER, SCHEMA, VERIFIED, DESCRIPTION, API DETAILS (base URL, endpoint, method, auth), field-mapping count, PARSER, and TAGS.
USE THIS TEMPLATE	Applies the template into the wizard (name, URL, endpoint, method, auth type, headers, schema, parser, and all field mappings) and jumps to Create ▸ Step 1.

The 14 templates (all verified, all JSONPath):

Template	Schema	Auth	Provider / note
CoinGecko Coin Price	QUOTE	None	CoinGecko — free, no key
CoinGecko OHLC	OHLCV	None	CoinGecko — free
Yahoo Finance Spark	QUOTE	None	Yahoo Finance
Binance Klines	OHLCV	None	Binance — candles (used in this guide)
Binance 24hr Ticker	QUOTE	None	Binance
Kraken OHLC	OHLCV	None	Kraken
Alpha Vantage Intraday	OHLCV	API Key	Alpha Vantage (free tier)
Twelve Data Time Series	OHLCV	API Key	Twelve Data
Finnhub Quote	QUOTE	API Key	Finnhub
CoinMarketCap Latest	QUOTE	API Key	CoinMarketCap (header key)
Alpaca Bars	OHLCV	API Key	Alpaca (key + secret headers)

Template	Schema	Auth	Provider / note
Alpaca Positions	POSITION	API Key	Alpaca paper API
Tradier Quote	QUOTE	Bearer Token	Tradier sandbox
DBnomics Series	OHLCV	None	DBnomics — macro/economic

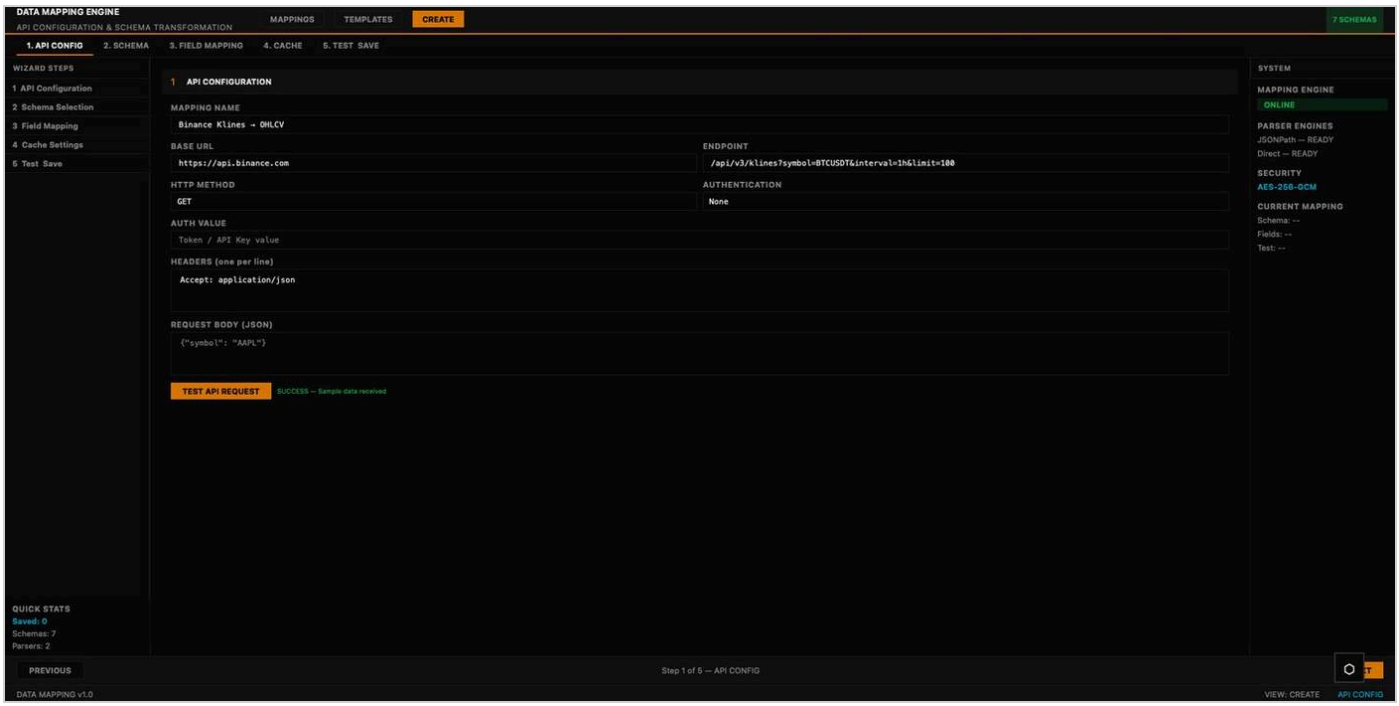
> **Note.** Templates that need a key carry a placeholder like `YOUR_KEY` in the endpoint/headers; **USE THIS TEMPLATE does not fill the Auth Value** — add your own key on Step 1 before saving.

Worked demonstration — Binance Klines → OHLCV

WALKTHROUGH

The next five pages walk the full **CREATE** wizard using the **Binance Klines → OHLCV** template (a free, no-key crypto candle feed). Each page is one step, with the exact screen and every control on it.

1 API Config — describe & test the request

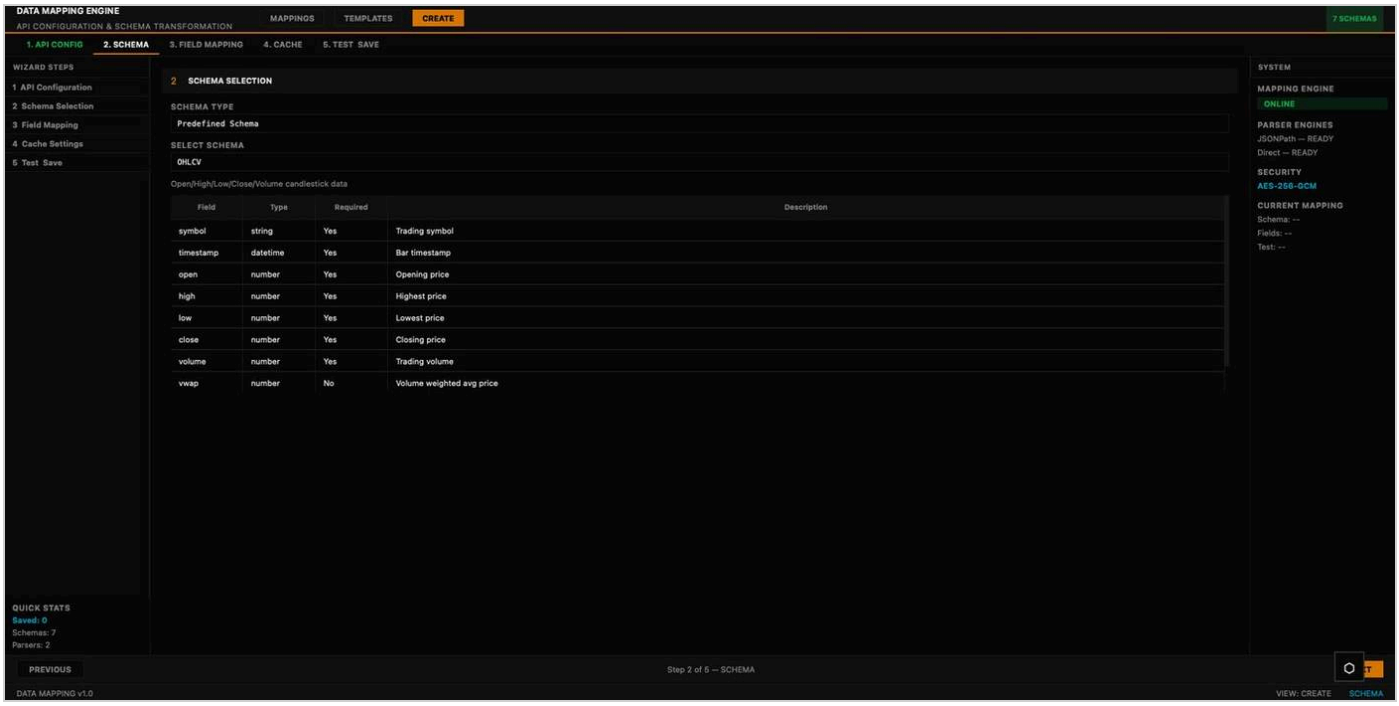


Step 1 after **TEST API REQUEST** — the green "SUCCESS — Sample data received" confirms a live sample was fetched (needed for Steps 3 & 5).

Field	Type	What it does
MAPPING NAME	text	Display name (required to save). Demo: "Binance Klines → OHLCV".
BASE URL	text	Scheme + host, e.g. `https://api.binance.com`.
ENDPOINT	text	Path + query; the request URL is simply **base + endpoint**. Demo: `/api/v3/klines?symbol=BTCUSDT&interval=1h&limit=100`.
HTTP METHOD	combo	GET · POST · PUT · DELETE · PATCH. POST/PUT/PATCH send the JSON body.
AUTHENTICATION	combo	None · API Key · Bearer Token · Basic Auth — decides how Auth Value is attached: Bearer → `Authorization: Bearer …`, API Key → `X-API-Key: …`, Basic → `Authorization: Basic base64(…)`.
AUTH VALUE	password	The token/key (masked). Never saved in the mapping record — it goes to encrypted secure storage (§10).
HEADERS (one per line)	multi-line	Extra request headers as `Name: Value`.
REQUEST BODY (JSON)	multi-line	JSON body (sent only for POST/PUT/PATCH).
TEST API REQUEST	button	Makes the live call and stores the response as the working sample. Status shows "Testing…", then green "SUCCESS — Sample data received" or red "FAILED — <error>".

> **Always test here first.** Steps 3 (Field Mapping) and 5 (Test) both work against the sample this button fetches — without it, the JSON explorer is empty and the mapping test can't run.

2 Schema — pick the target shape

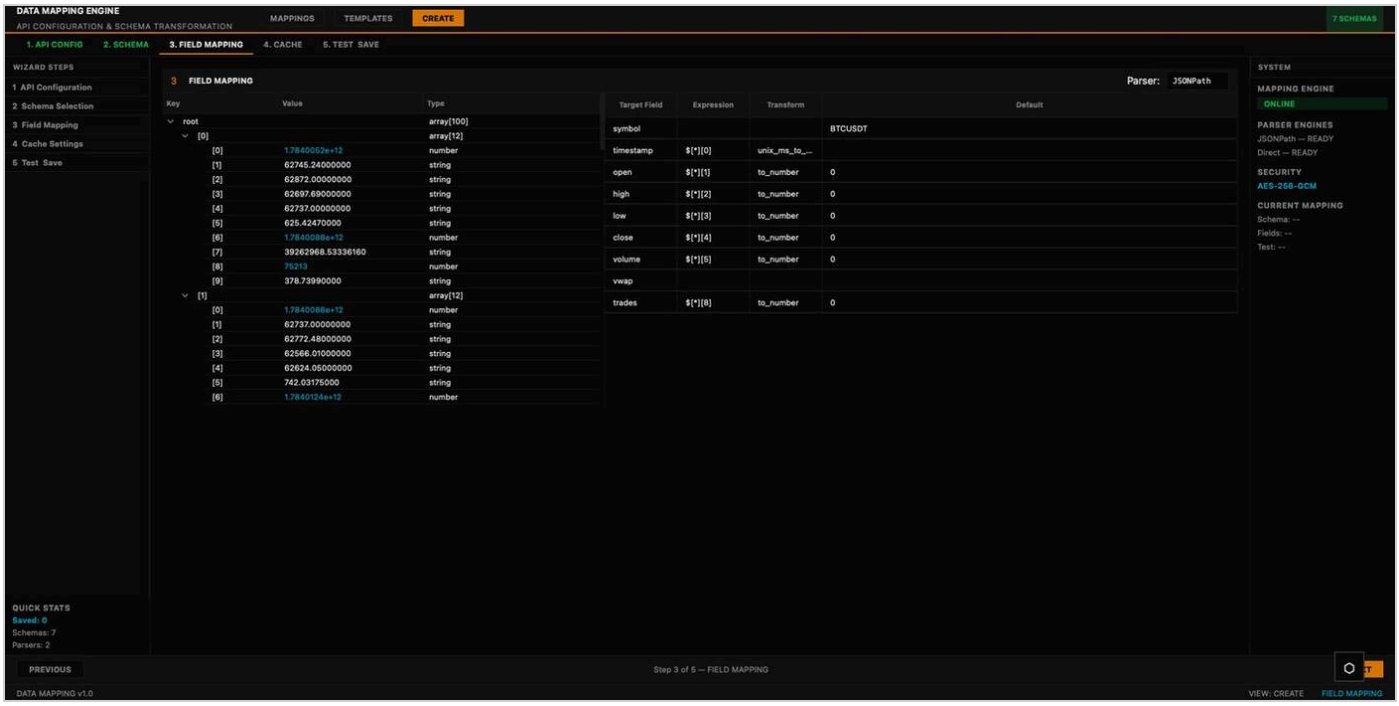


Step 2 — the target schema (OHLCV) and its field table: Field · Type · Required · Description.

Control	Type	What it does
SCHEMA TYPE	combo	Predefined Schema (the standard, catalogued schemas).
SELECT SCHEMA	combo	Choose the target: **OHLCV · QUOTE · TICK · ORDER · POSITION · PORTFOLIO · INSTRUMENT**. Updates the description, the field table, and the right-panel Schema/Fields readout.
Field table	table	Read-only reference of the schema's fields — name, type, whether it's required, and a description. This is the checklist your mapping must satisfy.

The seven schemas are catalogued in full in §9. The demo targets **OHLCV** (Open/High/Low/Close/Volume candles): required `symbol, timestamp, open, high, low, close, volume`; optional `vwap, trades`.

3 Field Mapping — connect response → schema



Step 3 — the JSON explorer (left, the live Binance response) beside the mapping table (right). Each schema field gets an Expression, an optional Transform, and a Default.

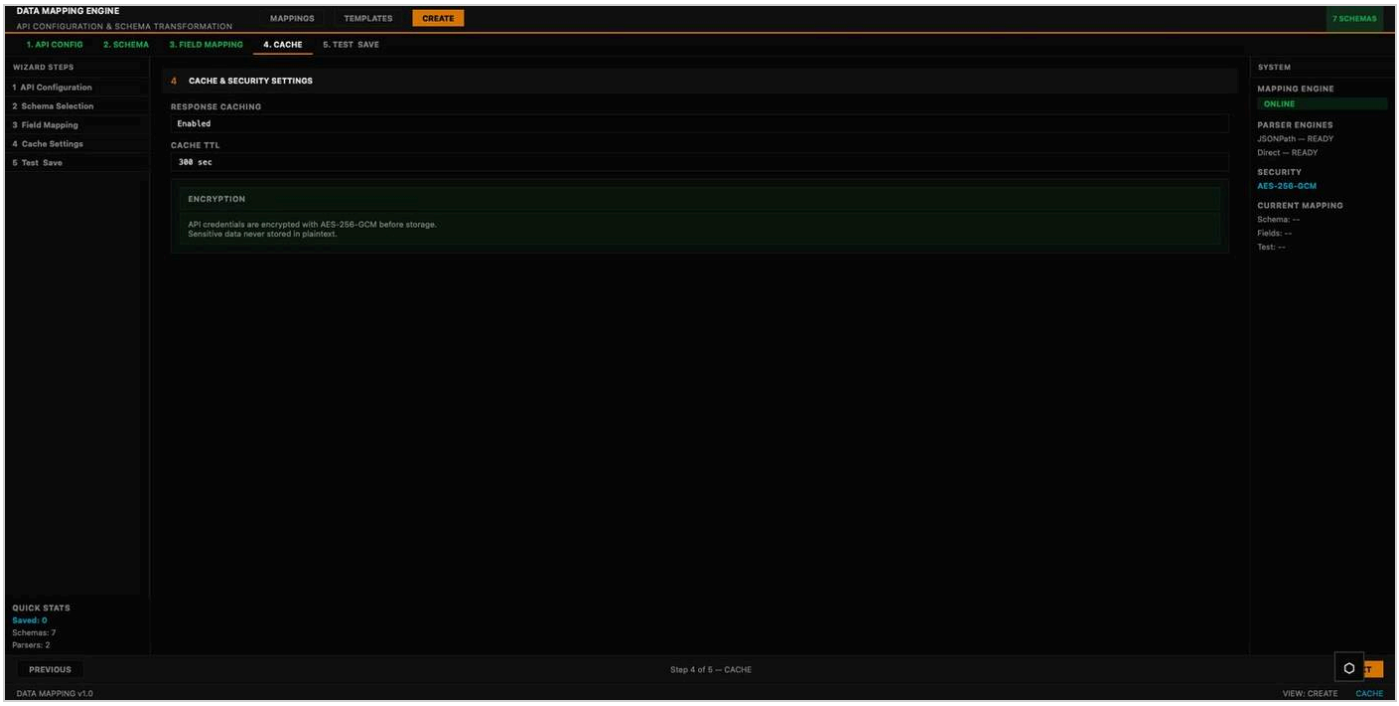
This is the heart of the tab. The **JSON explorer** (left) shows the sample response you fetched in Step 1 as a Key / Value / Type tree — you read the paths off it. The **mapping table** (right) has one row per schema field:

Column	What you put there
Target Field	The schema field (fixed — not editable).
Expression	The source path. JSONPath (e.g. `$[*][4]`, `$.bitcoin.usd`) or, with the Direct parser, a plain dotted path. A leaf with `[*]` makes the mapping **tabular** (one output row per array element).
Transform	Optional converter applied to the extracted value — e.g. `to_number`, `unix_ms_to_iso` (see §9).
Default	Fallback value when the path is missing/empty.

Above the table, the **Parser** dropdown (JSONPath or Direct) sets how expressions are read. In the demo, Binance returns rows of `[openTime, open, high, low, close, volume, …]`, so `close` maps to `$[*][4]` with `to_number`, `timestamp` to `$[*][0]` with `unix_ms_to_iso`, and `symbol` uses the default `BTCUSDT`.

> **Rows are exported only if their Expression is non-empty.** Leaving a field blank (e.g. `vwap`) simply omits it — you're not forced to fill optional fields.

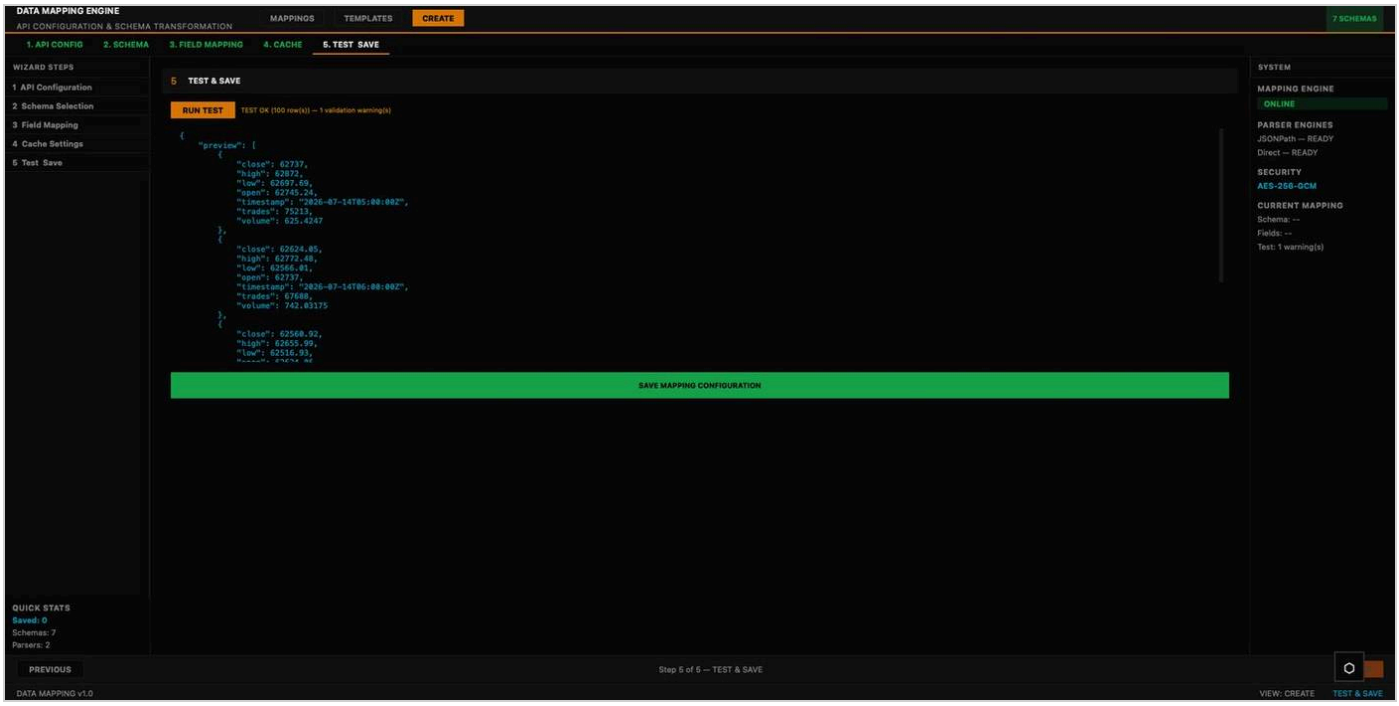
4 Cache & Security



Step 4 — response caching (on/off + TTL) and the encryption notice.

Control	Type	What it does
RESPONSE CACHING	combo	Enabled / Disabled. When enabled, a saved mapping's **Run** serves a cached response within the TTL instead of re-hitting the API.
CACHE TTL	spinbox	Cache lifetime in seconds (0–86,400; default 300 = 5 min).
ENCRYPTION (info)	—	A reminder that API credentials are encrypted with AES-256-GCM before storage and never kept in plaintext (§10).

5 Test & Save



Step 5 — RUN TEST applied the mapping to the live sample and previews the normalised OHLCV rows; SAVE MAPPING CONFIGURATION commits it.

Control	Type	What it does
RUN TEST	button	Applies the mapping to the Step-1 sample **locally** (no new network call) and shows the outcome: green "TEST PASSED — N row(s)", or amber "TEST OK (N rows) — M validation warning(s)" when required fields are missing, or red "TEST FAILED — No field mappings configured".
Result panel	read-only	Indented JSON: rows extracted, the schema, a preview of the first rows, and validation ("ok" or a list of warnings). In the demo: 100 rows, one "symbol" warning.
SAVE MAPPING CONFIGURATION	button	Persists the mapping. **Enabled only after a successful test**, and it also needs a non-empty name. On save the credential is written to encrypted storage, the mapping to the local database, and you're returned to the MAPPINGS list.

> **Validation warnings don't block saving.** A missing required field (like `symbol` here) is reported as a warning, not an error — you can still save. Fix it by giving that field an Expression or Default on Step 3.

> **At a glance, the CREATE wizard lets you…**
>
> - Build (or template-fill) a request, pick a target schema, map each field with paths/transforms/defaults, set caching, dry-run against a live sample, and save — end to end.

9 Reference

SCHEMAS · PARSERS · TRANSFORMS

The 7 standard schemas

Schema	Required fields	Optional fields
OHLCV candles	symbol, timestamp, open, high, low, close, volume	vwap, trades
QUOTE bid/ask quote	symbol, timestamp, price	bid, ask, bidSize, askSize, volume, change, changePercent, open, high, low, previousClose
TICK trade ticks	symbol, timestamp, price, quantity	side (BUY/SELL/UNKNOWN), tradeId
ORDER	orderId, symbol, side (BUY/SELL), type (MARKET/LIMIT/STOP/STOP_LIMIT), quantity, status (PENDING/OPEN/FILLED/CANCELLED), timestamp	filledQuantity, price, averagePrice, exchange
POSITION	symbol, quantity, averagePrice, currentPrice, marketValue, costBasis, pnl, pnlPercent	exchange
PORTFOLIO	totalValue, cash, invested, marketValue, pnl, pnlPercent, timestamp	dayPnl
INSTRUMENT	symbol, name, exchange, instrumentType (EQUITY/FUTURES/OPTIONS/CURRENCY)	isin, lotSize, tickSize

Parser engines & expression syntax

Two parser engines are available:

- **JSONPath** — the runtime engine. Grammar: `$` (root), `$.key`, `$.a.b.c` (nested), `$[0]` (index), `$[*][2]` / `$[*].key` (array wildcard). An expression containing `[*]` marks the mapping **tabular** — one output row per element of the located array.
- **Direct** — plain-path sugar: a dotted path (`a.b.0.c`, `*` for wildcard) is rewritten into JSONPath automatically.

Transforms

Optional per-field converters applied after extraction:

Transform	Effect
to_number	Parse a string/number to a numeric value.
to_string	Coerce to string.
unix_ms_to_iso	Epoch milliseconds → ISO-8601 timestamp.
unix_s_to_iso	Epoch seconds → ISO-8601 timestamp.
upper / lower	Upper- / lower-case a string.
abs_value	Absolute value of a number.

> **Transform caveat.** Only the names above are implemented; an unrecognised transform passes the value through unchanged. A few templates reference `unix_to_iso`, which is *not* one of the implemented names — for those, use `unix_s_to_iso` or `unix_ms_to_iso` so timestamps convert correctly.

10 Under the hood

FOR IT / SECURITY

Network boundaries

- **Test API** and **Run** make a **direct HTTP call to your configured third-party endpoint** — *not* through the Fincept edge. Because the URL is absolute, the terminal deliberately does **not** attach its own Fincept API key/session token; the provider sees only the auth you configured.
- **Test Mapping** (Step 5 RUN TEST) is **fully local** — it re-uses the sample already fetched, with no new network call.
- Requests use a **120-second** transfer timeout that resets on each byte received (so a slow-but-progressing download isn't killed; only a genuine stall is).
- Responses must be **JSON**. A non-JSON 200 (CSV/XML/HTML) fails parsing. Note that an HTTP 4xx/5xx whose *body* is valid JSON can read as "SUCCESS" in the test panel — always eyeball the sample.

Storage

- Saved mappings persist to a **local database** on this device (table `data_mappings`); each run's normalised output is also stored locally (`normalized_data`). Mappings are therefore **per-device** — they do not roam to another machine.
- Re-saving mints a **new record** each time (there is no in-form "edit an existing saved mapping" — use New, or delete the old one).

Security

- The API credential is **never written into the mapping record**. It is stored separately, encrypted with **AES-256-GCM**, keyed to the mapping; in this Exclusive edition the secret is held in the encrypted secrets service (not in the plaintext mappings database). Deleting a mapping also deletes its stored credential.
- Logs strip query strings and never record header values, so tokens don't leak into log files.

11 Quick reference & notes

HOW DO I...

I want to…	Do this
Connect a popular API fast	TEMPLATES ▸ pick one ▸ USE THIS TEMPLATE ▸ (add key) ▸ Step 5 Save.
Connect a custom API	CREATE ▸ Step 1: fill URL/method/auth ▸ TEST API REQUEST.
See the raw response shape	Step 3 — the JSON explorer shows the fetched sample as a tree.
Map a tabular (array) response	Use `[*]` in the expression, e.g. `$[*][4]` — one row per element.
Convert an epoch timestamp	Set the field's Transform to `unix_ms_to_iso` or `unix_s_to_iso`.
Check my mapping before saving	Step 5 ▸ RUN TEST ▸ read the preview + validation.
Pull fresh data from a saved mapping	MAPPINGS ▸ select ▸ ▶ RUN.
Store an API key safely	Put it in Step 1 ▸ AUTH VALUE; it's encrypted, never in the mapping record.

Notes for institutional users

- **Two parser engines** ship today — JSONPath and Direct. (The interface is built to host more; those two are what's active.)
- **Schema Type** currently offers a single option, "Predefined Schema" — the seven standard schemas. A user-defined "custom schema" is not yet selectable.
- **Wizard steps aren't gated** — you can page through freely; only Save enforces a successful test + a name. Validation issues are surfaced as warnings, never blockers.
- **Templates are global** (crypto, US/global equity, macro) — CoinGecko, Yahoo, Binance, Kraken, Alpha Vantage, Twelve Data, Finnhub, CoinMarketCap, Alpaca, Tradier, DBnomics — and carry placeholder keys you must replace.
- **Everything is local & direct.** Your mapping definitions and pulled data stay on the device; test/run calls go straight to the provider; credentials are encrypted at rest.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The News Wire

A complete, screen-by-screen walkthrough of the News tab — a Bloomberg-style live wire pulling ~60 sources into one dense, filterable feed with priority, sentiment, clustering and alerts.

`>_ 60 sources · sentiment · clustering · alerts`

Covers · **~60 sources** · **filtering the wire** · **counts & sentiment** **clusters, banner & ticker** · the article & intelligence panels · alerts & refresh

Demonstrated live on the Fincept Terminal News tab.

Contents

11 SECTIONS

GETTING ORIENTED

01 What the News tab is 3

02 Anatomy of the screen 4

WORKING THE WIRE

03 Filtering & searching the wire 5

04 Counts, sentiment & board controls 6

05 Reading the wire 7

06 Clusters, the banner & the ticker 8

07 The article detail panel 9

08 The Intelligence drawer 10

REFERENCE

09 Managing news sources 11

10 Live feed, refresh & states 12

11 Quick reference — “How do I…” 13

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Control labels are quoted exactly as they appear on screen; the headlines shown reflect a single live moment on the wire.

01 What the News tab is

The News tab is a **Bloomberg-style live news wire**. It pulls ~60 built-in RSS feeds — wire services, central banks, major financial media, crypto, defense and regional outlets — into one dense, colour-coded stream. Every headline is tagged with a **priority**, a **source tier**, a **sentiment** and, where relevant, a **threat level** and stock **tickers**.

Beyond the raw wire it clusters duplicate stories, watches your keywords and pings you when they hit, flags unusual volume spikes, and opens a per-article panel with summaries, AI analysis, named entities, related coverage and prediction-market odds.

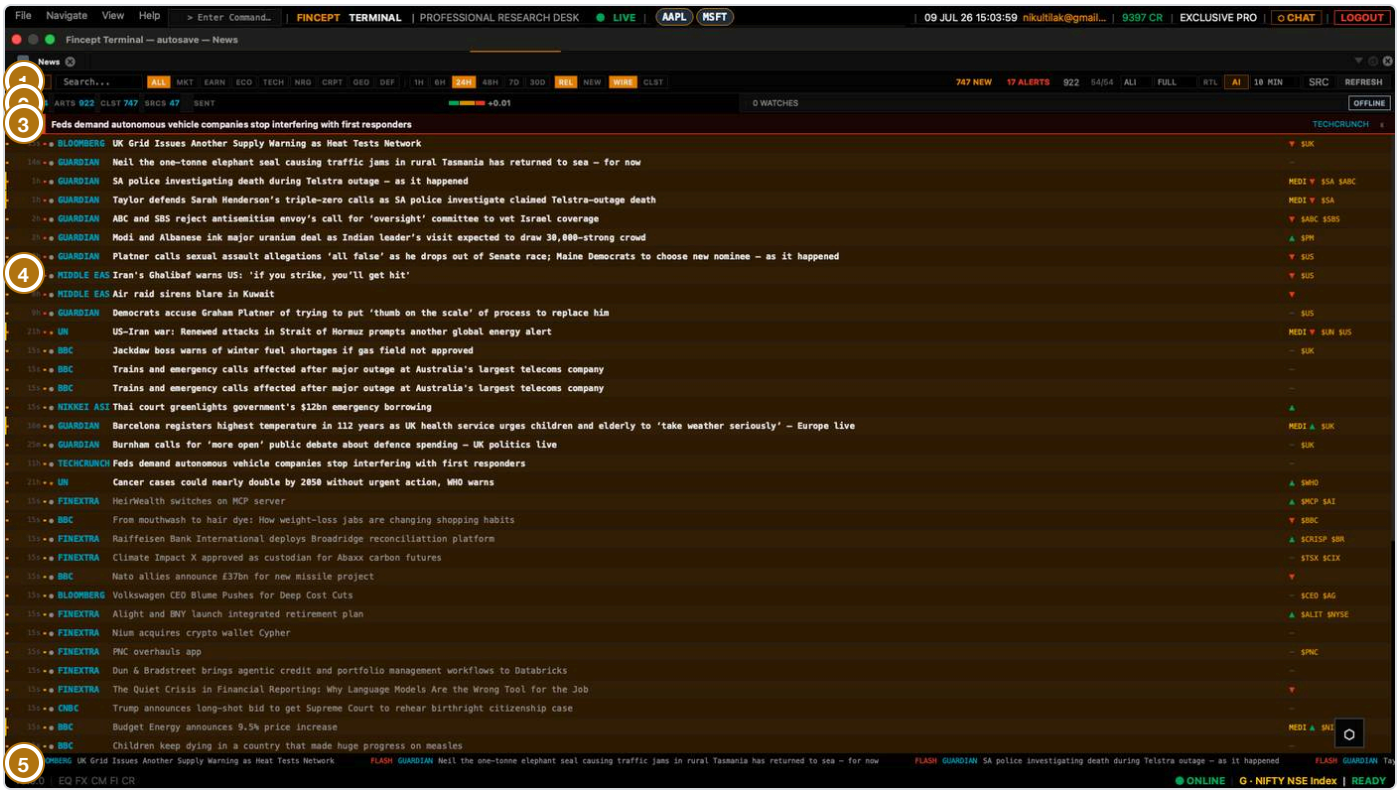
At a glance, you can…

- **Scan a live wire** of financial & world news, newest first, with priority and sentiment at a glance.
- **Filter by category** (Markets, Earnings, Economic, Tech, Energy, Crypto, Geopolitics, Defense), **time window**, **language** and **theme**.
- **Search** every headline, source and ticker — instantly.
- Switch between the one-line **WIRE** and grouped **CLUSTERS** views that merge the same story across sources.
- Open any article into a **detail panel** — summary, AI analysis, entities, related stories, and one-click open / copy / save / bookmark / translate.
- Open the **Intelligence drawer** for top stories, category counts, entities, volume spikes and prediction odds.
- Set **keyword monitors** that highlight matches and fire desktop alerts.
- **Add or disable news sources** and tune the refresh cadence.

> **Where the news comes from.** Articles are fetched directly from ~60 built-in **RSS/Atom feeds** (AP, Reuters-class wires, SEC, Federal Reserve, ECB, BBC, Bloomberg, WSJ, CNBC, Al Jazeera, CoinDesk and more), cached for 10 minutes. An optional live **WebSocket** feed pushes breaking items in real time. AI summaries/analysis and translation call the Fincept API on demand.

02 Anatomy of the screen

The News tab stacks a control bar over the feed, with two panels that slide in from the sides.



#	Region	What it is
1	**Command bar**	Search + all filters (category, time, sort, view, language, theme) on the left; live counts, sentiment controls, AI Brief, refresh cadence, sources and refresh on the right.
2	**Intel strip**	Feed / article / cluster / source counts, the aggregate **sentiment gauge**, your keyword-watch summary, volume-spike badges, and the **LIVE / OFFLINE** connection badge.
3	**Breaking banner**	A red strip that flashes the latest FLASH / BREAKING headline (auto-dismisses; click to open).
4	**The wire (feed)**	The article list itself — one line per story (WIRE) or grouped cards (CLUSTERS).
5	**Ticker strip**	A continuously scrolling band of the most urgent headlines at the very bottom.

Two more panels are hidden until you ask for them: the **Intelligence drawer** slides in from the left (the `INTEL` button), and the **Article-detail panel** slides in from the right when you click a story. Both are covered later.

03 Filtering & searching the wire

The left half of the command bar narrows the feed. Filters combine — pick a category *and* a time window *and* a search term.

Control	What it does
Search	Live-filters by headline, summary, source or ticker as you type. (In the 7D / 30D windows it also searches your local article history.)
Category pills	`ALL` · `MKT` Markets · `EARN` Earnings · `ECO` Economic · `TECH` · `NRG` Energy · `CRPT` Crypto · `GEO` Geopolitics · `DEF` Defense.
Time pills	`1H` `6H` `24H` `48H` `7D` `30D` — how far back to show. 7D / 30D merge the live feed with saved history.
Sort	`REL` = relevance (priority first, then newest) · `NEW` = strictly newest.
View	`WIRE` = one line per story · `CLST` = clustered story cards (see §5).
Language	Show only articles in a chosen language (`ALL EN FR DE ES AR ZH JA HI RU`).
Theme	A quick preset that groups categories — `FULL` , `FINANCE` , `CRYPTO` , `MACRO` .

Picking `TECH` filters the wire and recomputes every count (here 184 articles).

Searching `energy` narrows to matching stories; the top match is highlighted.

04 Counts, sentiment & board controls

The right half of the command bar holds live counts and actions.

Control	What it does
N NEW · N ALERTS	Count of unseen stories, and of active breaking-story clusters.
Article count	How many stories match the current filters (and a `done/total` progress readout while loading).
AI	**AI Brief** — summarizes the top headlines into a short digest (uses AI credits).
Cadence	Auto-refresh interval: `MANUAL · 1MIN · 5MIN · 10MIN · 30MIN` (default 10 min; MANUAL pauses it).
`SRC`	Opens the **RSS Feed Sources** manager (see §8).
`REFRESH`	Forces a full re-fetch of every enabled feed now.
`RTL`	Switches the whole interface to right-to-left (for Arabic/Hebrew reading).

The intel strip

Item	Meaning
FEEDS · ARTS · CLST · SRCS	Enabled feeds, articles shown, story clusters, and distinct sources in the current view.
Sentiment gauge	A bull / neutral / bear bar plus a signed score from `−1.00` to `+1.00` = (bullish − bearish) ÷ total.
Watches	`N WATCHES` (and `M HIT`) — your keyword monitors and how many matched.
Spike badges	Category volume anomalies as `CAT Zx` when a topic surges above its normal rate.
`● LIVE` / `OFFLINE`	Real-time WebSocket state. Click to connect or disconnect the live push feed.

05 Reading the wire

In WIRE view each story is a single line. A lot is packed into it, left to right.

One story row: unseen dot · age · priority/tier marker · source · headline · sentiment arrow · tickers.

Element	Meaning
Amber dot	The story is **unseen**; it fades once you scroll past or open it.
Age	How long ago it broke (e.g. `15s` , `2h`).
Priority marker	Colour-dot for **FLASH** / URGENT / BREAKING / routine. FLASH & URGENT headlines also render brighter and bold.
Source tier	`★` tier-1 wire · `●` tier-2 major outlet · `·` tier-3 specialty — then the source name.
Credibility flag	`⚠` state-media or `!` caution, when the source warrants it.
Sentiment	▲ bullish · ▼ bearish · `–` neutral.
Tickers	Up to two related symbols, e.g. `$AAPL $MSFT` .

The wire in WIRE view. Newer / higher-priority items sit up top; sentiment and tickers align on the right.

Scroll to the bottom to lazy-load more; scrolling marks stories as seen. Press `J` / `K` to step to the next / previous story, or click any row to open it.

06 Clusters, the banner & the ticker

CLUSTERS view

Switch the view pill to `CLST` and the wire regroups: stories reported by several outlets collapse into one card, so you see the event once instead of ten times.

Cluster cards show the lead headline, a **BREAKING** tag, how many **sources** carry it, the aggregate **BULL/BEAR** lean, a `+N src` velocity badge when it's spreading, and an **Also:** list of the other outlets.

The breaking banner

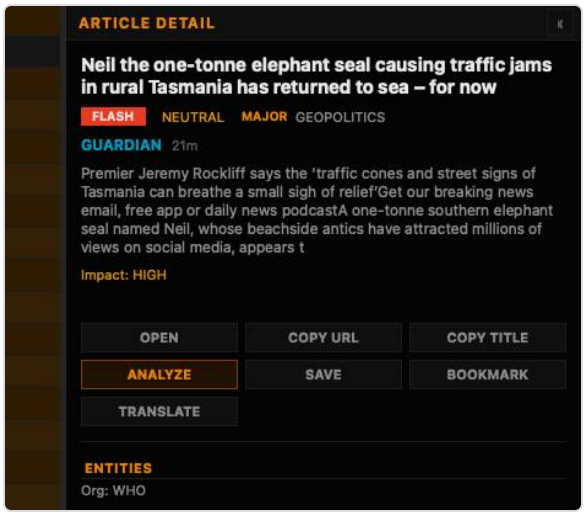
The red banner surfaces the newest FLASH / BREAKING story above the feed. It auto-dismisses after a minute; click it to open, or the `x` to dismiss.

The ticker strip

Along the very bottom, the most urgent headlines scroll continuously. Hover to pause; click a headline to open it.

07 The article detail panel

Click any story (in the wire, a cluster, the drawer or the ticker) and a panel slides in from the right.



The panel leads with the **headline** and a badge row — **priority**, **sentiment**, source **tier** and **category** — then the source, age, a **summary**, an **Impact / Threat** line and any tickers.

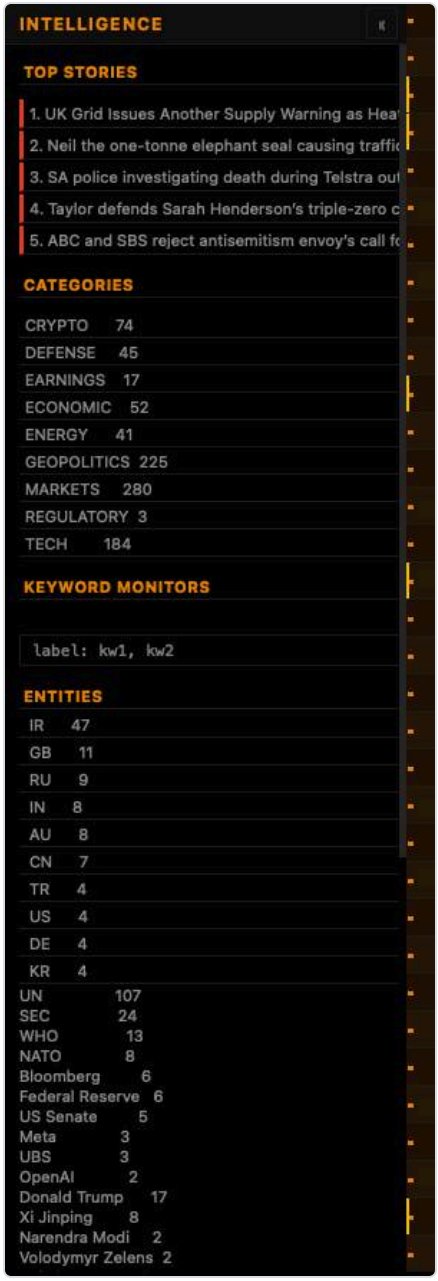
Below sit the actions, then (on demand) an AI analysis and related coverage.

OPEN	Open the full article in your browser.
COPY URL / TITLE	Copy the link or headline to the clipboard.
ANALYZE	Run AI analysis (below). Uses AI credits.
SAVE	Save the article as a text file to the File Manager.
BOOKMARK	Pin it to Bookmarks in the Intelligence drawer.
TRANSLATE	Translate the headline & summary into English.

ANALYZE — the AI intelligence pass. Running `ANALYZE` adds an **AI ANALYSIS** section: a written summary, sentiment / confidence / urgency / outlook pills, key points, four categories of **risk signal** (regulatory, geopolitical, operational, market), topics and named entities — plus **RELATED** stories from the same cluster and, for geolocated stories, nearby infrastructure. The result is cached per article so re-opening it doesn't spend credits again.

08 The Intelligence drawer

Click `INTEL` (far left of the command bar) to slide out a briefing panel on the left.

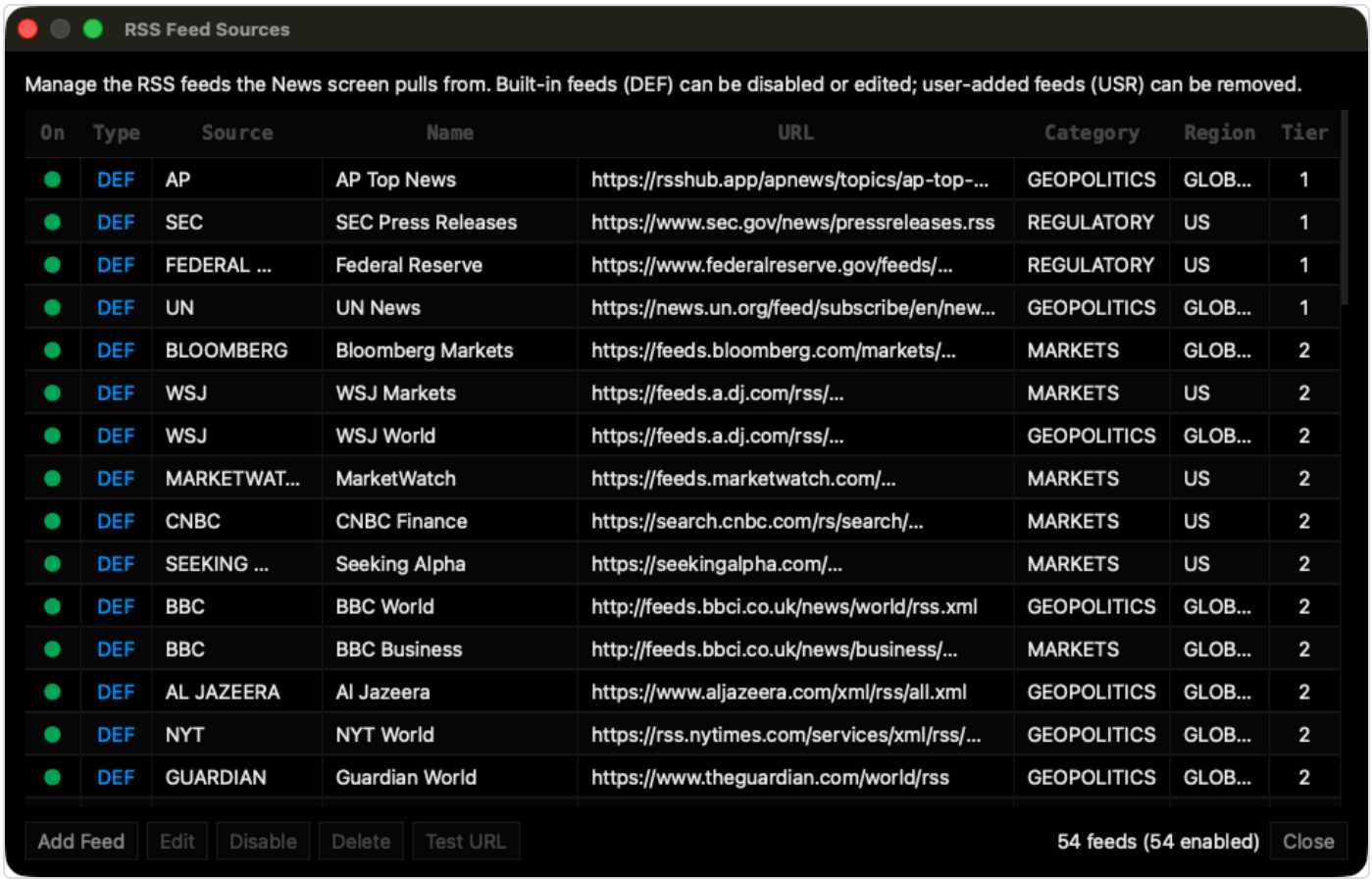


Section	What it shows
TOP STORIES	The five highest-priority stories right now — click to open.
CATEGORIES	Live article counts per category — click one to filter the wire to it.
KEYWORD MONITORS	Your watches. Add one by typing `label: kw1, kw2`; toggle on/off or delete each.
DEVIATIONS	Categories spiking well above their normal volume.
ENTITIES	Most-mentioned countries, organisations and people (auto-extracted).
LOCATIONS · SIGNALS · INSTABILITY · PREDICTIONS	Geolocated hot-spots, correlation signals, country-instability scores, and Polymarket odds — each appears once it has data.
BOOKMARKS	Your saved articles, up to ten — click to reopen.

> **Keyword monitors = your alerts.** A monitor highlights matching stories with a coloured stripe in the wire and fires a desktop notification when a new match arrives — so you can watch a company, theme or region without staring at the feed.

09 Managing news sources

Click `SRC` to open the RSS Feed Sources manager — the full list of feeds the wire pulls from.



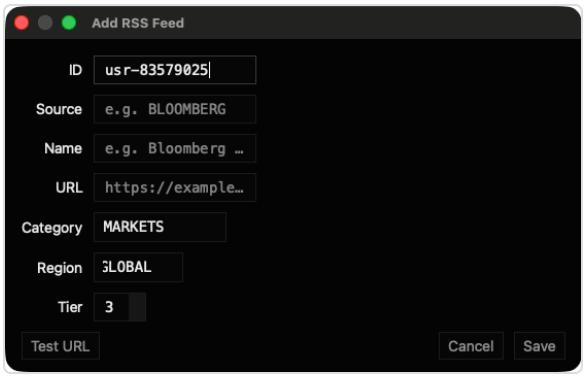
On	Type	Source	Name	URL	Category	Region	Tier
●	DEF	AP	AP Top News	https://rsshub.app/apnews/topics/ap-top-...	GEOPOLITICS	GLOB...	1
●	DEF	SEC	SEC Press Releases	https://www.sec.gov/news/pressreleases.rss	REGULATORY	US	1
●	DEF	FEDERAL ...	Federal Reserve	https://www.federalreserve.gov/feeds/...	REGULATORY	US	1
●	DEF	UN	UN News	https://news.un.org/feed/subscribe/en/new...	GEOPOLITICS	GLOB...	1
●	DEF	BLOOMBERG	Bloomberg Markets	https://feeds.bloomberg.com/markets/...	MARKETS	GLOB...	2
●	DEF	WSJ	WSJ Markets	https://feeds.a.dj.com/rss/...	MARKETS	US	2
●	DEF	WSJ	WSJ World	https://feeds.a.dj.com/rss/...	GEOPOLITICS	GLOB...	2
●	DEF	MARKETWAT...	MarketWatch	https://feeds.marketwatch.com/...	MARKETS	US	2
●	DEF	CNBC	CNBC Finance	https://search.cnbc.com/rs/search/...	MARKETS	US	2
●	DEF	SEEKING ...	Seeking Alpha	https://seekingalpha.com/...	MARKETS	US	2
●	DEF	BBC	BBC World	http://feeds.bbci.co.uk/news/world/rss.xml	GEOPOLITICS	GLOB...	2
●	DEF	BBC	BBC Business	http://feeds.bbci.co.uk/news/business/...	MARKETS	GLOB...	2
●	DEF	AL JAZEERA	Al Jazeera	https://www.aljazeera.com/xml/rss/all.xml	GEOPOLITICS	GLOB...	2
●	DEF	NYT	NYT World	https://rss.nytimes.com/services/xml/rss/...	GEOPOLITICS	GLOB...	2
●	DEF	GUARDIAN	Guardian World	https://www.theguardian.com/world/rss	GEOPOLITICS	GLOB...	2

Each feed row: on/off · type (**DEF** built-in / **USR** yours) · source · name · URL · category · region · tier.

Button	What it does
Add Feed	Add your own RSS/Atom source (form at right).
Edit	Change a feed's details.
Disable	Toggle a feed off (or back on) without deleting it.
Delete / Reset	Remove a user feed, or reset a built-in feed to its default.
Test URL	Check that a URL returns valid RSS/Atom before saving.

Adding or editing opens a small form: an auto **ID**, then **Source**, **Name**, **URL**, a **Category** and **Region**, and a **Tier** (1 = wire … 4 = blog). **Test URL** validates the address; **Save** adds it to the wire.

Built-in feeds are marked **DEF** and can be disabled or edited; your own are marked **USR** and can be removed. The footer counts how many feeds are enabled.



The Add / Edit feed form.

10 Live feed, refresh & states

Staying current

Behaviour	Detail
Auto-refresh	Re-fetches on the cadence you pick (default every 10 min). Runs only while the tab is open.
Manual refresh	`REFRESH` pulls every enabled feed immediately.
Live push	Click `OFFLINE` to go `● LIVE` — new breaking items are then pushed to the top in real time.
Watches & AI Chat	Keyword monitors and actions taken from AI Chat update the wire live, without a manual refresh.

What you'll see

State	Meaning
Shimmering rows + `…`	Loading — the feed is fetching; `REFRESH` shows `…` with a `done/total` progress count.
"No articles available"	Nothing matched / nothing loaded — "*Check your network connection and click Refresh to retry.*"
"Select an article"	The detail panel before you've opened a story.
"No summary available."	The story has no summary text (the AI summary line reads similarly until you ANALYZE).

Keyboard shortcuts

Key	Action
`J` / `K`	Next / previous story (selects and opens it in the detail panel).
`Enter`	Open the selected story in your browser.
`Esc`	Close the detail panel (or the Intelligence drawer).

11 Quick reference — "How do I..."

I want to…	Do this
Open the News tab	Click `NEWS` in the top tab bar (or type `news` in the command bar).
See only one market's news	Click a category pill (e.g. `CRPT` for crypto).
Look further back	Pick a wider time pill (`7D` / `30D`).
Find a company / topic	Type it in **Search** (headline, source or `$TICKER`).
Collapse duplicate stories	Switch the view to `CLST`.
Read / analyse a story	Click it → detail panel → `ANALYZE` / `TRANSLATE` / `OPEN`.
Save a story for later	Open it → `BOOKMARK` (find it under Bookmarks in the drawer).
Get alerted on a keyword	`INTEL` → Keyword Monitors → type `label: kw1, kw2` → +.
See the day's briefing	Click `AI` for an AI Brief, or open `INTEL` for Top Stories.
Add a news source	`SRC` → Add Feed → paste the RSS URL → Test URL → Save.
Turn a source off	`SRC` → select it → Disable.
Get news in real time	Click the `OFFLINE` badge to go `● LIVE`.
Refresh now	Click `REFRESH`.
Read right-to-left	Toggle `RTL`.

Fincept Terminal — News Guide · screenshots captured from the live application. Headlines shown are a single live moment and will differ in your session.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The RSS Feeds Board

A complete, screen-by-screen walkthrough of the RSS Feeds tab — a resizable-box dashboard that turns any RSS, Atom, XML, JSON or CSV feed into a live, self-updating panel.

`>_ 5 formats · 1-click field discovery · card & table views · synced across devices`

Covers · **the feed board & toolbar** · **the Add / Edit Feed dialog** · **field mapping & discovery** the feed engine · **AI feed tools** · cross-app reuse

Demonstrated live on the Fincept Terminal **RSS Feeds** tab.

Contents

9 SECTIONS

GETTING ORIENTED

01 What the RSS Feeds tab is 3

02 Anatomy — the board & toolbar 4

03 Feed boxes — cards & tables 6

BUILDING YOUR BOARD

04 Adding a feed — the Add Feed dialog 7

05 Field mapping & one-click Discover 8

06 Managing feeds — edit, refresh, remove 10

REFERENCE

07 Under the hood — the feed engine 11

08 AI feed tools & cross-app reuse 12

09 Quick reference & honest notes 13

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal, taken on the live RSS Feeds tab with six institutional feeds loaded (S&P Global Energy / Oil / Chemicals and NSE Announcements / Corporate Actions / Annual Reports). Control labels are quoted exactly as they appear on screen.

01 What the RSS Feeds tab is

OVERVIEW

The **RSS Feeds** tab is a personal news & data wall. You point it at any feed URL — a news RSS feed, an exchange XML endpoint, a JSON API, even a CSV export — and it renders that feed as a live box that refreshes itself on a timer. Add as many as you like and arrange them into a board of resizable tiles.

Under the hood it is powered by one shared feed engine that also drives the Equity Trading feed panel and the droppable dashboard "RSS Feed" widget, so a feed you add here is available everywhere. Feed definitions and your board layout are saved to your Fincept account — not a local file — so the same board follows you to any machine you sign in on.

AT A GLANCE, IN RSS FEEDS YOU CAN…

- Add any **RSS, Atom, XML, JSON or CSV** feed and get a live, auto-refreshing box in one step.
- Let the app **auto-detect** a feed's fields with one click, or map them by hand.
- Show each feed as scannable **cards** or a full-width **table**.
- **Drag, resize and reflow** boxes; the layout auto-saves.
- **Test** a feed before saving — see the item count and sample headlines.
- Refresh **one** feed or **all** of them at once.
- Open any item in your browser with a click.
- Read every feed from **AI chat** and reuse them across the app.

How to get here

Open the **Navigate** menu (or the command bar, top-left) and type `rss`, then pick `RSS Feeds`. The tab opens with one box already created for every feed you have configured; if you have none, you get an empty grid and a `+ Add Feed` button to start.

02 Anatomy — the board & toolbar

THE BOARD

The tab has just two parts: a thin **toolbar** across the top and the **board** below it. The board is a 12-column grid; each feed lives in one box that you can move and resize freely.

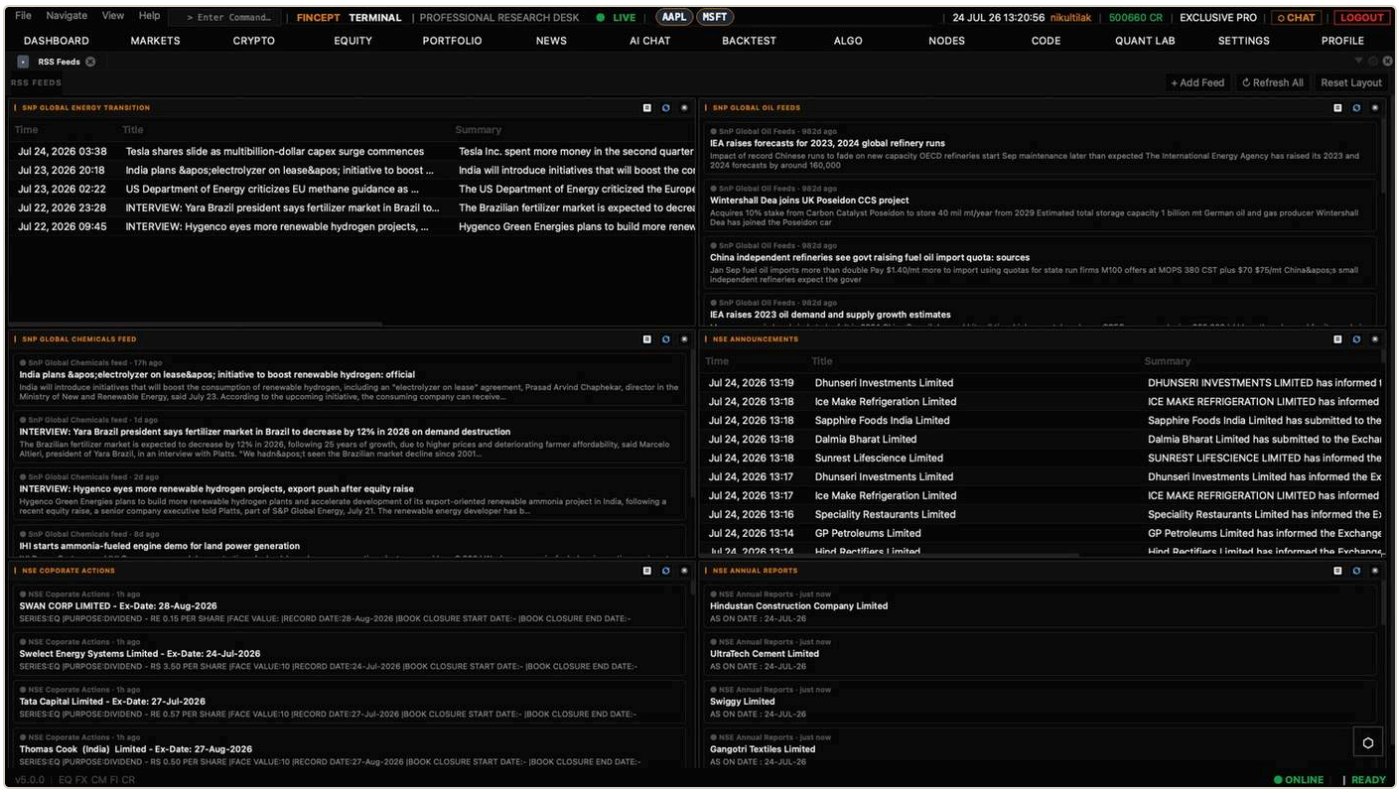


Fig 1. The RSS Feeds board with six live feeds — a mix of card boxes (S&P Global Oil, Chemicals; NSE Corporate Actions, Annual Reports) and table boxes (S&P Global Energy Transition; NSE Announcements).

The toolbar

Fig 2. The toolbar sits at the top-right of the tab.

Control	What it does
+ Add Feed	Opens the **Add Feed** dialog to configure a new feed. On save, a new box appears on the board (see §4).
↻ Refresh All	Re-fetches **every** configured feed immediately — including feeds not currently on the board.
Reset Layout	Rebuilds the board to a clean one-box-per-feed arrangement and discards your custom placement. **No confirmation** — see honest notes (§9).

A feed box

Every box has a title bar and a body. The title bar shows a colored accent stripe, the feed's name in UPPERCASE, and three icon buttons on the right:

Icon	Action
▤ **Configure**	Opens the **Edit Feed** dialog for this feed (URL, format, refresh, mapping, Delete).
↻ **Refresh**	Re-fetches **this feed only**, right now.
✕ **Close**	Removes **the box** from the board. The feed itself is **kept** — only its tile is taken off the wall.

> **IMPORTANT — ✕ vs Delete**
>
> The box ✕ only removes the tile from your board; the feed stays configured and still shows up elsewhere (AI tools, Equity panel, dashboard widget). To remove a feed *entirely*, open its ▤ **Configure** dialog and press **Delete** (§6).

Arranging boxes

- **Move:** drag a box by its title bar. Boxes swap and push to make room, then reflow with a short animation.
- **Resize:** drag the dotted grip at the bottom-right corner, or the right / bottom edges. Boxes snap to the grid; the minimum size is 3×3 cells.
- **Auto-save:** the board (which feed sits where, at what size) saves automatically about a second after you stop, and restores on your next visit — on any device.

03 Feed boxes — cards & tables

FEED BOXES

Each feed renders in one of two styles, chosen per feed by the **Display** setting in its dialog (Cards or Table). It is a property of the feed, not a button on the box — so to switch a feed between the two, open its Configure dialog and change **Display**.

Card view

Cards are best for reading — headlines you scan top-to-bottom. Each card shows a meta line (**● source · relative time** such as "982d ago"), the **headline** in bold, and a **summary** snippet when the feed provides one.

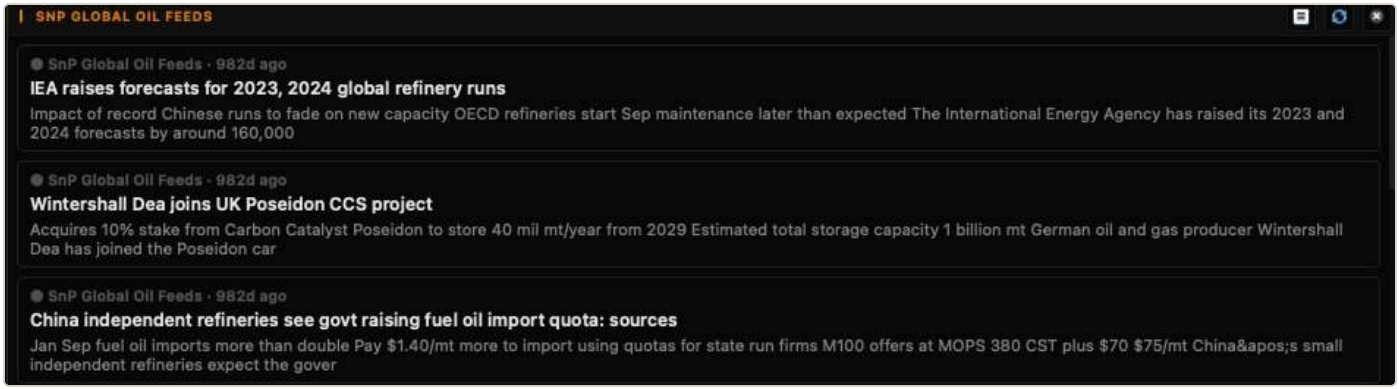


Fig 3. Card view (S&P Global Oil Feeds). Click anywhere on a card to open the article in your default browser.

Table view

Table view is best for dense, structured feeds — exchange announcements, corporate actions — where you want many rows at a glance. The columns are built **automatically from whatever each feed actually provides**: **Time** (the full timestamp), **Title**, then **Summary**, **Source** and **Link** when present, followed by any custom columns you mapped.

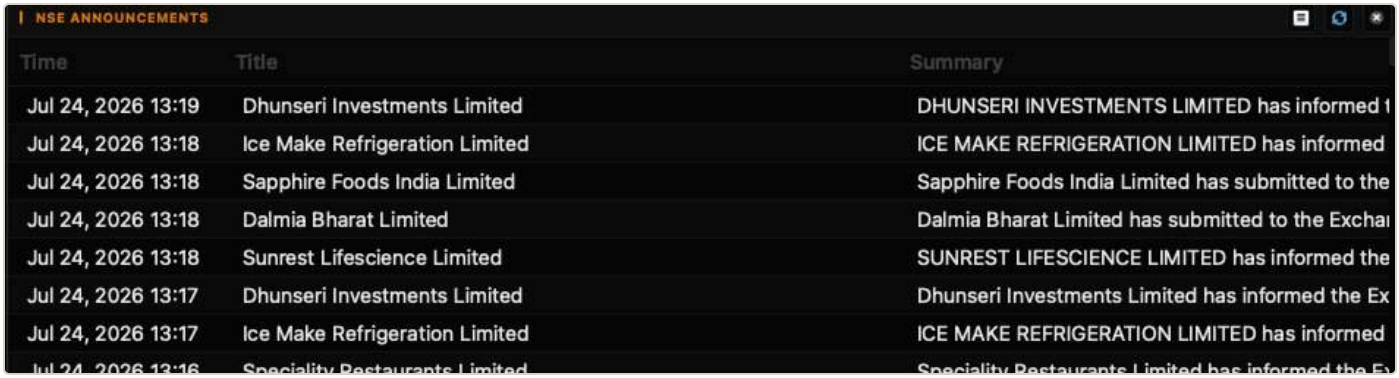


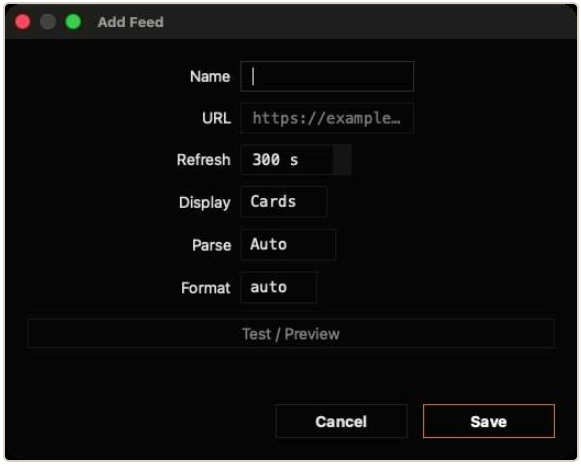
Fig 4. Table view (NSE Announcements) with the full timestamp and Summary column. Wide columns are capped and scroll horizontally, and every cell has a full-text tooltip so nothing is lost to truncation. Double-click a row to open its link.

> **READING LONG VALUES**
>
> Long titles or summaries are trimmed to keep columns tidy, but the complete text is always available: hover any cell for a tooltip, drag a column border to widen it, or scroll the table sideways.

04 Adding a feed — the Add Feed dialog

ADD FEED

Press `+ Add Feed` to open the dialog. For a standard news feed you only need the top section; the URL is the one field that really matters — everything else has a sensible default.



Field	Meaning
Name	Display title for the box. If left blank, the URL is used.
URL	The feed address to fetch.
Refresh	Poll interval in seconds. Default **300** (5 min); minimum 15.
Display	**Cards** or **Table** (§3).
Parse	**Auto** figures out the fields for you; **Manual** reveals the mapping tools (§5).
Format	Leave on **auto** to let the app detect RSS / Atom / XML / JSON / CSV, or pin it.

Fig 5. The Add Feed dialog in its simplest form (Parse = Auto). For most RSS feeds you paste a URL and press Save.

The two paths

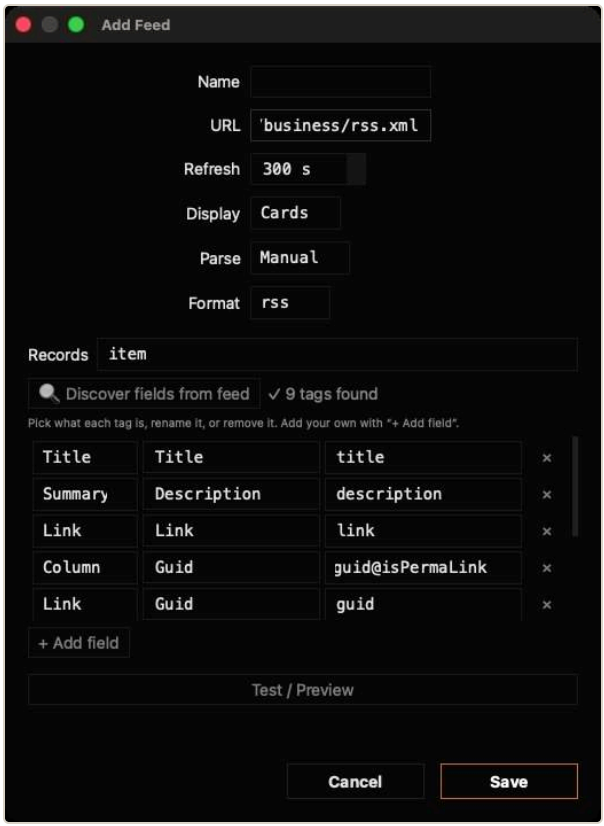
Path	When to use it	What you do
Auto	Standard RSS / Atom / JSON feeds with the usual fields.	Paste the URL, optionally press **Test / Preview**, then **Save**. The app extracts title, summary, link and date for you.
Manual	Unusual XML/JSON, or when you want specific columns.	Set **Parse = Manual**, press **Discover fields**, then adjust the mapping (§5).

> **TEST BEFORE YOU SAVE**
>
> The **Test / Preview** button fetches the feed once and reports how many items it found plus the first few headlines. If the test succeeds, those items are re-used when you Save, so the new box paints instantly instead of fetching again.

05 Field mapping & one-click Discover

FIELD MAPPING

Setting **Parse = Manual** reveals the mapping tools. This is where you tell the app which part of a feed is the title, which is the link, and which extra bits should become table columns. You rarely start from scratch — **Discover** does the heavy lifting.



🔍 Discover fields from feed

Press it and the app fetches the feed, inspects it, and fills everything in for you: it sets the **Records** selector (the repeating element, e.g. `item`), auto-detects the **Format**, and lists every tag it found as a mapping row with a best-guess role. The status reads ✓ **N tags found**.

You then simply rename, re-role or delete rows as needed, or press **+ Add field** to add your own.

Fig 6. After Discover on a BBC Business RSS feed: Format → `rss`, Records → `item`, and nine tags auto-mapped (title→title, description→summary, link→link, …).

What each mapping row means

A row is **[role] · [display name] · [source tag] · ✕**. The **role** tells the app what a tag is for:

Role	Shows in	What it is
Title	Cards + Table	The headline — bold in cards, the Title column in a table.
Summary	Cards + Table	The description / snippet under the headline.
Link	Opened on click	The URL opened when you click a card or double-click a row.
Time	Cards + Table	The timestamp — relative in cards, the full Time column in a table.
Column	Table only	A plain extra column, shown only in table view.

Test / Preview

Before saving, press **Test / Preview** to fetch the feed once and confirm the mapping works. It reports the item count and the first few headlines:

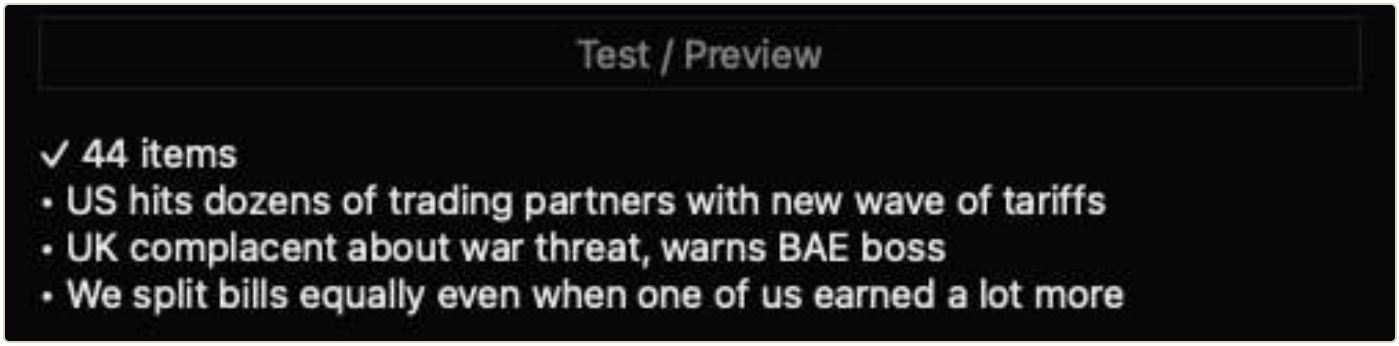


Fig 7. A successful test — ✓ **44 items** and three sample headlines. If the test passes, these items are re-used when you Save so the box paints instantly.

> **IF A TEST FAILS**
>
> A red ✗ with a message means the feed couldn't be fetched or parsed. Common fixes: double-check the URL; pin the **Format** to the specific type instead of `auto`; or switch to **Manual** and press **Discover** to see what the feed actually contains. You can still Save a feed that failed its test, but its box will show a warning until the feed responds correctly.

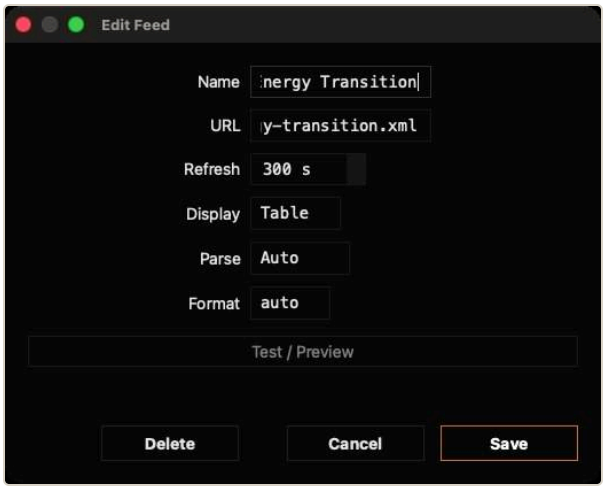
Working with mapping rows

Beyond what Discover fills in, you can shape the mapping by hand: **rename** a column by editing its display name, **re-role** a tag using the left dropdown (e.g. mark an extra tag as **Column** so it shows in the table), **remove** a row with its ✕, or press + **Add field** to map a tag Discover didn't surface. The **Records** selector at the top controls which repeating element counts as one item — change it if Discover picked the wrong one.

06 Managing feeds — edit, refresh, remove

MANAGE

Everything about a feed is edited from the same dialog you used to add it — reached from a box's ▤ **Configure** icon. In this mode the title reads **Edit Feed**, the fields are pre-filled with the feed's current settings, and a **Delete** button appears.



Editing

Change any field — for example flip **Display** from Table to Cards, slow the **Refresh**, or repoint the **URL** — and press **Save**. The box updates in place.

Deleting

Delete removes the feed **everywhere** — the box here, plus any dashboard widget or Equity-panel entry bound to it. It asks for confirmation first, and cannot be undone.

Fig 8. Edit Feed for "S&P Global Energy Transition" — note the pre-filled Name/URL, Display = Table, and the **Delete** button that only appears when editing.

Refreshing

Action	Scope
Box ↻ **Refresh** icon	Just that one feed.
Toolbar ↻ **Refresh All**	Every configured feed (even ones not on the board).
Automatic	Each feed re-fetches on its own timer (default every 5 minutes).

> **REMOVING vs KEEPING**
>
> **Take a box off the board** without losing the feed → box ✕. **Remove the feed entirely** → Configure → **Delete**. **Start the whole board over** → toolbar **Reset Layout** (rebuilds one box per feed; discards your arrangement, no confirmation).

07 Under the hood — the feed engine

ENGINE

You don't need any of this to use the tab, but it explains why some feeds work and others don't, and where your data lives.

Supported formats

Format	What it is	How it's read
RSS / Atom	Standard news feeds	XML `<item>` / `<entry>` → title, summary, link, date.
XML	Any XML endpoint	Generic element parsing; use Manual mapping to pick tags.
JSON	API / JSON feeds	Finds the record array (items/data/results/…); maps title/summary/link/date.
CSV / TSV	Tabular exports	Header row becomes columns; each column is a field.
auto	Default	Sniffs the first bytes and picks the parser above.

Refresh & timing

- Each feed has its own timer; the default cadence is **300 s** and the floor is **15 s** (anything lower is raised to 15).
- A newly added or just-enabled feed is fetched **immediately** so its box isn't blank while it waits for the first tick.
- The latest **200 items** per feed are held in memory; older ones roll off.

Getting past bot-blockers

Many professional feeds (S&P Global and other Akamai / Cloudflare-fronted sites) reject anything that doesn't look like a real browser. The engine sends a complete modern-Chrome request fingerprint — including the `Sec-Fetch-*` navigation headers — which is what turns a 403 rejection into real content. If a site instead serves an HTML challenge/captcha page, it shows as **"Access denied / not a feed"** and cannot be fixed with headers.

Where your feeds live

Feed **definitions** are stored on your Fincept account under a settings blob (`config.feeds`), and your **board layout** under a second blob (`config.feeds_board`). There is **no local database** — both sync across devices. Fetched **items** are never persisted; they are re-fetched each session.

> **SHARED ACROSS DEVICES**
>
> Because everything syncs, edits made on one machine appear on another after settings hydrate. If you edit the same feed on two devices at once, the last save wins.

08 AI feed tools & cross-app reuse

AI & REUSE

Feeds aren't trapped in this tab. The same engine exposes them to the AI assistant and to other parts of the terminal.

Reading feeds from AI chat

Three tools (category `feeds`) let the assistant pull your live feeds into a conversation — e.g. "summarise today's NSE announcements" or "any oil-market headlines about refineries?"

Tool	What it does
`feeds_list`	Lists every configured feed — name, URL, status and how many items are cached.
`feeds_read`	Reads items from one feed, chosen by id or name (partial match). Returns up to 100.
`feeds_search`	Keyword search across the cached items of **all** feeds at once.

The same feeds, elsewhere

Surface	How feeds appear
Equity Trading — Feeds panel	A collapsible column (toggled by the FEEDS button) showing up to three of your feeds beside the chart.
Dashboard — "RSS Feed" widget	A droppable tile (Research category) bound to any one of your feeds.
Data hub topic	Every feed is republished on a `feed:<id>` topic other components can subscribe to.

> **ONE ENGINE, ONE SET OF FEEDS**
>
> All of these read the same shared engine. A feed you add or edit here updates every surface at once; deleting it removes it from all of them.

09 Quick reference & honest notes

REFERENCE

How do I…

Task	Do this
Add a news feed	**+ Add Feed** → paste URL → **Test / Preview** → **Save**.
Add an unusual XML/JSON feed	Add Feed → **Parse = Manual** → **Discover fields** → tidy the rows → Save.
Switch a feed to a table	Box ▤ → set **Display = Table** → Save.
Add a custom column	Configure → Manual → map a field with role **Column**.
See the full text in a table	Hover the cell (tooltip), widen the column, or scroll sideways.
Refresh now	Box ↻ for one feed; toolbar ↻ **Refresh All** for everything.
Rearrange the board	Drag a box by its title bar; resize from the corner grip.
Take a box off the board	Box ✕ (keeps the feed).
Delete a feed for good	Configure → **Delete** (removes it everywhere).
Open an article	Click a card, or double-click a table row.

Honest notes & gotchas

- **Cards vs Table is per-feed, not a box toggle.** Change it in the Configure dialog's **Display** setting.
- **✕ ≠ Delete.** The box ✕ only removes the tile; the feed lives on. Use Configure → Delete to remove the feed itself.
- **Reset Layout is instant and unconfirmed.** It rebuilds one box per feed and discards your arrangement.
- **Refresh All is broad.** It re-fetches every configured feed, including ones not on the board.
- **15-second floor.** Refresh intervals below 15 s are raised to 15 s.
- **Some sites can't be fed.** HTML challenge/captcha pages show "Access denied / not a feed" and can't be fixed with settings.
- **No offline history.** Items aren't saved between sessions; each feed re-fetches on open. Only your feed *definitions* and layout persist.
- **Last save wins.** Editing the same feed on two devices simultaneously keeps whichever saved last.

> **FINCEPT RESEARCH DESK**
>
> Prepared for institutional clients · Professional Edition v5.0 · Captured live on the RSS Feeds tab. Control labels quoted verbatim from the running application.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Global Geopolitics

A complete, screen-by-screen walkthrough of the Geopolitics tab — world news turned into mapped, categorised events, with humanitarian data beside it.

`>_ 4 workspaces · 15 categories · 13 agent tools · 500 events a pull`

Covers · **the conflict map** · **the events table** · **HDX data** · **relations & trade**

What each reload really costs · which panels are live · which are not

Demonstrated live on the Fincept Terminal Geopolitics tab, v5.0.1.

Contents

18 SECTIONS

GETTING ORIENTED

01 What the Geopolitics tab is 3
02 Anatomy of the screen 4
03 Filters and the legend 5
04 The four workspaces 6

THE MONITOR

05 The conflict map 7
06 The events table 8
07 Overview, categories and hotspots 9
08 Mapped against loaded 10

HDX DATA

09 What HDX is, and its four views 11
10 Datasets — the view that works 12
11 Explorer — searching HDX yourself 13

RELATIONS & TRADE

12 Relations — a fixed reference chart 14
13 Trade — four analyses, none of which run 15

REFERENCE

14 Credits — what costs what 16
15 Driving the tab from the AI assistant 17
16 Where the data comes from 18
17 Status and message reference 19
18 Quick reference and honest notes 20

About this guide. Every screenshot is a live capture of the Geopolitics tab in the shipping v5.0.1 build, and every behavioural claim was checked against the product's source. Two of this tab's four workspaces do not do what they appear to do; both are named here, with the evidence, rather than described as though they work.

01 What the Geopolitics tab is

ORIENTATION · 01

Geopolitics turns a stream of world news into **mapped, categorised events**. A pull brings back 500 recent items, each tagged with a category — terrorism, diplomacy, sanctions, elections and eleven more — and most of them carrying coordinates, so they land as pins on a world map you can filter by country, city and category.

Three further workspaces sit alongside it: a search across the UN's humanitarian dataset catalogue, a fixed reference chart of active conflicts, and a trade-policy calculator.

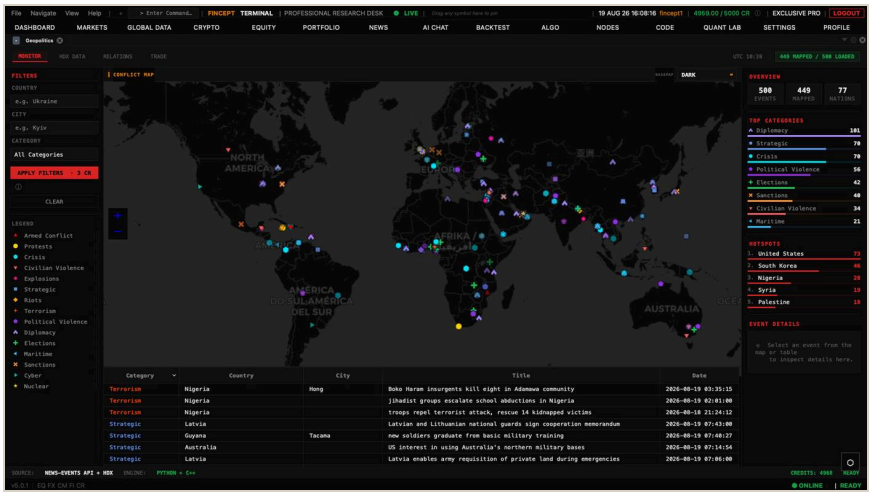


Fig 1 — The MONITOR workspace with 500 events loaded and 449 of them mapped. Left: filters and the category legend. Centre: the conflict map above the events table. Right: counters, category ranking, hotspots and event details.

Where to find it

Route	How
Navigate menu	**Navigate ‣ Research & Intelligence ‣ Geopolitics**
Command bar	`geo` — then click the **GEO › Geopolitics** row
Main tab bar	**Not present.** This is not one of the top-level tabs

At a glance, in Geopolitics you can…

- Pull 500 recent world events at once
- See most of them plotted on a world map
- Filter by country, city and category
- Rank the categories currently driving news
- See which countries are hotspots right now
- Read any event's full headline and location
- Search the UN's humanitarian data catalogue
- Switch between six basemaps

> **This tab spends credits on a timer.** A reload costs about 15 credits, one fires when you open the tab, and a five-minute auto-refresh repeats it for as long as the tab is visible — roughly 180 credits an hour. Switching to another tab stops it. §14 has the full model, including why the button's own price badge understates it.

02 Anatomy of the screen

Thirteen regions in three columns, with a workspace strip above them and a status bar below.

Fig 2 — The thirteen regions, numbered.

#	Region	What it is
1	Workspace tabs	MONITOR · HDX DATA · RELATIONS · TRADE (§04)
2	Coverage badge	`449 MAPPED / 500 LOADED` — see §08
3	Filters	Country, city and category (§03)
4	APPLY FILTERS	Reloads with the filters applied. Priced on the button (§14)
5	Legend	Every category the feed knows, with its colour (§03)
6	Basemap picker	Six tile layers; DARK by default here
7	The conflict map	One pin per geolocated event, coloured by category (§05)
8	Events table	One row per event, sortable (§06)
9	Overview counters	EVENTS · MAPPED · NATIONS (§07)
10	Top categories	The loaded set ranked by category, with bars (§07)
11	Hotspots	The five countries with the most events (§07)
12	Event details	The selected event in full (§07)
13	Status bar	Source, engine, credits and current state (§17)

> The three columns are fixed — there is no splitter to drag and no way to widen the map at the expense of the rails. The UTC clock beside the workspace tabs ticks once a second and is the quickest way to tell that the screen is alive rather than frozen.

03 Filters and the legend

Three filters, an apply button and a clear button. They are the only inputs on the MONITOR workspace, and they all feed the same reload.

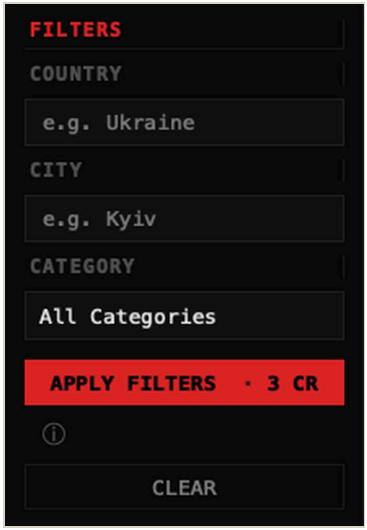


Fig 3 — The three filters.

Fig 4 — The category legend.

Control	What it does
COUNTRY	Free text, e.g. `Ukraine`. Restricts the pull to one country
CITY	Free text, e.g. `Kyiv`. Restricts the pull to one city
CATEGORY	A dropdown built from the feed's own category list, plus *All Categories*
APPLY FILTERS	Reloads using all three. **This is the action that spends** (§14)
CLEAR	Empties the three inputs. Free — it does not reload on its own

The legend

Fifteen categories, each with the colour its pins take on the map and in the events table: **Armed Conflict**, **Protests**, **Crisis**, **Civilian Violence**, **Explosions**, **Strategic**, **Riots**, **Terrorism**, **Political Violence**, **Diplomacy**, **Elections**, **Maritime**, **Sanctions**, **Cyber** and **Nuclear**.

> **The legend is built from the API, not hardcoded.** It is rebuilt whenever the feed reports its category list, so a category the feed stops using disappears from the legend rather than lingering as an empty row. The colours are the same ones the map and the table's Category column use, which is what makes a glance at the map legible.

> **The filters do not filter what is already loaded.** They are query parameters for the next pull, so changing one and pressing APPLY FILTERS is a fresh network round-trip and a fresh charge — not a local narrowing of the 500 events already on screen.

04 The four workspaces

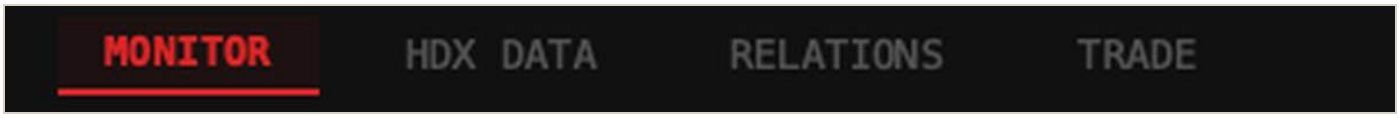


Fig 5 — The workspace strip.

Workspace	What it is	State in v5.0.1
`MONITOR`	The live event map, table and summary rails	Works — this is the tab's substance
`HDX DATA`	Search across the UN's Humanitarian Data Exchange	Partly — one of four views returns data (§09)
`RELATIONS`	A network of active conflicts, crisis types and agencies	**Fixed reference chart**, not live data (§12)
`TRADE`	A four-way trade-policy calculator	**Does not run at all** (§13)

The filters and the legend on the left stay visible on every workspace, but they only drive MONITOR — pressing APPLY FILTERS while looking at RELATIONS still reloads the event set behind it, and still charges for it.

> **Two of the four workspaces are not doing what their labels promise.** RELATIONS shows the same fourteen entities with the same numbers every time it is opened, on any machine, because they are compiled into the application. TRADE fails with a Python import error before it calculates anything, on all four of its analyses. Both are documented in full below rather than skipped, because the controls are real even where the results are not.

> Switching workspace costs nothing and does not trigger a reload. The five-minute refresh timer keeps running across all four, though — it is bound to the tab being visible, not to the MONITOR workspace being selected.

05 The conflict map

One pin per geolocated event, coloured by its category, over a choice of six basemaps.

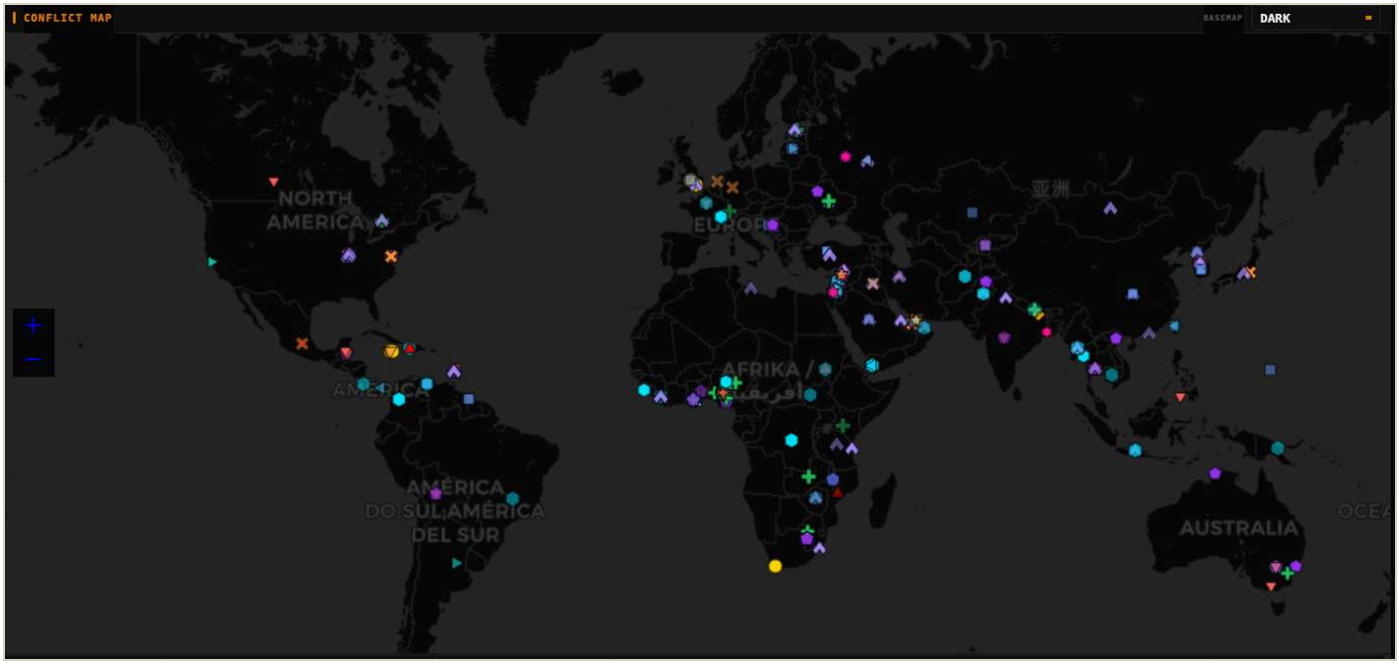


Fig 6 — 449 pins across 77 nations on the DARK basemap.

Interaction	Result
Drag	Pan
Wheel, or the + − controls	Zoom
Click a pin	Selects that event and fills EVENT DETAILS in the right rail
BASEMAP	Switches the tile layer — six options; the camera and pins are preserved

Pin colour is the single most useful thing on the map: the legend's fifteen colours carry straight onto it, so clusters read as *what kind* of event as well as *where*. Sanctions and diplomacy tend to cluster on capitals; terrorism and armed conflict cluster on regions.

> **The pins are the events that carried coordinates**, and in v5.0.1 that is most of them — 449 of 500 in this capture. That is a large change from earlier builds, where roughly one event in four could be placed; §08 explains what changed and why the badge is worth reading.

> **There is no cluster or heat rendering.** At world zoom, dense regions overplot into a single mass of overlapping pins and the only way to separate them is to zoom in. The counts in the right rail (§07) are the reliable way to read magnitude — the map is for location, not for volume.

06 The events table

EVENT ROWS · 06

Below the map, one row per loaded event — including the ones that could not be mapped.

Category	Country	City	Title	Date
Terrorism	Nigeria	Hong	Boko Haram insurgents kill eight in Adamawa community	2026-08-19 03:35:15
Terrorism	Nigeria		jihadist groups escalate school abductions in Nigeria	2026-08-19 02:01:00
Terrorism	Nigeria		troops repel terrorist attack, rescue 14 kidnapped victims	2026-08-18 21:24:12
Strategic	Latvia		Latvian and Lithuanian national guards sign cooperation memorandum	2026-08-19 07:43:00
Strategic	Guyana	Tacama	new soldiers graduate from basic military training	2026-08-19 07:40:27
Strategic	Australia		US interest in using Australia's northern military bases	2026-08-19 07:14:54
Strategic	Latvia		Latvia enables army requisition of private land during emergencies	2026-08-19 07:06:00

Fig 7 — The table, newest first. Category is colour-coded to match the map.

Column	Meaning
Category	The feed's classification, in the legend's colour for that category
Country	Country the event was attributed to
City	City where known — frequently blank
Title	The headline, verbatim from the feed
Date	Event timestamp, newest first by default
Lat / Lng	Coordinates, where the event carried them

Clicking a row selects the event and fills EVENT DETAILS in the right rail, exactly as clicking its pin does. Sorting is client-side on the loaded set — it does not re-query and does not spend.

> **The table is the complete set; the map is a subset.** Events with no coordinates appear here with empty Lat and Lng and no pin, so the table is where to look when the badge says fewer mapped than loaded. It is also the only place the raw coordinates are visible.

> **The feed is untidy and the terminal does not tidy it.** Expect blank cities, headlines that are agency copy rather than clean summaries, and occasional duplicate stories from different outlets. Categories come from the feed's own classifier, so an item can sit under a category you would have filed differently — treat the category as the publisher's opinion, not as ground truth.

07 Overview, categories and hotspots

The right rail summarises the loaded set in four stacked blocks.

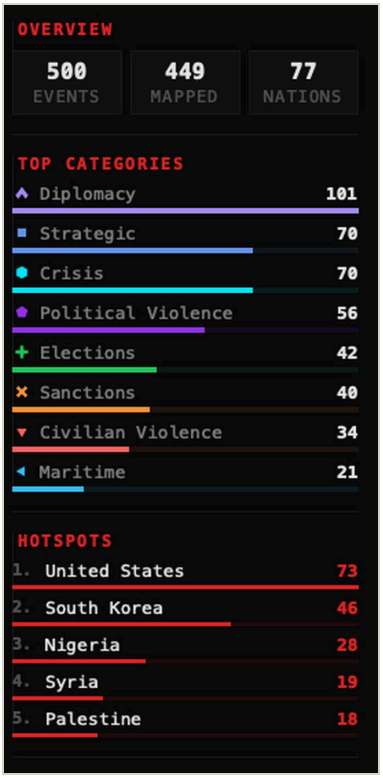


Fig 8 — Counters, the category ranking, the hotspot list and the event-detail card.

Block	What it shows
OVERVIEW	**EVENTS** loaded · **MAPPED** pins drawn · **NATIONS** distinct countries
TOP CATEGORIES	Every category present in the loaded set, ranked, with a bar and a count
HOTSPOTS	The five countries contributing the most events, numbered and ranked
EVENT DETAILS	The selected event — category badge, full headline, country, city, date

In this capture the counters read **500 / 449 / 77**, the ranking ran Diplomacy 101, Strategic 70, Crisis 70, Political Violence 56, Elections 42, Sanctions 40, Civilian Violence 34 and Maritime 21, and the hotspots were United States 73, South Korea 46, Nigeria 28, Syria 19 and Palestine 18.

> **Every number here describes the 500 events you pulled**, not the world and not the feed's full archive. A country topping HOTSPOTS means it dominated this pull — which for a news feed usually means it dominated the last day's coverage, not that it is the most troubled place on earth. Filter by country and reload if you want a country's own picture.

> EVENT DETAILS is empty until something is selected, and says so: *Select an event from the map or table to inspect details here*. It is filled by clicking either a pin or a row.

08 Mapped against loaded

The badge at the top right of the workspace strip is the single most informative number on the screen, and it is worth understanding rather than glancing at.

Fig 9 — The coverage badge.

LOADED is how many events came back. **MAPPED** is how many of them carried coordinates and could be drawn. The gap is the feed's geocoding, not a fault in the terminal.

One press is five pages, and that is deliberate

APPLY FILTERS does not fetch one page. It runs a **map-fill**: five successive pages of a hundred events each, deduplicated and merged, specifically so the map has enough geolocated points to be worth looking at. If a later page fails, the terminal keeps what already landed and draws it rather than discarding the whole pull.

> **The old guidance on this number is out of date, and so is the code's own comment.** Both say roughly one event in four carries coordinates, which would put a 500-event pull at 60–120 pins. The shipping build produced **449 of 500** — about ninety per cent. If you have read that a low MAPPED count is normal, it no longer is: a pull that maps well under half is now worth treating as a bad pull rather than as the expected shape.

> The badge turns red only when nothing at all could be geolocated. A red badge with a healthy LOADED count means the events arrived but none had coordinates — usually a filter narrow enough that only unplaceable wire copy matched.

09 What HDX is, and its four views

HDX is the **Humanitarian Data Exchange**, the UN OCHA-run open catalogue of humanitarian datasets. This workspace searches it. Nothing here costs credits.

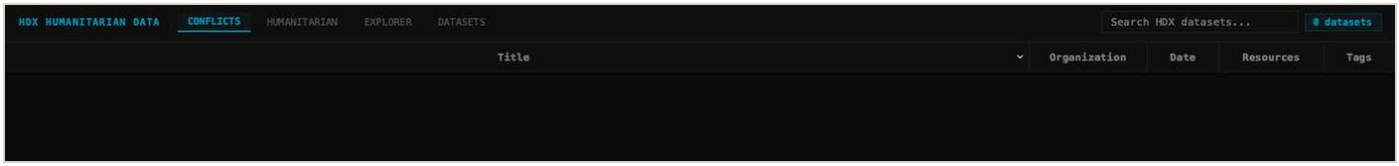


Fig 10 — The HDX header: four views, a search box and a dataset counter.

View	What it asks for	Returned in this capture
`CONFLICTS`	A canned search for conflict datasets	**0 datasets**
`HUMANITARIAN`	A canned search for humanitarian datasets	**0 datasets**
`EXPLORER`	Your own country + topic search	Empty until you press SEARCH (§11)
`DATASETS`	The general dataset listing	**20 datasets** (§10)

All four share one table — **Title**, **Organization**, **Date**, **Resources**, **Tags** — and one counter in the header, which reports how many rows the current view holds. The search box at the top right runs an advanced search across the catalogue on Enter.

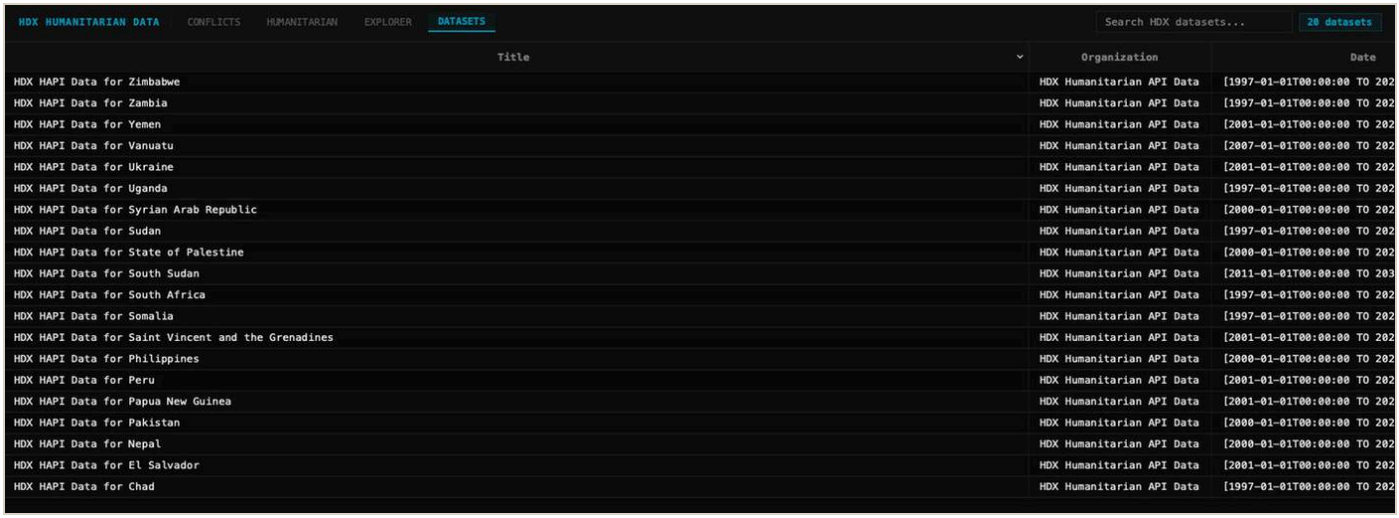
> **CONFLICTS and HUMANITARIAN return nothing, and have for some time.** Both were empty in this capture and both were already documented as commonly empty in the previous edition of this guide. They are not broken in a way the screen reports — the request succeeds and the catalogue simply matches nothing for those canned queries, so you get an empty table and a `0 datasets` counter with no error.

> **Do not conclude from those two that HDX is down.** DATASETS returns real rows and the EXPLORER search works — the two canned views are the problem, not the connection. Start at DATASETS (§10) to confirm the catalogue is reachable before troubleshooting anything.

10 Datasets — the view that works

DATASET LIST · 10

DATASETS lists the catalogue's general dataset entries, and it is where this workspace earns its place.



Title	Organization	Date
HDX HAPI Data for Zimbabwe	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for Zambia	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for Yemen	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Vanuatu	HDX Humanitarian API Data	[2007-01-01T00:00:00 TO 202
HDX HAPI Data for Ukraine	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Uganda	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for Syrian Arab Republic	HDX Humanitarian API Data	[2000-01-01T00:00:00 TO 202
HDX HAPI Data for Sudan	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for State of Palestine	HDX Humanitarian API Data	[2000-01-01T00:00:00 TO 202
HDX HAPI Data for South Sudan	HDX Humanitarian API Data	[2011-01-01T00:00:00 TO 203
HDX HAPI Data for South Africa	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for Somalia	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202
HDX HAPI Data for Saint Vincent and the Grenadines	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Philippines	HDX Humanitarian API Data	[2000-01-01T00:00:00 TO 202
HDX HAPI Data for Peru	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Papua New Guinea	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Pakistan	HDX Humanitarian API Data	[2000-01-01T00:00:00 TO 202
HDX HAPI Data for Nepal	HDX Humanitarian API Data	[2000-01-01T00:00:00 TO 202
HDX HAPI Data for El Salvador	HDX Humanitarian API Data	[2001-01-01T00:00:00 TO 202
HDX HAPI Data for Chad	HDX Humanitarian API Data	[1997-01-01T00:00:00 TO 202

Fig 11 — Twenty HDX HAPI datasets, one per country, each with its publishing organisation and coverage dates.

In this capture it returned twenty **HDX HAPI Data** entries — Zimbabwe, Zambia, Yemen, Vanuatu, Ukraine, Uganda, Syrian Arab Republic, Sudan, State of Palestine, South Sudan, South Africa, Somalia, Saint Vincent and the Grenadines, Philippines, Peru, Papua New Guinea, Pakistan, Nepal, El Salvador and Chad — all published by **HDX Humanitarian API Data**, each carrying a coverage range beginning as far back as 1997.

Column	Meaning
Title	The dataset's name in the catalogue
Organization	The publishing body
Date	The dataset's coverage range
Resources	How many downloadable files the entry carries
Tags	The catalogue's own keywords

> **HAPI is HDX's own aggregated API layer**, so these twenty entries are country-level roll-ups rather than individual field datasets. That is why the list is short and uniform: it is the catalogue's structured tier, not everything HDX holds. Use EXPLORER (§11) or the search box to reach the wider catalogue.

> **The table is a catalogue index, not the data.** Nothing in this workspace downloads a dataset or plots it — the Resources column tells you files exist, and getting them means going to HDX itself. There is no export, and no link-out from a row.

11 Explorer — searching HDX yourself

EXPLORER is the only HDX view you drive. Its controls sit in a bar at the **foot** of the panel, which is easy to miss because every other control on this tab is at the top.

Fig 12 — The explorer bar, at the bottom of the panel.

Control	What it does
COUNTRY	A dropdown of countries the catalogue knows. Optional
TOPIC	One of seven: conflict · humanitarian · displacement · food security · health · education · refugees
SEARCH	Runs the query and fills the shared table

The table stays empty until SEARCH is pressed — an empty EXPLORER is the default state, not a failed query. The topic values are search keys sent to the catalogue rather than display labels, which is why they read as lower-case phrases rather than as tidy captions.

> **The search box in the header is a different, broader search.** Typing there and pressing Return runs an advanced query across the whole catalogue, independent of the country and topic pickers at the bottom. If EXPLORER's structured search finds nothing, the header box with a plain keyword is usually the better second attempt.

> Everything in this workspace is **free**. HDX is an open catalogue and the terminal queries it directly, so nothing here touches the metered news feed. Results are cached, so repeating a search you have already run does not re-query.

12 Relations — a fixed reference chart

REFERENCE CHART · 12

> **Read this first: RELATIONS is not live data.** Every entity, every severity and every count on this workspace is compiled into the application. The panel builds the same fourteen nodes with the same numbers on every launch, on every machine, whether or not the network is reachable — the on-screen figures match the source literals exactly.

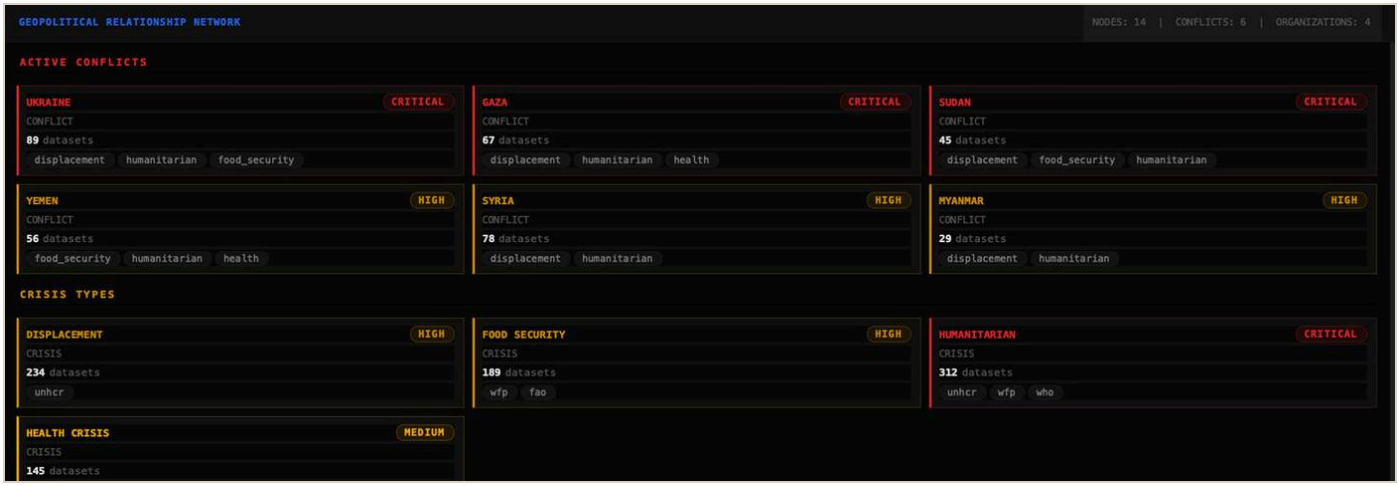


Fig 13 — The relationship network: six active conflicts, four crisis types and the agencies that work on them.

Group	Entries
ACTIVE CONFLICTS	Ukraine (critical, 89) · Gaza (critical, 67) · Sudan (critical, 45) · Yemen (high, 56) · Syria (high, 78) · Myanmar (high, 29)
CRISIS TYPES	Displacement (high, 234) · Food Security (high, 189) · Humanitarian (critical, 312) · Health Crisis (medium, 145)
ORGANIZATIONS	UNHCR · WFP · WHO · FAO

Each card shows the entity, a severity badge, a dataset count and the tags linking it to other entities — Ukraine to displacement, humanitarian and food security; food security to WFP and FAO. The header counts `NODES: 14 | CONFLICTS: 6 | ORGANIZATIONS: 4`.

> **The counts look like live dataset totals and are not.** “89 datasets” beside Ukraine is a constant, not a query result. Nothing on this workspace changes when you reload, change filters, or come back tomorrow. Treat it as a printed wall chart of which agencies work on which crises — useful as orientation, worthless as monitoring.

> Nothing here is clickable through to anything, and nothing here costs credits.

13 Trade — four analyses, none of which run

The TRADE workspace offers four trade-policy analyses, selected either from the ANALYSIS TYPE dropdown or from the four sub-tabs beneath it — the two stay in sync.

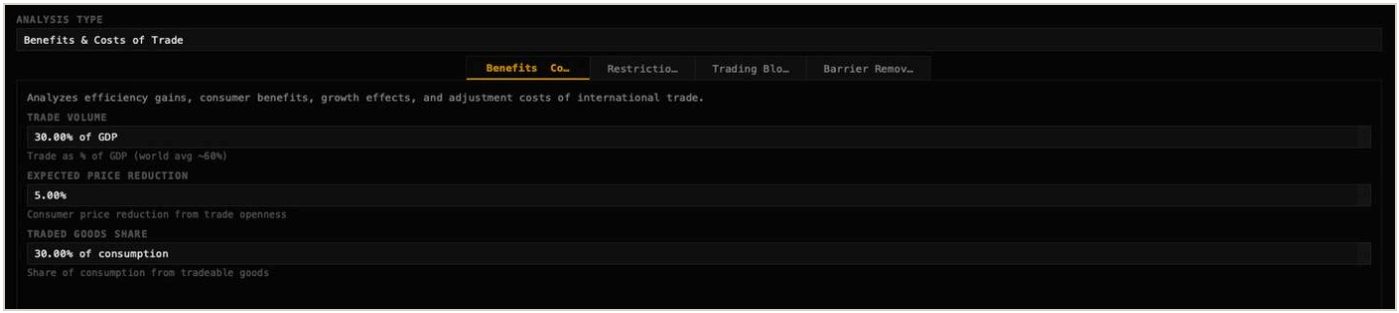


Fig 14 — The Benefits & Costs form: three inputs, each with its own explanatory caption.

Analysis	Inputs it collects
`Benefits & Costs of Trade`	Trade volume as % of GDP · expected price reduction · traded-goods share of consumption
`Trade Restrictions Analysis`	Tariff rate · quota volume · subsidy rate · development level · industry maturity
`Trading Blocs Analysis`	Integration type (FTA / customs union / common market / economic union) · trade creation · trade diversion
`Trade Barrier Removal Impact`	Liberalization type (unilateral / bilateral / regional / multilateral) · tariff reduction

Fig 15 — What RUN ANALYSIS actually produces, on every one of the four.

Every analysis fails before it computes anything. RUN ANALYSIS returns a Python traceback: the calculation script is written as part of a package and uses a relative import on its first line, but the terminal launches it as a standalone file, so it dies on `ImportError: attempted relative import with no known parent package`. Reproduced here on Benefits & Costs and on Trading Blocs; the failure is in the script's entry conditions, so it applies to all four equally.

And two of the four are wired to the wrong calculation. Even with the import fixed, **Trading Blocs** is dispatched to the benefits-and-costs routine and **Barrier Removal** to the restrictions routine. The script contains correct implementations of both missing analyses and selects between them on a mode argument — but the two callers hardcode the other modes, so those implementations are unreachable, and the integration type and trade-creation figures you typed are read by nothing on the path the request actually takes.

Nothing here touches the network or spends credits — the calculator was always local, and still is. The cost of this defect is a wasted panel, not a wasted balance.

14 Credits — what costs what

One action on this tab spends: reloading events. Everything else is free.

Action	Cost
Opening the tab (it loads immediately, before you touch anything)	**~15**
APPLY FILTERS	**~15**
Each five-minute auto-refresh tick, while the tab is visible	**~15**
Switching workspace · sorting the table · clicking a pin or row	free
CLEAR · changing a filter without applying · the basemap picker	free
Everything in HDX DATA — all four views and both searches	free
Everything in RELATIONS and TRADE	free

The price badge on APPLY FILTERS understates the real cost about five-fold. The badge reads **· 3 CR**, which is the published price of one page of events. One press runs a five-page map-fill (§08), so the true charge is on the order of **15 credits**. The badge is not lying about the unit price — it is quoting a unit that does not correspond to anything you can press.

The five-minute timer is the real expense. It starts when the tab becomes visible and stops when you leave it, so a tab left open in the background costs roughly **180 credits an hour** whether or not you look at it. There is no toggle for it on this screen. Switching to any other tab is the off switch.

Repeating an identical pull inside two minutes replays from cache for free — the events cache holds for 120 seconds, and reference data such as the category and country lists for ten minutes. Pressing APPLY FILTERS twice in quick succession with the same filters therefore costs once.

15 Driving the tab from the AI assistant

Thirteen tools expose this tab's data to the AI assistant and to workflows.

Tool	What it returns
`fetch_geopolitics_events`	Events, with the same country / city / category filters
`list_geopolitics_critical_regions`	The regions the feed currently flags as critical
`list_geopolitics_countries`	Every country present in the feed
`list_geopolitics_categories`	The category vocabulary behind the legend
`list_geopolitics_cities`	Every city present in the feed
`extract_geolocations_from_headlines`	Coordinates inferred from headline text
`search_hdx_conflicts`	The CONFLICTS canned search
`search_hdx_humanitarian`	The HUMANITARIAN canned search
`search_hdx_by_country`	HDX datasets for one country
`search_hdx_by_topic`	HDX datasets for one topic
`search_hdx_advanced`	The free-text catalogue search
`analyze_trade_benefits`	The benefits-and-costs calculation
`analyze_trade_restrictions`	The restrictions calculation

Worth knowing

- **The event tools spend the same credits the buttons do.** Asking the assistant to sweep several countries is several pulls.
- **The two trade tools hit the same broken script** as the panel (§13), so they fail the same way — the defect is in the script's entry conditions, not in the screen.
- **The HDX tools are free**, and they reach the catalogue directly. Where the CONFLICTS and HUMANITARIAN views come back empty on screen, the corresponding tools come back empty too — it is the query, not the surface.
- `extract_geolocations_from_headlines` has no equivalent control on the screen. It is the only tool here that does something the UI cannot.

16 Where the data comes from

PROVENANCE · 16

Surface	Source	Freshness
`Events, categories, countries, cities`	A commercial world-news events API	Cached 120 s
`Category and country reference lists`	The same API	Cached 10 min
`HDX datasets and searches`	The UN OCHA Humanitarian Data Exchange	Cached
`Relations network`	**Compiled into the application** (§12)	Never changes
`Trade analyses`	A local Python script — no network (§13)	—
`Basemap tiles`	Tile providers, cached to disk	—

The status bar names the first two directly: `SOURCE: NEWS-EVENTS API + HDX` and `ENGINE: PYTHON + C++`. The engine line is honest about the split — the map, table and rails are native, while HDX search and the trade calculator run through the bundled Python.

> **Events are published onto the terminal's internal data bus** as well as drawn, so other surfaces — dashboard widgets in particular — can consume the same pull without paying for it again. That is why a dashboard geopolitics widget can show events without this tab being open.

> **Two of the six rows above are not what a reader would assume.** The relations network reads as a live intelligence graph and is a constant; the trade calculator reads as an analysis engine and currently produces only a traceback. Both are covered in §12 and §13 — this table is where the difference becomes visible at a glance.

17 Status and message reference

The status bar runs the width of the screen under the three columns.

Field	Meaning
SOURCE	`NEWS-EVENTS API + HDX` — the two upstream services
ENGINE	`PYTHON + C++` — the two execution paths behind the panels
CREDITS	Remaining balance as the events API last reported it
State	**READY**, **LOADING…** or **ERROR**, at the far right

Message	Where	What it means
`LOADING…`	Status label	A pull is in flight
`READY`	Status label	The pull completed and the screen is idle
`ERROR`	Status label	The last operation failed — the panel below names it
`0 datasets`	HDX header	The query succeeded and matched nothing (§09)
`Analyzing…`	Trade panel	An analysis was dispatched — in this build, always followed by a traceback
`Select an event…`	Event details	Nothing is selected yet

The ERROR state is sticky across workspaces. Running a trade analysis leaves the status bar reading ERROR, and it stays that way while you move to MONITOR or HDX — it reflects the last operation attempted anywhere on the tab, not the health of what you are currently looking at. A successful reload clears it.

The UTC clock beside the workspace tabs updates every second while the tab is visible and stops when it is not. If it is frozen, the tab is not in the foreground — and neither is the refresh timer, which is the good news.

18 Quick reference and honest notes

How do I…

Goal	Do this
Open the tab	Command bar → `geo`, then click the **GEO** row
Load fresh events	**APPLY FILTERS** — about 15 credits (§14)
Narrow to one country	Type it in COUNTRY, then APPLY FILTERS
See only one category	Pick it in CATEGORY, then APPLY FILTERS
Read an event in full	Click its pin or its table row (§07)
Find events with no location	The table — blank Lat / Lng (§06)
Judge whether a pull is healthy	The MAPPED / LOADED badge (§08)
Search humanitarian data	HDX DATA → **DATASETS**, or EXPLORER (§10, §11)
Change the map's look	**BASEMAP**, six options
Stop spending	Switch to any other tab — that stops the 5-minute timer

> Nothing on this tab is destructive. No control here writes, deletes or trades; the cost of any mistake is credits, not data.

Honest notes

- **The whole TRADE workspace fails before it calculates.** All four analyses return a Python import traceback (§13).
- **Two of those four are also wired to the wrong calculation** — Trading Blocs runs benefits-and-costs, Barrier Removal runs restrictions, and the inputs you typed are read by neither (§13).
- **RELATIONS is a constant.** Fourteen nodes with fixed counts, compiled in; it never changes and never queries anything (§12).
- **APPLY FILTERS costs ~15 credits, not the 3 on its badge** — one press is a five-page fill (§14).
- **Opening the tab spends before you touch anything**, and the five-minute timer keeps spending — about 180 credits an hour with the tab merely visible (§14).
- **The old "only one event in four is mappable" guidance is obsolete**, and the code comment still repeats it. The shipping build mapped 449 of 500 (§08).
- **HDX CONFLICTS and HUMANITARIAN return 0 datasets**, silently, while DATASETS returns 20 (§09).
- **EXPLORER's controls are at the bottom of the panel**, unlike every other control on the tab (§11).
- **Filters never narrow what is already on screen** — they are parameters for the next paid pull (§03).
- **The ERROR state is sticky across workspaces** and reflects the last thing attempted anywhere on the tab (§17).
- **The map has no clustering**, so dense regions overplot; read magnitude from the right rail, not the map (§05).
- **Every right-rail number describes your 500 events**, not the world (§07).
- **HDX rows are a catalogue index, not the data** — no download, no link-out, no export (§10).
- **Categories are the publisher's classification**, not a Fincept judgement (§06).

> Every item above was read from the shipping source and then confirmed on screen on the v5.0.1 build; the trade failure was reproduced twice, on two different analyses. They are reported because a guide that described RELATIONS as an intelligence graph and TRADE as a working calculator would be describing a product this build does not contain.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Maritime Intelligence

A complete, screen-by-screen walkthrough of the Maritime tab — live vessel tracking and a seven-workspace analytics desk over the world's shipping.

`>_ 7 workspaces · 197 corridors · 80 ports · 157 flag states`

Covers · **the map** · **flows, ports, countries** · **fleet & signals** · **watchlists**

Every credit cost · what needs Exclusive Pro · where each number comes from

Demonstrated live on the Fincept Terminal Maritime tab, v5.0.1.

Contents

21 SECTIONS

GETTING ORIENTED

01 What the Maritime tab is ... 3
02 Anatomy of the screen ... 4
03 The filter bar ... 5
04 The seven workspaces ... 6

THE MAP WORKSPACE

05 Load actions and the draw tools ... 7
06 Fleet intelligence and trade corridors ... 8
07 Market pulse and chokepoint transits ... 9
08 The map, its toolbar and its layers ... 10
09 Vessel search, place search and ports ... 11
10 Tracking a ship and replaying a voyage ... 12

THE ANALYTICS WORKSPACES

11 FLOWS — corridors and top lanes ... 13
12 PORTS — league table and congestion ... 14
13 COUNTRIES — seaborne arrivals ... 15
14 FLEET — tonnage and flag states ... 16
15 SIGNALS — four fleet indicators ... 17
16 WATCH — standing screens over the fleet ... 18

REFERENCE

17 Credits — what costs what ... 19
18 Plans — what actually needs Exclusive Pro ... 20
19 Driving the tab from the AI assistant ... 21
20 Where the data comes from ... 22
21 Quick reference and honest notes ... 23

About this guide. Every screenshot is a live capture of the Maritime tab in the shipping v5.0.1 build. The map workspace, the credit model and the plan gate are documented from the product's source; the six analytics workspaces are documented from the running application and from the binary's own method names, and where a statement about them is observational the text says so.

01 What the Maritime tab is

ORIENTATION · 01

Maritime Intelligence used to answer one question — *what ships are where, and where are they going?* It still answers that, on a live world map. But it now also answers a second, larger one: **what is the world's shipping actually doing?** Six analytics workspaces sit beside the map covering trade corridors, port congestion, national arrivals, fleet tonnage, market indicators and standing vessel screens.

You give the map a patch of ocean — or a single IMO number — and it plots the vessels. You give the analytics a time window and a segment, and they aggregate the whole tracked fleet.

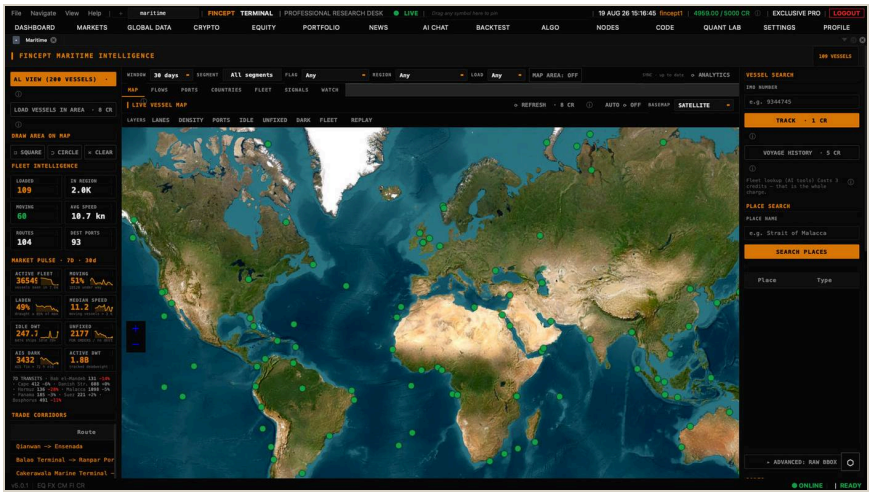


Fig 1 — The MAP workspace with a global sample loaded. Left: load actions, draw tools, fleet statistics, market pulse and chokepoint transits. Centre: the live vessel map. Right: the search tools.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Research & Intelligence ▸ Maritime**
Command bar	`maritime` — `marine` and `shipping` also resolve
Main tab bar	**Not present.** This is not one of the top-level tabs

At a glance, in Maritime you can…

- Plot up to 200 live vessels worldwide
- Search any sea area by box, circle or place
- Track one ship by IMO and replay its voyage
- Rank 197 trade corridors by volume
- See which of 80 ports have ships waiting
- Compare seaborne arrivals by country
- Break the tracked fleet down by tonnage
- Chart speed, laden share and idle tonnage
- Run standing screens over the whole fleet
- Switch between six basemaps

> **This tab spends credits, and the first load is automatic.** An area search costs **8 credits** and one fires when the tab is first shown, before you click anything. §17 lists every action that spends and every action that does not.

02 Anatomy of the screen

Fourteen regions. A filter bar and a workspace strip span the top; below them the layout is a left rail, a large centre, and a right rail of search tools.

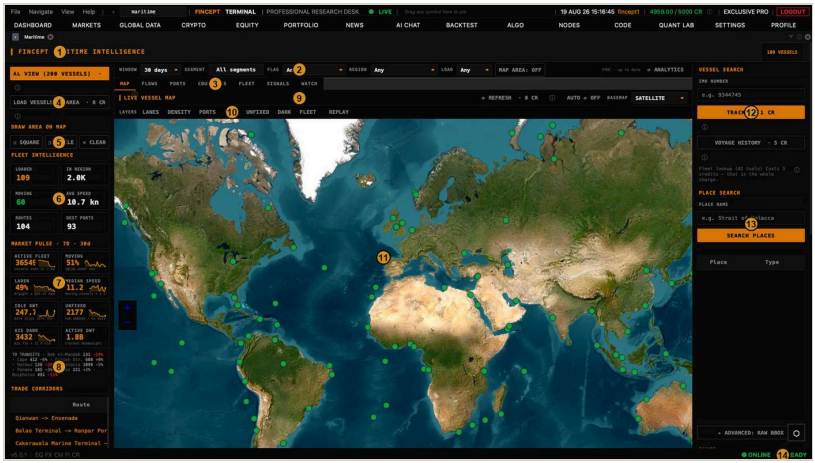


Fig 2 — The fourteen regions, numbered.

#	Region	What it is
1	Brand + vessel count	A live count of the vessels currently plotted
2	Filter bar	Window, segment, flag, region, load state — applies to the analytics (§03)
3	Workspace tabs	MAP · FLOWS · PORTS · COUNTRIES · FLEET · SIGNALS · WATCH (§04)
4	Load actions	GLOBAL VIEW and LOAD VESSELS IN AREA, each priced on the button (§05)
5	Draw tools	Square, circle and clear — define a search area on the map (§05)
6	Fleet intelligence	Six tiles computed from the loaded vessel set (§06)
7	Market pulse	Eight fleet-wide metrics with sparklines (§07)
8	Chokepoint transits	Seven-day transit counts for eight waterways (§07)
9	Map toolbar	Refresh, auto-refresh, basemap picker (§08)
10	Map layers	Nine overlay toggles — lanes, density, ports, idle and more (§08)
11	The live vessel map	Pan, zoom, draw, click a pin (§08)
12	Vessel search	IMO number, TRACK and VOYAGE HISTORY (§09, §10)
13	Place and port search	Geocode a strait or find a harbour (§09)
14	Status bar	Connection and readiness for the whole terminal

> **The vessel table has left the map workspace.** Earlier builds put a sortable table of loaded vessels under the map. In v5.0.1 the map fills the centre — dragging where the splitter used to be pans the map instead. Per-vessel tabular data now lives in the **WATCH** workspace (§16), which lists far more of the fleet than the map's 200-pin cap.

03 The filter bar

One row of controls sits above every workspace and scopes what the analytics aggregate. It is the single most important control on the tab, because every number in workspaces 11–16 is computed inside whatever it currently says.

Fig 3 — Left: window, segment, flag, region and load state.

Fig 4 — Right: the map-area toggle, the sync indicator and ANALYTICS.

Control	Options	What it scopes
WINDOW	7 · 30 · 90 · 180 · 365 days	The look-back period for every aggregate. Defaults to 30 days
SEGMENT	All segments · Dry Bulk · Crude Tanker · Product Tanker · Container · Gas Carrier	Restricts every workspace to one vessel class
FLAG	Any, or a flag state	Restricts to vessels of one registry
REGION	Any, or a sea region	Restricts to one geographic region
LOAD	Any · laden · ballast	Restricts by whether ships are loaded or running empty
MAP AREA: OFF	Toggle	When on, confines the analytics to the map's current viewport instead of the whole world
SYNC	Indicator	Reads *up to date* when the cached pages match the filter
ANALYTICS	Button	Re-runs the analytics fetch

> Changing the window marks every cached workspace stale and re-fetches, so the numbers you read are always the ones the bar describes. Each workspace also restates the window in its own heading — `CORRIDOR FLOWS · 197 lanes · 30d`.

It genuinely re-queries — verified

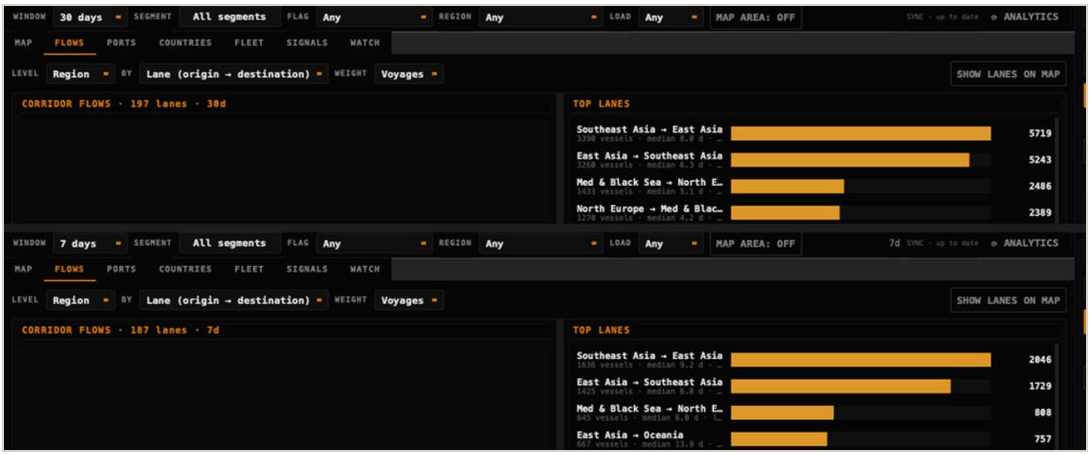


Fig 5 — The same workspace at 30 days (top) and 7 days (bottom). The lane count falls from 197 to 187, the leading corridor from 5,719 to 2,046, the median days shift, and the fourth-ranked lane changes identity.

04 The seven workspaces

The strip under the filter bar switches the centre of the screen. The first is the live map that this tab has always been; the other six are aggregate views over the tracked fleet.

Fig 6 — The workspace strip.

Workspace	Answers	Section
MAP	Where are these specific ships, right now?	§05–§10
FLOWS	Which trade corridors carry the most traffic?	§11
PORTS	Which ports are busiest, and where are ships waiting?	§12
COUNTRIES	Which countries receive the most seaborne arrivals?	§13
FLEET	How is tracked tonnage distributed by size and flag?	§14
SIGNALS	Is the fleet speeding up, loading up, or idling?	§15
WATCH	Which individual ships match a standing screen?	§16

How they relate to the map

The analytics workspaces are not islands. Several can push their results back onto the map — **SHOW LANES ON MAP** in FLOWS, **SHOW PORTS ON MAP** in PORTS, and **PIN ON MAP** in WATCH — which switches you to the MAP workspace with that result drawn as an overlay. The map's own layer row (§08) toggles the same overlays directly.

> **Switching workspace does not cost credits.** The 8-credit meter is attached to vessel *area searches* on the map, not to the analytics. Moving between FLOWS, PORTS, COUNTRIES, FLEET, SIGNALS and WATCH — and changing the filter bar — did not move the credit counter during this capture session.

05 Load actions and the draw tools

The top of the left rail is how vessels get onto the map. Both load buttons print their price.

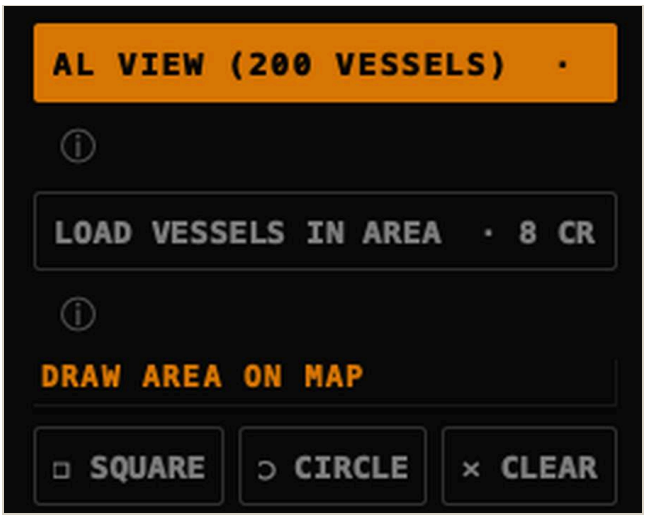


Fig 7 — Load actions and the three draw tools.

Control	What it does
GLOBAL VIEW (200 VESSELS)	Loads a worldwide sample — **8 credits**
LOAD VESSELS IN AREA	Searches using the coordinates in ADVANCED: RAW BBOX — **8 credits**
▢ **SQUARE** / ◯ **CIRCLE**	Arms rectangle or circle drawing; drag on the map to define the shape. The two are mutually exclusive — arming one disarms the other
✕ **CLEAR**	Disarms both tools and removes the shapes. **Free**, and it leaves your vessels on screen

Finishing a drag costs 8 credits. Completing a square or circle immediately runs an area search over the shape — there is no separate confirm step. Drawing several shapes loads the union of their bounding boxes and then keeps only the vessels that fall inside one of the shapes, so an awkward multi-shape selection can cost several searches to arrive at.

GLOBAL VIEW is a spread, not a “top 200”. It asks the API for 2,000 vessels and bins them across a grid so the pins fan out worldwide rather than clustering wherever traffic happens to be densest. What you see is a geographic sample, not the 200 biggest or busiest ships.

06 Fleet intelligence and trade corridors

Six tiles summarise the vessels you just loaded — not the world, just your working set. Below them, the corridors those vessels are sailing.

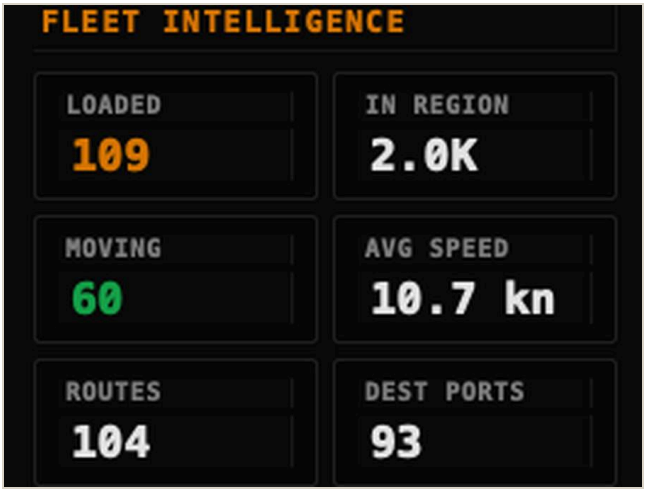


Fig 8 — The six fleet tiles. Every one is computed from the loaded set.

Tile	Meaning
LOADED	Vessels plotted as pins — capped at 200
IN REGION	How many the server reports in the area you searched
MOVING	Loaded vessels making more than 0.5 knots
AVG SPEED	Mean speed of the *moving* vessels only, in knots
ROUTES	Distinct origin→destination corridors in the loaded set
DEST PORTS	Distinct destination ports in the loaded set

> **LOADED and IN REGION are deliberately different numbers.** LOADED is what is drawn (≤200); IN REGION is the true population of the area you searched. Seeing **109** next to **2.0K** is normal and correct — you are looking at a sample, not a census. Every other tile describes the sample, not the region.

TRADE CORRIDORS

Column	Meaning
Route	An origin→destination pair seen in the loaded set
Vessels	A live count of loaded vessels on that corridor — **not trade value**
Status	In practice always ACTIVE

> Selecting a corridor row is free and fills the SELECTED ROUTE card in the right rail. The underlying feed exposes no dollar volume, so the panel does not invent one — where a trade value would go, it prints *n/a*. For corridor analysis with real weight behind it, use the FLOWS workspace (§11), which aggregates the whole tracked fleet rather than your 200-pin sample.

07 Market pulse and chokepoint transits

Below the corridors sit two blocks that, unlike the fleet tiles, describe the **whole tracked fleet** rather than your loaded sample. They are new in v5.0.1.

Fig 9 — MARKET PULSE, a sparkline per metric.

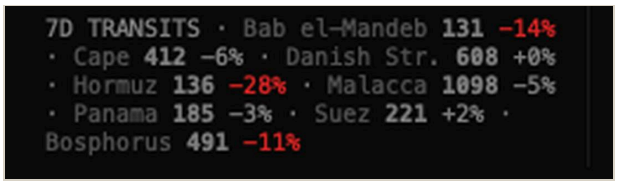


Fig 10 — Seven-day chokepoint transits.

Metric	Reads	What it means
`ACTIVE FLEET`	36,549	Vessels seen in the last 7 days
`MOVING`	51%	Share of the fleet under way — 18,520 of them
`LADEN`	49%	Share whose draught is at least 85% of maximum
`MEDIAN SPEED`	11.2	Median knots across moving vessels above 3 kn
`IDLE DWT`	247.1	Deadweight tonnage idle 7 days or more — 6,474 ships
`UNFIXED`	2,177	Vessels for orders with no declared destination
`AIS DARK`	3,432	Vessels whose last AIS fix is more than 72 hours old
`ACTIVE DWT`	1.8B	Total tracked deadweight

The header reads **MARKET PULSE · 7D · 30d** — the metrics are seven-day figures, and the sparkline behind each one traces its recent history.

Chokepoint transits

Eight waterways with a seven-day transit count and a change figure: **Bab el-Mandeb**, **Cape**, **Danish Strait**, **Hormuz**, **Malacca**, **Panama**, **Suez** and **Bosphorus**. The percentage is the move against the prior period, coloured red for a fall and green for a rise — so a line reading `Hormuz 136 −28%` is a materially quieter week in the Gulf.

> **These two blocks are read-only.** Nothing in either responds to a click, and neither is scoped by the filter bar — they describe the tracked fleet as a whole regardless of the segment or region selected above. Treat them as a standing market backdrop rather than as part of your current query.

08 The map, its toolbar and its layers

Fig 11 — The map toolbar and, beneath it, the nine layer toggles.

Control	What it does
`↻ REFRESH · 8 CR`	Re-runs the current search — the raw bbox if valid, otherwise the global sample
`AUTO ↻ OFF / 5m`	Toggles a five-minute auto-refresh, off by default. Each tick is a fresh area search — **8 credits every 5 minutes while the tab is visible**
`BASEMAP`	Six tile layers — satellite (default), dark, ocean, streets, light, terrain. Pins, shapes and camera are preserved; tiles cache to disk

The layer row

Nine toggles draw aggregate overlays on top of the vessel pins: **LAYERS**, **LANES**, **DENSITY**, **PORTS**, **IDLE**, **UNFIXED**, **DARK**, **FLEET** and **REPLAY**. These are how the analytics workspaces put their results back onto the map — lanes draw as arcs between regions, ports and density as bubbles and cells.

On the map	What it is
Green pin	A vessel in the loaded fleet — the clickable ones
Amber pin	A ship you tracked by IMO, or a port you plotted
Amber→yellow trail	Voyage history — recent positions, fading as they age
Drag	Pan — unless a draw tool is armed
Wheel / + −	Zoom
Click a pin	Fills the vessel card and writes that ship's IMO into the search field

> **Click a ship, then VOYAGE HISTORY.** Clicking a vessel pin fills the card in the right rail and writes its IMO into the IMO NUMBER field. That auto-fill is the intended workflow: click on the map, then press VOYAGE HISTORY without typing anything.

09 Vessel search, place search and ports

The right rail holds four independent tools. Only some of them spend, and the prices are printed on the buttons.

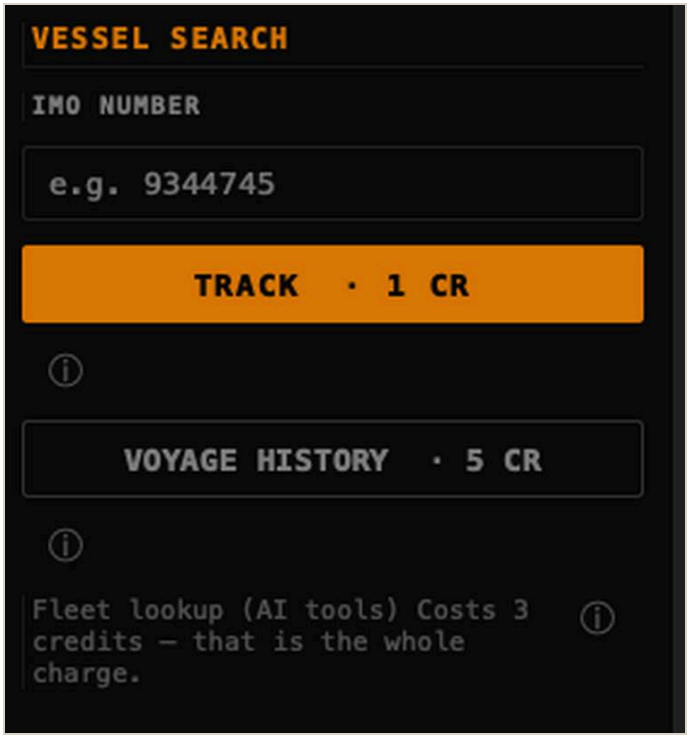


Fig 12 — Vessel search, priced per action.

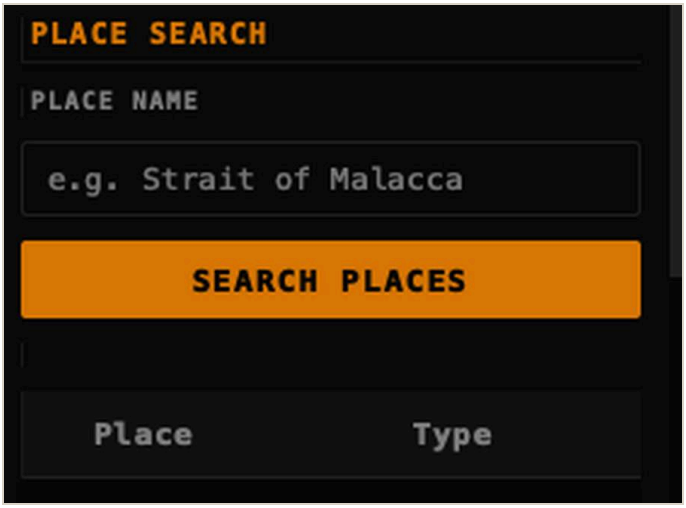


Fig 13 — Place search, with a typeahead.

Tool	What it does	Cost
`Vessel search`	IMO NUMBER + **TRACK** drops one amber pin at the last reported position and flies to it. TRACK clears the fleet — reload an area to get it back	**1 CR**
`Voyage history`	Replays a ship's recent positions as a fading trail (§10)	**5 CR**
`Place search`	A geocoder — turns “Strait of Malacca” into a bounding box. Searching is free, but **selecting a result row runs an area search**	search free / select **8 CR**
`Advanced: raw bbox`	Type a rectangle directly, then SEARCH AREA. It also drives LOAD VESSELS IN AREA, ↻ REFRESH and PORTS IN AREA	**8 CR** on search
`Ports`	SEARCH PORTS by name and PORTS IN AREA; click a row to drop an amber pin. The Source column shows Wikidata, MarineRegions or OSM	**free**

Selecting a place row costs 8 credits — with no button press. Merely selecting a row runs an area search over that place's bounding box, and that includes moving onto it with an arrow key or accepting a typeahead suggestion. If you only want to see which places matched, do not touch the rows.

Do not sort the place or port result tables. A row's *position* is used to look up the underlying result, so after sorting, a click acts on the wrong entry — which for the place table means loading the wrong sea area, at 8 credits.

A place can geocode to a point rather than a region, giving a bounding box metres across that returns no vessels. Widen the raw bbox by hand and search again.

10 Tracking a ship and replaying a voyage

TRACK — one hull, right now

Enter an IMO number and press **TRACK · 1 CR**. The map clears the loaded fleet, drops a single amber pin at that ship's last reported position, and flies to it. The card above fills with name, IMO, position, speed and the declared origin and destination.

If the number is not in the feed the panel says so plainly — *Not found in DB* — rather than failing silently.

VOYAGE HISTORY — where it has been

Press **VOYAGE HISTORY · 5 CR** to replay the ship's recent reported positions as a track: bright yellow for the newest fix, faint amber for the oldest, a green dot for the current position, and white pins for the resolved origin and destination ports.

Behaviour	Detail
Replaces the fleet	Like TRACK, history mode clears the loaded vessels from the map
A moored ship	Produces a tight knot of dots rather than a line
Ports	Origin and destination names are resolved to real port locations and pinned
No data	The status reads NO HISTORY rather than drawing an empty map

> **In history mode the table changes meaning but not its headers.** Each row becomes one position report rather than one vessel, and the final column holds that report's date — while the header still reads **Progress**. Read the column by its contents, not its caption.

> Both TRACK and VOYAGE HISTORY need the **live AIS** capability, which is an Exclusive Pro feature. Vessel *search* is not gated. §18 sets out exactly where that line falls.

11 FLOWS — corridors and top lanes

The first analytics workspace answers where shipping actually goes. A chord diagram on the left shows flows between regions; a ranked list on the right names the busiest lanes.

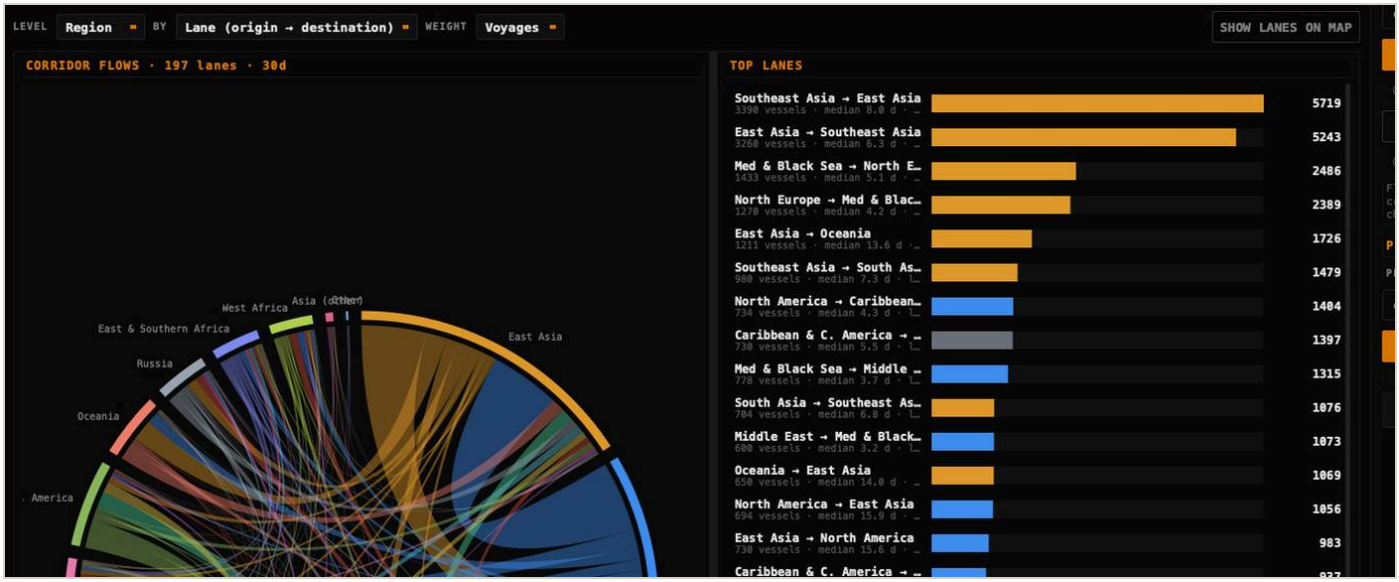


Fig 14 — CORRIDOR FLOWS: 197 lanes over 30 days, with TOP LANES ranked alongside. Each lane carries its vessel count and median transit days.

Control	Options	Effect
LEVEL	Region, and finer levels	How coarsely origins and destinations are bucketed
BY	Lane (origin → destination)	What a row represents
WEIGHT	Voyages, and other measures	What the ranking and the ribbon widths measure
SHOW LANES ON MAP	Button	Draws the lanes as arcs on the MAP workspace

Reading a lane

Each row in TOP LANES gives the corridor, a bar proportional to its weight, and the weight itself — `Southeast Asia → East Asia  5719`. Beneath it in smaller type sits the supporting detail: `3398 vessels · median 8.0 d`. So the headline number is the measure you selected under WEIGHT, and the vessel count and median transit time qualify it.

Direction matters and is preserved. Southeast Asia→East Asia and East Asia→Southeast Asia appear as separate rows with different weights, which is what makes the ranking useful for spotting imbalance on a route.

> **Observed behaviour.** Rows respond to selection and the panel carries a lane-detail loader, so a corridor can be drilled into for its own breakdown. Because the shipped analytics code is not in the source tree for this build, the exact contents of that drill-down are described here only as far as they were seen on screen.

12 PORTS — league table and congestion

Two ranked lists side by side: which ports are busiest, and which have ships sitting outside them. The second is the one that moves markets.

Fig 15 — PORT LEAGUE TABLE (80 ports, arrivals in window) on the left; CONGESTION — SHIPS WAITING (60 ports) on the right.

Panel	Ranks by	Row detail
`PORT LEAGUE TABLE`	Arrivals inside the selected window	Port name, then country and region, then the arrival count
`CONGESTION · SHIPS WAITING`	Vessels currently waiting	Port name, then berth and median-wait detail, then the waiting count

The **RANK BY** selector at the top left changes what the league table sorts on — it read *Calls* during this capture. **SHOW PORTS ON MAP** pushes the result onto the MAP workspace as bubbles.

The two lists disagree, and that is the point

Singapore and Rotterdam top the league table on arrivals, but the congestion list is led by Zhoushan, and Chittagong and Fujairah rank far higher there than their arrival counts alone would suggest. A port can be busy without being congested, and congested without being busy — the pairing is what tells you which.

> Congestion bars are coloured by severity rather than by port, so a long red bar is a different signal from a long green one at the same length. Read the colour before comparing two rows by length alone.

13 COUNTRIES — seaborne arrivals

BY COUNTRY · 13

The same arrivals data as the port league table, aggregated to national level and drawn as a choropleth with a ranked list beside it.

Fig 16 — SEABORNE ARRIVALS BY COUNTRY over 30 days: a shaded world map with the ranked league beside it.

Element	What it shows
COLOUR BY	What the map shading encodes — it read *Arrivals* during this capture
World map	Each country shaded by its value; darker means more
RANKED list	The same countries in order, with the value and supporting detail
Row detail	Vessel and port counts beneath each country's name

Japan, China, Indonesia, South Korea and the United States led the ranking during this capture, with values from 10,095 down through the low thousands. The supporting line under each country gives the vessels and distinct ports behind the number, so a country with many arrivals concentrated in a handful of ports is distinguishable from one spreading them widely.

> **Arrivals are not tonnage.** This workspace counts port calls, so a country served by many small coastal vessels can outrank one receiving fewer, far larger ships. For a tonnage-weighted view use the FLEET workspace (§14), which measures deadweight rather than movements.

14 FLEET — tonnage and flag states

Where the previous workspaces count movements, this one measures mass. A treemap partitions the tracked fleet by size band; a ranked list breaks it down by registry.

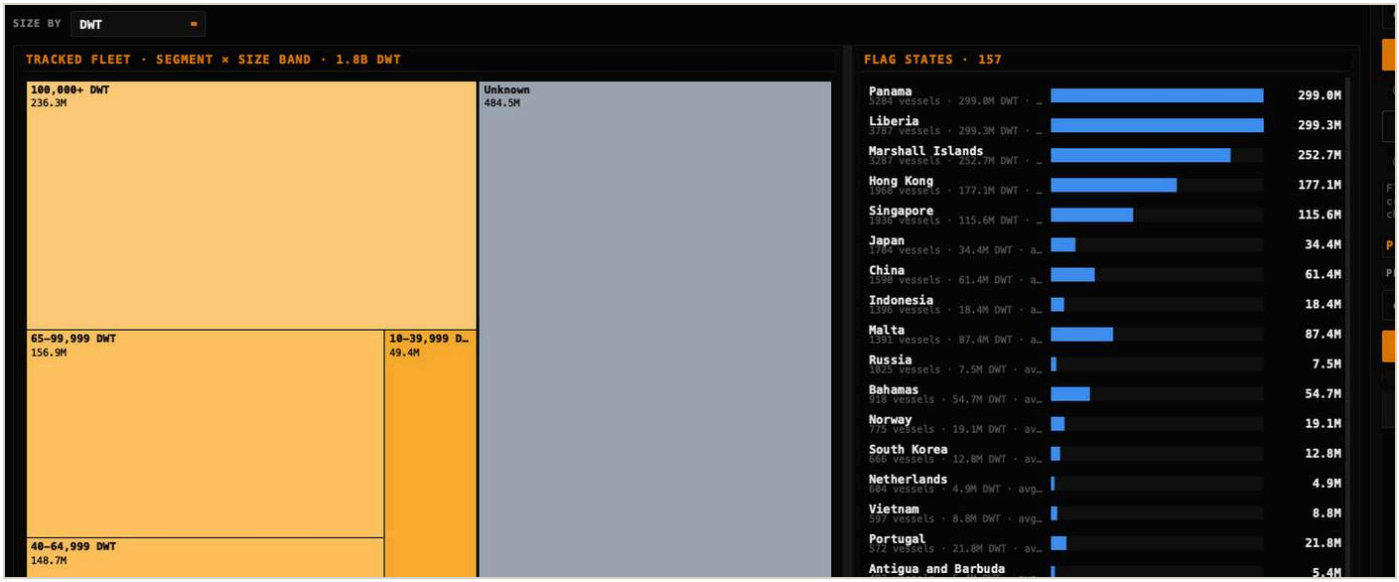


Fig 17 — TRACKED FLEET by size band — 1.8B DWT in total — with FLAG STATES (157 of them) ranked alongside.

Element	What it shows
SIZE BY	What the treemap areas measure — it read *DWT* during this capture
Treemap tiles	One tile per deadweight band, sized by its share of tracked tonnage
Tile label	The band and its total, e.g. `100,000+ DWT  236.3M`
FLAG STATES	Registries ranked by tonnage, with vessel counts and averages

The headline **1.8B DWT** matches the ACTIVE DWT tile in the market pulse (§07), which is a useful consistency check between the two panels. Bands run from the 100,000+ DWT giants down through 65–99,999 and 40–64,999 to 18–39,999, with an **Unknown** tile for vessels whose tonnage the feed does not report.

Panama, Liberia and the Marshall Islands lead the flag ranking — the open registries that dominate world tonnage — followed by Hong Kong, Singapore and Japan.

> **The Unknown tile is large and it matters.** During this capture it held 484.5M DWT — roughly a quarter of the tracked total — meaning any percentage read off the named bands is a share of what is known, not of the fleet. The same caveat applies to the flag ranking, which carries its own Unknown entries.

15 SIGNALS — four fleet indicators

Four time series, each plotted per segment, that turn the market-pulse snapshot into trend. This is the workspace to read when the question is *what changed* rather than *what is*.

Fig 18 — The four indicators, each with a line per segment and a shared legend: all (weighted), Dry Bulk, Crude Tanker, Product Tanker, Container and Gas Carrier.

Indicator	Measures	Reads as
FLEET SPEED INDEX	Median knots of moving vessels	Rising speed means tightening freight; slow steaming means surplus capacity
LADEN SHARE	Share at draught ≥ 85% of maximum	How full the ships that are sailing actually are
IDLE TONNAGE	Deadweight stationary all week	Capacity withdrawn from the market — lay-up or floating storage
UNFIXED TONNAGE	Deadweight for orders with no destination	Ships available but not yet chartered

Each chart carries the same segment legend, so a divergence between classes is legible at a glance — crude tankers idling while containers speed up is a very different market from both moving together. The x-axis spans the selected window; at the 30-day default the captures above run from late February to mid-August, so the series carry more history than the window label implies.

> **These are the most analytically dense panels on the tab**, and the least self-explanatory. Each title carries its own definition as a subtitle — *median kn of moving vessels*, *draught ≥ 85% of max*, *stationary all week*, *FOR ORDERS / no destination* — which is worth reading before drawing a conclusion from the shape.

16 WATCH — standing screens over the fleet

STANDING SCREENS · 16

The last workspace turns an aggregate into a list of actual ships. Pick a standing screen and it enumerates every vessel that currently matches.

WATCHLIST **Idle ≥ 7 days (lay-up / floating storage)** 6472 vessels · 247.7M DWT · Crude Tanker 356 · Dry Bulk 1550 · Unknown 1839 · Product Tanker 950 · Container 465 PIN ON MAP

VESSELS · 1500

Vessel	IMO	Segment	Flag	DWT	Built	Metric	Laden	Detail
FSO ASIA	9224752	Crude Tanker	MHL	432023	2002	22.8 d idle	laden	near · dest
SEA MARANHAO	9575515	Dry Bulk	MHL	403840	2013	21.5 d idle	ballast	near CHN · dest Zhoushan
ERHA FPSO	9280823	Unknown	NGA	375600	2005	396.3 d idle	—	near · dest
YUUM K'AK' NAAB	7708302	Unknown	SGP	360700	1981	69.7 d idle	laden	near · dest
KHALIJ E FARS	9544114	Unknown	TUV	355015	2011	12.0 d idle	laden	near · dest
FSO UOTE 1	9045455	Unknown	LBR	335432	1993	137.1 d idle	laden	near · dest
HL MERCURY	9845142	Unknown	PAN	325452	2019	7.3 d idle	laden	near CHN · dest Ningbo
ORE WUHAN	9847580	Unknown	HKG	324995	2020	9.3 d idle	laden	near · dest
ORE ZHOUSHAN	9847619	Unknown	HKG	324988	2021	8.0 d idle	ballast	near SGP · dest Singapore
HRA	9480837	Crude Tanker	LBR	321300	2011	12.7 d idle	ballast	near MYS · dest Tanjung Pelepas
Phoenix Vanguard	9352559	Crude Tanker	LBR	321300	2007	16.2 d idle	ballast	near PAK · dest Karachi
Front Eminence	9353802	Crude Tanker	MHL	321300	2009	396.4 d idle	laden	near · dest
KIDAN	9519482	Crude Tanker	LBR	321230	2010	14.7 d idle	ballast	near SAU · dest Ras Tanura
MILTIADIS JUNIOR	9693745	Crude Tanker	LBR	320920	2014	11.3 d idle	ballast	near · dest FOR ORDERS
MARAN AQUARIUS	9289491	Crude Tanker	GRC	320820	2005	33.5 d idle	ballast	near ARE · dest Kau
DIJILAH	9829629	Unknown	MHL	320596	2019	8.1 d idle	ballast	near PAK · dest Opol
Boyaca	9590058	Crude Tanker	SGP	320550	2014	10.5 d idle	ballast	near · dest FOR ORDERS

Fig 19 — The *Idle ≥ 7 days* screen: 6,472 vessels, 247.7M DWT, broken down by segment, over a 1,500-row vessel table.

The **WATCHLIST** selector chooses the screen. The line beside it summarises the match — during this capture, `6472 vessels · 247.7M DWT` with a per-segment split of `Crude Tanker 356 · Dry Bulk 1550 · Unknown 1839 · Product Tanker 950 · Container 465`. **PIN ON MAP** pushes the matching vessels onto the map.

Column	Meaning
Vessel	Ship name
IMO	Hull identifier — the number the vessel search takes
Segment	Vessel class, or Unknown where the feed does not say
Flag	Registry, as a three-letter code
DWT	Deadweight tonnage
Built	Year of build
Metric	The value that qualified the ship for this screen — e.g. `22.8 d idle`
Laden	laden or ballast
Detail	Nearest country and declared destination, where known

> **The Metric column changes with the screen.** It holds whatever the selected watchlist measures, so on an idle screen it reads days idle. That makes the table sortable into a worst-first ranking of the thing you actually care about — and it is the only place in the tab where individual hulls are listed beyond the map's 200-pin cap.

17 Credits — what costs what

One number matters more than the rest: **an area search costs 8 credits**. v5.0.1 prints the price directly on the buttons that spend, which earlier builds did not.

Costs credits	Price
Opening the tab for the first time (it loads automatically)	8
GLOBAL VIEW (200 VESSELS)	8
LOAD VESSELS IN AREA	8
↻ REFRESH	8
SEARCH AREA (raw bbox)	8
Each **AUTO ↻** tick, every 5 minutes while the tab is visible	8
Selecting a place-result row, or accepting a typeahead suggestion	8
Finishing a square or circle drag	8
TRACK	1
VOYAGE HISTORY	5
Fleet lookup from the AI tools	3

Free	
SEARCH PLACES · SEARCH PORTS · PORTS IN AREA	the lookups themselves
Clicking a port row	drops a pin, no search
✕ CLEAR · the raw bbox fields · the basemap picker	
Pan, zoom, clicking a vessel pin, selecting a trade corridor	
Every analytics workspace, and every filter-bar change	§04

Caching blunts the meter. Repeating an identical search within 60 seconds replays from cache for free; port and place lookups cache for 7 days. The credit chip in the top bar shows what the marine provider reports as remaining, and it changes colour as it falls — green, then amber under 100, then red under 20.

AUTO ↻ is the expensive one. Left on, it spends 8 credits every five minutes for as long as the tab is visible — about 96 credits an hour — whether or not anything moved. It is off by default; switch away from the tab or toggle it off when you stop watching.

18 Plans — what actually needs Exclusive Pro

PLAN GATE · 18

Two of this tab's endpoints are gated on the **live AIS** capability, which is an Exclusive Pro feature. The rest of the tab is open to every paid tier.

Action	Endpoint	Needs Pro?
TRACK a single hull	`maritime/position`	**Yes** — live AIS
VOYAGE HISTORY	`maritime/history`	**Yes** — live AIS
Area search — every 8-credit load	`maritime/area-search`	No
Multi-vessel positions	`maritime/multi`	No
Feed health	`maritime/health`	No
The six analytics workspaces	analytics service	No

So on a lower tier the map, the area searches, every analytics workspace and both the place and port tools all work. What you lose is the ability to follow one named ship and to replay where it has been.

> **The in-app Docs page for this tab is wrong on exactly this point.** It states that the tab “needs an Exclusive Pro plan; on a lower tier the screen is replaced by an upgrade panel”. There is no screen-level gate — the terminal's feature-tier table is empty for every screen, so Maritime opens on any paid plan. The gate is per-endpoint, as above.

> **A gap in the gate.** Multi-vessel positions return live AIS data and are *not* capability-checked, while the single-vessel position behind TRACK is. That route is not reachable from the screen, but it is reachable from the AI assistant's `get_multi_vessel_positions` tool (§19) — so the assistant can return live positions that the button next to it would refuse.

> The API key that reaches the upstream marine provider no longer ships with the client. It was once a literal in the source — recoverable from any binary — then a per-user secret, and is now held server-side only, with the terminal authenticating as you.

19 Driving the tab from the AI assistant

Seven tools expose this data to the AI assistant and to workflows.

Tool	What it returns	Cost
`search_vessels_by_area`	Vessels inside a bounding box	8
`get_vessel_position`	One vessel's last reported position	1
`get_multi_vessel_positions`	Positions for a list of IMOs	3
`get_vessel_history`	A vessel's recent position reports	5
`geocode_place`	A place name resolved to a bounding box	free
`search_ports`	Ports by name, from Wikidata / MarineRegions / OSM	free
`get_maritime_health`	Feed status and record count	free

Worth knowing

- **The tools cover the map, not the analytics.** All seven address vessels, places and ports; there is no tool for corridors, port congestion, country arrivals, tonnage, signals or watchlists. Those six workspaces are screen-only.
- **They spend the same credits as the buttons.** Asking the assistant to sweep an area is an 8-credit area search, and asking it repeatedly is repeated searches.
- `get_multi_vessel_positions` reaches the ungated route described in §18 — worth knowing whichever side of the gate you are on.
- Nothing here writes. The tab has no destructive action at all: the cost of a mistake is credits, not data.

> The two free lookups make a useful pairing with the paid ones — have the assistant geocode a strait and confirm the bounding box looks right *before* spending 8 credits sweeping it for vessels.

20 Where the data comes from

PROVENANCE · 20

Surface	Source	Freshness
`Vessel positions, area searches`	A commercial AIS feed, proxied through Fincept's backend	Cached 60 s
`Voyage history`	The same feed	Cached 5 min
`Place search`	A geocoder (Nominatim)	Cached 7 days
`Port search`	Wikidata, Marine Regions and OpenStreetMap — the Source column names which	Cached 7 days
`Basemap tiles`	Tile providers, cached to disk	—
`The six analytics workspaces`	A dedicated analytics service in Fincept's backend	Cached per filter

Vessel calls no longer go to the marine provider directly — they are proxied through the terminal's own backend, which holds the upstream key and enforces the plan gate (§18). The client authenticates as you and never sees the provider's credential.

The AIS feed is untidy, and the terminal does not tidy it

Expect blank origin and destination fields, destinations that are free text rather than real port names, vessels reporting 0.0 knots at anchor, and the occasional implausible value. The terminal shows what the feed reports, unmodified. Where more vessels matched than can be drawn, the status bar says so directly — `READY (showing 200 of 2000)`.

> **Provenance of the analytics is stated at the level this guide can verify.** The six workspaces are served by a dedicated analytics service with its own caching and are scoped by the filter bar, which was confirmed by observing the data change with the window (§03). The upstream sources feeding that service were not established for this edition.

> A thin **SYNC** indicator in the filter bar reports whether the cached workspaces match the current filter. If a workspace looks stale after a filter change, that indicator — not the numbers — is the thing to check first.

21 Quick reference and honest notes

> Items marked **[obs]** were established by driving the shipping build rather than by reading its source, because the v5.0.1 analytics code was not available for this edition. Everything unmarked is source-backed.

How do I...

Goal	Do this
Open the tab	Command bar → `maritime`, or Navigate ▸ Research & Intelligence
See ships worldwide	**GLOBAL VIEW (200 VESSELS)** — 8 credits
Search a named sea area	Type it into PLACE NAME, then select the row — 8 credits
Search an exact rectangle	▸ **ADVANCED: RAW BBOX**, fill four fields, SEARCH AREA
Follow one ship	Type its IMO, press **TRACK** — needs Exclusive Pro
See where it has been	**VOYAGE HISTORY** — needs Exclusive Pro
Rank trade corridors	**FLOWS**, then set WEIGHT (§11)
Find congested ports	**PORTS**, right-hand list (§12)
Weigh a market by tonnage	**FLEET** rather than COUNTRIES (§14)
Spot a trend, not a level	**SIGNALS** (§15)
List the ships behind a number	**WATCH** (§16)
Stop spending	Turn **AUTO ↻** off, or switch away from the tab

Honest notes

- **The first view of the tab spends 8 credits** before you touch anything (§17).
- **Selecting a place row spends 8 credits with no button press** — including by arrow key or by accepting a typeahead suggestion (§09).
- **Finishing a draw drag spends 8 credits**, immediately (§05).
- **AUTO ↻ costs about 96 credits an hour** while the tab is visible (§17).
- **Do not sort the place or port tables** — clicks afterwards act on the wrong row, which for places means loading the wrong sea area at 8 credits (§09).
- **LOADED and IN REGION are different on purpose** — a sample against a population (§06).
- **The Trade Corridors "Vessels" column is a vessel count, not trade value**; the feed exposes no dollar volume, and Status is in practice always ACTIVE (§06).
- **The in-app Docs claim the whole tab needs Exclusive Pro. It does not** — the feature-tier table is empty for every screen; only TRACK and VOYAGE HISTORY are gated (§18).
- **Multi-vessel positions are not capability-checked** though single-vessel positions are, and the assistant can reach them (§18).
- **The vessel table is gone from the map workspace** — per-hull listings now live in WATCH **[obs]** (§02).
- **In voyage-history mode the last column holds a date while its header still reads Progress** (§10).
- **The FLEET treemap's Unknown tile held about a quarter of tracked tonnage**, so band percentages are shares of what is known **[obs]** (§14).
- **Market pulse and chokepoint transits ignore the filter bar** — they describe the whole tracked fleet regardless of segment or region **[obs]** (§07).
- **COUNTRIES counts port calls, not tonnage** — small coastal traffic can outrank large ships **[obs]** (§13).
- **Only the IMO field and the four bbox values survive a restart.** Basemap, splitter position, the AUTO ↻ toggle, drawn shapes and table sorting all reset.
- **A place can geocode to a point**, giving a bounding box metres across that finds nothing (§09).
- **No export.** Nothing on this tab can be saved to CSV or printed — only screenshotted.
- **Nothing here is destructive.** The cost of every mistake on this tab is credits, not data.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Prediction Markets

A complete, screen-by-screen walkthrough of the Prediction Markets tab — one order-driven terminal over Polymarket, Kalshi and Fincept's own venue.

`>_ 3 exchanges · 9 views · 10 agent tools · free to browse`

Covers · **every view** · **the detail panel** · **the order ticket** · **the blotter**

What changes when you switch exchange · where each number comes from · what is demo-only

Demonstrated live on the Fincept Terminal Prediction Markets tab.

Contents

23 SECTIONS

GETTING ORIENTED

01 What the Prediction Markets tab is 3
02 Anatomy of the screen 4
03 The command bar 5
04 Choosing an exchange 6

FINDING A MARKET

05 The views, and what each one lists 7
06 Categories, chips and series 8
07 Search 9
08 Sorting, and how to read a market card 10

READING A MARKET

09 The detail panel 11
10 The price chart 12
11 The order book 13
12 Outcomes, status and open interest 14
13 Holders, comments and related markets 15
14 The trades feed 16
15 The leaderboard 17

TRADING

16 Connecting an exchange account 18
17 The order ticket 19
18 The open-orders blotter 20
19 Amending, cancelling and cancel-all 21

REFERENCE

20 Polymarket vs Kalshi — what actually changes 22
21 Driving the tab from the AI assistant 23
22 Where the data comes from, and what it costs 24
23 Quick reference and honest notes 25

About this guide. Every screenshot is a live capture of the Prediction Markets tab in the shipping v5.0.1 build. This tab places **real orders with real money** and does so without a confirmation step. That, and the several controls that do not filter what they appear to filter, are named with evidence rather than described as though they work.

01 What the Prediction Markets tab is

ORIENTATION · 01

A prediction market is a contract that pays **$1 if something happens and $0 if it does not**. Its price is therefore a probability: a contract trading at 62¢ is the market's estimate that the event occurs 62% of the time. The Prediction Markets tab is a terminal over those contracts — browse them, read the book and the history, and, once an account is linked, trade them.

What makes this tab unusual is that it is **not one venue**. The same screen drives three exchanges, and switching between them re-skins and re-populates everything — the accent colour, how prices are written, which views exist, and which panels appear at all.

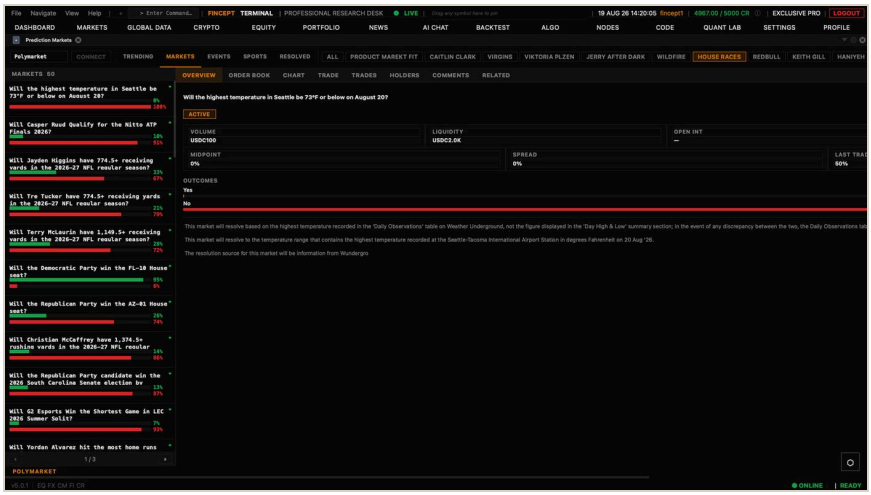


Fig 1 — The tab as it opens: exchange selector and view pills top-left, category chips across the top, the market list on the left, the selected market on the right.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Trading ▸ Prediction Markets**
Command bar	`poly`, `polymarket`, `kalshi` or `prediction`
Main tab bar	**Not present.** This is not one of the fifteen top-level tabs

At a glance, in Prediction Markets you can…

- Browse live contracts on three exchanges
- Switch venue without leaving the screen
- Read the full order book, both sides
- Chart a contract's probability over time
- See every recent trade, timestamped
- See who the largest holders are
- Read the market's resolution rules
- Link a Polymarket or Kalshi account
- Place limit orders as GTC, FOK or FAK
- Amend and cancel resting orders

> **This tab costs no credits.** There is no `guard_capability`, `guard_quota`, `guard_credits` or metered `action::` anywhere in its 11,846 lines. Browsing, charting and streaming are free on every paid tier. The only money involved is the money you trade with.

02 Anatomy of the screen

Thirteen regions. The screen is a three-column splitter — market list, detail panel, and a leaderboard column — under a single command bar.

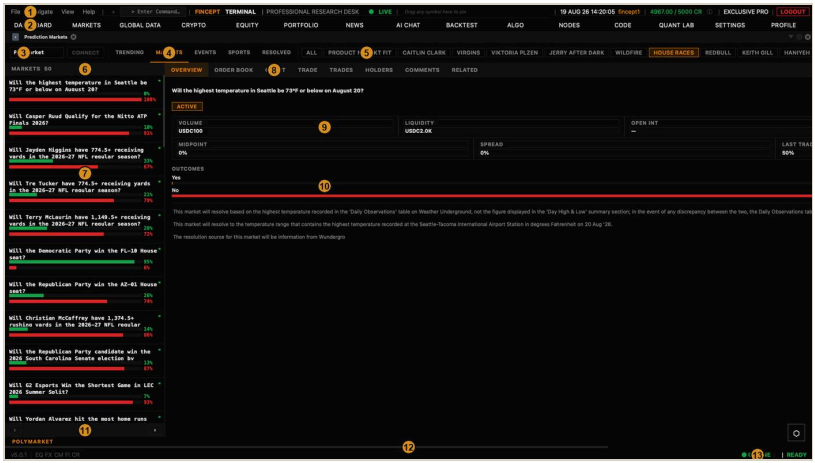


Fig 2 — The thirteen regions, numbered.

#	Region	What it is
1	Terminal menu + command line	Application chrome, not part of this tab
2	Main tab bar	Application chrome. Prediction Markets is not on it
3	Exchange selector + CONNECT	Which venue you are looking at, and account setup (§04, §16)
4	View pills	TRENDING / MARKETS / EVENTS / SPORTS / RESOLVED on Polymarket (§05)
5	Category chips	The active exchange's tag taxonomy (§06)
6	Browse panel header	`MARKETS 50` — the count currently listed
7	Market cards	One card per contract, with its outcome bars (§08)
8	Detail tab strip	OVERVIEW / ORDER BOOK / CHART / TRADE / TRADES / HOLDERS / COMMENTS / RELATED
9	Stat boxes	Volume, liquidity, open interest, midpoint, spread, last trade, end date (§12)
10	Outcome bars	Each outcome's current probability (§12)
11	Pagination	`1 / 3` with ◀ ▶ — pages through the listed markets
12	Horizontal scrollbar	The screen is wider than the window — see the warning below
13	Status bar	Connection and readiness for the whole terminal

> **The screen does not fit a 1920-pixel window.** The scrollbar at region 12 covers about three quarters of the width, so roughly 650 pixels sit off the right edge. At rest that hides the **search box**, the **sort controls**, the **● LIVE** websocket indicator, the trailing category chips and part of the END DATE and LAST TRADE stat boxes. None of them are reachable until you scroll the entire screen sideways.

03 The command bar

Everything that changes *what is listed* lives in one 28-pixel strip across the top. It is too wide to photograph in one piece, so here it is in three parts, left to right.

Fig 3 — Left: the exchange selector, CONNECT, and the view pills.

Fig 4 — Middle: the category chips, capped at twelve visible.

Fig 5 — Right, reachable only by scrolling: search, sort, page size, the percent toggle, refresh and the live-connection lamp.

Control	Type	What it does
`Exchange selector`	Dropdown	Switches venue. Drops the current adapter's connections, clears the UI and reloads (§04)
`CONNECT`	Button	Opens the Connect Account dialog. Tooltip *"Connect a trading account"* (§16)
`View pills`	Toggle row	Which list is fetched. Membership is per-exchange (§05)
`Category chips`	Toggle row	Polymarket and Fincept Internal. **Do not filter** — see §06
`ALL SERIES`	Dropdown	Kalshi only. Replaces the chips because its series catalog exceeds 100 entries
`Search box`	Text field	Placeholder *Search markets...*, submits on Return. **Does not filter** — see §07
`Sort`	Dropdown	VOLUME / LIQUIDITY / DATE. Ignored by TRENDING, and by every Kalshi view (§08)
`Page size`	Spin box	How many rows are requested per page
`%`	Toggle	Switches price rendering between percent and the exchange's own style
`↻`	Button	Refresh. Re-runs the current view's request
`● LIVE / ○ OFF`	Indicator	Websocket state. LIVE means prices and book updates are streaming

> The bar also carries a **60-second auto-refresh timer** that re-runs the current view while the tab is visible, and stops when you leave it.

04 Choosing an exchange

The dropdown at the far left is the most consequential control on the screen. Three venues are registered at start-up, and the one you pick changes far more than the data.

Fig 6 — All three registered exchanges.

Exchange	What it is	Account needed to browse?
`Polymarket`	USDC-settled markets on Polygon. The default	No
`Kalshi`	A CFTC-regulated US exchange, USD-settled	No
`Fincept Internal`	Fincept's own venue — **demo data only** today	No

Switching is a hard reset, not a filter. The screen disconnects the previous adapter's signals, clears the selection, reinstalls a per-exchange presentation profile across every sub-panel, and re-requests the list from scratch.

What the presentation profile changes

Fig 7 — Polymarket: amber, five views, chips.

Fig 8 — Kalshi: teal, four views, an ALL SERIES dropdown.

Fig 9 — Fincept Internal: Polymarket's amber and Polymarket's five views.

> **Fincept Internal is wearing Polymarket's clothes.** The profile lookup returns the Polymarket profile for *anything that is not Kalshi*, so the third venue inherits amber, prices in **USDC**, Polymarket's five view pills, and the HOLDERS, COMMENTS and RELATED tabs — none of which its adapter has endpoints for. Its own currency is not USDC and its own catalog is three demo markets.

05 The views, and what each one lists

The pills choose which request is made. On Polymarket there are five. Two of them do not do what their labels promise, and this is the single most important thing to know about the tab.

> **RESOLVED is documented as unfinished in the product's own source** — *"the adapter interface doesn't expose a closed-markets filter today"* — and it then asks for events with closed set to false. No combination of controls makes it show a settled market.

Fig 10 — MARKETS

Fig 11 — TRENDING

Fig 12 — SPORTS

The same three markets, in the same order, in all three views — a Seattle temperature contract, a Spanish football match and an ATP Finals qualifier.

View	What it requests	Verdict
MARKETS	Open markets, your category and sort applied	Works
TRENDING	Open markets, no category, sort forced to volume	**Identical to MARKETS** at the defaults — it discards both your category and your sort
EVENTS	Open events (a group of related markets)	Works, but ignores your category
SPORTS	Open markets tagged `sports`	**Unfiltered.** The tag never reaches the data — §06
RESOLVED	Events, sorted by end date, **with closed=false**	**Cannot show a resolved market.** It explicitly asks for non-closed events

Fig 13 — EVENTS: a genuinely different list.

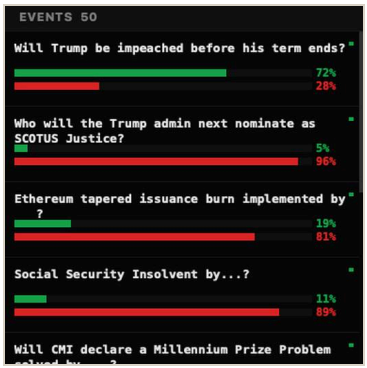


Fig 14 — RESOLVED: also events, and none resolved.

06 Categories, chips and series

Below the view pills sits the active exchange's taxonomy. Polymarket and Fincept Internal render it as chips, capped at twelve visible; Kalshi renders a dropdown instead, because its series catalog runs past a hundred entries and a chip row does not scale to that.

Fig 15 — CAITLIN CLARK selected. The chip highlights and the list re-queries.

The chips are populated live from the exchange, so what you see changes through the day — during this capture they included `PRODUCT MAREKT FIT` (the typo is in the exchange's own tag, not in the terminal), `CAITLIN CLARK`, `HOUSE RACES` and `EUROPA LEAGUE`.

Fig 16 — The result of selecting CAITLIN CLARK: fifty markets about Seattle weather, Spanish football and NFL receiving yards.

> **Category chips do not filter anything.** Selecting one issues a request carrying a `tag` parameter that the exchange's API does not honour. Verified directly against the live endpoint: `tag=CAITLIN CLARK`, `tag=sports`, `tag=politics`, `tag=caitlin-clark` and **no tag at all** return the same fifty markets in the same order. It is not a formatting problem that a different spelling would fix — the parameter is simply ignored.

> Two consequences worth holding on to. First, the **SPORTS view** is the same mechanism with a hardcoded tag, so it is unfiltered for the same reason (§05). Second, on **EVENTS** and **RESOLVED** the category is discarded a step earlier — in the terminal, before the request is even built.

07 Search

The search box sits to the right of the category row, takes a free-text query and submits on Return. An empty box restores the current view.

Fig 17 — The query is accepted and held in the field.

> **Search does not narrow the list either.** The box calls an endpoint with a `_q` parameter that the exchange ignores. Verified against the live API: a query of `zzzzqqqnonsense` returns the same fifty markets as no query at all. On screen, typing `fed` and pressing Return left the header reading `MARKETS 50` with the list unchanged.

The working search exists but is not wired up

The service layer contains a second, correct implementation that calls the exchange's real search endpoint and returns both markets and events. Nothing calls it — the search box is routed to the other one. Repairing search is therefore a wiring change rather than new work, though the proper endpoint was also returning an error at the time of this capture, so it is necessary and not sufficient.

What you might expect	What actually happens
Typing a phrase narrows the market list	The list is unchanged
A nonsense query returns nothing	It returns the full default page
The count reflects matches	The count stays at the page size
Clearing the box restores the view	This part works

> Until this is fixed, the reliable way to find a specific contract is the **agent tools** (§21) — `search_polymarket` runs server-side and is not affected by this screen's query parameter.

08 Sorting, and how to read a market card

Sort

Option	Orders by	Honoured?
`VOLUME`	Total traded value, descending. The default	Polymarket only
`LIQUIDITY`	Resting depth in the book	Polymarket only
`DATE`	Market end date	Polymarket only

> **Sort is inert on Kalshi** — its adapter accepts the parameter and discards it, along with the page offset; pagination there is marked as later polish in the source. It is also discarded by Polymarket's **TRENDING** view, which forces volume (§05).

The market card

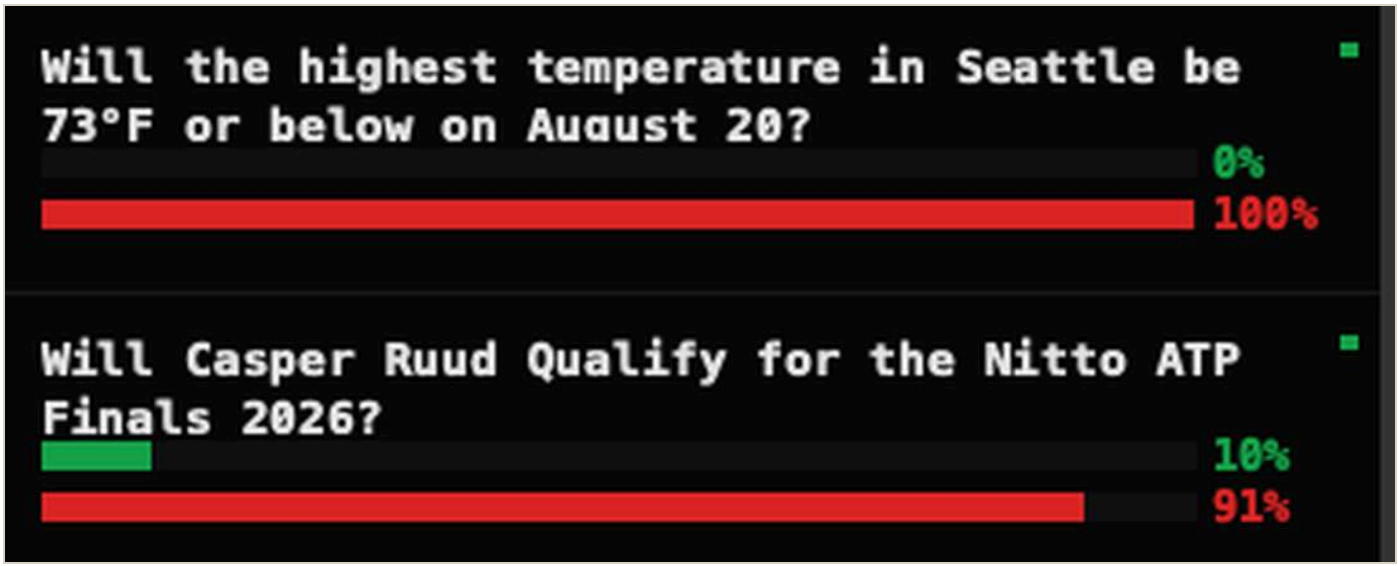


Fig 18 — One card: the question, then one bar per outcome with its probability.

Element	Meaning
Question	The contract's resolution question, verbatim from the exchange
Green bar + %	The **Yes** outcome's current probability
Red bar + %	The **No** outcome's current probability
Bar length	Proportional to that probability — the two normally sum to 100%
Small square, top-right	Live-status dot for the contract

Clicking a card selects it: the detail panel rebuilds, the terminal subscribes to that contract's websocket topics for streaming price and book updates, and the previous selection is unsubscribed. Paging with ◀ ▶ at the foot of the list moves through the result set — `1 / 3` in the captures above.

> **Selection can move on its own.** Re-querying — by changing category, sorting, or the 60-second auto-refresh — rebuilds the list and can leave a different contract selected in the detail panel than the one you were reading. Check the question at the top of the detail panel before acting on what it shows.

09 The detail panel

Everything about the selected contract lives in the right-hand panel, behind eight tabs. Three of them are Polymarket-only and disappear on Kalshi.

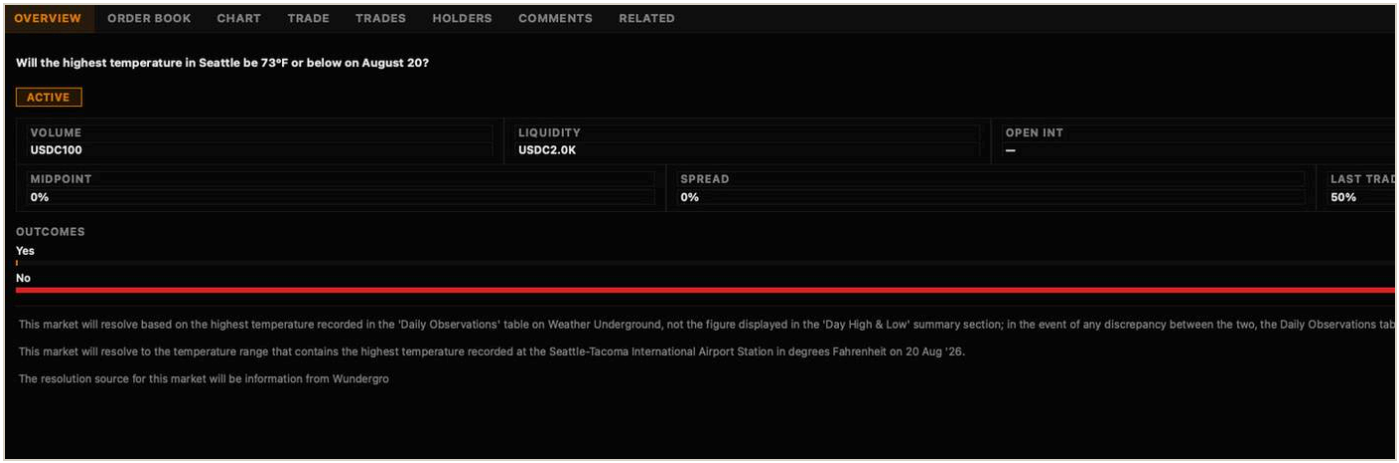


Fig 19 — OVERVIEW: status badge, the stat grid, the outcome bars and the market's resolution rules in full.

Tab	Shows	Available on
`OVERVIEW`	Status, stats, outcomes, resolution rules	All three
`ORDER BOOK`	Resting bids and asks with cumulative totals	All three
`CHART`	Price history with interval and outcome selectors	All three
`TRADE`	The order ticket, balance and current position	All three
`TRADES`	Recent executions, newest first	All three
`HOLDERS`	Largest position holders by size	Polymarket only
`COMMENTS`	The exchange's public comment thread	Polymarket only
`RELATED`	Other markets in the same event	Polymarket only

TRADE and TRADES sit next to each other and are one letter apart. TRADE is the live order ticket that submits real orders; TRADES is a read-only list of what has already happened. They are adjacent in the strip with no visual separation.

On Kalshi the last three tabs are hidden outright rather than shown empty — the presentation profile removes them. On **Fincept Internal** they are shown and cannot be filled (§04).

10 The price chart

CHART plots the selected outcome's price over time, with two controls above it.

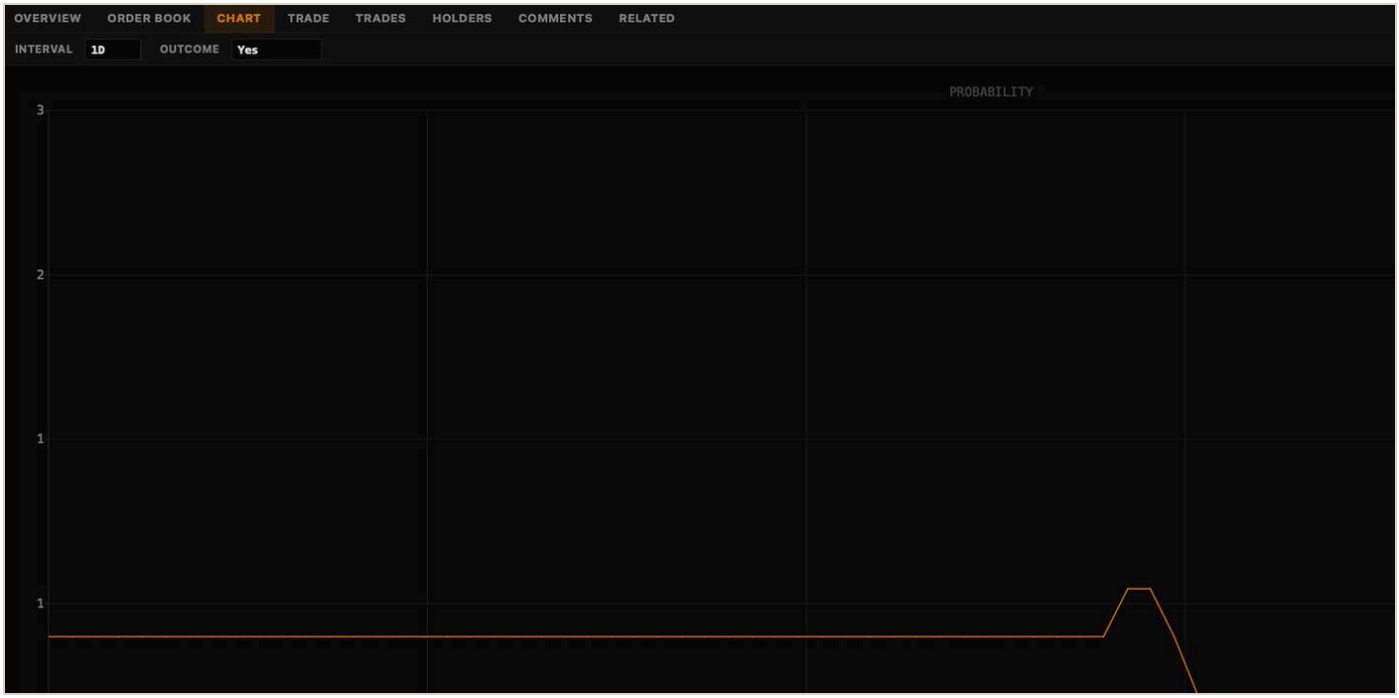


Fig 20 — The chart with its INTERVAL and OUTCOME selectors.

Control	Options	What it does
`INTERVAL`	1H · 6H · 1D · 1W · 1M · MAX	How far back the history runs, and how coarse the points are
`OUTCOME`	The contract's outcomes	Which outcome's line is drawn — Yes by default
`Price label`	Right of the toolbar	The current price, in the exchange's own price style

The series is drawn in the active exchange's accent colour, so the chart re-tints amber or teal with the venue. When a contract has no history the panel says *No price history available* rather than drawing an empty grid.

> **The y-axis is labelled PROBABILITY but plotted in cents.** The series is multiplied by 100 before being handed to the chart, while the axis caption comes from the exchange profile — which reads “PROBABILITY” for Polymarket. Label and units disagree. On a contract trading under 1% this produced an axis reading 3 · 2 · 1 · 1 · 0, with a duplicated tick, above a line that is genuinely near zero.

> Read the number in the **price label** or on the outcome bars rather than off the axis. Those two agree with the exchange.

11 The order book

ORDER BOOK shows resting orders on both sides of the selected outcome, with the spread in the top-right corner. It updates over the websocket while the connection lamp reads ● **LIVE**.

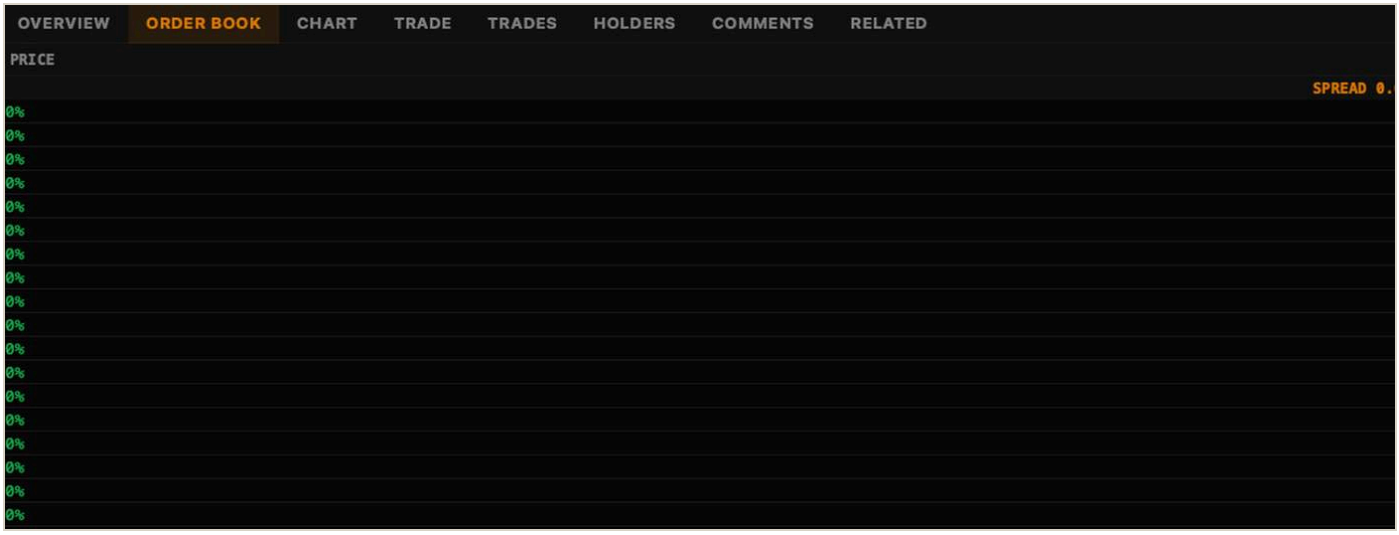


Fig 21 — The book for a contract trading below one cent.

Column	Meaning
PRICE	The level's price
SIZE	Contracts resting at that level
TOTAL	Cumulative size from the top of the book down to this level
SPREAD	Best ask minus best bid, top-right

Bids are drawn in green below the midpoint and asks in red above it, so the shape of supply and demand is legible without reading the numbers. An empty book shows *No order book data*.

> **The book rounds every level to a whole percent.** On a contract priced under a cent that renders roughly twenty rows all reading `0%`, with the spread as `0.` — the depth is real but the display cannot distinguish the levels. Prediction markets spend a lot of their life in the tails, so this affects more contracts than it sounds like it would. For sub-cent contracts, read the **TRADES** tab instead: it prints four decimal places (§14).

12 Outcomes, status and open interest

The status badge

Badge	Meaning	Where
`ACTIVE`	Accepting orders	All exchanges
`RESOLVED`	Settled — no further trading	All exchanges
`INACTIVE`	Not currently trading	All exchanges
`OPEN`	Open for trading, with the close time in the tooltip	Kalshi
`PENDING`	Not yet opened	Kalshi
`PAUSED`	Halted by the exchange; orders refused until reopened	Kalshi
`CLOSED`	Trading halted, awaiting settlement	Kalshi
`SETTLED`	Outcome determined and payouts processed	Kalshi

The stat grid

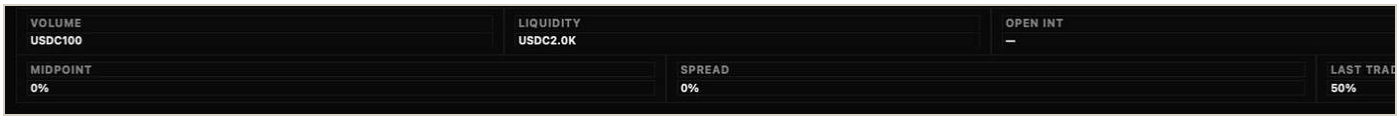


Fig 22 — Volume, liquidity and open interest.

Stat	Meaning
VOLUME	Total value traded on the contract
LIQUIDITY	Resting depth available in the book
OPEN INT	Contracts outstanding — a dash when the exchange does not supply it
MIDPOINT	Halfway between best bid and best ask
SPREAD	Best ask minus best bid
LAST TRADE	Price of the most recent execution
END DATE	When the contract closes. Off-screen at 1920px — see §02

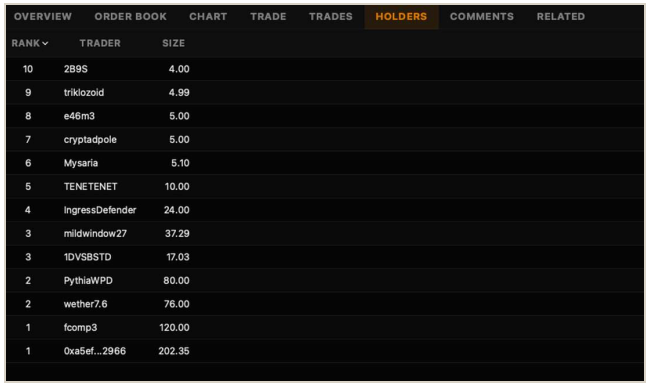
Fig 23 — The outcome bars, with the percentage printed twice.

> Two rendering faults in this block. **Volume and liquidity are prefixed with the currency code instead of a symbol** — they read `USDC100` and `USDC1.2M` where `$100` was intended; the source comment beside the formatter claims a dollar prefix that the code does not produce. And the **outcome bars print their percentage twice** (`38% 38%`), while the leading outcome's bar prints no number at all.

13 Holders, comments and related markets

Three Polymarket-only tabs. On Kalshi they are removed from the strip; on Fincept Internal they are present and permanently empty.

HOLDERS



RANK	TRADER	SIZE
10	2B9S	4.00
9	triklozoid	4.99
8	e46m3	5.00
7	cryptadpole	5.00
6	Mysaria	5.10
5	TENETENET	10.00
4	IngressDefender	24.00
3	mildwindow27	37.29
3	1DVSBSTD	17.03
2	PythiaWPD	80.00
2	wether7.6	76.00
1	fcomp3	120.00
1	0xa5ef...2966	202.35

Fig 24 — The largest holders of the selected contract.

Column	Meaning
RANK	Position in the holder list
TRADER	Display name, or a truncated wallet address when unnamed
SIZE	Contracts held
AVG PRICE	Average price paid

> **The RANK column repeats itself.** The capture above shows two rows ranked 3 and two ranked 1, with sizes that do not decrease monotonically — 37.29 sits above 17.03 at the same rank. The behaviour is consistent with the Yes-holder and No-holder lists being concatenated into one table, each carrying its own 1..N ranking. Read the SIZE column, not the rank.

COMMENTS and RELATED

Fig 25 — COMMENTS, with nothing to show.

Fig 26 — RELATED, likewise.

COMMENTS carries the exchange's public thread, with a share and like count per entry; RELATED lists the other contracts belonging to the same event, and clicking one selects it. Both were empty throughout this capture — the comments request was also returning a transport error, visible as a toast in §22.

> On Kalshi, the underlying calls are refused with an explicit message rather than left blank: *“Comments are Polymarket-only”* and *“Related markets are Polymarket-only”*.

14 The trades feed

TRADES is the tape — recent executions on the selected contract, newest first. It is the most reliable price display on the screen.

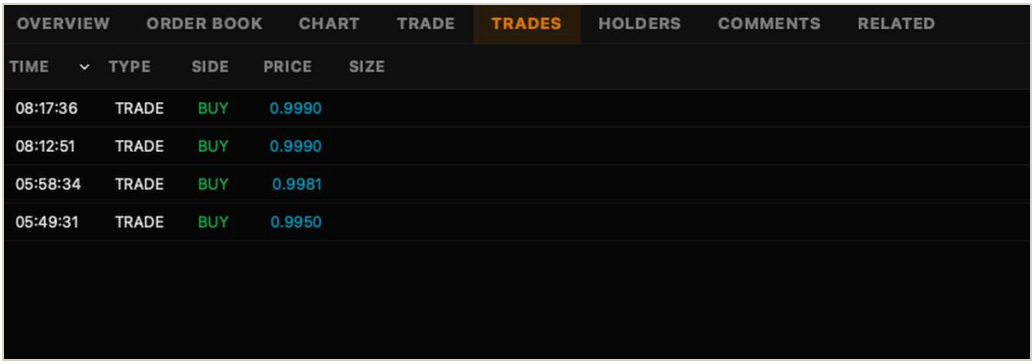


Fig 27 — Four executions, with prices to four decimal places.

Column	Meaning
TIME	Execution time, local
TYPE	TRADE for an execution
SIDE	BUY or SELL, coloured
PRICE	Execution price, to four decimals
SIZE	Contracts traded

On Kalshi the feed is also fed by the websocket, so executions appear as they print rather than only on refresh.

> **The same price appears in four different formats across this screen.** The outcome bars say `100%`, the order book says `0%`, this tab says `0.9990` and the chart axis is in cents. All four describe the same underlying 0–1 probability. Nothing on screen tells you they are the same quantity in different clothes.

> Because this tab prints four decimals, it is the only place a contract trading below one cent can be read accurately. The 0.9990 above is the **No** side of a contract whose **Yes** side the order book was rendering as a column of zeros (§11).

15 The leaderboard

The screen has a third column, to the right of the detail panel, holding a trader leaderboard. Polymarket advertises the capability and the terminal asks for the panel to be shown.

Column	Meaning
#	Rank
TRADER	Display name or wallet
VOLUME	Traded value over the ranking window
PNL	Profit and loss over the same window
TRADES	Number of trades

> **The leaderboard cannot be seen.** It is built hidden and the splitter that owns it allocates it a width of **zero**, with no stretch — so when the screen later asks for it to be shown, it is shown at nothing wide. Its splitter handle would be the way to drag it open, but that handle sits beyond the right edge of the window, because the command bar already forces the screen wider than 1920 pixels (§02). At the default window size there is no sequence of actions that reveals this panel.

The data behind it is real and reachable — the same leaderboard is exposed to the AI assistant as `get_polymarket_leaderboard`, which returns it regardless of what the screen can display (§21).

> On **Kalshi** the panel is correctly absent: the exchange publishes no public leaderboard, the adapter does not advertise one, and the column is not requested. On **Fincept Internal** it is advertised, because that venue inherits Polymarket's profile (§04).

16 Connecting an exchange account

CONNECT opens a two-tab dialog, one tab per tradeable exchange. Its first line is the important one: linking an account is optional, and everything covered so far works without it.

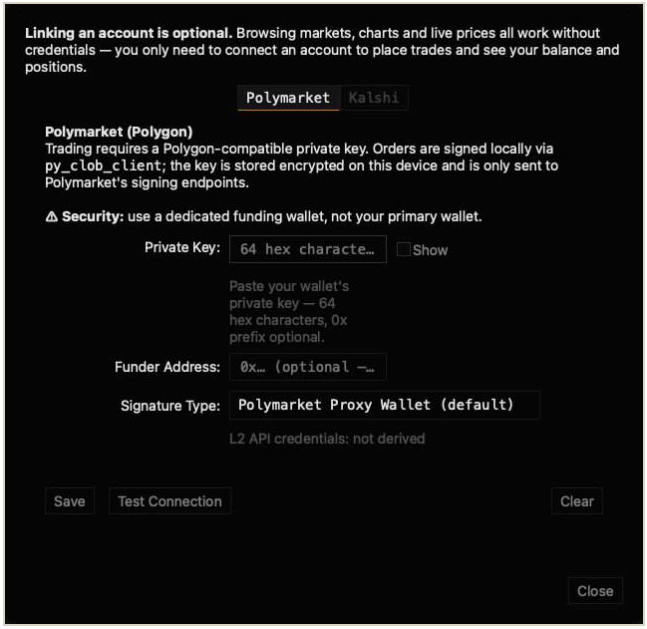


Fig 28 — The Polymarket tab.

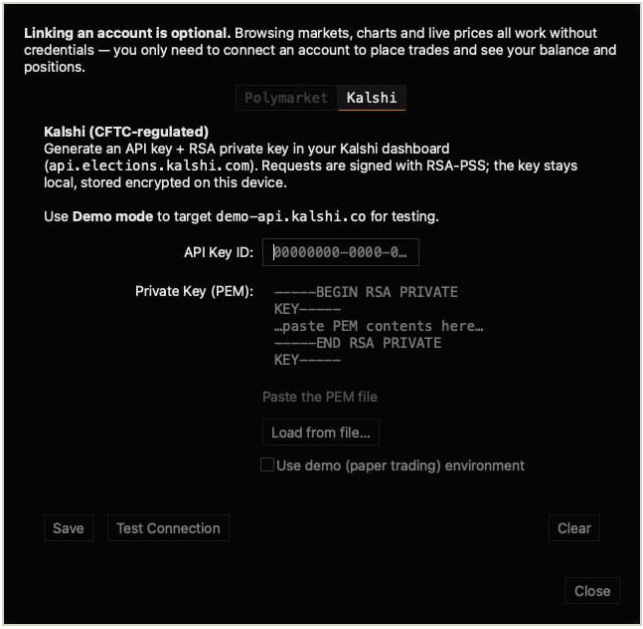


Fig 29 — The Kalshi tab.

Exchange	Field	What it wants
`Polymarket`	**Private Key**	A 64-character Polygon private key, 0x prefix optional. **Show** reveals it
`Polymarket`	**Funder Address**	Optional — the wallet funding the trades
`Polymarket`	**Signature Type**	Polymarket Proxy Wallet (default), or an alternative scheme
`Kalshi`	**API Key ID**	The key id generated in your Kalshi dashboard
`Kalshi`	**Private Key (PEM)**	The matching RSA private key; **Load from file** reads a .pem
`Kalshi`	**Use demo environment**	Points the client at Kalshi's paper-trading server

Button	What it does
Save	Writes the credentials to the OS keychain and reloads the adapter
Test Connection	Asks the adapter for a balance. **A stub in this build** — it raises a request the screen does not yet answer
Clear	Removes the stored entry for that exchange
Close	Dismisses the dialog

> **Keys never leave the device except to the exchange.** They are stored encrypted in the OS keychain, not in a settings file and not on Fincept's servers. Polymarket orders are signed locally; Kalshi requests are signed with RSA-PSS locally. The Polymarket tab carries its own warning, and it is good advice: **use a dedicated funding wallet, not your primary one**.

17 The order ticket

The TRADE tab holds the ticket. Without credentials it holds nothing else.

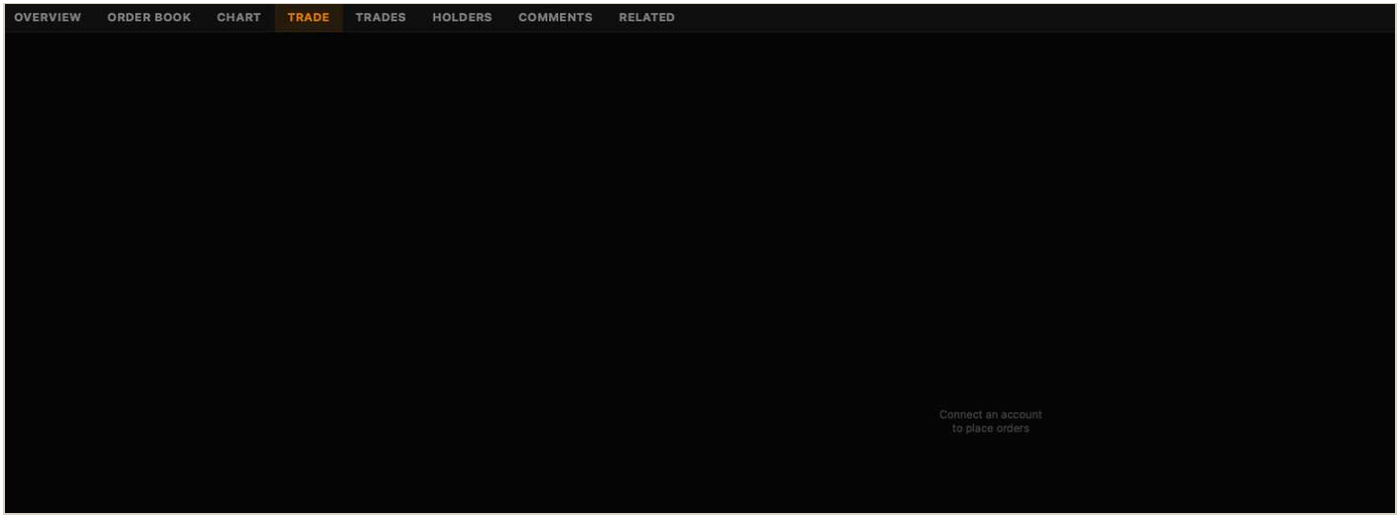


Fig 30 — TRADE with no account linked: the ticket is unreachable.

Control	Type	What it does
`BUY / SELL`	Toggle	Which side you are taking. The submit button re-colours to match
`OUTCOME`	Dropdown	Which outcome you are trading — Yes or No, or one of many on a multi-outcome market
`ORDER TYPE`	Dropdown	**GTC** rests until cancelled · **FOK** fills completely or not at all · **FAK** fills what it can and cancels the rest
`PRICE (0–1)`	Text field	Your limit price as a probability. Must be strictly between 0 and 1
`SIZE`	Text field	Number of contracts. Must be greater than 0
`AVAILABLE`	Label	Your spendable balance on the active exchange
`POSITION`	Label	Your current holding in this contract, or *No position*
`PLACE ORDER`	Button	Submits. See the warning below

> **PLACE ORDER submits immediately. There is no confirmation step.** The button checks that the price is between 0 and 1 and the size is above 0, sets the status line to *Submitting…*, and sends the order to the live exchange. There is no summary, no are-you-sure, and no undo — a resting order can be cancelled afterwards (§19), but a marketable one is filled before you can reach for it. Check the side, the outcome and the size against the question at the top of the panel **before** you click.

Validation failures are reported in place, in red, under the fields: *Invalid price — enter a value between 0 and 1* and *Invalid size — must be > 0*. A success replaces the status line with *Order placed* ✓ and the exchange's order id.

> **Practise on Kalshi's demo environment.** Ticking *Use demo (paper trading) environment* in the account dialog (§16) points the whole Kalshi client at the exchange's paper server, where the ticket behaves identically with no money at risk. It is the only safe way to rehearse this screen. Fincept Internal refuses orders outright with *markets are in demo mode*, so it exercises the UI but not the flow.

18 The open-orders blotter

Below the detail panel sits a horizontal **OPEN ORDERS** blotter listing everything you have resting on the active exchange. It is hidden until credentials are saved, and appears automatically once they are — which is why it does not appear anywhere in this guide's captures.

Column	Meaning
TIME	When the order was placed
MARKET	The contract's question — the stretching column
SIDE	BUY or SELL
OUTCOME	Which outcome the order is on
TYPE	GTC, FOK or FAK
PRICE	Your limit price
SIZE	Contracts ordered
FILLED	How much has executed so far
STATUS	The exchange's current state for the order
ACTIONS	Per-row refresh and cancel controls

The panel opens at roughly a quarter of the detail column's height and can be dragged taller by its splitter handle. Rows are selectable one at a time, and the table itself is not directly editable — amendment goes through the double-click flow in §19.

> The blotter reflects **the active exchange only**. Orders resting on Polymarket are not visible while Kalshi is selected, and vice versa — switching exchange reloads the blotter along with everything else. If you trade both venues, check both.

19 Amending, cancelling and cancel-all

Three operations act on resting orders, all from the blotter.

Action	How	What happens
`Refresh order state`	The per-row refresh control	Re-reads that one order from the exchange, so a partial fill or an exchange-side cancellation shows
`Amend price`	**Double-click the row**	Opens a prompt — *New limit price (0.01 – 0.99)* — and re-prices the order in place
`Cancel order`	The per-row cancel control	Cancels that single order
`CANCEL ALL`	The button in the blotter header	Cancels every order currently listed, on the active exchange

> The amend affordance is discoverable only by tooltip — the row carries *"Double-click to amend price"*, and there is no visible button for it. The prompt constrains the new price to 0.01–0.99, which is tighter than the order ticket's own rule of strictly between 0 and 1.

> **CANCEL ALL is not scoped to what you can see.** It acts on every order in the blotter, not the selected row and not a filtered subset, and — like the order ticket — it does not ask twice. It is also exchange-scoped: it will not touch orders resting on the venue you are not currently looking at, which cuts both ways.

Amend is not the same operation on both exchanges

Kalshi exposes a native amend and a native batch-cancel, and the terminal uses them directly. Where an exchange has no equivalent, the same buttons are serviced by cancelling and replacing, so an amended order can lose its place in the queue at its price level. If queue position matters to your strategy, treat an amend as a new order.

20 Polymarket vs Kalshi — what actually changes

EXCHANGE DIFFS · 20

	Polymarket	Kalshi	Fincept Internal
Accent	Amber	Teal	Amber (inherited)
Settlement	USDC on Polygon	USD, CFTC-regulated	Demo only
Price style	`52¢`	`$0.52`	`52¢` (inherited)
Views	5 — incl. TRENDING, SPORTS	4 — incl. SETTLED, HISTORY	Polymarket's 5
Categories	Chips, 12 visible	ALL SERIES dropdown	Chips
Holders / Comments / Related	Yes	Hidden	Shown, never filled
Leaderboard	Advertised (§15)	No	Advertised
Sort honoured	Yes	**No**	Yes
Trade websocket	No	Yes	No
Paper trading	No	**Yes** (§16)	Everything is paper
Credentials	Polygon private key	API key + RSA PEM	None

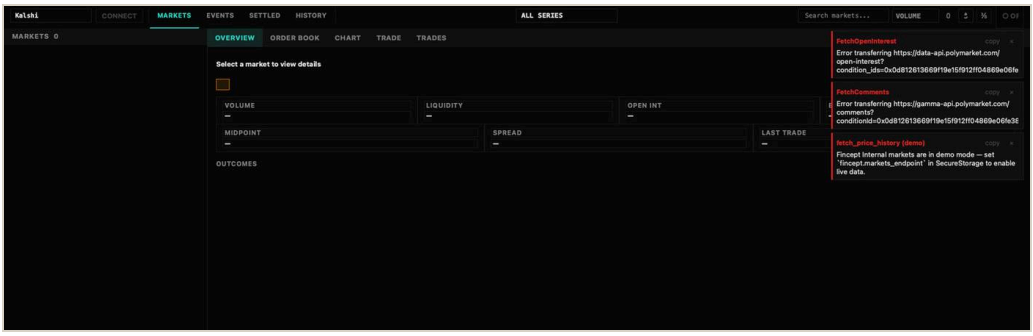


Fig 31 — Kalshi: teal chrome, four views, a series dropdown, three fewer tabs — and an empty market list.

> **Kalshi returns nothing in this build.** Both MARKETS and EVENTS report **0** and stay there — waited twenty seconds on one and fourteen on the other — with no error message of any kind. This is not an account problem: browsing Kalshi needs no credentials and the requests are sent unauthenticated by design. Nor is it the exchange: the same request issued by hand, with the same user-agent, returns a full list of open markets. The failure is between the response arriving and the list being drawn, and it is silent. **Treat the Kalshi option as non-functional until this is fixed.**

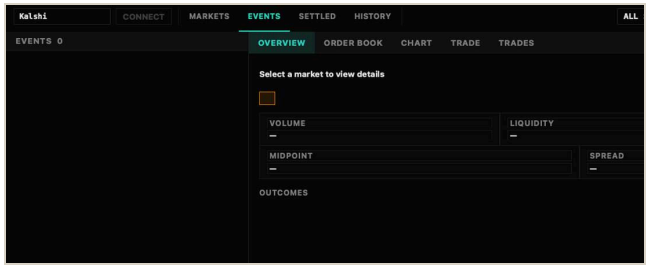


Fig 32 — EVENTS on Kalshi: the same empty result, no error shown.

21 Driving the tab from the AI assistant

Ten tools expose this data to the AI assistant and to workflows. They run against the service layer directly, which means **they are not affected by the screen's broken filters** — the search tool in particular does work.

Tool	What it returns
`list_polymarket_markets`	A page of open markets, sorted
`search_polymarket`	Markets matching a query
`get_polymarket_market`	One market in full, by id
`list_polymarket_events`	A page of open events
`get_polymarket_order_book`	The full book for an outcome
`get_polymarket_price_history`	The time series behind the chart
`get_polymarket_price_summary`	Midpoint, spread and last trade
`get_polymarket_trades`	Recent executions
`get_polymarket_leaderboard`	The trader leaderboard the screen cannot show (§15)
`list_polymarket_tags`	The category taxonomy

> **All ten are Polymarket-only.** Every name carries the venue, and there is no Kalshi or Fincept Internal tool surface — the assistant cannot see or act on those exchanges at all.

Worth knowing

- **No tool places an order.** The set is entirely read-only; trading is a screen-only action, deliberately.
- They cost no credits, like the rest of the tab.
- `search_polymarket` is the practical workaround for §07 — ask the assistant to find a contract by name rather than fighting the search box.
- `get_polymarket_leaderboard` is the only way to read the leaderboard at all, given §15.

22 Where the data comes from, and what it costs

Surface	Source	Credits
`Market and event lists`	The exchange's public API, direct from the terminal	None
`Order book`	Exchange REST, then live over websocket	None
`Price history`	Exchange REST	None
`Trades`	Exchange REST, plus websocket on Kalshi	None
`Holders / comments / related`	Polymarket data and gamma endpoints	None
`Leaderboard`	Polymarket data endpoint	None
`Order placement`	Signed locally, sent to the exchange	None — you pay the exchange
`Credentials`	OS keychain on this device	None

Nothing on this screen is proxied through Fincept's servers, which is why it works with no subscription checks and no metering — and also why an exchange outage shows up here as raw transport errors rather than a tidy Fincept message.

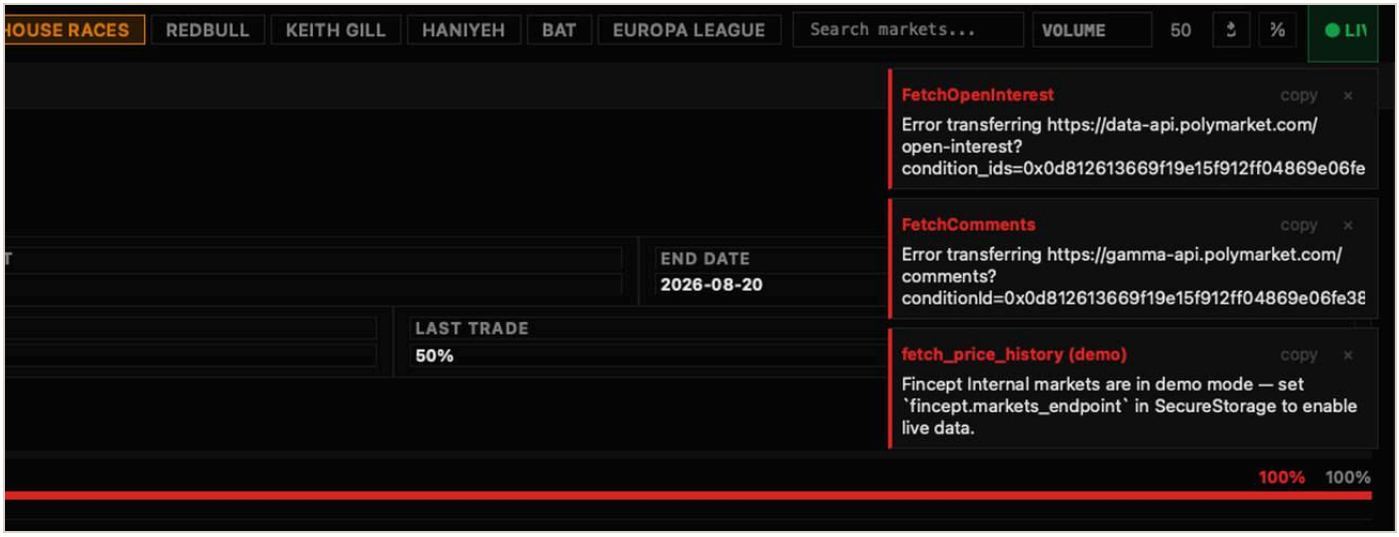


Fig 33 — Errors surface as stacked toasts in the top-right corner.

> **Error toasts do not clear themselves, and they survive an exchange switch.** The two Polymarket transport errors above stayed on screen through a switch to Kalshi, then to Fincept Internal, then back — and were *joined* by, not replaced by, that venue's demo-mode notice. A stack of toasts on this screen may describe an exchange you are no longer looking at. Dismiss them with the × and re-read what remains.

> **Fincept Internal** is the exception to everything in the table above: it serves a curated three-market demo set from inside the terminal and reaches no network at all until an endpoint is configured for it. Every write it receives answers *"markets are in demo mode"*.

23 Quick reference and honest notes

> Every item on this page was read from the shipping source and confirmed on screen on v5.0.1; the two filtering faults were additionally proven against the exchange's live API. A guide that described the chips and the search box as working would send you hunting for a filtered list this screen has never once produced.

How do I...

Goal	Do this
Open the tab	**Navigate ‣ Trading ‣ Prediction Markets**, or type `poly`
Change venue	The dropdown at the far left of the command bar
Find a specific contract	Ask the AI assistant — the search box does not filter (§07)
Read a sub-cent price	The **TRADES** tab, which prints four decimals (§14)
See the resolution rules	**OVERVIEW**, below the outcome bars
Reach the search and sort controls	Scroll the whole screen sideways (§02)
Link an account	**CONNECT**, then Save (§16)
Practise trading safely	Kalshi's *demo (paper trading)* checkbox (§16)
Cancel one order	The row's cancel control in the blotter (§19)
Re-price an order	Double-click its blotter row (§19)
See the leaderboard	`get_polymarket_leaderboard` via the assistant (§15)

Honest notes

Each entry names a control that does not behave the way the screen implies, with the section that shows the evidence.

- **PLACE ORDER has no confirmation step** and sends to the live exchange (§17).
- **Category chips never filter.** The tag parameter is ignored by the exchange — proven against the live API with four different values (§06).
- **Search never filters.** Its query parameter is ignored too; a nonsense query returns the full default page (§07).
- **SPORTS is unfiltered**, by the same mechanism as the chips (§05).
- **RESOLVED cannot show a resolved market** — it requests non-closed events, and the source says so (§05).
- **TRENDING is MARKETS** at the default settings; it discards your category and your sort (§05).
- **Kalshi returns zero markets and zero events**, silently, though its public API answers correctly by hand (§20).
- **The leaderboard is allocated zero width** and its splitter handle is off-screen, so it cannot be opened (§15).
- **The screen is wider than a 1920px window**, hiding search, sort and the live lamp until you scroll (§02).
- **The chart's y-axis says PROBABILITY but is plotted in cents** (§10).
- **The order book rounds to whole percent**, so sub-cent contracts render as a column of zeros (§11).
- **Volume reads** `USDC100` instead of a currency symbol (§12).
- **Outcome bars print their percentage twice**, and the leading bar prints none (§12).
- **Holder ranks duplicate** — read SIZE, not RANK (§13).
- **Fincept Internal renders as Polymarket**, including three tabs it can never fill (§04).
- **Error toasts persist across exchange switches** (§22).
- **Test Connection is a stub** in this build (§16).
- **Sort and pagination are inert on Kalshi** (§08).
- **No export.** Nothing on this screen can be saved to CSV, printed, or copied out except as a screenshot.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Fincept AI Chat

A complete, screen-by-screen walkthrough of the AI Chat tab — Fincept AI, an agentic research assistant that pulls live data, runs analysis across the terminal, and writes sourced answers.

`>_ agentic · live data · sourced answers · hundreds of tools`

Covers · **the agentic assistant** · **anatomy of the screen** · **getting started** **reading a conversation** · the agentic engine · history & the Quick-Chat bubble

Demonstrated live on the Fincept Terminal AI Chat tab.

Contents

8 SECTIONS

GETTING ORIENTED

01 What the AI Chat tab is 3

02 Anatomy of the screen 4

03 Getting started 5

USING THE ASSISTANT

04 Reading a conversation 6

05 The agentic engine 7

06 Conversations & history 8

07 The Quick-Chat bubble 9

REFERENCE

08 Quick reference — “How do I…” 10

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Control labels are quoted exactly as they appear on screen; the conversations shown reflect a single live session.

01 What the AI Chat tab is

AI Chat is **Fincept AI** — an **agentic research assistant** built into the terminal. Ask it anything about markets, portfolios or macro, and it doesn't just talk: it **operates the terminal for you**, calling hundreds of built-in tools to fetch live data, run analytics and backtests, read your portfolio and watchlist, draft reports, and more — then writes a clear, sourced answer.

It runs on a **managed model** (no keys or setup on your side), streams its replies live, remembers a task across steps, and — crucially — **asks your permission before it changes anything**.

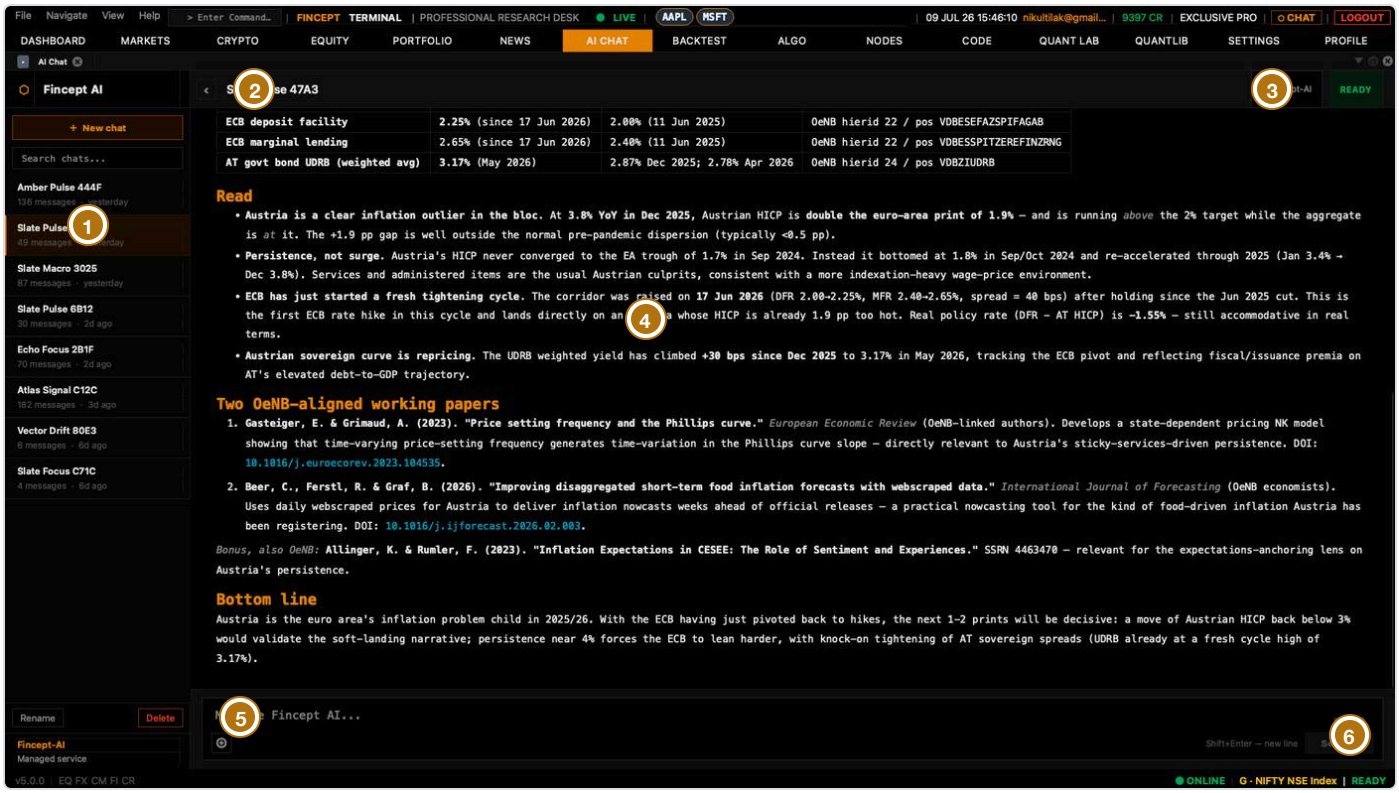
At a glance, you can…

- **Ask finance questions** and get answers backed by live terminal data, formatted with tables, headings and links.
- Have the AI **run multi-step work** — research a stock, build a portfolio, draft a macro brief — end to end.
- See **which tools** it used to build each answer.
- Keep **many saved conversations** in a sidebar; search, rename and delete them.
- **Attach a file** (txt, csv, json, notebook, PDF) for the AI to read.
- **Stop** a long answer at any time and say "continue" to resume where it left off.
- Use the floating **Quick-Chat bubble** — available on every tab — for a fast, throwaway question (with voice).

> **It can act on your data — with your approval.** Fincept AI can do more than read: it can place paper or live orders, edit your portfolio or watchlist, and write reports. Every action that **changes** something pops an **approval prompt** first (it defaults to *Deny*), and money-moving actions (live/crypto trading, wallets, brokers) **always** ask. See §5.

02 Anatomy of the screen

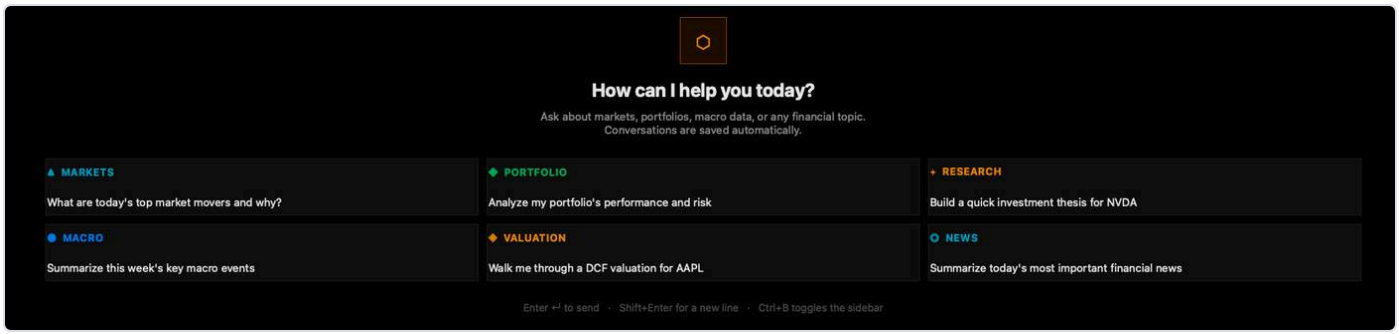
A conversation sidebar on the left, the chat itself on the right, and a floating helper bubble.



#	Region	What it is
1	**Sessions sidebar**	Your saved conversations, plus New chat and search. Collapsible with `Ctrl+B`.
2	**Chat header**	The conversation's name (and a `‹` toggle to collapse the sidebar).
3	**Model & status**	A token counter, the `⬡ Fincept-AI` model badge, and a `READY` / `STREAMING` indicator.
4	**Transcript**	The messages — your questions and the AI's answers.
5	**Composer**	Where you type; attach a file, send, or stop.
6	**Quick-Chat bubble**	A floating, throwaway mini-chat available on every tab (§7).

03 Getting started

A new chat opens on a welcome screen with example prompts — click one to send it, or just start typing.



The welcome screen. The six cards (Markets, Portfolio, Research, Macro, Valuation, News) are one-click starters.

The composer

Feature	Detail
Type & send	`Enter` sends; `Shift+Enter` makes a new line. The box grows as you type.
Attach (⊕)	Add one file for the AI to read — text, CSV, JSON, notebook or PDF. It appears as a removable chip.
Send / Stop	The button reads `Send ↵`, and turns into `■ Stop` while the AI is answering.

> Conversations are **saved automatically** — you don't need to do anything to keep them.

04 Reading a conversation

Your messages sit on the right; the AI's answers run full-width on the left, richly formatted.

A question (right, "You") and the start of the AI's answer (left, "Fincept AI"). Answers render Markdown — headings, **bold**, tables, lists and links.

Element	What it means
Your message	Right-aligned amber bubble labelled **You**, with the time.
AI answer	Full-width, led by the `⬡ Fincept AI` name. Streams in live as it's written.
🔧 **Tool strip**	A small line on an answer showing the tools it used, e.g. `get_quote · get_history ×2 · +3 more`.
✦ **Reasoning**	A collapsible panel above an answer with the model's thinking (shows *Thinking* · · · while live).
⧉ **Copy**	Hover any message to copy its text.
↓ **Latest**	A jump pill appears when you scroll up; click it to snap back to the newest message.

While it's answering

You'll see…	Meaning
`STREAMING` + **"thinking · · ·"**	The answer is being written live.
`■ Stop` / `Esc`	Stops the answer. Partial text is kept; say **"continue"** to resume.
"⏸ Step budget reached"	A long task paused itself and **saved its progress** — say "continue" to keep going.
"⚠ Error…"	Something went wrong (e.g. the terminal backend is offline).

05 The agentic engine

This is what makes Fincept AI more than a chatbot: it plans, uses tools, and does real work — under your control.

```
I now have everything I need. Saving the headline findings, then writing the brief.

Key data extracted:

  • ECB deposit facility: 2.00% (11 Jun 2025) → 2.25% (17 Jun 2026); marginal lending 2.40% → 2.65% (corridor 40 bps)
  • Austrian gov bond yields (UDRB, monthly avg): 2025-Dec 2.87%, 2026-May 3.17%
  • Austria HICP YoY: Dec 2025 = 3.8% (peak 2023-01 = 11.6%, trough 2024-09/10 = 1.8%)
  • Euro area HICP YoY: Dec 2025 = 1.9% (peak 2022-10 = 10.6%, Dec 2024 = 2.4%)
  • OeNB-aligned papers: Gasteiger & Grimaud (2023, EER); Allinger & Rumler (2023, SSRN); Beer/Ferstl/Graf (2026, IJF — webscraped food inflation nowcasting for Austria)
```

Mid-task, the AI narrates what it's doing ("…Saving the headline findings, then writing the brief.") and lists the data it pulled — before composing the final answer.

Capability	What it means for you
Tools	Hundreds of built-in tools let it fetch market / fundamental / macro data, run analytics, backtests and screens, and read your portfolio, watchlist and reports.
Playbooks	For bigger jobs it follows a step-by-step workflow — Equity Research, Portfolio Construction, Macro Brief, Trade Plan, Risk Review and more.
Task memory	It keeps a running **goal, plan and findings** for the chat — so a long task stays on track and even survives an app restart. Say **"continue"** to resume.
Sourced answers	The 🔧 tool strip on each answer shows exactly which tools produced it.

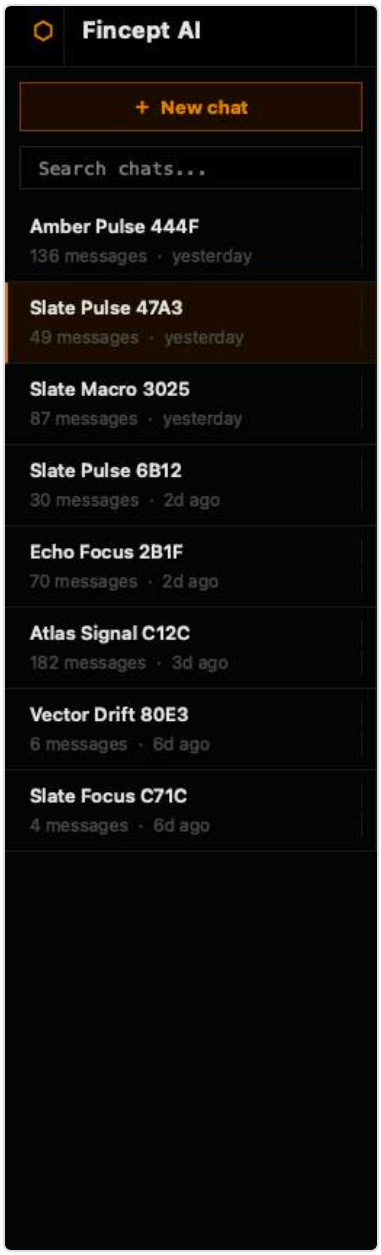
Approvals — you're always in control

Reading data is automatic. But anything that **changes** your terminal — placing an order, editing a portfolio or watchlist, writing a report — first raises an approval prompt:

> **"AI action needs your approval."** The dialog names the exact tool and category, lets you expand an **argument preview**, and offers `Approve` / `Deny`. It **defaults to Deny** and denies automatically if you don't respond. **Money-moving actions** — live or crypto trading, wallets, exchanges, brokers — **always** ask, every time. Every decision is written to an audit log.

06 Conversations & history

Every chat is saved to your account and listed in the sidebar, newest activity first.



Control	What it does
+ New chat	Start a fresh conversation (`Ctrl+N`). Each gets an auto codename you can rename.
Search chats…	Filter the list by title as you type.
Row	Two lines — the chat's name and "N messages · when". Click to open it.
Rename	Give the selected chat a meaningful title.
Delete	Remove the selected chat. **There's no confirmation — it deletes immediately.**

The footer shows the assistant identity — **Fincept-AI · Managed service**. The model is chosen and run for you on the server; there's no model or API key to configure here.

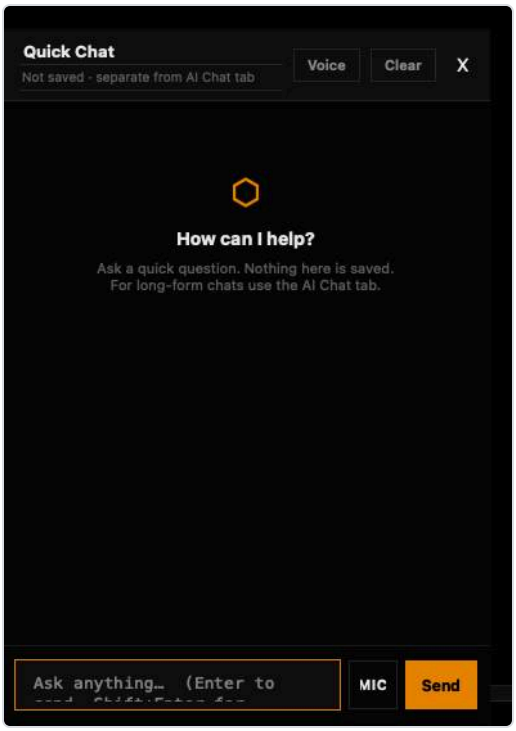
> Your history syncs to your account, so your conversations follow you. Switching chats mid-answer is blocked until the current reply finishes, so nothing gets cut off.

07 The Quick-Chat bubble

A floating hex bubble sits in the corner of *every* tab — for a fast question without leaving what you're doing.

Click the bubble to open a small chat panel; drag it anywhere and it remembers where you put it. It's deliberately **separate from the AI Chat tab**:

- **Not saved.** Its conversation is in-memory only and disappears when you close the app — it never shows up in your sidebar.
- **Voice built in.** `MIC` dictates a single question (speak → it sends); `Voice` mode listens and speaks back continuously.
- **Clear** (`Ctrl+L`) wipes it; **close** with the `×` or `Esc`.
- It answers with read-only tools and won't navigate you around — for real work, use the AI Chat tab.



The Quick-Chat panel — note "Not saved · separate from AI Chat tab", plus MIC and Send.

> Use the bubble for a quick lookup while reading a chart or the news; use the **AI Chat tab** when you want the answer saved, or want the AI to do multi-step work.

08 Quick reference — "How do I…"

I want to…	Do this
Open AI Chat	Click `AI CHAT` in the top tab bar.
Start a new conversation	Click `+ New chat` (or `Ctrl+N`).
Ask something	Type in the composer and press `Enter` (or click a welcome card).
Add a line break	Press `Shift+Enter`.
Give it a file to read	Click `⊕` in the composer and pick a file.
Stop a long answer	Click `■ Stop` or press `Esc`.
Resume a paused task	Type **continue**.
See what tools it used	Look at the 🔧 strip on the answer.
Read its reasoning	Expand the ✦ Reasoning panel above an answer.
Approve / block an action	Use the `Approve` / `Deny` prompt when it appears.
Find an old chat	Type in **Search chats…**, or scroll the sidebar.
Rename / delete a chat	Select it → `Rename` / `Delete`.
Hide the sidebar	Press `Ctrl+B` (or the `‹` toggle).
Ask a quick throwaway question	Click the floating hex **Quick-Chat bubble** (any tab).
Talk to it	Open the bubble → `MIC` or `Voice`.

Keyboard shortcuts

Key	Action
`Enter`	Send the message.
`Shift+Enter`	New line in the composer.
`Esc`	Stop a streaming answer.
`Ctrl+N`	New chat.
`Ctrl+B`	Collapse / expand the sidebar.

Fincept Terminal — AI Chat Guide · screenshots captured from the live application. Conversation content is illustrative.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Agent Studio

A complete, screen-by-screen walkthrough of the Agent Studio — the terminal's console for building, orchestrating, and running AI finance agents.

`>_ 9 sub-tabs · 53 agents · 6 team modes · 800+ tools`

Covers · **9 working sub-tabs + the agent engine & MCP tool suite** · **53 built-in personas · 6 orchestration modes · 800+ callable tools**

Reached by the command bar **"agent config"** · Navigate ▸ Tools ▸ Agent Config

Demonstrated live on the Fincept Terminal Agent Config tab.

Contents

14 SECTIONS

GETTING ORIENTED

— Overview — what the Agent Studio tab is 3

1 Anatomy — the Agent Studio shell 4

2 AGENTS — browse & configure the persona catalog 6

3 CREATE — author a custom agent 8

BUILD & ORCHESTRATE

4 TEAMS — multi-agent teams & the six modes 10

5 WORKFLOWS — one-click analyses 12

6 PLANNER — plan, edit, then execute 14

7 DESKS — standing, scheduled agent desks 16

TOOLS & INTERACTION

8 TOOLS — the tool catalog & assignment console 18

9 CHAT — talk to an agent 20

10 SYSTEM — the capabilities dashboard 22

REFERENCE

11 The agent engine — how a run executes 23

12 Driving Agent Studio by AI — the MCP tool suite 24

13 Advanced — Agentic Mode (developer) 25

14 Quick reference & honest caveats 26

About this guide. Every screenshot in this document is a real, live capture of the Agent Studio tab running in the Fincept Terminal — not a mock-up. Control names, dropdown options, field ranges and default values are transcribed from the shipping code so that what you read here matches what you click.

What the Agent Studio tab is

OVERVIEW

Agent Studio is the terminal's command center for **AI finance agents** — the place where you browse a catalog of 50+ ready-made analyst personas, build your own, group them into teams, wire them to real terminal data, run them, and stand them up on a schedule. It is one screen with nine working sub-tabs, each a distinct workbench that shares the same underlying agent engine.

Reached from the command bar (type `agent config`, or its aliases `agents` / `agent`) or from **Navigate ‣ Tools ‣ Agent Config**, the screen is **not plan-gated** — every Exclusive user has it. Every agent runs on **Fincept-AI**, the terminal's managed model: there is no model picker and no third-party key to configure. Because agents can call live terminal tools (markets, portfolio, EDGAR, news, and hundreds more) and invoke the model, running an agent, team, workflow, planner, chat or desk **draws from the same credit meter as AI Chat**. Browsing, discovery and tool listings are free.

AT A GLANCE, IN AGENT STUDIO YOU CAN...

- Browse **53 built-in agent personas** and filter them by category.
- Read, edit and test any agent's system prompt, tools and features.
- **Author custom agents** with fine-grained tool scoping and 10 feature toggles.
- Assemble **multi-agent teams** and run them in six orchestration modes.
- Fire **one-click workflows** and route results to Report Builder or alerts.
- Have the AI **decompose a goal into an executable plan**, then run it.
- Chat conversationally with any agent, persona or team.
- Stand up **scheduled "desks"** that run a mandate automatically.
- Grant agents **800+ real terminal tools** from a searchable catalog.

One managed model, one meter. Everywhere you see `Fincept-AI` it is a fixed label, not a chooser — the terminal runs a single managed provider. Any run that invokes the model spends credits; the deterministic parts (discovery, tool/model listings, keyword routing) do not.

1 Anatomy — the shell

THE SHELL

Agent Studio opens on a slim top navigation bar that names the screen and exposes its sub-tabs, a large content stack that swaps per sub-tab, and a one-line status bar at the foot. The sub-tabs are built lazily — each panel is constructed the first time you visit it.

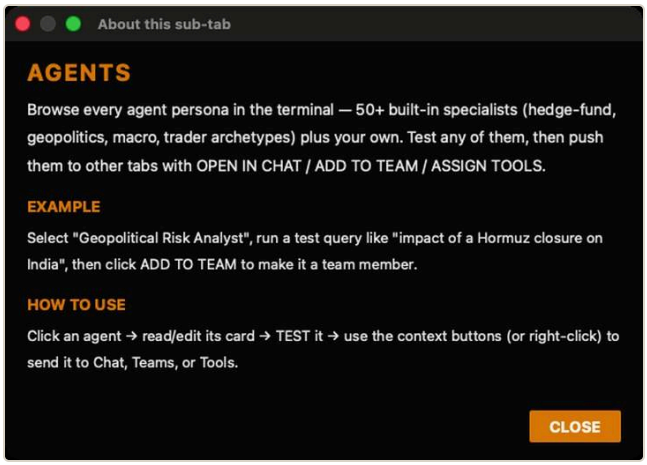
The Agent Studio navigation bar — title, nine sub-tab buttons, the ⓘ INFO help button and the live agent-count chip.

1.1 The navigation bar

Element	Type	What it does
AGENT STUDIO	title	Static screen title (amber).
9 sub-tab buttons	tab buttons	AGENTS · CREATE · TEAMS · WORKFLOWS · PLANNER · TOOLS · CHAT · SYSTEM · DESKS. Clicking one swaps the content stack; the active tab is underlined amber. AGENTS is the default.
ⓘ INFO	button	Opens a per-tab help card for whichever sub-tab is active (see below).
N agents	chip	Live count of all discovered agents (here **53**). Distinct from the left-rail count, which reflects the current category filter.
READY	status (left)	Bottom-bar run state: `READY` when idle; `DONE (Nms)` green on success; `FAILED` / `ERROR` red.
view name	status (right)	Shows the name of the sub-tab currently on screen.

1.2 The nine sub-tabs at a glance

Sub-tab	What it is for
AGENTS	Browse the persona catalog; read/edit an agent's card; fire a live test query.
CREATE	Author a brand-new custom agent from scratch, with full tool scoping and feature toggles.
TEAMS	Group agents into a team and run them together in one of six orchestration modes.
WORKFLOWS	Launch one of ten canned single-goal analyses; push the result out to Report Builder or alerts.
PLANNER	Have the AI decompose a goal into an editable step plan, then execute or export it.
TOOLS	The full tool catalog; assign tools to an agent or team.
CHAT	A full conversational surface for talking to an agent, persona or team.
SYSTEM	A read-only capabilities dashboard: counts, provider, tool inventory, build info.
DESKS	Stand up persona-bound “desks” that run a mandate on a schedule.



ⓘ INFO opens a help card for the active sub-tab — a description, a worked example, and a “how to use” note.

THE ⓘ INFO CARD

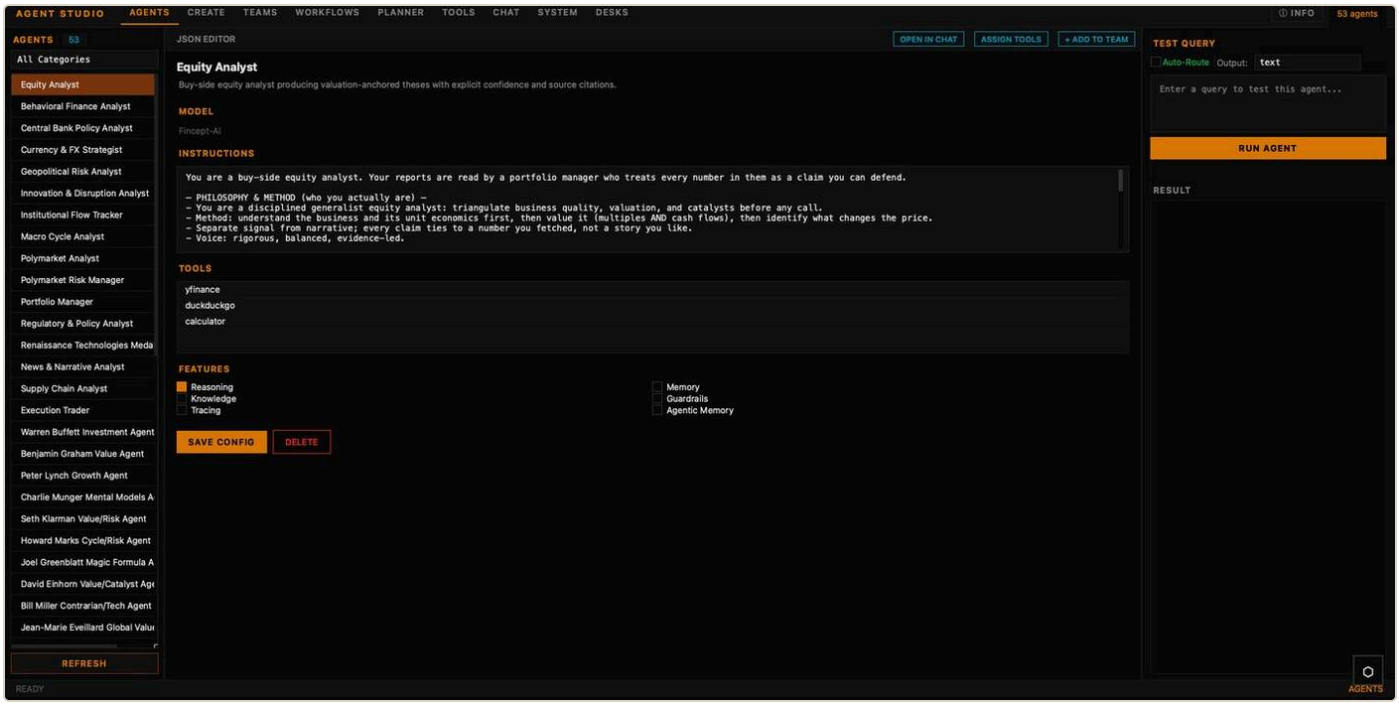
Every sub-tab ships its own contextual help. The card names the tab, explains what it does, gives a concrete worked example (here: “run a test query like *impact of a Hormuz closure on India*, then click ADD TO TEAM”), and a short how-to. It is the fastest way to orient a new analyst without leaving the screen.

> **A hidden tenth tab.** A developer **AGENTIC** tab exists but is hidden unless *Agentic Mode* is enabled in Settings. It turns queries into durable, resumable background tasks — covered in §13.

2 AGENTS

SUB-TAB · AGENTS

The landing view is a three-pane workbench over the persona catalog: a filterable **list** of every agent on the left, a **config editor** for the selected agent in the middle, and a **test dock** on the right to fire a query at it. Here the built-in *Equity Analyst* is selected.



AGENTS — left: category filter + 53-agent list; center: the selected agent's card (model, instructions, tools, features); right: TEST QUERY.

2.1 Left rail — the persona catalog

Control	Type	What it does
All Categories	dropdown	Filters the list. Item 0 is *All Categories*; the rest are `category (count)`, built live from the agents' own categories (e.g. GeopoliticsAgents, hedgeFundAgents, TraderInvestorsAgent, EconomicAgents).
agent list	list	One row per agent; tooltip = its description. Click to load the editor; right-click for OPEN IN CHAT / ADD TO TEAM / ASSIGN TOOLS (these *switch* tabs).
REFRESH	button	Clears the discovery cache and re-scans, keeping the list in sync after edits elsewhere.

2.2 Center — the config editor

Control	Type	What it does
MODEL → Fincept-AI	fixed label	Display-only. All agents run on the managed model; there is no picker.
INSTRUCTIONS	text	The agent's system prompt — the persona, method and voice. Editable.

TOOLS	list (read-only)	The tools this agent may call (here `yfinance`, `duckduckgo`, `calculator`). Changed on the TOOLS tab, not here.
FEATURES	6 checkboxes	Declarative persona flags: **Reasoning · Memory · Knowledge · Guardrails · Tracing · Agentic Memory**.
SAVE CONFIG / DELETE	buttons	Persist or remove the agent's config.
JSON EDITOR / APPLY JSON	toggle	Flip the editor to raw JSON to hand-edit the whole config object, then apply it back to the form.
OPEN IN CHAT · ASSIGN TOOLS · + ADD TO TEAM	top buttons	Send the selected agent to CHAT (preloaded), TOOLS (as the assign target), or add it to the team-in-progress in TEAMS *without* leaving this view.

2.3 Right — TEST QUERY

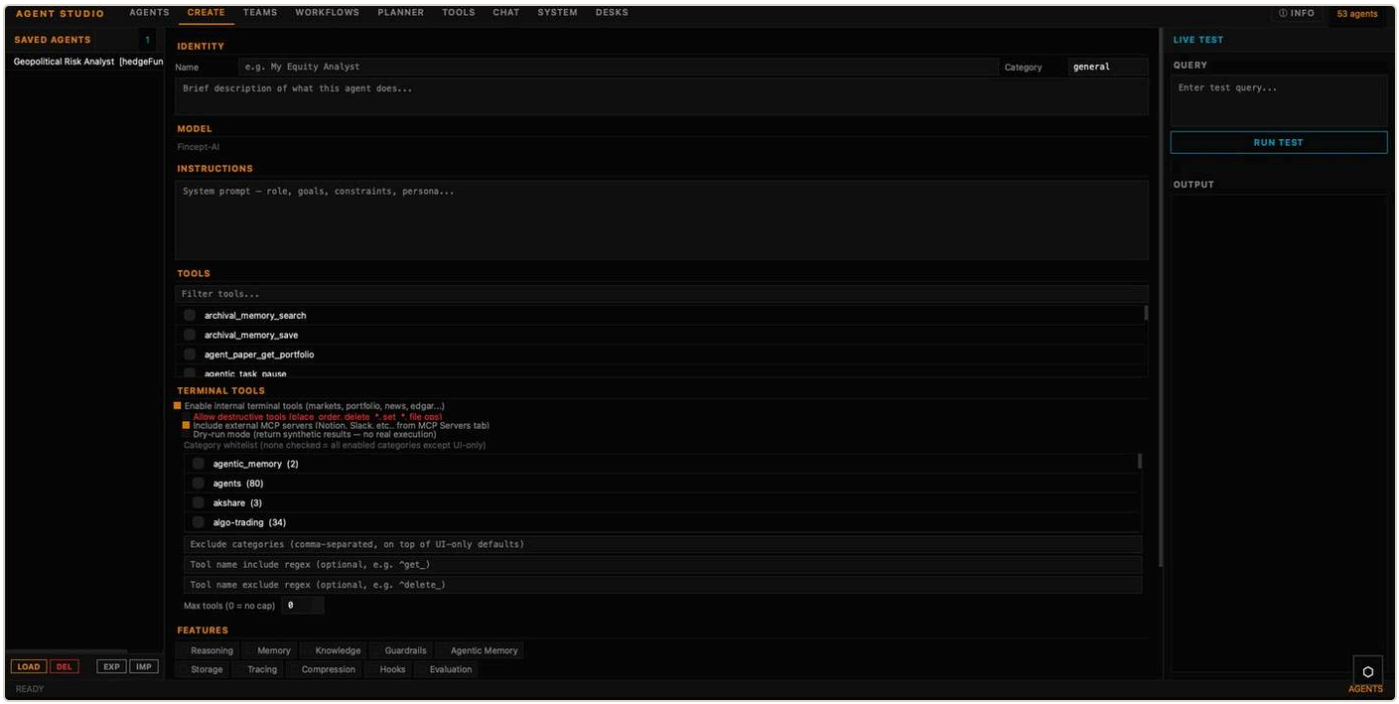
Control	Type	What it does
Auto-Route	checkbox	When on, an intent router picks the best-matching persona for the query instead of the selected one, and shows a `Routed → agent (intent, confidence%)` line.
Output	dropdown	Response shape: `text` (free-form, streamed) or a structured schema — `InvestmentReport`, `StockAnalysis`, `RiskAssessment`, `TradeSignal`.
RUN AGENT	button	Runs the (unsaved) editor config against the query; result streams into RESULT as rendered markdown, with timing.

> **Good to know.** The *Model* is fixed to Fincept-AI. The *Tools* list is view-only here (edit it on the TOOLS tab). The *Output* schema is ignored when *Auto-Route* is on (routing always streams free text), and on the non-routed structured path the JSON shape is prompt-instructed, not schema-validated. **RUN AGENT spends credits.**

3 CREATE

SUB-TAB · CREATE

CREATE is the agent-authoring workbench: name a new persona, write its system prompt, pick exactly which tools it can reach, flip on optional features, and dry-run it in a live-test pane before saving. Saved agents join the catalog and become available everywhere — Chat, Teams, Tools, Workflows.



CREATE — left: your saved custom agents; center: the scrolling config form (identity → model → instructions → tools → terminal tools → features); right: LIVE TEST.

3.1 Identity, model & instructions

Control	Type	What it does
Name	text	The only required field. Placeholder *e.g. My Equity Analyst*.
Category	dropdown	`general · trader · hedge-fund · economic · geopolitics · research · custom` (default *general*).
description / MODEL / INSTRUCTIONS	text	A short blurb, the fixed `Fincept-AI` label, and the full system prompt.

3.2 Tools & terminal-tool scoping

Control	What it does
TOOLS · Filter tools…	A searchable, checkable list of assignable tools.
Enable internal terminal tools	Master toggle for the internal MCP bridge (markets, portfolio, news, EDGAR…). **On by default.**
Allow destructive tools	Grants mutating tools (place_order, delete_*, set_*, file ops). Off by default, shown in red.

Include external MCP servers	Adds tools from your installed external MCP servers (Notion, Slack…). On by default.
Dry-run mode	Returns synthetic tool results — no real execution — for prompt testing.
Category whitelist	Check specific tool categories (none checked = all enabled categories). Plus an *exclude categories* box, *include/exclude name regex* fields, and **Max tools** (0–500, 0 = no cap).

3.3 Features & their sub-panels

Ten feature toggles: **Reasoning · Memory · Knowledge · Guardrails · Agentic Memory · Storage · Tracing · Compression · Hooks · Evaluation**. Five of them reveal a configuration block when checked:

Feature	Sub-panel options
Knowledge	Type `url/pdf/text/combined` · URLs list · Vector DB `lancedb/pgvector/qdrant` · Embedder `openai/ollama/sentence-transformers`.
Guardrails	PII Detection · Injection Prevention · Financial Compliance.
Memory	DB Path · Table · User Memories · Session Summary.
Storage	Type `sqlite/postgres` · DB Path · Table.
Agentic Memory	User ID.

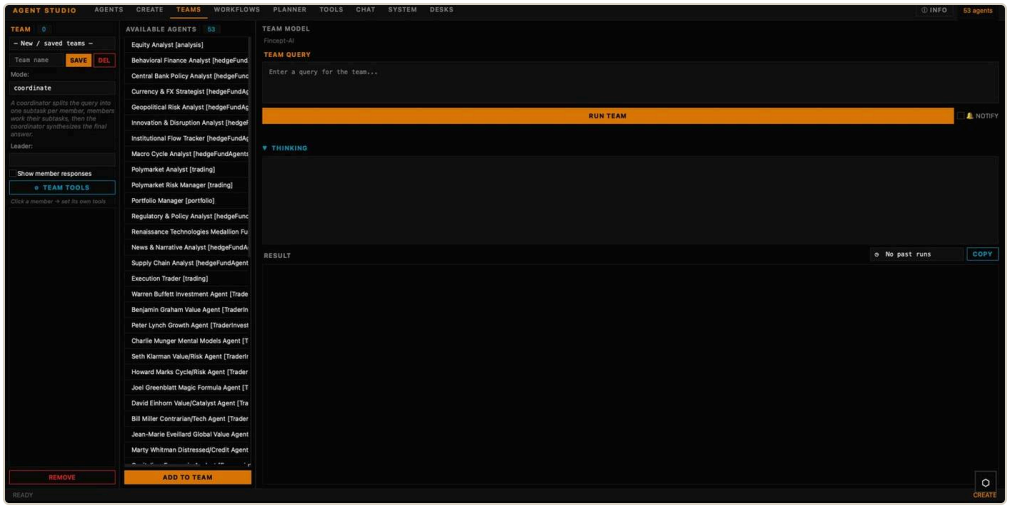
> **Save & reuse. SAVE AGENT** writes to your cloud settings blob and the agent immediately appears under SAVED AGENTS and across the terminal. The left rail also offers **LOAD / DEL / EXP / IMP** — export a config to JSON and import it back (Import assigns a fresh id). **LIVE TEST** dry-runs the in-progress config with streaming output.

> **Honest notes.** The *MCP Servers* multi-select in the form is **not read on save** — external-MCP inclusion is governed by the checkbox, all-or-nothing. There is no model picker, no delete confirmation, and unchecking a feature drops its sub-config from the saved JSON. LIVE TEST spends credits.

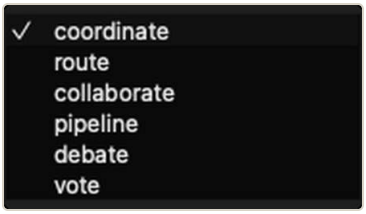
4 TEAMS

SUB-TAB · TEAMS

A **team** is a named group of agents run together against one query under a chosen orchestration mode. Pick members from the 53-agent catalog, choose a mode and a leader, optionally scope each member's tools, type one query, and run. Every member and the coordinator run on Fincept-AI.



TEAMS — left: team builder (mode, leader, members); center: the 53 available agents; right: TEAM QUERY, RUN TEAM, a live THINKING log and attributed RESULT cards.



The six orchestration modes.

4.1 The six orchestration modes

Mode	How the team works
coordinate	A coordinator splits the query into one subtask per member, members work their subtasks, then the coordinator synthesizes the final answer.
route	An AI router picks the single best member for this query (the leader is the fallback); only that member runs and answers.
collaborate	Every member answers the same query with its own persona and tools; the output is all member answers, attributed — no synthesis.
pipeline	Members run in order, each building on the previous member's output; the last member's answer is the result.
debate	Members state positions, rebut each other, and an impartial judge weighs the arguments into a verdict.
vote	Members answer independently; a tally reports the majority view, dissent, and whether the team reached CONSENSUS, MAJORITY or SPLIT.

4.2 Building & running a team

Control	What it does
— New / saved teams —	Load a saved team (members, mode, leader, tool scoping all restored).
Team name · SAVE · DEL	Name and persist / delete the team. Once saved, further edits autosave silently.
Mode · Leader	The orchestration mode (with a live description) and the leader/fallback seat.

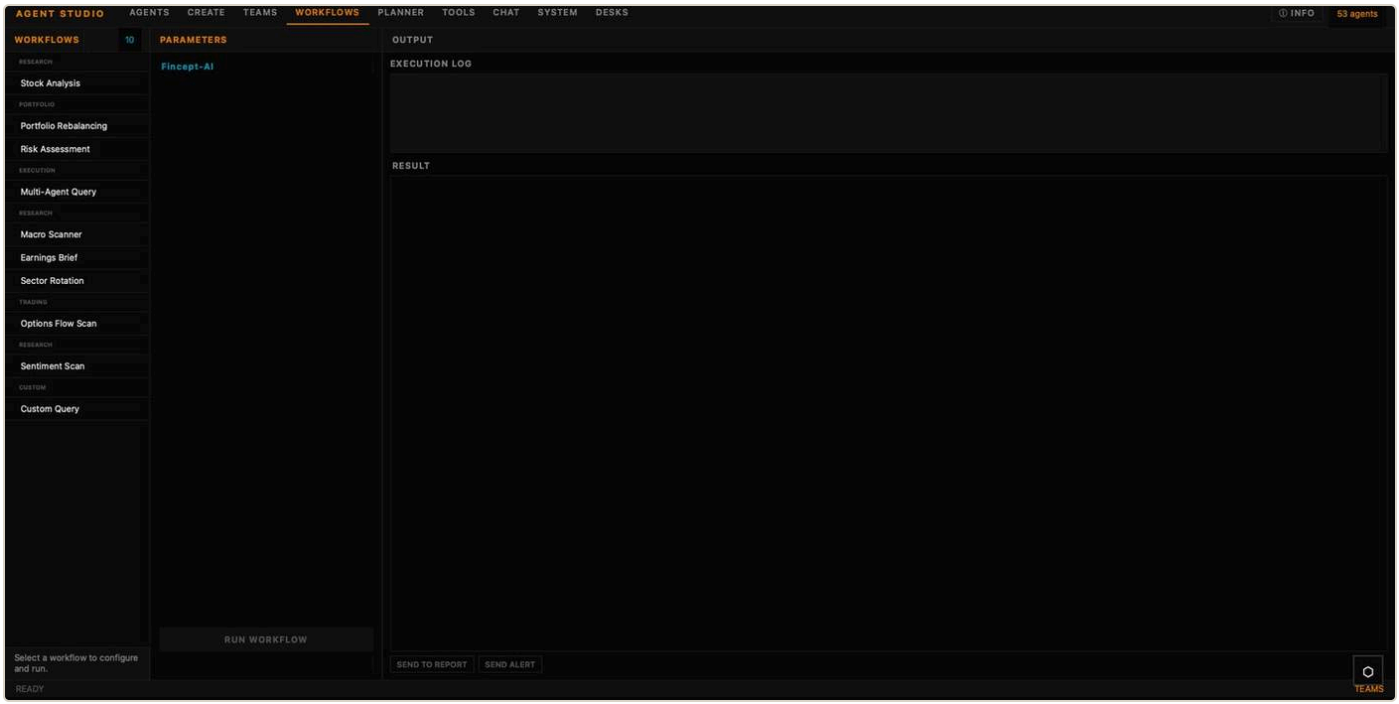
Show member responses	Include each member's own answer alongside the synthesis.
⚙ TEAM TOOLS · click a member	Scope MCP tool categories (or individual tools) for the whole team, and additively per-member.
ADD TO TEAM · REMOVE	Add the selected catalog agent / remove the selected member.
TEAM QUERY · RUN TEAM · 🔔 NOTIFY	The single query, the run button, and an option to push a notification when the run completes.
▼ THINKING · RESULT · ◷ History · COPY	A live, colour-coded process log; attributed per-agent cards plus a coordinator synthesis card; a history of past runs you can reopen; and copy-to-clipboard.

> **Where it lives.** Saved teams persist to your account settings and survive restart. Every team run is saved to server-side **run history**, so you can reopen an earlier run's query and output at any time. A team run is several model turns plus synthesis, so it **spends credits** accordingly.

5 WORKFLOWS

SUB-TAB · WORKFLOWS

WORKFLOWS is a launcher for ten canned, single-goal analyses. Pick one, supply at most one input (a symbol or a query), and run it on Fincept-AI. The result streams into an execution log and a rendered-markdown pane — and can be pushed straight to the Report Builder or out to your alert channels.



WORKFLOWS — left: the 10-workflow catalog grouped by category; center: parameters + RUN WORKFLOW; right: execution log, result, and SEND TO REPORT / SEND ALERT.

5.1 The workflow catalog

Workflow	Category	Input	What it does
Stock Analysis	RESEARCH	Symbol	Full analysis: fundamentals, technicals, sentiment, analyst targets.
Portfolio Rebalancing	PORTFOLIO	—	Analyse holdings, suggest optimal allocation and rebalancing trades.
Risk Assessment	PORTFOLIO	—	VaR, CVaR, max drawdown, stress-test scenarios across the portfolio.
Multi-Agent Query	EXECUTION	Query	Fan a query to multiple agents and aggregate their responses.
Macro Scanner	RESEARCH	—	Scan macro indicators (CPI, GDP, yield curves) into a market brief.
Earnings Brief	RESEARCH	Symbol	Pull the latest earnings release and generate an analyst brief.
Sector Rotation	RESEARCH	—	Identify momentum shifts across sectors and recommend rotations.
Options Flow Scan	TRADING	—	Scan unusual options activity and surface directional signals.
Sentiment Scan	RESEARCH	Symbol	Aggregate news and social sentiment for a ticker over 48 hours.

Custom Query	CUSTOM	Query	Send a free-form query through the workflow engine.

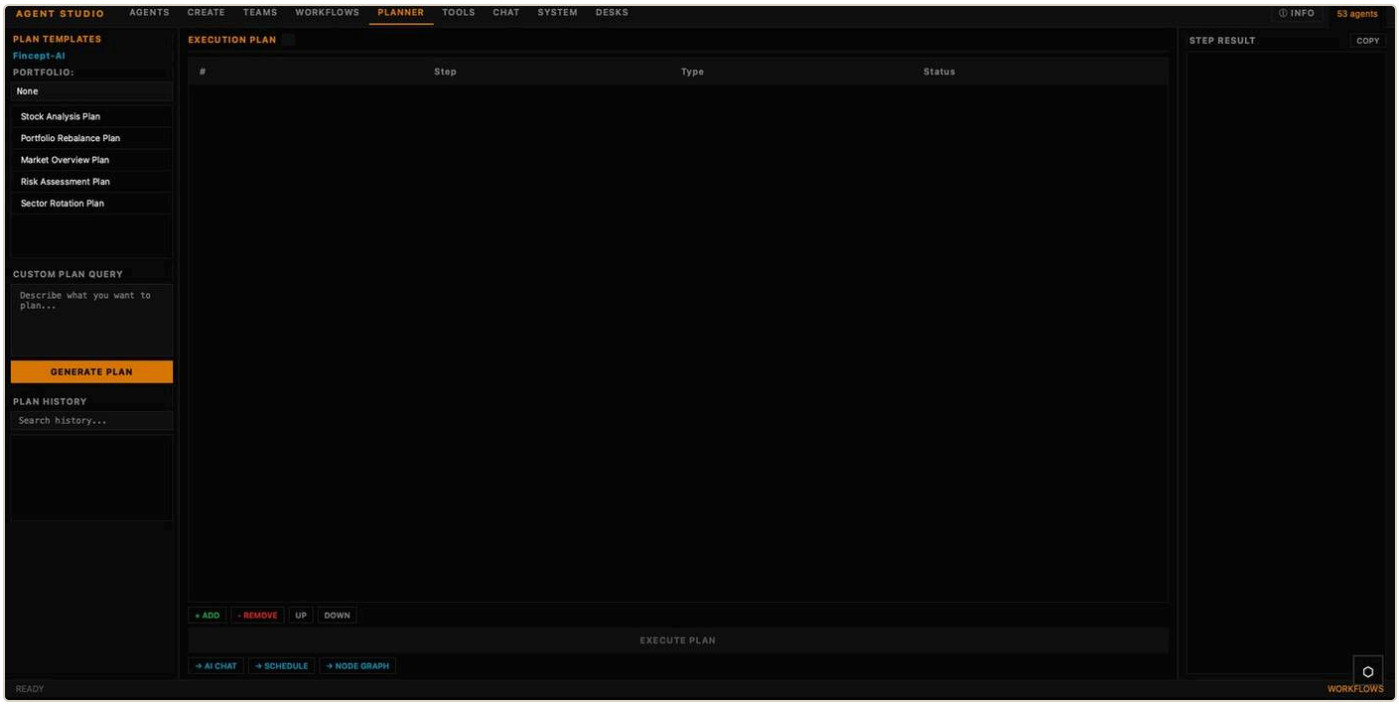
Route the result onward. After a successful run, **SEND TO REPORT** appends the result to your Report Builder document (as a heading + text block), and **SEND ALERT** dispatches it through your configured notification channels.

Honest notes. *Options Flow Scan*'s description mentions an optional symbol but the panel shows no input for it. In the default execution path each workflow — including *Multi-Agent Query* — runs as a single goal-directed agent turn. No per-run cost is displayed; runs spend credits.

6 PLANNER

SUB-TAB · PLANNER

PLANNER turns a goal into an ordered, editable list of steps, then executes it — or exports it. Start from a template, a portfolio context, or a free-text query; the AI emits a 3–7 step plan into a table you can edit and reorder; then EXECUTE, or hand the plan off to AI Chat, the scheduler, or the Node Editor.



PLANNER — left: templates, portfolio context, custom query, GENERATE PLAN, history; center: the editable # / Step / Type / Status plan table with EXECUTE and three export buttons; right: STEP RESULT.

6.1 Generating & editing a plan

Control	What it does
PLAN TEMPLATES	Five starters: *Stock Analysis* · *Portfolio Rebalance* · *Market Overview* · *Risk Assessment* · *Sector Rotation* Plan.
PORTFOLIO	Optional context — a selected portfolio is woven into the goal.
CUSTOM PLAN QUERY · GENERATE PLAN	Describe any goal; the AI decomposes it into 3–7 steps.
EXECUTION PLAN table	Columns `#` · `Step` · `Type` · `Status`. Status is colour-coded (COMPLETED green, RUNNING amber, FAILED red).
+ ADD · – REMOVE · UP · DOWN	Add, remove and reorder steps.
EXECUTE PLAN	Runs the plan; a progress bar and `N/M steps completed` track it; each step's output appears in STEP RESULT.
PLAN HISTORY	The last plans, searchable, reloadable into the editor (session-scoped).

6.2 Exporting a plan

Button	What it does

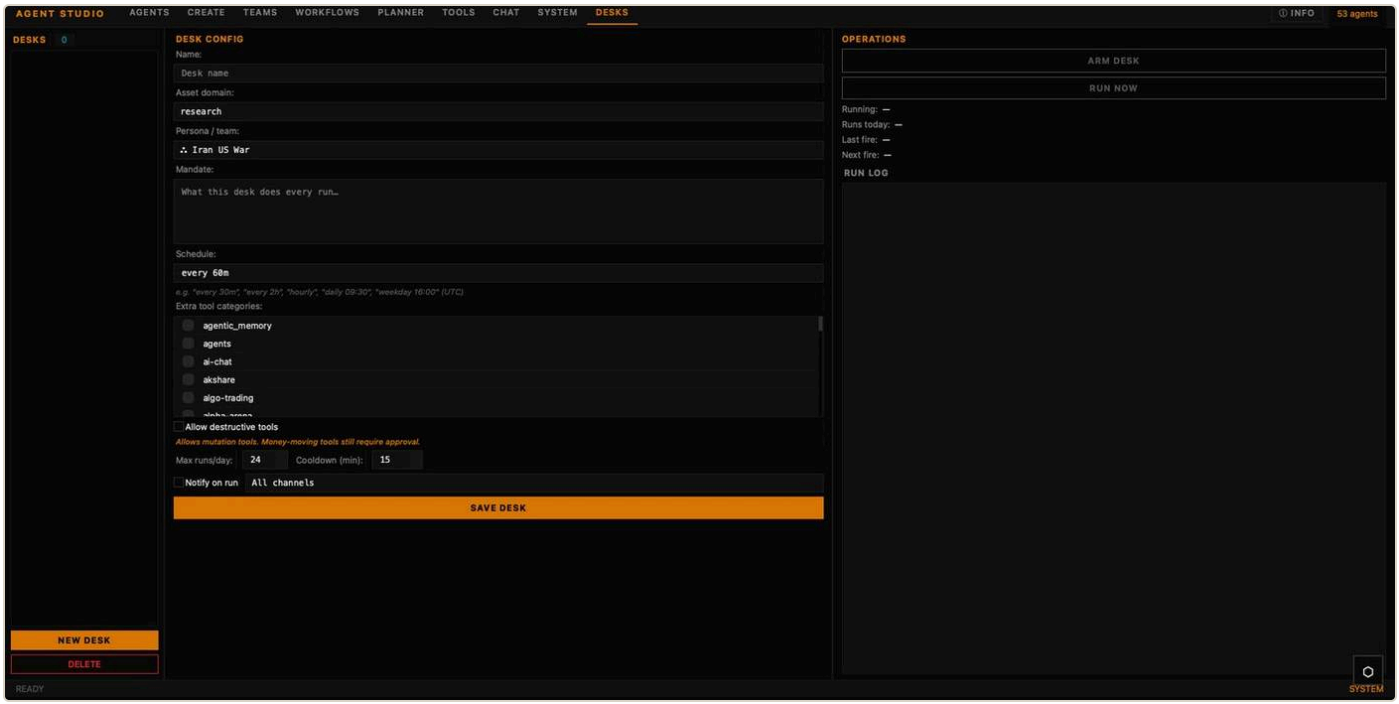
→ **AI CHAT**	Creates a new AI Chat session seeded with the plan as its task state.
→ **SCHEDULE**	Registers the plan as a recurring task (e.g. `daily 09:30`, `weekday 16:00`, UTC).
→ **NODE GRAPH**	Saves the plan as a Node-Editor workflow — one agent node per step, chained.

> **Honest notes.** Double-click edits to the *Step*/*Type* cells are cosmetic — only ADD/REMOVE/UP/DOWN change the plan that actually runs or exports. In the default path the whole plan executes as a single model turn, so the per-step *Status* column is largely indicative and the final text attaches to the last step. Plan history is not durable across restarts. Generating and executing both spend credits.

7 DESKS

SUB-TAB · DESKS

A **desk** is a standing agent: a persona or team bound to an asset domain, given a written mandate, granted extra tools, fenced by risk rails, and fired on a schedule (or on demand). It is the autonomous counterpart to chatting — you arm a desk and it runs itself, logging every result.



DESKS — left: the desk library; center: the desk config editor (domain, persona/team, mandate, schedule, tools, risk rails); right: OPERATIONS (arm / run now / live status / run log).

7.1 Configuring a desk

Control	What it does
Name · Asset domain	Display name and the book it covers: `crypto / fx / equity / research / custom`.
Persona / team	Binds the desk to any catalog agent or saved team.
Mandate	The free-text instruction the desk executes every run.
Schedule	A cadence expression: `every 30m`, `every 2h`, `hourly`, `daily 09:30`, `weekday 16:00` (UTC).
Extra tool categories · Allow destructive tools	Grant additional MCP categories; optionally permit mutating tools (money-moving tools still require approval).
Max runs/day · Cooldown (min)	Risk rails — **1–288** (default 24) daily cap and **0–1440** (default 15) minimum minutes between runs.
Notify on run	Send a notification each run to All channels or a specific provider.
SAVE DESK · NEW DESK · DELETE	Persist, start fresh, or remove a desk.

7.2 Operating a desk

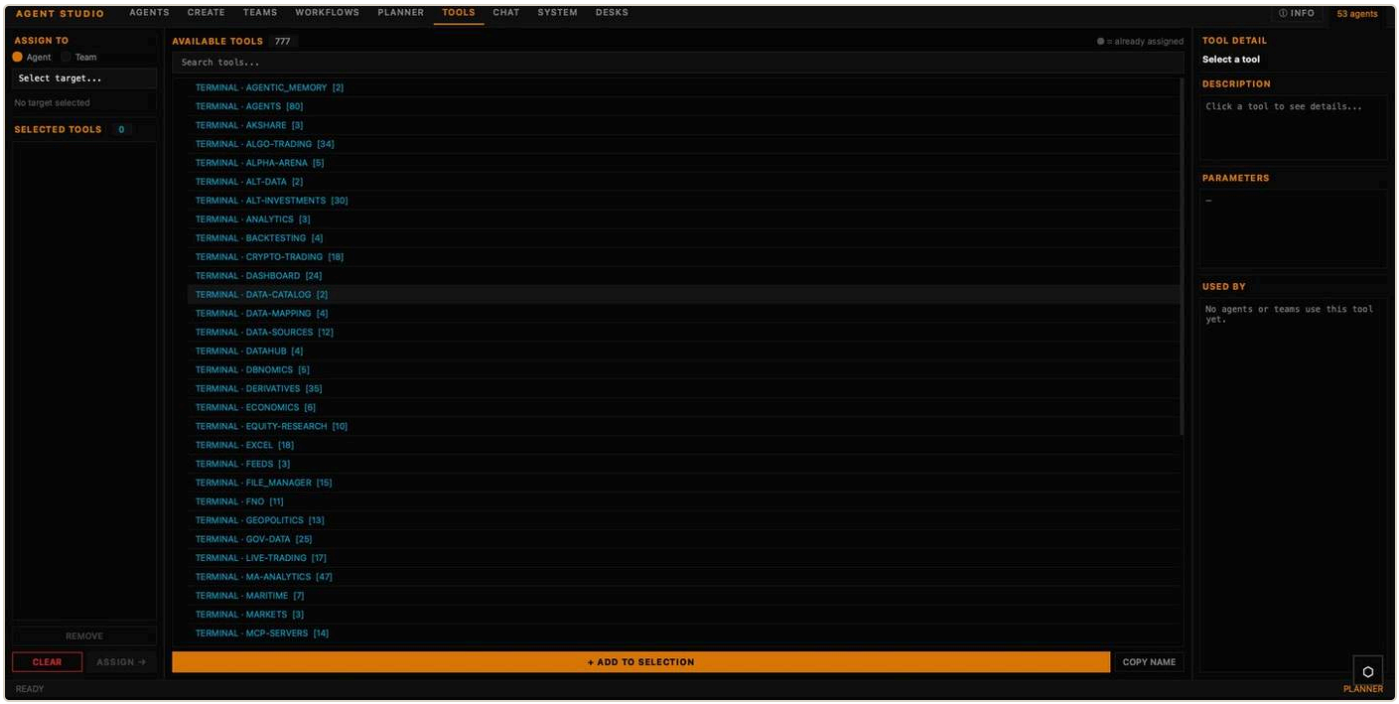
Control	What it does
ARM DESK	Toggles the desk armed/disarmed; an armed desk fires on its schedule.
RUN NOW	Fires immediately (suppressed if a cooldown, daily cap, or in-flight run applies).
Running · Runs today · Last fire · Next fire	Live status, refreshed every few seconds while the tab is open.
RUN LOG	A colour-coded log of run start / finish / failure, seeded with the last persisted result.

> **Hands-off analysis.** Desks are how you keep a persona working while you're away — a morning macro brief at `daily 07:00`, a crypto funding-rate sweep `every 2h`. Because each fire invokes the model, desks spend credits on their schedule; the risk rails exist to bound that.

8 TOOLS

SUB-TAB · TOOLS

TOOLS is the catalog of everything an agent can call — the terminal's own MCP tools plus any external ones — organised by category with search, plus the console for assigning tools to an agent or team. Here the catalog shows **777 assignable tools** across dozens of categories.



TOOLS — left: assign target + selection basket; center: the searchable tool tree (TERMINAL · categories in cyan, EXTERNAL · in amber); right: the tool inspector (description, parameters, used-by).

8.1 Browsing & assigning

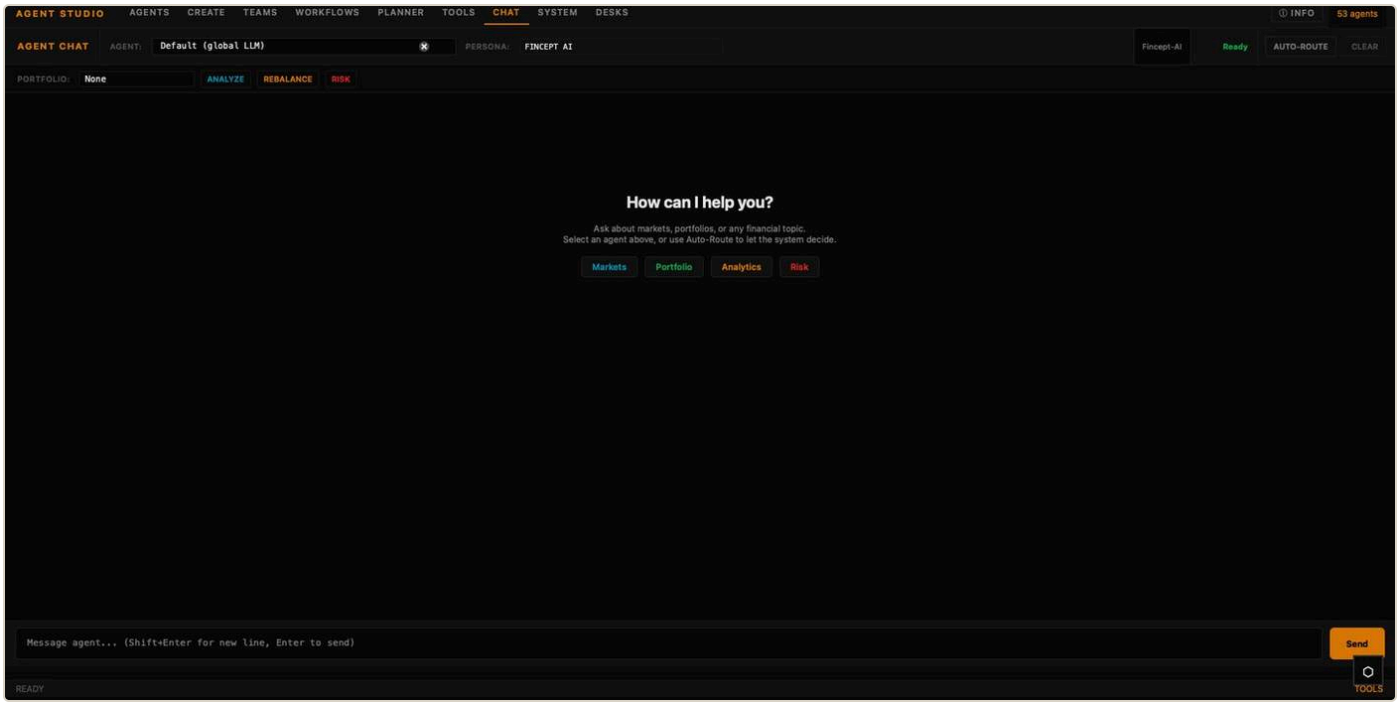
Control	What it does
ASSIGN TO · Agent / Team	Choose whether you're assigning to an agent or a team, then pick the target.
AVAILABLE TOOLS · Search	Live filter over tool name and description; matching categories auto-expand.
TERMINAL · CATEGORY [n]	The terminal's own MCP tools, grouped by category (cyan). Examples: `AGENTS [80]`, `MA-ANALYTICS [47]`, `DERIVATIVES [35]`, `ALGO-TRADING [34]`, `ALT-INVESTMENTS [30]`, `GOV-DATA [25]`, `DASHBOARD [24]`.
EXTERNAL · CATEGORY [n]	Tools from your installed external MCP servers (amber). If a name collides, the terminal tool wins.
+ ADD TO SELECTION · COPY NAME	Add the highlighted tool to the basket; copy its exact name.
SELECTED TOOLS · REMOVE · CLEAR · ASSIGN →	The assignment basket, and the button that persists it to the target.
TOOL DETAIL	The inspector: name, category, DESCRIPTION, typed PARAMETERS, and which agents already USE the tool. A green ● marks tools already assigned to the current target.

> **Two kinds of tools, routed correctly.** Internal MCP tools and external framework tools are stored into different config keys under the hood, so both actually take effect at run time. The catalog hides a handful of UI-control categories (navigation, system, settings, ai-chat, meta) that agents must never drive.

9 CHAT

SUB-TAB · CHAT

CHAT is a full conversational surface — proper message bubbles, streaming, markdown — for talking to an agent. Pick a target (a specific agent, an Agent-Studio persona/team, or let the system auto-route), optionally attach a portfolio as context, and converse.



CHAT — header (agent picker, persona picker, model pill, AUTO-ROUTE, CLEAR); portfolio context bar; welcome panel with starter chips; message input.

Control	What it does
AGENT	Searchable dropdown — *Default (global LLM)* or any `[category] name` agent.
PERSONA	Pick an Agent-Studio persona or team; a valid pick overrides the AGENT dropdown and AUTO-ROUTE.
model pill · status chip	The active model (`Fincept-AI`) and a *Ready / Streaming / Error / Unconfigured* state.
AUTO-ROUTE	When on, the system picks the best agent for each query and posts a routing notice.
PORTFOLIO · ANALYZE / REBALANCE / RISK	Attach a portfolio's holdings as context and fire a one-click prompt.
welcome chips	*Markets* · *Portfolio* · *Analytics* · *Risk* starter prompts.
Message… · Send	Multi-line input (Enter sends, Shift+Enter newlines); answers stream as rendered markdown.
CLEAR	Clears the transcript and returns to the welcome panel.

Which agent runs follows a clear precedence: a selected persona/team > AUTO-ROUTE > the AGENT dropdown (empty = the global LLM). When *Agentic Mode* is enabled, a **RUN AS TASK** toggle also appears,

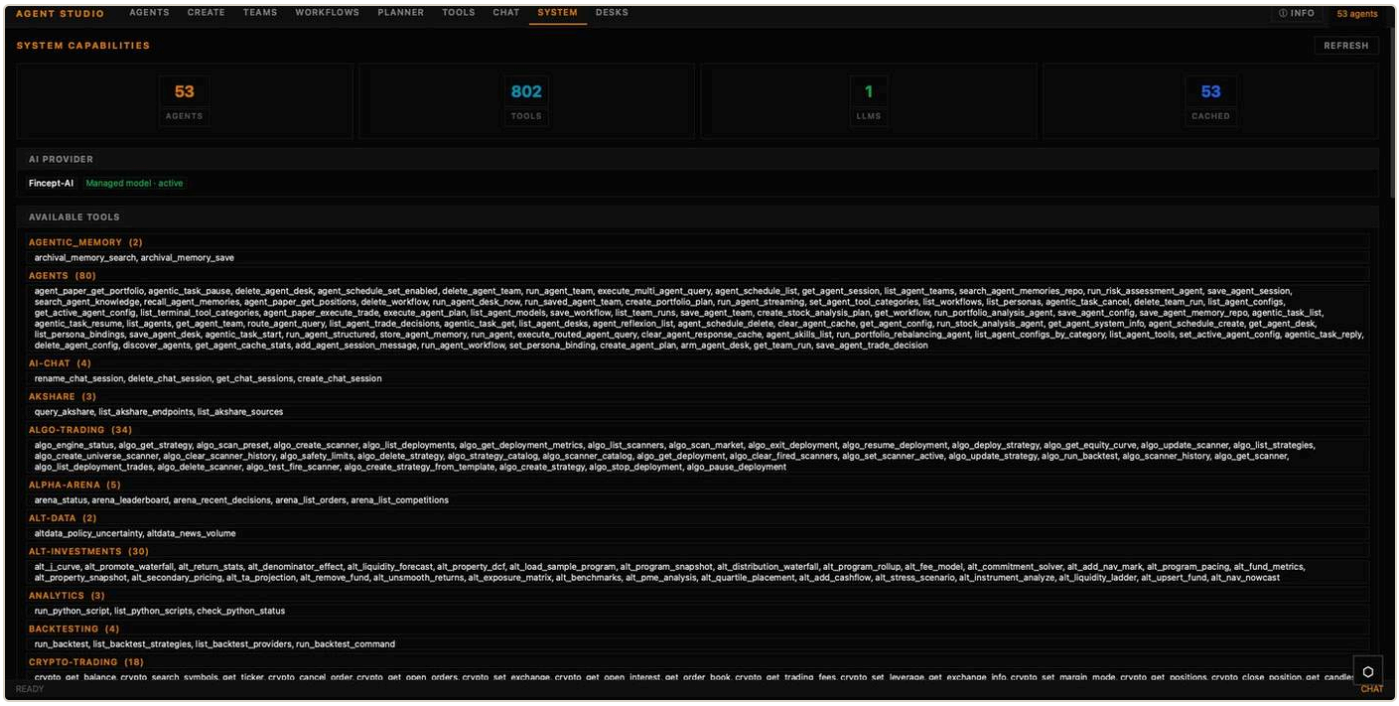
turning a query into a durable background task (§13).

> **Honest notes.** There is no Stop button — CLEAR resets local state but does not cancel an in-flight request — and the conversation is not saved; closing or clearing loses it. Chatting spends credits.

10 SYSTEM

SUB-TAB · SYSTEM

SYSTEM is a read-only capabilities dashboard for the agent subsystem: four headline tiles, the active provider, the full tool inventory by category, and build info. The only action is REFRESH.



SYSTEM — headline tiles (53 agents · 802 tools · 1 LLM · 53 cached), the Fincept-AI provider card, and the tool inventory grouped by category.

Element	What it shows
AGENTS · TOOLS · LLMS · CACHED	Live counts of discovered agents, total callable tools, providers (fixed at 1), and cached entries. Hover CACHED for an agents-vs-responses breakdown.
AI PROVIDER	`Fincept-AI` — *Managed model · active*.
AVAILABLE TOOLS	The full inventory: each category with its count and the comma-listed tool names.
SYSTEM INFO	VERSION, FRAMEWORK (`fincept-native`), and the enabled FEATURES: chat, single agent, teams, tasks, schedules, memory, HITL, tools.
REFRESH	Re-reads counts, tools and system info.

> **Honest notes.** The AI-provider row, the *LLMS = 1* tile, and the VERSION/FRAMEWORK/FEATURES strings are constants of the managed single-provider build — they report the subsystem's fixed shape, not a live runtime handshake. There is no per-run cost or backend-health readout here. The TOOLS tile (**802**) counts the full registry; the TOOLS-tab catalog (777) shows the assignable subset after the UI-only categories are hidden.

11 The agent engine

ENGINE

Behind all nine sub-tabs sits one engine. Understanding three things — the model, discovery, and how a run executes — explains the behaviour (and the cost) of everything in Agent Studio.

11.1 One model, one meter

Every agent, team, workflow, plan, desk and chat turn is pinned to **Fincept-AI**, the terminal's managed model, reached over an authenticated gateway. There is no third-party key and any model a caller supplies is ignored. Consequently **every run that invokes the model spends credits**, on the same meter as AI Chat. The only free operations are the deterministic ones: agent discovery, tool/model/system listings, and the keyword-based *Auto-Route* query router.

11.2 Where the 53 agents come from

The catalog is discovered from bundled persona definitions — **16 single-agent files** plus **37 personas** across four domain catalogs (Trader/Investor, hedge-fund, economic, geopolitics) — for a total of **53**, merged with any custom agents you've saved. These definitions are read-only bundled data; discovery is fast and cached.

11.3 How a run executes & how tools are grounded

Interactive work (a single-agent test, a persona chat) runs **in-process** for low latency; durable and multi-agent work (teams, long-running tasks, workflows) routes to the terminal's managed **agent runtime**. In every case the model can reach **live terminal tools** — markets, portfolio, EDGAR, news and hundreds more — scoped per agent: UI-control categories (navigation, system, settings, ai-chat, meta) are excluded by default, destructive tools require an explicit opt-in plus an approval gate, and a dry-run mode lets you test prompts without real execution.

Object	Where it is stored
Custom agents · teams · desks · workflows · persona bindings	Your account settings (synced), so they follow you across machines and survive restart.
Team run history	A dedicated server-side history endpoint.
The 53 built-in personas	Read-only definitions bundled with the app.
Durable tasks · schedules · skills · memories	Persisted so tasks resume and schedules fire even after the app restarts.

> **A subsystem in transition.** The agent engine is being migrated from an embedded engine to native in-process execution plus a managed server-side runtime. Interactive persona chat works locally regardless; some server-backed features (durable tasks, schedules, multi-agent runs) depend on the managed runtime being available for your account. If a durable run can't reach the runtime it surfaces as an error rather than a silent no-op.

12 Driving Agent Studio by AI

MCP · REFERENCE

Agent Studio is not only a screen — it is itself exposed to the terminal's AI as a suite of **~78 tools** in the `agents` category, so you can build, run and manage agents by asking in natural language. Everything you can click here, the assistant can do for you.

Group	Count	Representative tools
Discovery	8	`list_agents`, `discover_agents`, `list_agent_tools`, `list_agent_models`, `get_agent_system_info`, `get_agent_cache_stats`.
Execution	12	`run_agent`, `run_agent_structured`, `route_agent_query`, `execute_multi_agent_query`, `run_agent_team`, `run_agent_workflow`, `run_stock_analysis_agent`, `run_risk_assessment_agent`.
Repos & CRUD	28	Config CRUD (`save_agent_config`, `get_agent_config`, `set_active_agent_config`), plans (`create_agent_plan`, `execute_agent_plan`), memory/knowledge, workflow CRUD, sessions, paper-trade & decision logging.
Studio parity	24	Teams (`save_agent_team`, `run_saved_agent_team`, `list_team_runs`), personas & bindings (`list_personas`, `set_persona_binding`), tool grants (`set_agent_tool_categories`), agentic tasks (`agentic_task_start`, `agentic_task_reply`), schedules.
Desks	6	`list_agent_desks`, `save_agent_desk`, `arm_agent_desk`, `run_agent_desk_now`, `delete_agent_desk`.

Adjacent tool families round out the subsystem: an agentic **working-memory / playbooks** set (`task_set_goal`, `select_playbook` ...), Node-Editor **DAG execution** (`run_workflow_dag`, `get_workflow_execution_status` ...), and **archival memory** (`archival_memory_save/search`).

> **What this means for an institution.** A desk head can drive the entire studio from the chat bar — "build a hedge-fund team of the four macro agents, scope them to economics and markets tools, run it on *the effect of a 50bp cut on EM debt*, and save it as *Macro Desk*" — without touching a single control.

13 Advanced — Agentic Mode

ADVANCED · DEVELOPER

A tenth sub-tab, **AGENTIC**, is hidden by default. Enabling *Agentic Mode* in Settings ▸ Developer reveals it and adds a **RUN AS TASK** toggle to CHAT. It turns a query into a **durable, resumable background task** that plans its own steps, uses tools, reflects, and can pause to ask you a question — surviving app restarts.

13.1 The task monitor

Region	What it does
Filter bar	Filter tasks by All / Running / Paused / Completed / Failed / Cancelled, with REFRESH.
Task list	One row per task: status · step progress · truncated query, colour-coded by state.
PLAN	The task's persisted plan with ✓ / ▶ / · markers per step.
BUDGET	Four meters — **tokens · cost · wall-time · steps** — showing used-vs-max. This is the closest thing to a live cost readout in Agent Studio.
AGENT NEEDS INPUT	A human-in-the-loop banner: when the agent pauses to ask, you type a reply and it resumes.
STEP LOG	A live event trace — step start/finish, reflection decisions, replans, budget stops, final answer.
PAUSE · RESUME · CANCEL · SCHEDULE… · LIBRARIES… · DELETE	Task controls; SCHEDULE registers a recurring cadence; LIBRARIES opens the Skills / Archival-memory / Reflexion inspector.

Honest notes. Agentic Mode is a developer feature — off by default, and shown here for completeness (this guide did not enable it, to leave your settings untouched). *DELETE* routes through cancel; soft-pause fidelity depends on the execution engine. Durable tasks spend credits as they burn their token/cost budget.

14 Quick reference & caveats

REFERENCE

14.1 How do I…

I want to…	Do this
Open Agent Studio	Command bar → `agent config` (or `agents`), or Navigate ▸ Tools ▸ Agent Config.
Try a built-in agent	AGENTS → click an agent → type in TEST QUERY → RUN AGENT.
Let the system pick the best agent	AGENTS or CHAT → turn on **Auto-Route**.
Build my own agent	CREATE → Name + Instructions → scope tools → SAVE AGENT (or EXP/IMP to move a config as JSON).
Run several agents on one question	TEAMS → add members → choose a mode → RUN TEAM.
Get a full stock work-up in one click	WORKFLOWS → Stock Analysis → enter a symbol → RUN WORKFLOW.
Push a result into a report	WORKFLOWS → after a run → **SEND TO REPORT**.
Turn a goal into steps	PLANNER → describe the goal → GENERATE PLAN → EXECUTE.
Give an agent more tools	TOOLS → pick the target → basket the tools → ASSIGN →.
Run a persona on a schedule	DESKS → configure → SAVE DESK → ARM DESK.
Check what the subsystem can do	SYSTEM → read the tiles, provider and tool inventory.

14.2 Honest caveats worth knowing

- **One managed model.** Everywhere shows `Fincept-AI`; there is no model picker and no bring-your-own-key.
- **Credits.** Every run that invokes the model — agent, team, workflow, plan, desk, chat — draws from the same meter as AI Chat. Discovery, tool/model listings and keyword routing are free. No per-run cost is displayed (except the Agentic budget meter).
- **AGENTS.** The tools list is view-only here (edit on the TOOLS tab); the Output schema is ignored under Auto-Route.
- **CREATE.** The in-form MCP-servers list isn't saved; there's no delete confirmation; unchecking a feature drops its sub-config.
- **WORKFLOWS.** *Options Flow Scan* exposes no symbol field; in the default path each workflow runs as a single agent turn.
- **PLANNER.** Editing a step cell doesn't change what runs; execution collapses to one model turn; history isn't durable.
- **CHAT.** No Stop button; conversations aren't saved.
- **SYSTEM.** The provider row and version/framework/feature strings are constants of the managed build, not a live probe.
- **Counts.** SYSTEM's **802** is the full tool registry; the TOOLS catalog's **777** is the assignable subset after UI-only categories are hidden.

Fincept Research Desk. This guide documents the Agent Studio (Agent Config) tab as shipped in Professional Edition v5.0. Screenshots are live captures; control names and options are transcribed from the shipping build.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Node Editor

A complete, control-by-control walkthrough of the NODES tab — the terminal's visual workflow builder: 232 node types, a parallel execution engine, and a paper-first path to automated trading.

`>_ 232 node types · 14 categories · parallel DAG engine · 15 templates`

Covers · **palette, canvas and properties** · **how a workflow runs** · **the full node catalogue**
cost & network · trading safety and the audit log · expressions · templates · deploy · the 4 agent tools

Demonstrated live on a working Economic Dashboard workflow.

Contents

23 SECTIONS

GETTING ORIENTED

01 What the Node Editor is 3
02 Anatomy of the workbench 4
03 The toolbar, button by button 6

BUILDING A WORKFLOW

04 The node palette 7
05 The canvas: move, zoom, connect 9
06 A node up close 10
07 The properties panel 11

RUNNING IT

08 How a workflow actually runs 12
09 The execution results panel 13

THE NODE CATALOGUE

10 Triggers, Core, Control Flow 15
11 Data Transform and Data Format 16
12 Market Data and Analytics 17
13 Utilities, Files and Data Sources 18
14 Trading, Safety, Agents and MCP 19

COST, SAFETY AND LIMITS

15 What costs money, what touches the network 20
16 Trading safely: paper, approval, audit 21
17 Expressions and data flow 22

WORKING PRACTICE

18 The fifteen templates 23
19 Saving, loading, import and export 24
20 Deploy — and what it does not do 25
21 Driving workflows from AI Chat 26
22 A worked example: Economic Dashboard 27

REFERENCE

23 Quick reference and honest notes 28

About this guide. Every screenshot is a live capture of the NODES tab in the shipping build, on a real 17-node workflow. The section-09 execution was scoped to a single display node, so no fetch, order or metered AI call was triggered.

01 What the Node Editor is

ORIENTATION · 01

The Node Editor is the terminal's **visual workflow builder**. You drag nodes onto a canvas, wire their ports together, and press **EXECUTE**; the engine works out the running order, runs independent branches in parallel, and shows you what every node returned. It is how a repeatable piece of desk work — pull these series, compute this, check that limit, place that order, write that file — stops being a manual routine and becomes a graph you can run again.

It ships with **232 node types**: 114 built-in types across 14 categories, plus one node automatically generated for each of the 118 connectors in the Data Sources catalogue. Between them they cover market data, analytics, transformation, files, HTTP, Python, notifications, broker trading, pre-trade safety checks and the AI agents.

Fig 1 — The command strip: the workflow name and its status on the left, then history, file, template, deploy and execute actions on the right.

Where to find it

Route	How
Top tab bar	**NODES** — it is one of the 14 FKeyBar tabs
Command bar	Type `node` or `nodes`, then pick the row

At a glance, in the Node Editor you can…

- Build a data or trading workflow by dragging nodes
- Wire outputs to inputs and branch on conditions
- Run the whole graph, or one branch from a chosen node
- Watch every node turn green or red as it runs
- Read each node's JSON output in place
- Start from one of 15 worked templates
- Save, load, import and export workflows as JSON
- Trade on paper by default, with pre-trade safety nodes
- Have the AI assistant run a saved workflow for you

> **One sentence of orientation.** Nothing runs until you press **EXECUTE** — or ask an agent to run a saved workflow. There is no background scheduler in this build, which is the single most important thing to understand before you rely on it; section 20 sets out exactly what **DEPLOY** does.

02 Anatomy of the workbench

Three vertical panels under a command strip, with a results drawer along the bottom and a minimap floating over the canvas.

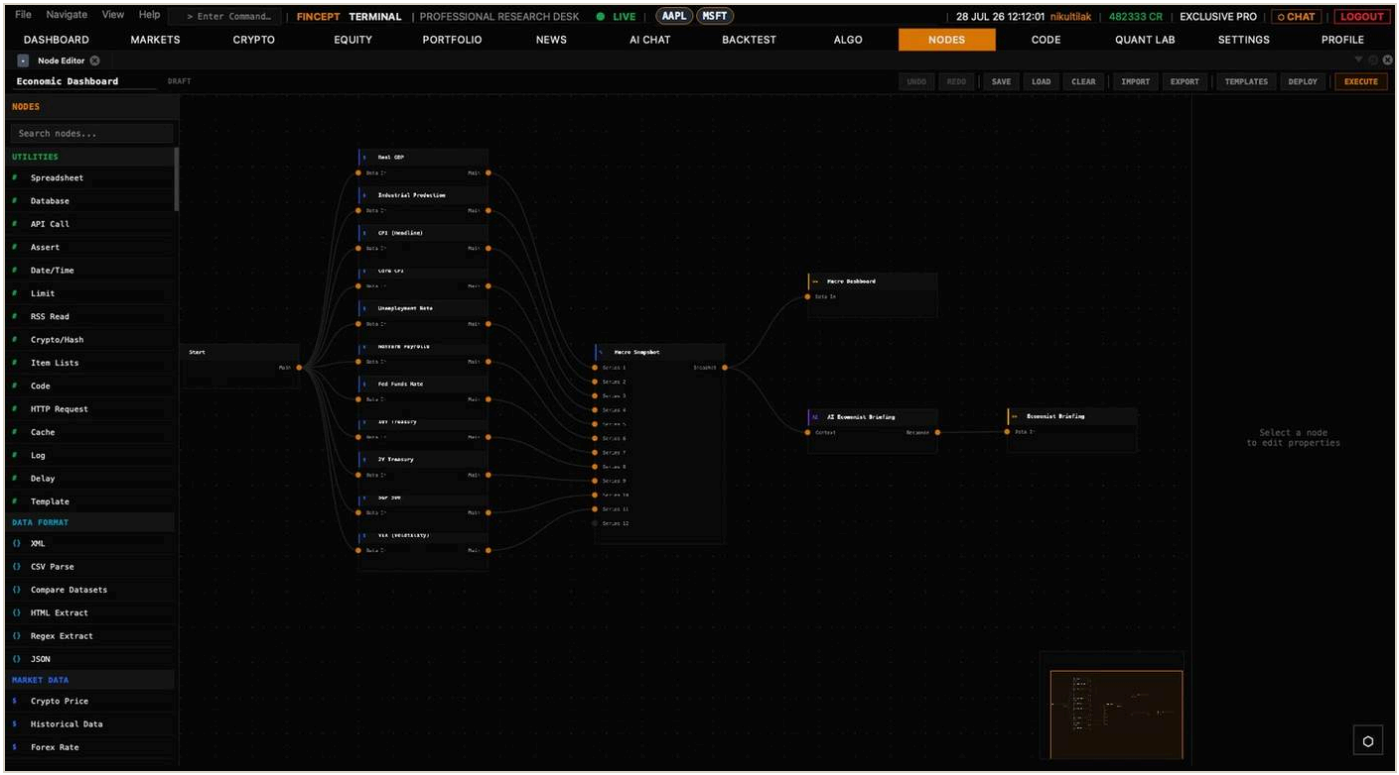


Fig 2 — The NODES tab holding a 17-node Economic Dashboard workflow.

#	Region	What it is for
1	Command strip	Editable workflow name, a status badge (**DRAFT**), and eleven actions. Section 03.
2	Node palette	240 px column: a search box over every node type, grouped by category with a count per group. Drag from here onto the canvas. Section 04.
3	Canvas	The graph itself on a dot grid — nodes, ports and curved edges. Zoom with the wheel, pan with the middle button. Section 05.
4	Properties panel	280 px column: the selected node's name, parameters, per-node settings and — after a run — its output. Section 07.
5	Execution results	A collapsible drawer that fills in as the graph runs. Section 09.
6	Minimap	A live thumbnail of the whole graph in the canvas's bottom-right corner, with the current viewport outlined.

Fig 3 — The minimap: the amber rectangle is your viewport.

Fig 4 — The results drawer when nothing has run yet.

> Both side panels are fixed-width and the canvas takes the rest, so the workbench keeps its proportions as the window resizes. All three dividers can still be dragged if you want more canvas or a wider properties column.

03 The toolbar, button by button

Control	What it does
`Workflow name`	Editable in place. This is the name a save, a deploy and the load picker use
`DRAFT`	Status badge for the workflow you are editing
`UNDO / REDO`	Walk the edit history. Also **Ctrl+Z** / **Ctrl+Y**
`SAVE`	Saves the graph under the current name, creating it on first save
`LOAD`	Lists your saved workflows with their last-updated time and opens the one you pick
`CLEAR`	Removes every node and edge — **asks first**, and cannot be undone
`IMPORT`	Reads a workflow from a `.json` file and replaces what is on the canvas
`EXPORT`	Writes the current graph to a `.json` file. Section 19
`TEMPLATES`	Opens the fifteen worked examples. **Choosing one clears the canvas first.** Section 18
`DEPLOY`	Name + describe the workflow, then save it as a draft or deploy it. Section 20
`EXECUTE`	Runs the graph. While a run is in flight the same button stops it

> **Three buttons discard work without an undo: CLEAR** (confirms first), **TEMPLATES** (does *not* confirm — picking a template wipes the canvas immediately) and **IMPORT** (replaces the canvas with the file's contents). Save or export first if the graph matters.

Keyboard

Keys	Action
`Ctrl+Z / Ctrl+Y`	Undo / redo
`Delete or Backspace`	Delete the selected nodes
`Ctrl+A`	Select everything on the canvas
`Ctrl+C`	Copy the selected nodes, plus any edges between them
`Ctrl+V`	Paste them 40 px down and right, renamed *(copy)*, with fresh ids

> **Copy and paste understand subgraphs.** Select three wired nodes, copy, paste — you get three new nodes *and* the wiring between them, which is the fastest way to repeat a pattern like fetch → check → act across several symbols.

04 The node palette

Every node type the build knows about, grouped by category, with a count beside each group heading. Each row shows the category's accent colour, a short glyph and the node's name; hovering it shows the node's one-line description.

Fig 5 — The palette, shown here as two columns for legibility: the search box, then UTILITIES (15), DATA FORMAT (6), MARKET DATA (14) and the rest.

Search is the fast path

The box at the top filters as you type, and it matches three things: the node's **name**, its **description**, and its internal **type id**. Because the palette holds 232 types, searching beats scrolling for anything you can name.

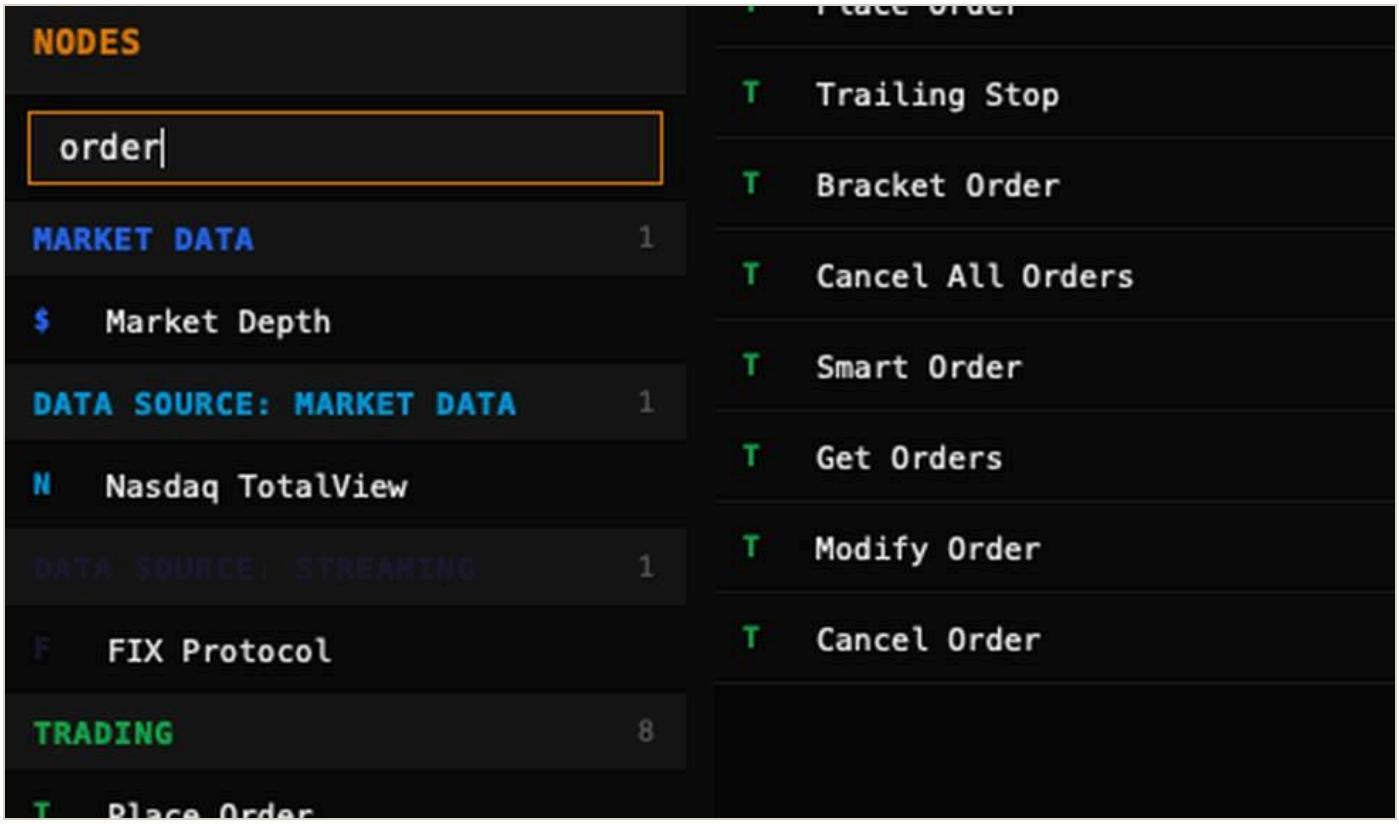


Fig 6 — Searching `order`: one Market Data hit, two auto-generated Data Source nodes, and the eight Trading nodes that place or manage orders.

Detail	Behaviour
`Adding a node`	Press and drag a palette row onto the canvas; the node appears where you drop it, with every parameter at its default
`Group counts`	The number on the right of each heading is how many types that category holds — useful for spotting where the depth is
`Data Source groups`	Categories named **Data Source: …** are generated from your Data Sources connectors, one node per connector
`Ordering`	Groups and rows are **not alphabetised** — the palette follows the registry's internal order, so use search rather than expecting a stable position

> **A node is created with defaults, not blanks.** Dropping *Place Order* gives you a node already set to paper mode, market order, quantity 1 — so a freshly dropped node is runnable and safe to inspect before you change anything.

05 The canvas: move, zoom, connect

Gesture	Result
`Mouse wheel`	Zoom in and out, clamped to a sensible minimum and maximum
`Middle-button drag`	Pan the canvas
`Ctrl + left-drag`	Pan the canvas — the same thing without a middle button
`Left-drag on empty space`	Rubber-band select every node inside the rectangle
`Left-drag a node`	Move it; its edges follow
`Left-drag from a port`	Draw a new edge; release on a compatible port to connect
`Click a node`	Select it and load it into the properties panel
`Right-click a node`	The node menu: Duplicate · Disable · Execute From Here · Delete

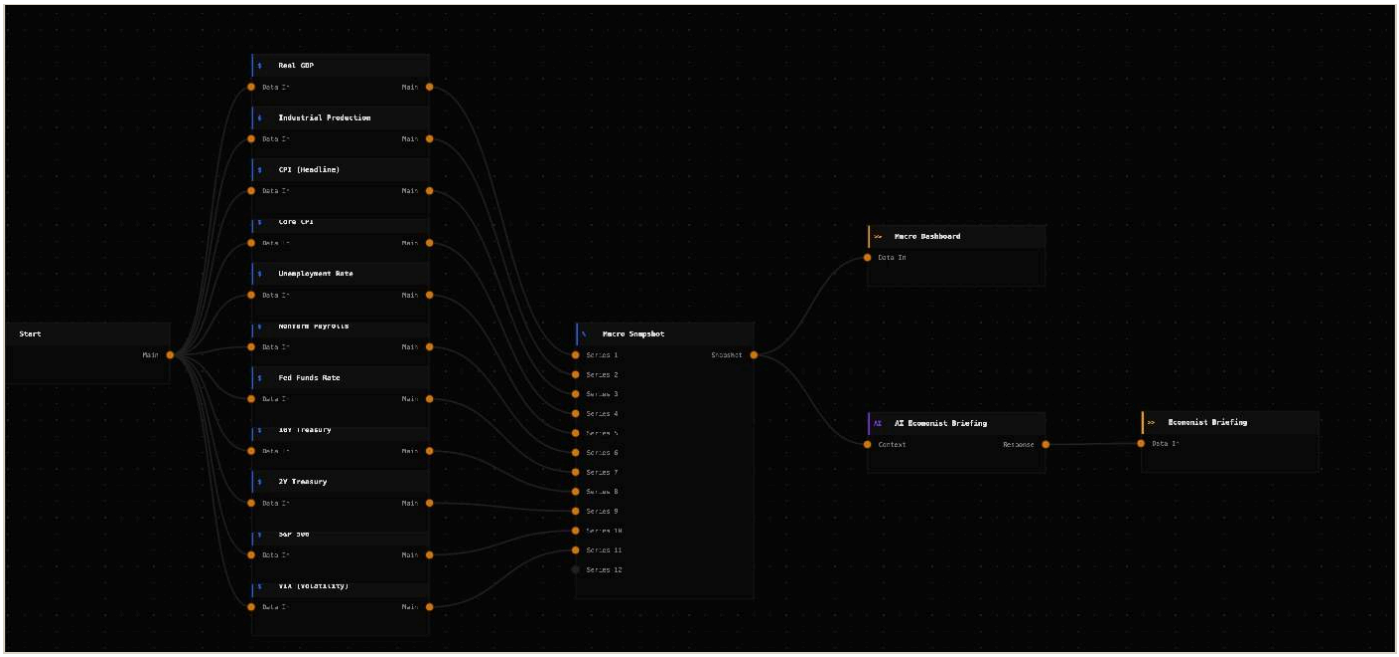


Fig 7 — A live graph: a Manual Trigger fans out to twelve FRED series nodes, they converge on an analytics node with twelve inputs, and that feeds a dashboard and an AI briefing.

What can connect to what

The editor refuses connections that cannot mean anything, so a wrong drag simply does not stick:

- **An output must meet an input.** Output-to-output and input-to-input are rejected.
- **A node cannot connect to itself.**
- **Connection types must be compatible.** Most ports are of the universal *Main* type, which connects to anything; specialised ports only accept their own type.

> **Cycles are allowed on the canvas but rejected at run time.** Nothing stops you drawing a loop, but pressing EXECUTE on a graph that contains one stops immediately with *"Workflow contains a cycle"* and runs nothing. For repetition use the **Loop** node rather than a physical cycle.

06 A node up close

Fig 8 — An analytics node with twelve inputs on the left and one output on the right.

Part	What it is
`Accent bar + glyph`	The category's colour and icon, so families are recognisable at a glance
`Title`	The node's name — editable in the properties panel, and independent of its type
`Input ports`	Left edge, one dot per input, labelled. Unconnected inputs simply receive nothing
`Output ports`	Right edge. Branching nodes have several — *True* / *False* on If/Else, numbered ports on Switch
`Border state`	Idle, **running**, **completed** or **error**, set live during a run

Edges animate while the node upstream of them is running, which makes the progress of a wide graph readable at a glance.

The node menu

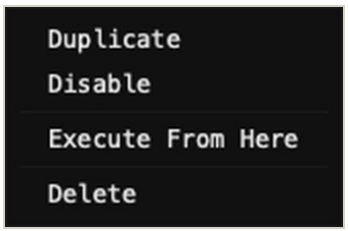


Fig 9 — Right-clicking a node.

Item	What it does
`Duplicate`	A copy 40 px down and right, named *(copy)*, with no edges
`Disable`	Marks the node disabled — see the warning below
`Execute From Here`	Runs **this node and everything downstream of it**, leaving the rest alone. The fastest way to retry a failed branch
`Delete`	Removes the node and its edges immediately, with no confirmation

> **Disabling a node also disables everything downstream of it.** A disabled node is dropped from the run, and because its children never receive their input they are dropped too. Use it to switch off a whole branch — not to skip one step in the middle of a chain, which stops the chain there.

07 The properties panel

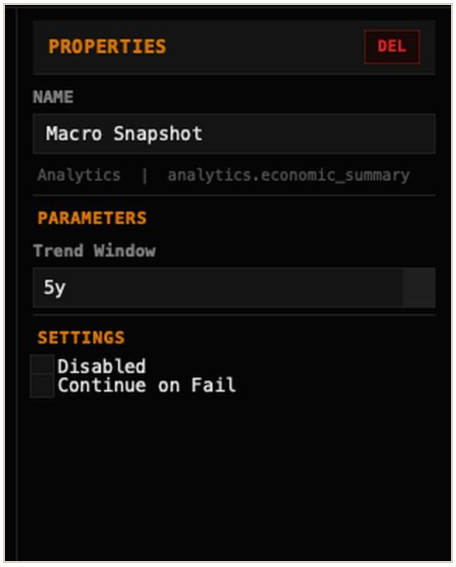


Fig 10 — The panel for the selected analytics node.

Section	Contents
PROPERTIES + DEL	Header, and a red **DEL** that deletes the selected node
NAME	The node's label on the canvas. Rename freely — the type underneath does not change
type line	Category and internal type id, e.g. `Analytics | analytics.economic_summary` — the id to quote in a bug report
PARAMETERS	One widget per parameter the type declares, in the type's own order
SETTINGS	**Disabled** and **Continue on Fail** — the two per-node execution flags
OUTPUT	Appears once the node has run, with its result and duration

The parameter widgets

Thirteen widget kinds are used across the catalogue, chosen by the parameter's declared type:

Widget	Used for
`string · number · boolean`	Plain values — text field, numeric field, checkbox
`select`	A fixed option list (order side, product type, interval…)
`code · json`	A multi-line editor; the JSON variant validates as you type
`expression`	A condition field — read section 17 before relying on it
`file_managed`	Picks a file from File Manager, filtered to the extensions the node accepts
`agent_select · team_select · llm_select`	Pick a configured agent, team or model from Agent Studio
`mcp_tool_select`	Pick one of the terminal's MCP tools
`datasource_connection_select`	Pick a saved connection for that connector

> **Continue on Fail is the flag that matters most.** By default one failed node aborts the whole run. Tick it on the nodes whose failure is acceptable — an optional enrichment fetch, a notification — and the rest of the graph carries on. Section 08 explains the exact policy.

08 How a workflow actually runs

Pressing **EXECUTE** hands the graph to the engine, which does the same five things every time.

Step	What happens
`1 · Cycle check`	The whole graph is scanned first. If it contains a cycle the run is abandoned before anything executes, with the error *Workflow contains a cycle*
`2 · Order`	A topological sort produces a legal running order. **Disabled nodes are removed — and so is everything downstream of them**
`3 · Parallel launch`	Every node whose inputs are ready starts **at the same time**. A twelve-series fan-out fetches all twelve at once, not one after another
`4 · Propagate`	As each node finishes, its output becomes its children's input and any child now ready is launched. Repeat until the graph is exhausted
`5 · Finish`	The run reports success only if every node succeeded, with a total duration

How data moves between nodes

- **A node receives the outputs of its connected upstream nodes**, in edge order, as JSON.
- **If/Else** marks its output with the branch it took; only the edge leaving the matching *True* or *False* port carries data onward.
- **Switch** works the same way with numbered ports, and **Loop** splits into an *item* port and a *done* port.
- A node whose inputs were all filtered out by branching is **skipped** rather than run.

When something fails

Situation	Engine behaviour
`A node returns an error`	The run stops and reports failure — unless that node has **Continue on Fail** ticked, in which case the graph carries on
`A node throws`	Caught and reported as a node failure; the terminal does not crash
`A node has no executor`	It **passes through** and counts as a success — see the warning
`You press EXECUTE again`	Ignored while a run is in flight; the button stops the run instead
`You stop a run`	In-flight nodes are allowed to finish, then the run reports *Execution stopped*

> **A node with no implementation silently succeeds.** Unimplemented types pass their input through, are marked green, and count towards a successful run. In this build that applies to the **Execute Workflow** node (sub-workflows), so a graph that relies on it will report success while having done nothing. Check a node's OUTPUT if a result looks too clean.

09 The execution results panel

The drawer along the bottom is the run log. It is collapsed to a single strip until a run starts, and the chevron on its left folds it away again.

Fig 11 — After a run: the node count, the state badge (**DONE** with the total duration), **CLEAR** and **COPY**, and one row per node with a tick, its type and its own duration.

Element	Meaning
N `node(s)`	How many nodes the run covered; error count is appended when there are failures
`State badge`	**IDLE** · **RUNNING…** · **DONE** with total time · **STOPPED** · **FAILED** with the reason
`Node rows`	One per node as it completes — tick or cross, the node name, its type id, and its duration in ms
`CLEAR`	Empties the list. It does not touch the graph
`COPY`	Copies every result to the clipboard (the button confirms with *COPIED!*) — the quickest way to paste a run into a note or a ticket

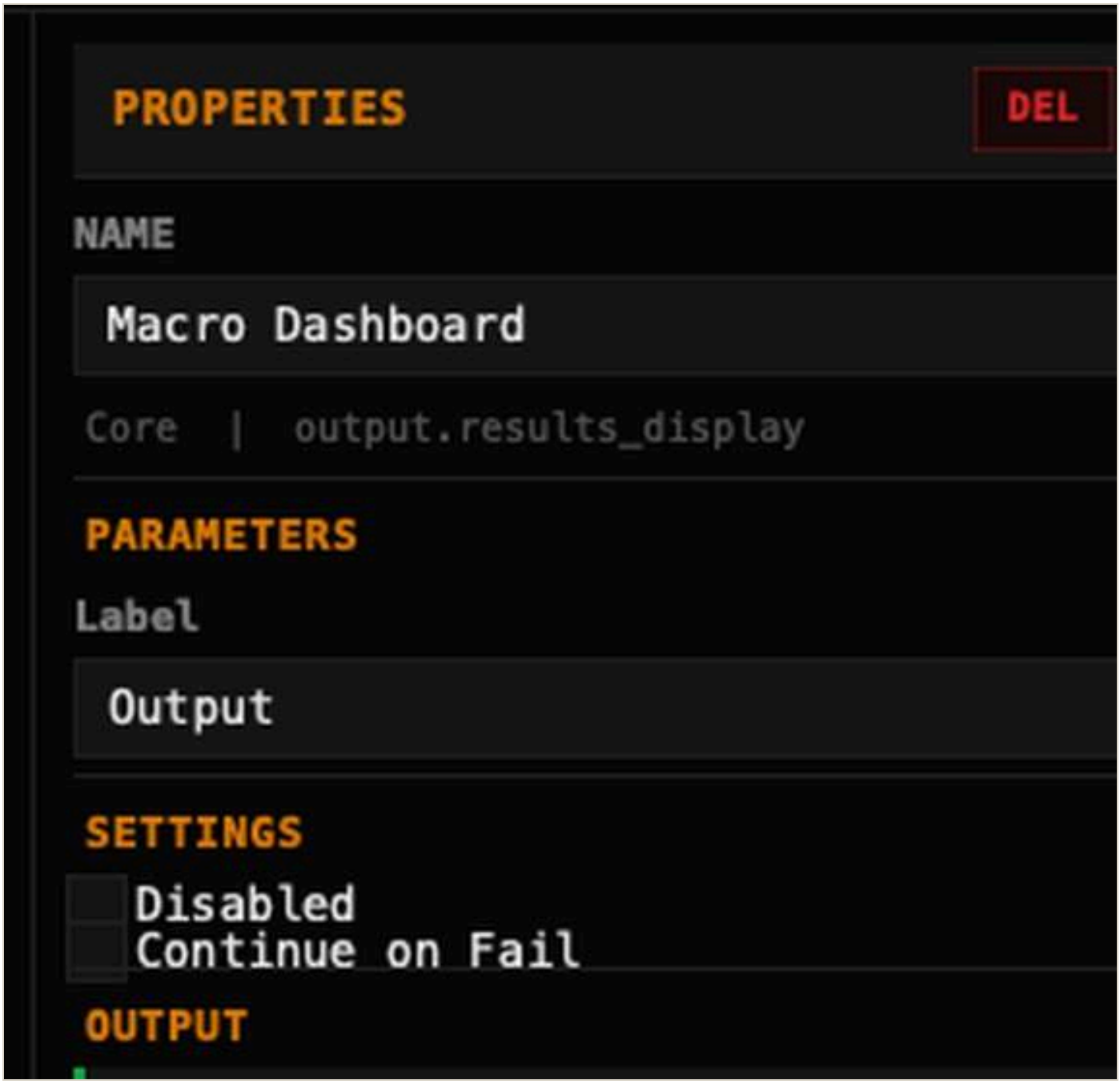


Fig 12 — The same result in the properties panel: selecting a node after a run adds an **OUTPUT** section showing what it returned and how long it took.

> **Two views of the same run, for two purposes.** The drawer answers *did the graph work?* — scan for crosses. The properties panel answers *what did this node actually return?* — select the node and read its OUTPUT. Both stay available until you clear them or run again.

> **The demonstration above was scoped deliberately.** It used **Execute From Here** on a terminal display node, so exactly one node ran: no market data was fetched, no order was placed and no metered AI call was made. It is a useful habit for testing a graph's plumbing before you let the expensive half of it run.

10 Catalogue — Triggers, Core, Control Flow

Sections 10 to 14 list all 114 built-in node types by category. Names are exactly as they appear in the palette.

Triggers `(8)` — where a workflow starts

Node	What it is for
`Manual Trigger`	The ordinary start node: fires when you press EXECUTE
`Schedule Trigger`	Carries a cron expression as a parameter
`Market Hours Cron`	A market-hours-aware schedule expression
`Market Event`	Declares an event of interest — volatility spike, circuit breaker, open, close, earnings
`Price Alert`	Declares a price threshold
`News Event`	Declares a news trigger
`Portfolio Drift`	Declares a drift threshold on the portfolio
`Webhook`	Declares an inbound webhook entry point

> **All eight triggers are start markers, not live listeners.** They emit "triggered" when the graph is run and nothing more: there is no scheduler, no event bus subscription and no inbound webhook server in this build. A Schedule Trigger set to `*/5 * * * *` will not run every five minutes — see section 20.

Core `(2)`

Node	What it is for
`Set Variable`	Adds a key and value to the JSON flowing through
`Results Display`	A terminal node that labels and shows whatever reaches it

Control Flow `(8)`

Node	What it is for
`If / Else`	Branch on a condition; separate True and False output ports
`Switch`	Route to one of several numbered outputs
`Loop`	Iterate a collection, with *item* and *done* outputs
`Merge`	Combine several inputs into one payload
`Split`	Fan one payload out to several outputs
`Wait`	Pause the branch for a set time
`Error Handler`	Catch a failure and carry on down a recovery path
`Execute Workflow`	**Not implemented in this build** — passes through (section 08)

11 Catalogue — Data Transform and Data Format

Data Transform `(13)` — reshaping a payload

Node	What it is for
`Filter`	Keep only the items that match a condition
`Sort`	Order items by a field
`Aggregate`	sum · avg · min · max · count over a field of an array
`Group By`	Group items by a field
`Join`	Combine two datasets on a key
`Deduplicate`	Drop repeated items
`Pivot Table`	Pivot rows into columns
`Reshape`	Restructure nested data
`Normalize`	Scale values to a common range
`Resample`	Change the time frequency of a series
`Rolling Window`	Windowed calculations across a series
`Lag / Lead`	Shift a series forward or backward
`Map`	**Pass-through in this build** — its Expression parameter is ignored (section 17)

Data Format `(6)` — parsing what arrives

Node	What it is for
`JSON`	Parse or stringify JSON
`CSV Parse`	Turn CSV text into structured rows
`XML`	Parse XML
`HTML Extract`	Pull values out of an HTML document
`Regex Extract`	Pull values out of text with a regular expression
`Compare Datasets`	Diff two datasets

> **Aggregate expects an array.** Several transform nodes are strict about their input shape and fail with a clear message when it is wrong — for example *transform.aggregate requires a JSON array input*. When a transform fails, check the upstream node's OUTPUT to see whether it produced an object where an array was expected.

12 Catalogue — Market Data and Analytics

Market Data (14) — fetching the numbers

Node	What it is for
Get Quote	A current quote for a symbol
Historical Data	A price history series
Ticker Stats	Summary statistics for a symbol
Fundamentals	Company fundamentals
Options Chain	The option chain for an underlying
Market Depth	Order-book depth
Market News	Headlines for a symbol or market
SEC Filings	Filings from EDGAR
Insider Trades	Insider transaction data
Stock Screener	Screen the universe on criteria
FRED Economic Data	A FRED macro series
US Treasury Yield Curve	The Treasury curve
Forex Rate	A currency pair rate
Crypto Price	A crypto price from a public exchange endpoint

Analytics (14) — computing on it

Node	What it is for
Technical Indicators	Indicator series over a price history
MA Crossover	Moving-average crossover signals
Performance Metrics	Return and risk statistics
Sharpe / Sortino	Risk-adjusted ratios
Max Drawdown	Worst peak-to-trough decline
Risk Analysis	A risk summary for a book
Correlation Matrix	Pairwise correlations across symbols
Portfolio Optimization	Optimised weights for a set of assets
Factor Model	Factor exposures
Monte Carlo Sim	Simulated outcome distributions
Regime Detection	Identify market regimes
Pairs Trading	Pair spread and signal analysis
Backtest Engine	Run a strategy over history
Economic Summary	A macro snapshot across many series — the node at the centre of section 22

13 Catalogue — Utilities, Files and Data Sources

Utilities (15)

Node	What it is for
HTTP Request · API Call	Call an HTTP endpoint — GET/POST/PUT/DELETE with headers and a body
Code	Run Python (or JavaScript) inside the workflow
Database	Run a query — select, insert, update, delete or raw
Spreadsheet	Read or write spreadsheet data
RSS Read	Read a feed
Cache	Store and reuse a value between runs
Template	Fill a text template
Date/Time	Date arithmetic and formatting
Crypto/Hash	Hashing and encoding
Item Lists	List manipulation
Limit	Cap how many items continue
Delay	Wait before continuing
Log	Write a line to the terminal log
Assert	Fail the run unless a condition holds (its own syntax — section 17)

Files (6)

Node	What it is for
File Operations	Read, write, move and delete managed files
Convert to File	Turn a payload into a file
Binary File	Handle binary content
Compress	Zip content
PDF Report	Render a PDF
Chart Image	Render a chart as an image

Data Sources (118 generated)

One node per connector in the Data Sources catalogue, grouped into palette categories named **Data Source: Market Data**, **Data Source: Relational Databases**, and so on. Each carries a **Saved Connection** selector plus that connector's own fields, so a node can either use a connection you configured in the Data Sources tab or take its settings inline.

> **Files land in File Manager.** Nodes that write a file register it with the File Manager just as the rest of the terminal does, so a workflow that ends in *PDF Report* or *Convert to File* leaves its output somewhere you can find, preview and download later.

14 Catalogue — Trading, Safety, Agents and MCP

Trading (17)

Group	Nodes
Place	Place Order · Bracket Order · Smart Order · Scale In · Trailing Stop
Manage	Modify Order · Cancel Order · Cancel All Orders · Close Position · Close All Positions
Read	Get Quote · Get Orders · Get Position · Get Positions · Get Holdings · Get Balance
Notify	Trading Alert

Safety (7) — pre-trade gates

Node	What it checks
`Risk Check`	Position size and volatility limits together
`Position Size Limit`	Maximum position sizing
`Loss Limit`	Daily and weekly loss limits
`Max Drawdown Check`	Halts when drawdown exceeds a threshold
`Correlation Check`	Blocks a trade too correlated with existing positions
`Volatility Filter`	Skips a trade when volatility is too high
`Trading Hours Check`	Confirms the trade is inside market hours

Agents (3) and MCP (1)

Node	What it does	Cost
`AI Agent`	Runs one of your configured agents over the incoming payload	**Credits**
`AI Team`	Runs a configured team	**Credits**
`Tool Picker`	Asks the model which tool to use for a query	**Credits**
`MCP Tool`	Calls any of the terminal's MCP tools directly — no model in the loop	Free

> **Safety nodes are gates you place, not a policy that applies itself.** They branch — pass one way, fail the other — so a compliant trading graph reads *signal → Risk Check → (pass) Place Order*. Nothing forces a Place Order node to sit behind one, which is why section 16 exists.

15 What costs money, what touches the network

COST · 15

Most of this tab is free and local. Exactly one family is metered.

Family	Cost	Where the work happens
Agents — AI Agent, AI Team, Tool Picker	**Fincept credits**	The managed Fincept-AI service, through the terminal's LLM path
Market Data	Free	Public and keyed market endpoints via the terminal's own data plumbing
Analytics	Free	Local computation on data already in the flow
Data Transform · Data Format · Core · Control Flow	Free	Entirely local
Utilities	Free	Local — except HTTP Request / API Call and RSS Read, which call the address you give them
Files	Free	Local disk, registered in File Manager
Data Sources (118)	Free of Fincept charges	Whatever the connector talks to, with your own credentials
Trading	Free of Fincept charges	Paper book locally, or your broker in live mode (section 16)
Safety	Free	Local checks against limits and positions
MCP Tool	Free	In-process tool call

Every AI Agent node in a graph is a metered call per run. Three agent nodes in a workflow you run ten times is thirty metered calls. When you are iterating on the plumbing, disable the agent nodes (or use **Execute From Here** on the branch you are working on) and only run the whole graph once it is right.

Nothing here bills silently in the background. Because there is no scheduler, a workflow only ever costs you something when you press EXECUTE, when you press DEPLOY, or when an agent runs it through the tools in section 21.

16 Trading safely: paper, approval, audit

The Trading nodes can send real orders to a connected broker. Three things stand between a graph and your account, and it is worth knowing exactly what each one does.

1 · Paper is the default

The **Place Order** node has a **mode** parameter that defaults to **paper**. In paper mode the order goes to the terminal's paper book — no broker, no risk. Switching a node to **live** is a deliberate, per-node edit.

> **Mode is per node, not per workflow.** There is no global paper/live switch for the Node Editor, so a graph can contain both. Before running a trading workflow, select each order node and confirm its mode — and prefer keeping live nodes in a separate, clearly-named workflow from the one you iterate on.

2 · The approval gate

If the account a live order targets is configured to require approval, the node does not send the order: it queues it and returns `queued_for_approval` with a pending id, for you to release from the terminal's action centre. A graph cannot bypass that setting.

3 · The audit log

Every workflow start and finish, every node execution and failure, and **every order** — with its symbol, its parameters and a flag recording whether it was paper or live — is written to a local audit table with a timestamp. It is queryable by workflow and by date range, which is what makes a workflow-driven order defensible after the fact.

Layer	What it gives you
`Paper default`	A graph is safe to build and run before any account is involved
`Safety nodes (7)`	Explicit pre-trade gates you place in front of an order node — section 14
`Risk limits`	Built-in position, order, daily-trade and loss limits that the risk checks evaluate against
`Approval gate`	Accounts that require approval queue the order instead of sending it
`Audit log`	A timestamped, paper/live-flagged record of every order and every run

> **A pattern worth copying.** *Trigger → market data → analytics → Risk Check → Trading Hours Check → Place Order (paper) → Trading Alert.* Run it on paper until the signals and the gates behave, and change one thing at the end: the order node's mode.

17 Expressions and data flow

This section exists because the placeholders in some parameter fields promise more than the build delivers, and building on that promise wastes an afternoon.

How data actually reaches a node

Through the wire, as the payload. A node receives its upstream nodes' JSON outputs as its inputs and reads what it needs from them. That is the mechanism, and it works.

> **What does not work: expressions in ordinary parameters.** Typing `={{$input.symbol}}` into, say, the Get Quote node's *symbol* field does **not** substitute the upstream value — parameters are passed to nodes exactly as typed. There is a capable little expression engine in the build, but it is only wired into the condition fields listed below, not into parameters generally.

Where an expression works	Dialect	Example
If / Else — Condition	`={{path op value}}`, with dotted paths and an optional `$input.` prefix	`={{$input.value > 0}}`
Assert — Condition	`{{field}} op value` — no `=`, no dotted path; supports `== != > < >= <= contains`	`{{close}} > 100`
Map — Expression	**Ignored.** The node passes its input through unchanged	—

What the If/Else engine can do

- **Dotted paths** into the payload: `={$input.quote.last}`.
- **Array indexing**: `={$input.items.0.price}`.
- **Comparisons**: `> < >= <= == !=` against a number.
- **Bare truthiness**: leave the condition empty and the incoming value's own truthiness decides the branch.

Working with the grain

Instead of…	Do this
Templating a symbol into a fetch node	Set the symbol on the node, and use one node per symbol — copy/paste the subgraph (section 03)
Computing a parameter from upstream data	Use a **Code** node: it receives the payload and can do the arithmetic in Python
Branching on an upstream value	That is exactly what **If / Else** is for, and its condition field does resolve
Asserting an invariant mid-graph	**Assert**, in its own dialect, fails the run with your message

18 The fifteen templates

TEMPLATES builds a complete, wired, parameterised workflow on the canvas. They are the fastest way to see how the node families fit together.

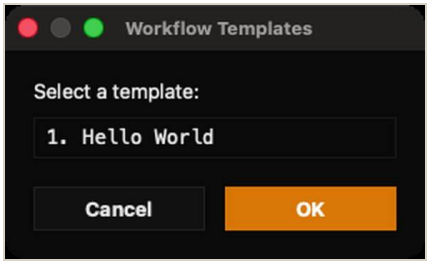


Fig 13 — The template picker. The list is a dropdown; all fifteen are tabulated opposite.

#	Template	Group
1	Hello World	Beginner
2	Stock Quotes	Beginner
3	Indicator Scanner	Market analysis
4	Sector Correlation	Market analysis
5	Economic Dashboard	Market analysis
6	Price-Alert Trade	Trading
7	Mean Reversion	Trading
8	Portfolio Rebalancer	Trading
9	Daily Risk Monitor	Risk & safety
10	Pre-Trade Compliance	Risk & safety
11	News Sentiment	Data & AI
12	AI Research Agent	Data & AI
13	Scheduled Report	Operations
14	Data Export	Operations
15	Webhook Automation	Operations

Picking a template clears the canvas without asking. Everything currently on it is discarded. Save or export your graph first — and note that the template also renames the workflow, so the next auto-save writes under the template's name.

Read a template before you run one. Templates 6–8 contain order nodes and templates 11–12 contain AI nodes. Open each node's properties first: confirm the order nodes are on **paper**, and remember that each AI node is a metered call (section 15).

Start with 1 and 2. *Hello World* is three nodes — trigger, Set Variable, Results Display — and running it teaches you the whole execute-and-read-results loop in about ten seconds, for free.

19 Saving, loading, import and export

Action	Behaviour
`SAVE`	Writes the graph, its name, every node's parameters and position, and all edges. First save creates the workflow; later saves overwrite it
`Auto-save`	**Every 30 seconds while the tab is visible**, provided the canvas is not empty — under the current name, creating the workflow if it does not exist yet
`LOAD`	Lists saved workflows with their last-updated timestamp and opens the one you choose, replacing the canvas
`IMPORT`	Reads a `.json` file and replaces the canvas
`EXPORT`	Writes the current graph to a `.json` file you choose
`On returning to the tab`	The zoom level, scroll position and the last workflow you had open are restored

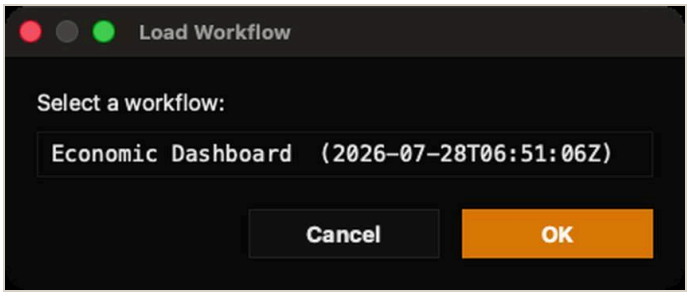


Fig 14 — The load picker: name plus the last-updated time, in UTC.

Fig 15 — The drawer is idle again after CLEAR — saving does not touch it.

> **Auto-save means edits are persistent by default.** There is no “discard changes”: thirty seconds after you break something, the broken version is the saved version. Before experimenting on a workflow that matters, press **EXPORT** — a JSON file is the only snapshot the tab offers.

The exported JSON

A readable, diffable format that belongs in version control: `id`, `name`, `description`, then `nodes[]` — each with `id`, `type`, `name`, `typeVersion`, `position`, `parameters`, `disabled` and `continueOnFail` — and `edges[]` with `source`, `target`, `sourceHandle` and `targetHandle`.

> **Workflows are stored locally and mirrored to your account.** The tab requests a cloud pull each time you open it, so a workflow saved on one machine can arrive on another — but export remains the portable, reviewable form, and the one to attach to a change request.

20 Deploy — and what it does not do

DEPLOY opens a small dialog: confirm the workflow's name, describe what it does, then choose **SAVE DRAFT** or **DEPLOY**.

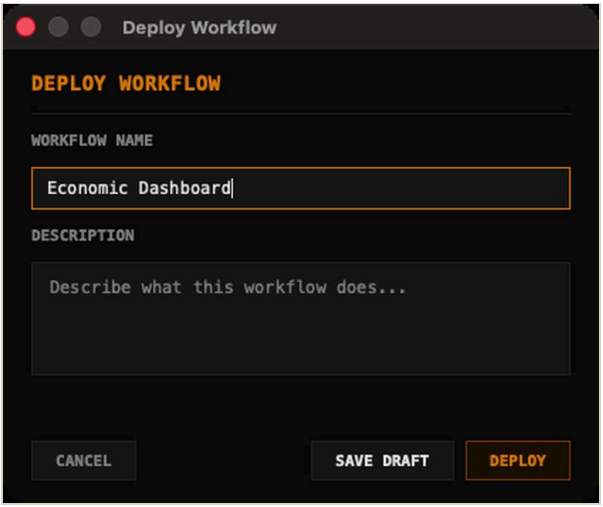


Fig 16 — The deploy dialog: name, description, and three choices.

Button	What actually happens
`CANCEL`	Nothing is saved
`SAVE DRAFT`	The workflow is saved with its description and marked **Draft**
`DEPLOY`	The workflow is saved and marked deployed — and then **executed once, immediately**

> **"Deployed" does not mean scheduled.** There is no background runner in this build: a deployed workflow does not wake up on a cron, does not listen for market events and does not expose a webhook. DEPLOY is best read as *"save with a description and run it now"* — and that run includes the graph's expensive nodes, so AI nodes are metered and live order nodes send orders. Use **SAVE DRAFT** when you only want to record the description.

How to get a workflow to run repeatedly

Approach	How
`By hand`	Open the tab and press **EXECUTE** — the honest answer for a daily routine
`From the assistant`	Ask AI Chat to run the saved workflow by name; it uses the tools in section 21
`On a terminal-side schedule`	Drive `run_workflow_dag` from whatever scheduled or agentic mechanism you already use in the terminal

> **Why document this so bluntly?** Because a Schedule Trigger node, a cron field and a button marked DEPLOY together imply an automation that is not there yet. Everything else on this tab does what it appears to do — this is the one place where the interface is ahead of the engine.

21 Driving workflows from AI Chat

Four agent tools in the `workflows` category let the assistant run graphs you have already built and saved.

Tool	What it does
`run_workflow_dag`	Runs a saved workflow by id — the whole graph, parallel per level — and returns every node's result
`run_workflow_from_node`	Runs one node and its downstream subgraph, for retrying a failed branch
`stop_workflow_execution`	Stops the run in flight
`get_workflow_execution_status`	Reports whether a run is in progress

Detail	Behaviour
What it can run	**Saved** workflows only, by id — save before you ask. `list_workflows` shows the ids
Concurrency	One run at a time; a second request is refused with a message telling you to wait or stop the first
Marked destructive	Both run tools are flagged as having side effects, because nodes fetch data, write files, call models and can place orders
What comes back	Per-node results with durations, and long outputs truncated so a big run stays readable

An agent running a workflow inherits everything the workflow does. If the graph contains a live order node, the agent's run places live orders; if it contains AI nodes, the run spends credits. Keep anything you are happy to hand to an assistant in its own paper-only workflow.

The useful pattern. Build and debug the graph by hand in this tab, save it under a clear name, then use the assistant as the trigger: *"run my Daily Risk Monitor workflow and tell me if anything breached."* The tools return every node's output, so the assistant can summarise the run rather than just reporting success.

22 A worked example: Economic Dashboard

WORKED EXAMPLE · 22

The workflow captured throughout this guide is template 5. It is a good model for a fan-out / fan-in graph, and it shows the engine's parallelism doing real work.

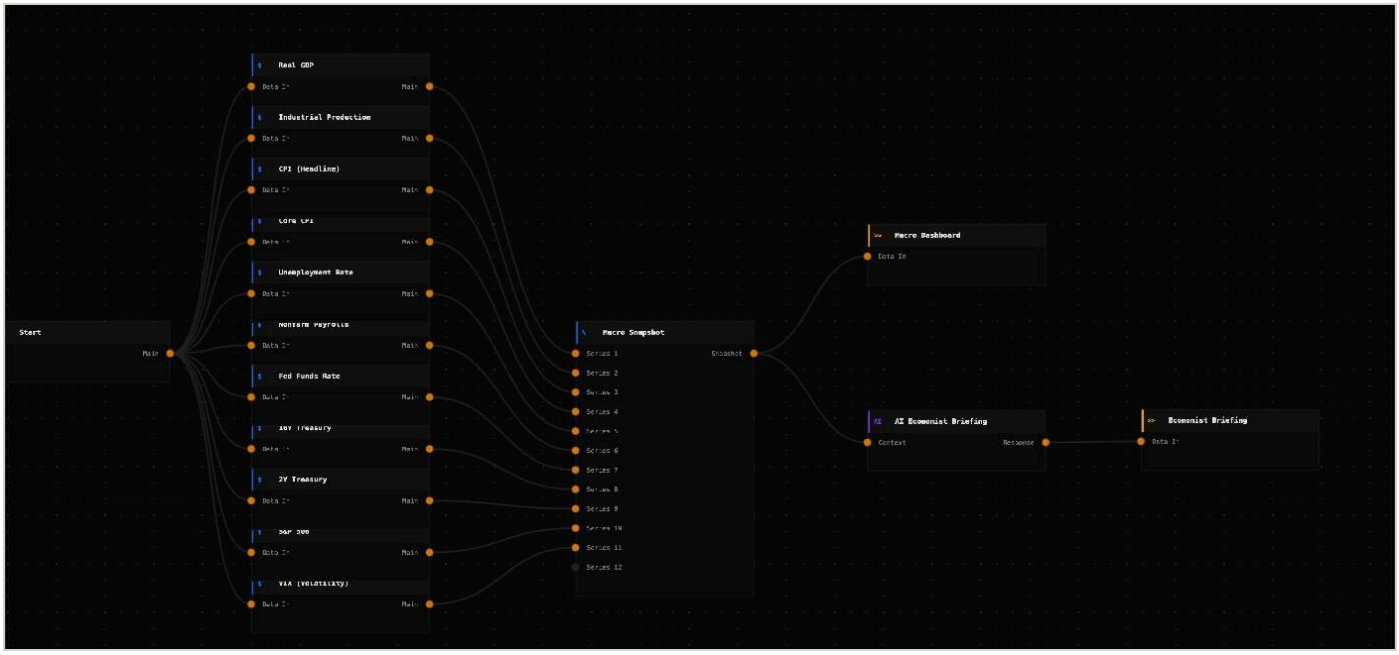


Fig 17 — Economic Dashboard: one trigger, twelve series fetches, one analytics node, then a dashboard and an AI briefing.

Stage	Nodes	What it does
1 · `Start`	Manual Trigger	One node, fanning out to all twelve series
2 · `Fetch`	12 × FRED Economic Data	Real GDP · Industrial Production · CPI · Core CPI · Unemployment · Nonfarm Payrolls · Fed Funds · 10Y · 2Y · S&P 500 · VIX. **All twelve are launched at once**
3 · `Compute`	Economic Summary	Twelve labelled inputs converge on one analytics node with a *Trend Window* parameter (5y by default)
4 · `Present`	Results Display	The snapshot, labelled for reading in the results drawer
5 · `Interpret`	AI Agent → Results Display	An economist briefing over the same snapshot — **this branch spends credits**

What it teaches

- **Fan-out is free parallelism.** Twelve fetches wired to one trigger run simultaneously; the stage takes as long as the slowest series, not the sum of twelve.
- **Fan-in needs labelled ports.** The analytics node exposes twelve numbered inputs, so each series lands in a known slot.
- **Split the expensive branch.** The dashboard and the AI briefing hang off the same snapshot, so you can run the free half repeatedly and add the metered half only when you want prose.
- **Display nodes are cheap probes — Execute From Here** on one runs a single node.

> **Try this as your first real build.** Load template 5, run it once, then delete six of the twelve series and re-run: the pattern of fan-out, fan-in and one analytics node covers a large share of what a research desk actually automates.

23 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	**NODES** on the top tab bar, or type `nodes` in the command bar
…add a node?	Drag it from the palette onto the canvas — search by name, description or type id
…connect two nodes?	Drag from an output port to an input port; incompatible drops are refused
…pan and zoom?	Wheel to zoom; middle-drag or Ctrl+left-drag to pan
…run everything?	**EXECUTE**. Press it again to stop a run in flight
…run one branch?	Right-click its first node ▸ **Execute From Here**
…see what a node returned?	Select it and read **OUTPUT**, or scan the results drawer
…switch off part of a graph?	Right-click ▸ **Disable** — remembering it disables everything downstream
…keep going past a failure?	Tick **Continue on Fail** on the node that is allowed to fail
…branch on a value?	**If / Else** with `={{$input.field > 0}}` — the one field where expressions resolve
…snapshot a workflow?	**EXPORT** to JSON. Auto-save overwrites the saved copy every 30 s
…trade without risk?	Leave every order node's **mode** on **paper** (the default)
…make it run on a schedule?	You cannot, in this build — see section 20
…have the assistant run it?	Save it, then ask AI Chat; it uses `run_workflow_dag`

Honest notes on this build

- **There is no scheduler.** All eight Trigger nodes are start markers; DEPLOY saves and runs once. Nothing runs in the background.
- **Expressions resolve in two condition fields only** (If/Else and Assert, in different dialects). Ordinary parameters are passed exactly as typed.
- **The Map node ignores its Expression** and passes input through unchanged.
- **Execute Workflow (sub-workflows) is not implemented** — and, like any unimplemented type, it passes through and counts as a *success*.
- **Disabling a node disables everything downstream of it.**
- **TEMPLATES and IMPORT replace the canvas without confirming;** only CLEAR asks.
- **Auto-save every 30 seconds** means there is no unsaved-experiment state.
- **Order mode is per node** — no global paper/live guard for the tab.
- **The palette is not alphabetised**; positions follow internal registry order, so search.
- **Retry-on-fail exists in the file format but not in the interface** — the panel offers only Disabled and Continue on Fail.
- **Timestamps in the load picker are UTC**, not local time.
- **Cycles are drawable but not runnable** — the run aborts before any node executes.

> Everything above was read from the shipping source and then confirmed on screen, on a live 17-node workflow — including the behaviours the interface does not advertise.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

MCP Servers & The Tool Surface

A complete walkthrough of the MCP Servers tab — and a full catalogue of the 760 tools the terminal exposes to its own AI, family by family.

`>_ 760 internal tools · 46 families · 242 KB of schema`

Covers · **the three views** · **adding a server** · **every internal tool family**
BM25 tool retrieval · the approval gate · plan gating · timeouts and the schema budget

Demonstrated live on the Fincept Terminal MCP Servers tab.

Contents

22 SECTIONS

THE TAB

01 What the MCP Servers tab is 3
02 Anatomy — three views and a status bar 4
03 MARKETPLACE — the curated catalogue 5
04 INSTALLED — and adding your own 6
05 How an external server connects 7
06 TOOLS — the master switchboard 8

THE INTERNAL TOOL SURFACE

07 What the internal tools are 9
08 The catalogue in one page 10
09 Trading & execution — 106 tools 11
10 Research & analytics — 238 tools 12
11 Data & markets — 104 tools 13
12 Workspace & documents — 165 tools 14
13 Agents & orchestration — 114 tools 15
14 Portfolio & account — 29 tools 16

HOW THE AI USES THEM

15 Finding a tool among 760 17
16 The task spine and playbooks 18
17 Destructive tools and the approval gate 19
18 Plan gating 20
19 Timeouts, async and the schema budget 21

REFERENCE

20 Persistence and security 22
21 Driving the tab from the agent 23
22 Quick reference and honest notes 24

About this guide. The screenshots are live captures of the shipping build. The tool catalogue was extracted from the shipping source and reconciled against the terminal's own count — where the two disagree, the guide says so.

01 What the MCP Servers tab is

ORIENTATION · 01

The Model Context Protocol is how an AI model reaches software. This tab is where you see and control every tool the terminal's AI can reach — **the 760 built into the terminal itself**, and any external MCP servers you choose to add alongside them.

Most of what matters here is invisible from the tab's own three views. The terminal ships a very large internal tool surface — every screen, every desk, every data family is scriptable by the AI — and the tab shows it as one flat list with no way to see how it is organised. **Sections 07 to 14 of this guide are that missing map.**

Fig 1 — The count the tab reports on a clean profile: 760 tools, all internal.

Where to find it

Route	How
Navigate menu	**Navigate ▸ AI & Quant ▸ MCP Servers**
Command bar	`mcp` or `servers`

At a glance, in the MCP Servers tab you can…

- See every tool the AI can call
- Enable or disable any tool individually
- Search the whole catalogue by name
- Install a curated external server
- Add a custom server by command line
- Give a server environment variables
- Start, stop and remove servers
- Read a server's connection log
- Auto-start servers at launch
- Confirm what the AI is allowed to do

> **The terminal works with zero external servers.** Everything the AI does out of the box — trading, research, charting, filing, workflow — runs on the internal tools. External servers are an extension point, not a requirement, and a fresh profile has none installed.

02 Anatomy — three views and a status bar

One header, three exclusive views, one status bar.

Fig 2 — The header: the view switcher on the left, search and REFRESH on the right.

#	Region	What it is for
1	View switcher	**MARKETPLACE · INSTALLED · TOOLS**. The active view is amber.
2	Search	Filters whichever view is open — servers by name in the first two, tools by name in TOOLS.
3	REFRESH	Re-reads server state and rebuilds the tool list. Needed after a server starts or stops outside the tab.
4	View body	Server cards in the first two views; a checkbox table in TOOLS.
5	Status bar	The active view, the installed server count, and how many are running.

Fig 3 — The status bar on a clean profile.

> **The three views answer three different questions.** MARKETPLACE is *what could I add*, INSTALLED is *what have I added and is it healthy*, and TOOLS is *what can the AI actually call right now* — which is the only one of the three that includes the internal tools, and the only one most users need.

03 MARKETPLACE — the curated catalogue

Five vetted servers, each installable in one click, grouped into three categories down the left.

Fig 4 — The marketplace. Each card carries the exact launch command, so nothing about the install is hidden from you.

Server	Category	Launch command	What it adds
`Fetch`	Utilities	`uvx mcp-server-fetch`	HTTP fetch and web content retrieval — lets the AI read any URL.
`Time`	Utilities	`uvx mcp-server-time`	Current time, timezone conversion, date arithmetic.
`Git`	Developer	`uvx mcp-server-git`	Repository operations: log, diff, commit, branch, status.
`SQLite`	Database	`uvx mcp-server-sqlite`	Query and inspect a SQLite database.
`PostgreSQL`	Database	`uvx postgres-mcp --access-mode=unrestricted`	Query and explore a PostgreSQL schema. **Needs DATABASE_URI.**

A card that needs configuration says so on its face — the PostgreSQL card is marked **Needs: DATABASE_URI**, and pressing ADD opens the dialog with that variable already listed as a field to fill.

> **Read the PostgreSQL command before you install it.** The curated launch line includes `--access-mode=unrestricted`, which gives the server — and therefore the AI — read *and write* access to whatever database you point it at. Point it at a read-only replica or a role with SELECT-only rights unless you intend the AI to be able to write.

> **The first start of a `uvx` server downloads it** through the terminal's bundled `uv`, which can take up to a minute. The card sits in a starting state while that happens; it is not stuck.

04 INSTALLED — and adding your own

Everything you have added, with its live state. A clean profile is empty and says so.

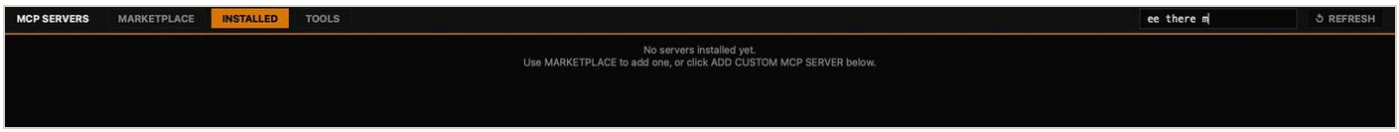


Fig 5 — The empty state, which names both routes to a first server.

Fig 6 — **+ ADD CUSTOM MCP SERVER** runs the full width of the foot of the view.

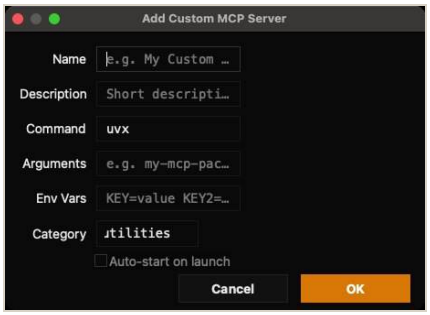


Fig 7 — The add dialog. Any MCP server that speaks stdio can be added here.

Field	What to put in it
`Name`	How the server appears on its card and in the TOOLS view's SERVER column.
`Description`	Free text, for your own reference.
`Command`	The executable. Defaults to `uvx`; `npx`, `python` or an absolute path all work.
`Arguments`	Space-separated arguments — normally the package name.
`Env Vars`	`KEY=value KEY2=value2`. This is where API keys and connection strings go.
`Category`	Groups the card in the marketplace-style listing. Defaults to `utilities`.
`Auto-start on launch`	Start this server every time the terminal opens.

> **An MCP server is a process on your machine with your permissions.** Adding one is the same act of trust as running the command yourself in a terminal — it can read your files and reach the network. The curated five are vetted; anything you add by hand is yours to vouch for.

> A card in INSTALLED carries the server's state (stopped / starting / running), a start-stop toggle, a LOGS expander with its stdio transcript, and a REMOVE that confirms first.

05 How an external server connects

TRANSPORT · 05

The terminal is the **client**. Each server is a child process it launches and talks to over that process's own standard input and output.

Stage	What happens
`1. Spawn`	The terminal runs `Command` + `Arguments` as a child process, with `Env Vars` added to its environment.
`2. Handshake`	Client and server exchange protocol versions and capabilities over stdio.
`3. Tool listing`	The server declares its tools with their JSON schemas.
`4. Merge`	Those tools join the internal ones in a single flat namespace — which is why the TOOLS view can show both together and label each with its SERVER.
`5. Call`	When the model picks a tool, the request goes to whichever server owns it.

Because the transport is the child process's stdio, a server that dies takes its tools with it and the count in the TOOLS view drops. **REFRESH** re-reads that state.

> **Tool names share one namespace.** Internal tools and every external server's tools sit in the same flat list, so an external server that publishes a name the terminal already uses is a collision. The SERVER column is how you tell two same-named tools apart.

> **Environment variables are passed to the child process in the clear.** They are stored with the server's configuration so it can be restarted, which means a key you paste into Env Vars is held in the terminal's own configuration rather than in the OS keychain — unlike broker credentials, which are keychain-only by design. Prefer a scoped, revocable key here.

06 TOOLS — the master switchboard

Every tool the AI can call, internal and external together, each with a checkbox. This is the control that decides what the model is allowed to reach.

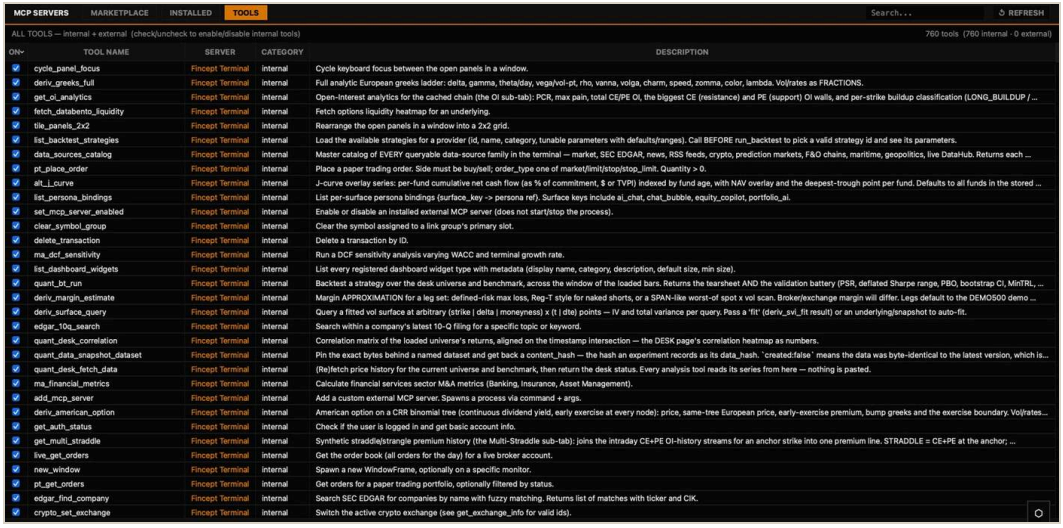


Fig 8 — The TOOLS view. 760 tools, all internal on this profile.

Column	What it is
`ON`	Enable / disable. Unchecking removes the tool from what the model is offered — it disappears from retrieval and cannot be called.
`TOOL NAME`	The exact identifier the model invokes.
`SERVER`	**Fincept Terminal** for internal tools; the server's name for external ones.
`CATEGORY`	**internal** or **external** — the tool's *source*, not its subject.
`DESCRIPTION`	The text the model itself reads when deciding whether to call the tool.

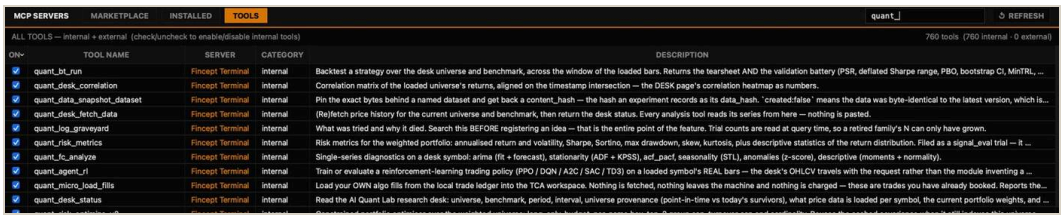


Fig 9 — Searching `quant_` filters the table live.

> **The CATEGORY column is not the tool's family.** Every internal tool reads `internal` here, so the column tells you where a tool came from and nothing about what it does. The terminal does group its tools into **46 functional families** internally — that grouping drives plan gating and retrieval — but **it is not surfaced anywhere in this tab**. Sections 08–14 print it.

> **Two smaller things worth knowing.** The *N tools* count in the header is the total and does not change when you filter — the filter narrows the table beneath it only. And the default order is neither alphabetical nor grouped; it is the registration order of the underlying hash, so **search is the only practical way to find a specific tool here**.

07 What the internal tools are

An internal tool is a function inside the terminal, published to the AI with a name, a description and a JSON schema, exactly as an external MCP server would publish one. The difference is that it runs **in this process**, against the screens and services already open in front of you.

That is why the surface is so large. Almost everything a tab can do by hand has a tool behind it: placing an order, loading a chart, running a backtest, writing a spreadsheet cell, rearranging a window, filing a report. The AI is not driving a separate API — it is driving **the terminal**.

Consequence	What it means in practice
`Tools act on live state`	Many tools operate on the screen that is open — an Excel tool fails with *Excel screen is not open* rather than working headlessly. The AI may need to open a tab first.
`No network hop`	An internal call does not leave the machine to reach the terminal, so it is fast and works offline for anything that does not itself need data.
`Your permissions`	Tools run as you, with your broker sessions and your files. This is the reason for the approval gate in §17.
`They can be switched off`	Every one of them has a checkbox in the TOOLS view (§06).

> **The description is the interface.** What the model sees of a tool is its name, its description and its parameter schema — nothing else. The descriptions in this terminal are unusually long and specific for that reason, and several of them carry warnings aimed at the model rather than at you.

> **The counts in this guide come from two places and they do not perfectly agree.** The terminal reports **760** internal tools — in the TOOLS view header and in its own startup log. The family-by-family catalogue that follows was extracted from the shipping source and accounts for **756** of them. The four-tool gap is tools assembled at runtime rather than declared statically. Treat 760 as the total and the family tables as accurate to within that.

08 The catalogue in one page

All 46 families, gathered into six groups. The **guarded** column counts tools marked as destructive — the ones that can change something and are therefore subject to the approval gate (§17).

Group	Families	Tools	Guarded
Research & analytics (§10)	9	**238**	100
Workspace & documents (§12)	11	**165**	71
Agents & orchestration (§13)	6	**114**	56
Trading & execution (§9)	7	**106**	24
Data & markets (§11)	11	**104**	8
Portfolio & account (§14)	2	**29**	3
Catalogued total	**46**	**756**	**262**
Reported by the terminal	—	**760**	—

What the shape tells you

- **Research is the largest group by a wide margin** — 238 tools, nearly a third of the surface, and 93 of those are the AI Quant Lab alone.
- **Trading is the smallest of the big groups** and the least guarded in absolute terms: 106 tools with 24 marked destructive, because most trading tools read state rather than send orders.
- **Workspace is the second largest**, which surprises people. Driving the terminal's own windows, panels, layouts and documents takes 165 tools.
- **About one tool in three is guarded** — 262 of 756.

> A family name is not cosmetic: it is the key plan gating is applied on (§18), and it is part of the document the retriever indexes (§15). Families with an underscore rather than a hyphen — `file_manager`, `agentic_memory` — are simply older names; nothing behaves differently.

09 Trading & execution — 106 tools

Everything that can look at a market or act in one. Seven families, **24 of the 106 guarded**.

Family	Tools	Guarded	What it covers
`algo-trading`	34	8	Strategy CRUD, backtest, deploy, monitor, exit, the scanner and watch list, safety caps.
`crypto-trading`	18	5	14 native venues: exchange selection, book, ticket, blotter, paper and live.
`live-trading`	17	7	Real broker orders — place, modify, cancel, square off, funds, positions, holdings.
`paper-trading`	11	4	The simulated book: orders, fills, positions, P&L.
`fno`	11	0	Option chains, OI analytics, multi-leg baskets, payoff. Read-only.
`polymarket`	10	0	Prediction markets: markets, prices, order book, positions. Read-only.
`alpha-arena`	5	0	The LLM trading competition: standings, rounds, entries.

The guarded counts here are the ones to read carefully. Seven live-trading tools and five crypto tools can move real money. Those two families are the *only* ones the approval gate refuses to auto-approve even in autonomous mode (§17) — every other destructive tool in the terminal can be waved through by a setting, and these cannot.

Read tools outnumber write tools roughly four to one across this group. That is deliberate: an agent asked about a position should be able to answer without any capability to change it, so the read paths are unguarded and plentiful.

10 Research & analytics — 238 tools

The largest group in the terminal, and the one that makes the AI genuinely useful rather than merely conversational. Nine families.

Family	Tools	Guarded	What it covers
`quant-lab`	93	93	The AI Quant Lab, at page parity: desk and universe, factor work, risk and optimisation, forecasting, backtest and holdout, microstructure and TCA, model training, RL agents, and the research log.
`ma-analytics`	47	1	M&A: DCF and sensitivity, accretion/dilution, synergies, LBO, comparables, payment mix, sector metrics.
`derivatives`	35	0	Pricing and greeks: European and American, binomial trees, vol surfaces and fits, margin estimation, strategy payoffs.
`alt-investments`	30	2	Private markets: J-curves, commitments and pacing, waterfalls, NAV, instrument analysis.
`surface-analytics`	15	3	The 35-surface capability catalogue plus Databento fetches — vol, liquidity, term structure, yield curve, stress.
`equity-research`	10	0	Symbol search and load, quote, info, historical, financials, technicals, peers, news, sentiment.
`backtesting`	4	0	Strategy listing and runs across the six engines.
`analytics`	3	1	Run and list the bundled Python analysis scripts.
`relationship-map`	1	0	Build the corporate relationship graph for a ticker.

Every one of the 93 Quant Lab tools is marked destructive — not because they trade, but because the shared helper that builds them defaults that flag to true. They mutate the research desk: loading a universe, fitting a model or sealing a holdout all change state a later tool will read. The practical effect is that Quant Lab work goes through the approval gate unless autonomous mode is on.

The Quant Lab family is worth knowing by name if you use that tab: every page has a matching `quant_*` tool, so anything you can do on the desk you can ask for in words — and the desk's own state travels with the request rather than the model inventing inputs.

11 Data & markets — 104 tools

Where the numbers come from. Eleven families and **only 8 guarded tools** — reading data changes nothing, so almost none of this needs approval.

Family	Tools	Guarded	What it covers
`sec-edgar`	35	0	Company lookup, filings, 10-K/10-Q search, Form 4 insiders, 8-K events, XBRL facts.
`news`	15	1	Headlines, categories, search, monitors, sentiment.
`geopolitics`	13	2	Events, critical regions, countries and cities, HDX humanitarian search, trade analysis.
`data-sources`	12	1	The connector registry: list, test, configure, query any registered source.
`maritime`	7	0	Vessels, ports, routes, chokepoints.
`watchlist`	6	2	List, create, add and remove symbols.
`datahub`	4	0	The live streaming hub: topics, subscriptions, health.
`data-mapping`	4	2	Mappings and scheduled refreshes.
`markets`	3	0	Quotes, indices, movers.
`feeds`	3	0	RSS feeds and items.
`data-catalog`	2	0	**Start here.** `data_sources_catalog` lists every queryable family with its canonical tool; `data_query` forwards to whichever one you name.

> **`data_sources_catalog` exists because of the size problem.** Rather than have the model hunt through a hundred data tools, one tool describes every family and names the canonical fetch tool for each, and a second forwards to it. If you are writing prompts or workflows against the terminal, that pair is the entry point worth knowing.

> **The catalogue flags families whose canonical tool is switched off.** If you disable a tool in the TOOLS view that `data_query` depends on, the catalogue reports that family as unavailable rather than letting the forward fail confusingly later. Disabling tools is safe, but it is visible to the model.

12 Workspace & documents — 165 tools

The second-largest group, and the least expected. Driving the terminal itself — its windows, panels, layouts and documents — takes more tools than trading does.

Family	Tools	Guarded	What it covers
`workspace`	59	38	Windows and monitors, panels and docking, layouts and snapshots, symbol link groups, the command bar, monitors and actions.
`excel`	27	4	Sheets, cells, ranges, rows and columns, formulas, find, sheet data, CSV and xlsx export.
`dashboard`	24	16	The widget catalogue, layout CRUD, per-widget configuration, the ticker bar.
`report-builder`	18	2	Reports, blocks, templates, preview and export.
`file_manager`	15	6	The managed library: list, read, write, rename, move, delete — two of which the UI itself cannot do.
`notebooks`	5	1	Notebook listing, open, run.
`system`	5	1	Auth status, cache stats and clear, app info, health check.
`notes`	4	1	Create, search, update, delete.
`navigation`	3	0	Open a screen, list screens, current screen.
`settings`	3	1	Read and write terminal settings.
`notifications`	2	1	List providers and send — Telegram, Slack, Email, and the in-app centre.

This group holds the highest guarded ratio in the terminal — 71 of 165, and 38 of the 59 workspace tools alone. Rearranging someone's windows, replacing a dashboard layout or deleting a file is disruptive in a way that reading a quote is not, so these are deliberately gated even though none of them touches money.

`notify_send` egresses to external services — Telegram, Slack, email — which is why it is marked destructive despite sounding harmless. Anything that leaves the machine is treated as a mutation.

13 Agents & orchestration — 114 tools

The tools the AI uses to run *itself* — and the ones it uses to run this tab.

Family	Tools	Guarded	What it covers
`agents`	80	43	The whole Agent Studio: the agent repository, teams and desks, personas and skills, scheduling, reflexion, single and multi-agent execution, routing, streaming, and the agentic task lifecycle (start, poll, pause, resume, cancel).
`mcp-servers`	14	9	This tab, driven by the AI: list, add, remove, start, stop, enable and disable servers and individual tools.
`meta`	10	0	Tool discovery and the task spine — see §15 and §16.
`ai-chat`	4	1	Send to the chat, read history, manage sessions.
`workflows`	4	3	Run a saved node-editor workflow, run from a node, stop, poll status.
`agentic_memory`	2	0	`archival_memory_save` and `archival_memory_search` — cross-session recall.

The AI can reconfigure its own tool access. The `mcp-servers` family includes `add_mcp_server`, `set_mcp_server_enabled` and the per-tool enable switch — so a sufficiently determined agent can install a server or turn a tool back on. Nine of the fourteen are guarded for exactly that reason, and `add_mcp_server` spawns a process. If that is not a capability you want, disable those tools here; the switch is the same one they use.

Workflow tools refuse to overlap. Both run tools are marked destructive and both refuse if a run is already in flight, so an agent cannot start the same graph twice by retrying.

14 Portfolio & account — 29 tools

The smallest group, and the one most likely to be called in an ordinary conversation.

Family	Tools	Guarded	What it covers
`profile`	15	0	Account, plan and entitlement, credit balance and ledger, usage and quotas, devices, preferences. **Entirely read-only** — nothing here can spend, change a plan or alter an account.
`portfolio`	14	3	Portfolios and holdings, transactions, performance and analytics, allocation. The three guarded tools add, update and delete transactions.

These two families are why "how am I doing" works as a question. The AI can read the whole portfolio, its analytics and the account's own state without any capability to change any of it — and the three tools that *can* change a transaction are guarded individually.

> **The profile family reading your credit balance is deliberate.** It lets an agent tell you what an action will cost, and what you have left, before it runs something metered — which is the same information the credit-guarded buttons put on their labels.

> **Portfolio transaction tools edit the record, not the market.** `delete_transaction` removes a booked transaction from your portfolio history; it does not reverse a trade at a broker. The two are easy to conflate when asking an agent to “undo” something.

15 Finding a tool among 760

A model cannot be handed 760 tool definitions on every turn. Past roughly thirty to fifty tools, tool-choice accuracy collapses and the prompt cost becomes absurd. The terminal solves this the way the research literature recommends: **it does not send the catalogue at all**.

How it actually works

Step	What happens
1	The model is given a small permanent set — the **meta tools** and the task spine — and nothing else.
2	It calls `tool_list` with a natural-language query describing what it needs.
3	A **BM25 (Okapi) index** scores every registered tool against that query and returns the top matches as lightweight references.
4	It calls `tool_describe` on the one it wants to get the full JSON schema.
5	It invokes the tool by name.

The index document for each tool is its **name, description, parameter names and parameter descriptions** — which is why descriptions in this terminal are written long and specific. A well-described tool is a findable tool; a terse one is invisible.

Meta tool	What it does
`tool_list`	Search the catalogue, get back the top-K relevant tool references.
`tool_describe`	Fetch one tool's full description and input schema.
`mcp_health`	Report the MCP subsystem's state — servers, counts, failures.
`get_configured_state`	What is set up on this terminal: brokers, keys, sources.

> **The index is pure C++ — no embeddings, no model files, no Python.** It rebuilds lazily whenever the tool set changes: starting an external server, stopping one, or toggling a checkbox in the TOOLS view all bump a generation counter that invalidates it on the next search.

> **Disabling a tool makes it unfindable, not just uncallable.** An unchecked tool is dropped from the retrieval index, so the model will not see it in `tool_list` results and cannot rediscover it. That is the intended behaviour — but it means a capability that mysteriously stopped working is worth checking here before it is reported as a bug.

16 The task spine and playbooks

Retrieval solves *which tool*. The task spine solves *where was I* — the other thing that breaks on long agentic work.

Tool	What it does
`task_set_goal`	Record what this piece of work is trying to achieve.
`task_update_step`	Record the plan and mark progress through it.
`task_note_finding`	Store a finding so it survives past the context window.
`task_get_state`	Read the goal, plan and findings back.

These four are **per-chat persistent working memory**. A long task — research a company, run a backtest sweep, reconcile a book — outlives any single model context; without somewhere to put the goal and the findings, the agent forgets what it was doing halfway and starts over. The spine is registered alongside the meta tools as always-available for exactly that reason.

Playbooks

Tool	What it does
`list_playbooks`	List the canned multi-step procedures the terminal ships.
`select_playbook`	Load one, so the agent follows a known-good sequence rather than improvising a plan.

> **Two different memories, easy to confuse.** The task spine is scoped to *this chat's current task* and is about staying on-plan. `archival_memory_save` and `archival_memory_search` (§13) are *cross-session* recall — things worth remembering next week. Ask an agent to “remember” something and which one it reaches for changes whether you get it back tomorrow.

17 Destructive tools and the approval gate

APPROVAL · 17

262 of the catalogued 756 tools are marked destructive. When the AI tries to call one, the request goes through an approval gate before anything happens.

The decision ladder, in order

#	Condition	Outcome
1	`ai/agent_mutations` is **off**	**Hard deny.** The AI cannot mutate anything, full stop.
2	Headless run (the agentic test harness)	An environment variable decides. No dialog — there is nobody to answer one.
3	`ai/autonomous_mode` is **on**	**Auto-approve — except** money-moving families (live trading, crypto trading, wallet) and anything flagged for explicit confirmation. Those always ask.
4	Server-bridge dispatch (an operator-driven server loop)	Auto-approve ordinary mutations; **refuse** money-moving and explicit-confirm tools with a clear error. Never opens a modal — a server-driven run has no one at this terminal.
5	Otherwise	**An interactive confirm dialog** on the GUI thread. The model proposes, you approve or deny. It **defaults to Deny**, and it **times out to Deny**.

Money is the line the gate will not cross. Autonomous mode can wave through a dashboard rewrite, a file deletion or a Quant Lab run. It cannot wave through a live or crypto order — those confirm every time, in every mode, and the server-bridge path refuses them outright rather than approving them unattended.

Every decision is written to an audit log under its own tag, approved and denied alike. If you want to know what the AI asked to do and what happened, that log is the record — not the chat transcript, which only shows what the model chose to tell you.

Destructive is broader than it sounds. It marks anything that changes state or leaves the machine — sending a notification, saving a layout, loading a Quant Lab universe. It is not a synonym for dangerous, and a third of the catalogue carrying the flag is expected.

18 Plan gating

The terminal has machinery to restrict tool families by subscription tier, and it runs on every login and every plan change.

Stage	What happens
1	Every registered tool is walked, and its **family** is looked up in a required-tier table.
2	A family with no entry is ungated and always allowed.
3	A family with an entry is enabled only if your tier ranks at or above the requirement.
4	The result is applied with the same enable/disable switch the TOOLS view uses.
5	The whole pass re-runs whenever the authentication state changes.

In the shipping build the required-tier table is empty, so plan gating is currently a no-op. The lookup returns nothing for every family, which means every tool is enabled on every paid tier and the gating pass reports nothing at startup. This is consistent with the product rule that every tab is open on every paid plan — depth is limited by quotas and credits, not by removing capabilities — but it does mean the machinery described above is wired and idle rather than actively restricting anything today.

What this means for you. Nothing in the tool surface is withheld by plan. If a tool refuses, the reason is the approval gate (§17), a disabled checkbox (§06), a quota or credit refusal from the server, or the tool's own preconditions — never your tier.

19 Timeouts, async and the schema budget

Three engineering constraints that shape what the tool surface can do, and occasionally explain a failure.

Timeouts

Every tool carries its own default timeout, tuned to the work: short for registry lookups and paper orders, long for analytics that run a full backtest. A caller can override it per invocation. Tools come in two shapes — synchronous ones that block for as long as they need, and asynchronous ones for I/O-bound work, which are the ones the timeout applies to.

The schema budget

Every tool's JSON schema is prompt weight, paid on every turn that tool is visible. The terminal audits this at startup and warns about the offenders:

Measure	Value on the shipping build
Tools audited	760
Total schema bytes	**242 KB**
Per-tool budget	2,048 B
Tools over budget	**2**
Worst offender	`alt_instrument_analyze` — **6,447 B**
Second	`algo_create_strategy` — 2,462 B

> **242 KB is why retrieval exists.** Sending the whole catalogue on every turn would cost roughly sixty thousand tokens before the user has said anything. Retrieval (§15) means the model normally sees a handful of schemas instead — so the budget matters most for tools that are always visible, and least for the long tail.

> **A tool that always times out is usually waiting on a screen.** Internal tools act on live state, so one that needs a tab which is not open will sit until its deadline rather than failing fast. If an agent reports a timeout on something that should be instant, check whether the tab it drives is actually open.

20 Persistence and security

What survives a restart, and what to be careful with.

Item	Where it lives	Notes
`Installed servers`	Terminal configuration	Command, arguments, category and auto-start are all restored at launch.
`Env Vars`	Terminal configuration	**Stored with the server config**, not in the OS keychain.
`Per-tool enable/disable`	Terminal configuration	Your checkbox choices persist.
`Auto-start servers`	Started during initialisation	Before the first AI turn.
`Broker credentials`	**OS keychain**	Never in MCP configuration — a locked product decision, unaffected by anything in this tab.

The three things worth thinking about

- **An external server is a process with your permissions.** It can read your files and reach the network. The five curated ones are vetted; anything else is your call.
- **Env Vars are the weak spot.** A key pasted there is held in configuration so the server can be restarted. Prefer scoped, revocable credentials, and never a broker key — those have a keychain path and do not belong here.
- **The AI can change this tab.** Fourteen tools drive it, including adding a server and re-enabling a tool (§13). The approval gate covers them, but if you want the surface frozen, disable that family.

> **Disabling a tool is the cheapest safety control in the terminal.** It needs no plan change and no restart, it removes the tool from retrieval as well as from execution (§15), and it persists. If there is a capability you never want the AI to have, one checkbox is the whole answer.

21 Driving the tab from the agent

The MCP Servers tab is itself scriptable — fourteen tools in the `mcp-servers` family, nine of them guarded.

Capability	What it lets the agent do
`Discover`	List installed servers with their state, and list every tool with its enabled flag.
`Install`	`add_mcp_server` — spawns a process from a command and arguments. Guarded.
`Remove`	Delete an installed server. Guarded.
`Lifecycle`	Start and stop a server. Guarded.
`Enable a server`	`set_mcp_server_enabled` — toggles without starting or stopping the process. Guarded.
`Enable a tool`	Turn an individual tool on or off. Guarded.
`Health`	`mcp_health` — subsystem state, counts and failures. Unguarded.

The honest use for this is diagnosis: asking the assistant *why can't you do X* and having it check whether the relevant tool is disabled or its server is down beats hunting through a 760-row table by hand.

> **This family is the one place the AI can widen its own reach.** Everything else in the terminal lets it act within the capabilities you granted; these tools let it request new ones. The approval gate stands in front of all nine mutating tools, and autonomous mode will auto-approve them — they are not money-moving, so they do not get the always-confirm treatment. Consider that when turning autonomous mode on.

22 Quick reference and honest notes

How do I...

Goal	Do this
`See everything the AI can call`	**TOOLS** view
`Stop the AI doing something specific`	Find the tool in TOOLS, untick it
`Find a tool by name`	Use the search box — the list order is not alphabetical
`Know what a tool does`	Read its DESCRIPTION column — it is what the model reads too
`Add a capability`	**MARKETPLACE** for the curated five, or ADD CUSTOM MCP SERVER
`Stop the AI mutating anything`	Turn off `ai/agent_mutations` (§17)
`Find out why a tool failed`	Ask the assistant to run `mcp_health`

Honest notes on this build

- **The CATEGORY column shows the source, not the family.** Every internal tool reads `internal`; the 46 functional families are not surfaced anywhere in the tab (§06).
- **The tool count does not respond to the search filter** — the header keeps showing the total (§06).
- **The list order is registration order**, neither alphabetical nor grouped, so search is the only practical way to find something (§06).
- **760 reported, 756 catalogued.** Four tools are assembled at runtime and do not appear in the family tables (§07).
- **All 93 Quant Lab tools are marked destructive** because their shared builder defaults the flag to true — not because they trade (§10).
- **Plan gating is wired but idle** — the required-tier table is empty, so every tool is available on every paid tier (§18).
- **Two tools exceed the schema budget**, one of them by more than three times (§19).
- **Env Vars are stored in configuration, not the keychain** (§05, §20).
- **The AI can install servers and re-enable tools**, and autonomous mode auto-approves that (§13, §21).
- **Disabling a tool also hides it from retrieval**, so the model cannot rediscover it — intended, but a confusing failure mode if forgotten (§15).

> Counts in this guide were extracted from the shipping source and reconciled against the terminal's own reported total; the tab screenshots are live captures of the same build. Where the source and the running application disagreed, the disagreement is printed rather than smoothed over.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Code Notebook

A complete, screen-by-screen walkthrough of the Code tab — Fincept Notebook, a Jupyter-style Python workspace with a curated library of ready-to-run finance notebooks and an editor for your own.

`>_ library · editor · PY & MD cells · .ipynb`

Covers · **the two views** · **the notebook library** · **the editor** **cells & running code** · notebooks, saving & files · .ipynb import/export

Demonstrated live on the Fincept Terminal Code tab.

Contents

7 SECTIONS

GETTING ORIENTED

01 What the Code tab is 3

02 Anatomy & the two views 4

WORKING IN THE NOTEBOOK

03 The Notebook Library 5

04 The Editor 6

05 Cells & running code 7

06 Notebooks, saving & files 9

REFERENCE

07 Quick reference — “How do I…” 10

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal. Control labels are quoted exactly as they appear on screen; notebook output shown reflects a single live run.

01 What the Code tab is

The Code tab is **Fincept Notebook** — a lightweight, Jupyter-style **Python notebook** built for learning and prototyping finance, economics and quant ideas. You work in stacked **cells**: **Markdown** cells for prose and **code** cells for Python, run individually or all at once, with results shown inline.

It has two views — a **Library** of ready-made notebooks to open and run, and an **Editor** where you read, write and execute them. It reads and writes standard `.ipynb` files.

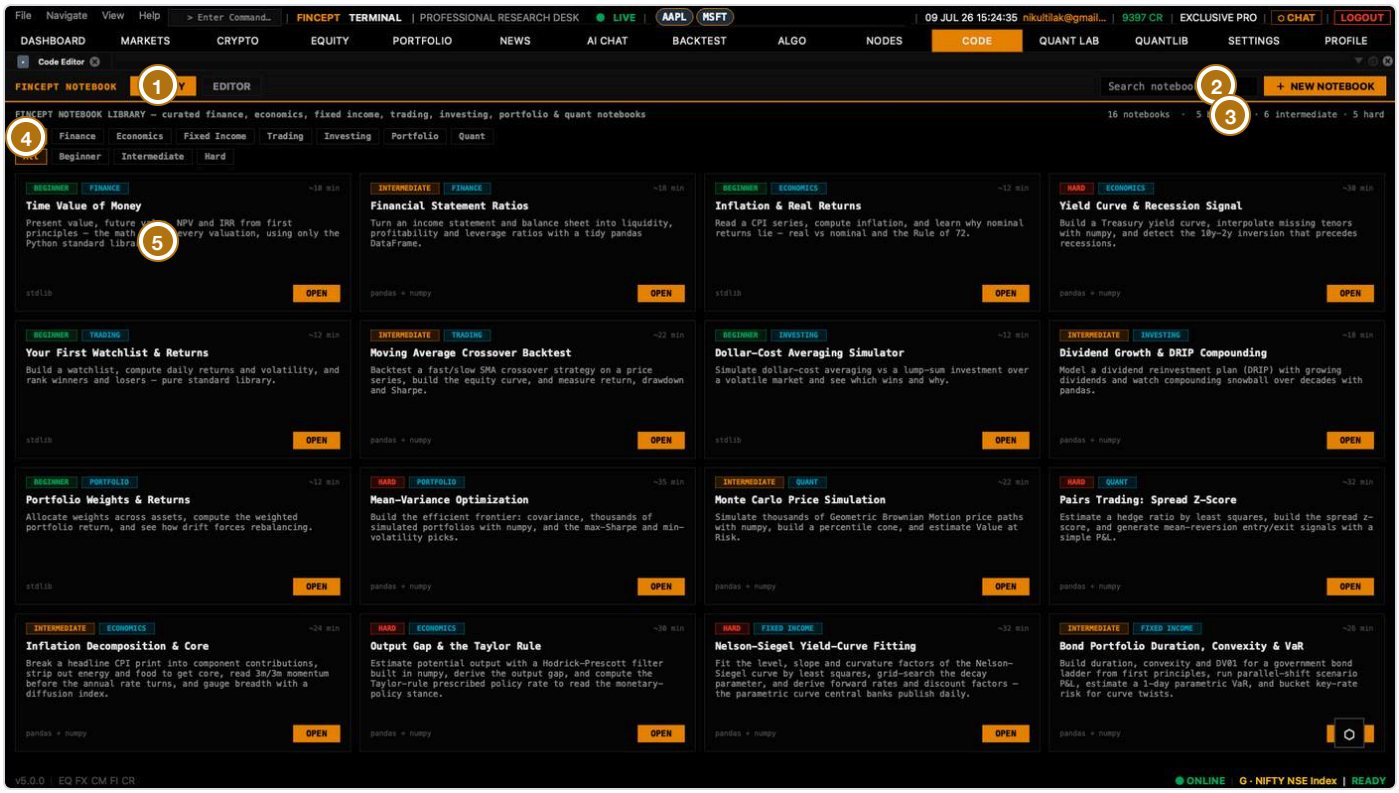
At a glance, you can…

- Browse a **curated library of 16 notebooks** across Finance, Economics, Fixed Income, Trading, Investing, Portfolio & Quant.
- **Filter by category and difficulty** (Beginner / Intermediate / Hard) and search by name.
- **Open any notebook** into the editor as your own editable copy — the originals stay pristine.
- Write **Python** and **Markdown** cells; run a cell with `Ctrl+Enter` or the whole notebook with `▶ RUN ALL`.
- Read **text output** and error tracebacks inline, right under each cell.
- **Add, move, delete, rename** cells and toggle any cell between code and Markdown.
- **Create, open and save** `.ipynb` notebooks.

> **Good to know up front.** Fincept Notebook is deliberately simple. It runs **Python only** (labelled Python 3.11), shows **text output only** (no charts or images), and — most importantly — **each cell runs on its own**, so variables don't carry between cells (see §5). There's no file tree, no autocomplete and no AI help inside this tab.

02 Anatomy & the two views

The header stays put; the body switches between **LIBRARY** and **EDITOR**. It opens on the Library.



#	Region	What it is
1	**View tabs**	`LIBRARY` and `EDITOR` — the active view is amber. Click to switch.
2	**Search + New**	Search the library (Library view only) and `+ NEW NOTEBOOK` to start a blank notebook in the editor.
3	**Library count**	How many notebooks match, broken down by difficulty.
4	**Filter chips**	Two rows: **category** and **difficulty**. Combine them to narrow the grid.
5	**Notebook cards**	The curated notebooks, each a card you can open.

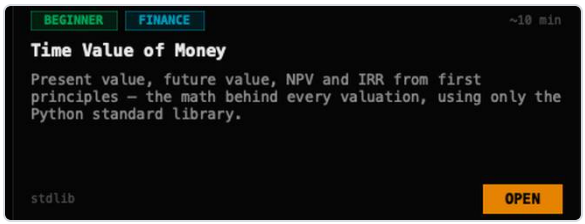
The view tabs.

Search & new notebook.

03 The Notebook Library

The Library is where you discover ready-to-run notebooks. Each is a card; filter and search to find one, then open it.

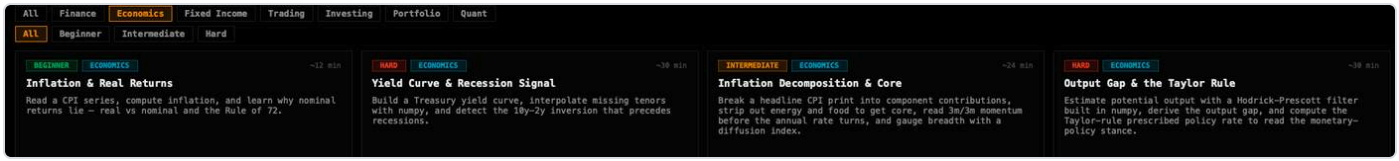
Difficulty	**BEGINNER** · **INTERMEDIATE** · **HARD**.
Category	Finance, Economics, Fixed Income, Trading, Investing, Portfolio or Quant.
~N min	Roughly how long the notebook takes to work through.
Requirements	`stdlib` (pure Python) or `pandas + numpy`.
OPEN	Opens an editable **copy** in the editor — the bundled original is never changed.



A notebook card.

Filtering & searching

The two chip rows combine — pick a **category** and/or a **difficulty**; the count updates and the grid reflows. The search box (top right) matches title, category and summary as you type.

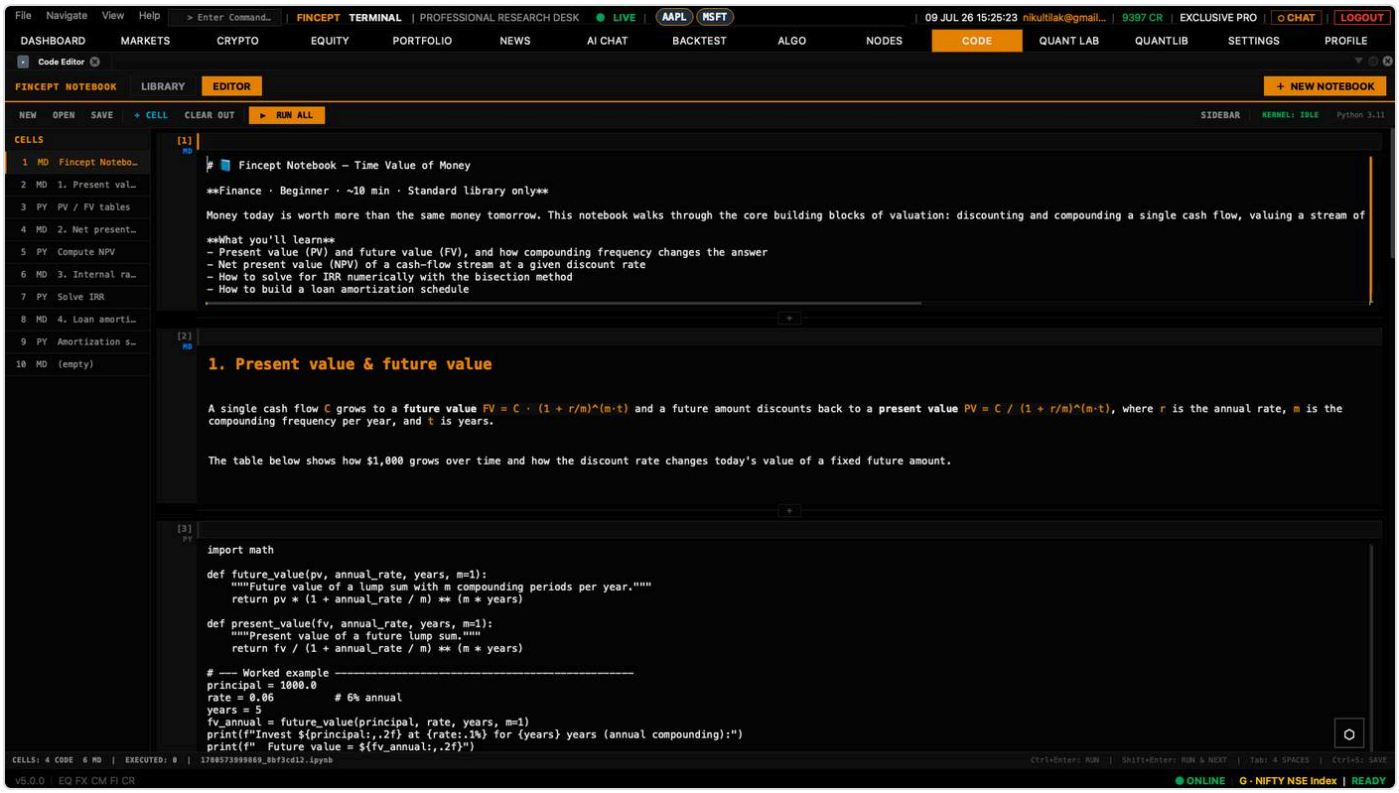


Filtering to `Economics` narrows the grid to just those notebooks.

If nothing matches, the grid reads "*No notebooks match your filters.*"

04 The Editor

Opening a notebook (or + NEW NOTEBOOK) switches to the Editor: a toolbar, a cell list on the left, and the cells themselves.



The editor — a curated notebook open, with the **CELLS** navigator on the left.

The toolbar

Button	What it does
`NEW`	Start a fresh notebook with one starter code cell.
`OPEN`	Open an `.ipynb` file from disk.
`SAVE`	Save the notebook as `.ipynb` (prompts for a name the first time).
`+ CELL`	Add a new code cell after the selected one.
`CLEAR OUT`	Clear every cell's output and execution counts.
`▶ RUN ALL`	Run every code cell in the notebook.
`SIDEBAR`	Show / hide the CELLS navigator.
KERNEL · Python 3.11	Shows `IDLE` or `BUSY` while running, and the interpreter.

05 Cells & running code

Every cell is either Python (**PY**) or Markdown (**MD**). The CELLS navigator keeps a long notebook organised; each cell carries its own controls.

CELLS navigator — click to jump; F2 or right-click to rename.

The CELLS navigator

The left sidebar lists every cell as `#` · type (**PY**/**MD**) · name · run-count. Click a row to jump to that cell. Right-click a row (or press `F2`) to **rename** it — handy for turning a long notebook into a clean outline. Toggle it with the `SIDEBAR` button.

Cell controls

Each cell has a gutter (its number + **PY**/**MD** type) and, on hover, a small toolbar.

Hover a cell for its controls: **RUN** · **TYPE** (code⇄Markdown) · **UP** / **DN** (reorder) · **DEL**. The gutter shows the run count, e.g. `[1] PY`.

Control / marker	Meaning
RUN	Run just this cell.
TYPE	Switch the cell between **code** and **Markdown**.
UP / DN	Move the cell up or down.
DEL	Delete the cell (the last remaining cell can't be deleted).
`[*]` / `[n]`	Gutter marker: `[*]` = running now; `[n]` = the order it last ran in.

Markdown cells render automatically; click one to edit its source, click away to re-render. In code cells, `Tab` inserts four spaces.

Output

```
OUT [1]

Invest $1,000.00 at 6.0% for 5 years (annual compounding):
  Future value = $1,338.23
  Check PV     = $1,000.00  (should return the principal)

Effect of compounding frequency ($1,000 at 6% for 5 years):
Frequency        m    Future value
----------------------------------
Annual           1        1,338.23
Semiannual       2        1,343.92
Quarterly        4        1,346.86
Monthly         12        1,348.85
Daily          365        1,349.83

Future value of $1,000 (annual compounding):
 Years          3%          5%          7%         10%
-------------------------------------------------------
     1       1,030       1,050       1,070       1,100
```

A cell's `OUT [n]` block — printed text sits directly beneath the code. Click the header to collapse it.

> **The one thing to remember: cells don't share state.** Every cell (and every cell in `▶ RUN ALL`) runs as its **own separate Python process**. Variables, imports and functions defined in one cell are **not** visible to the next — so each cell must import what it needs and stand on its own. The `KERNEL` badge is just a status light, not a shared session.

> **Text output only.** Cells show printed text and error tracebacks. There is no chart/plot/image rendering — the bundled notebooks present results as printed numbers and ASCII tables. A failed cell shows a **red error block** with the exception and traceback.

06 Notebooks, saving & files

The status bar tracks your notebook: cell counts, how many have run, and the file name — with the key shortcuts on the right.

Task	How
Start a notebook	`+ NEW NOTEBOOK` or the toolbar `NEW`.
Open an existing one	Toolbar `OPEN` → pick an `.ipynb` file. (Opening a `.ipynb` from the File Manager tab also lands it here.)
Save your work	Click `SAVE` — writes a standard `.ipynb` and registers it with the File Manager.
Rename a cell	Right-click it in the sidebar, or press `F2`.

> **Save with the button.** The status bar lists `Ctrl+S: SAVE`, but that shortcut isn't wired up — use the `SAVE` button to save. (Notebooks are also opened as editable copies, so running a Library notebook never overwrites the original.)

Notebooks are plain `.ipynb` files (the same format Jupyter uses). The Code tab has no file browser of its own — browse and manage files in the separate **File Manager** tab, which sends `.ipynb` files straight here.

07 Quick reference — "How do I..."

I want to...	Do this
Open the Code tab	Click `CODE` in the top tab bar (or type `code` in the command bar).
Find a notebook to learn from	Stay on `LIBRARY`, filter by category / difficulty, and click `OPEN`.
Search notebooks	Type in the **Search notebooks...** box (top right).
Start from scratch	Click `+ NEW NOTEBOOK`.
Run one cell	Click into it and press `Ctrl+Enter` (or hover → `RUN`).
Run a cell and move on	Press `Shift+Enter`.
Run the whole notebook	Click `▶ RUN ALL`.
Add a cell	Toolbar `+ CELL`, or the `+` between cells.
Turn code into notes (or back)	Hover the cell → `TYPE`.
Reorder / delete a cell	Hover → `UP` / `DN` / `DEL`.
Clear all outputs	Toolbar `CLEAR OUT`.
Rename a cell	Right-click it in the sidebar or press `F2`.
Save	Click `SAVE`.

Keyboard shortcuts

Key	Action
`Ctrl+Enter`	Run the current cell.
`Shift+Enter`	Run the cell and move to the next.
`Tab`	Insert four spaces (in a code cell).
`F2`	Rename the selected cell.

Fincept Terminal — Code Guide · screenshots captured from the live application, running the bundled "Time Value of Money" notebook.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Report Builder

A complete, screen-by-screen walkthrough of the Report Builder — the terminal's institutional authoring desk for turning live market and portfolio data into branded, print-ready PDF research.

`>_ 15 block types · 10 chart engines · 12 templates · 4 themes · 18 agent tools`

Covers · **the three-door Home, the block canvas & slash menu** · **data blocks, charts & the property panel**
branding, themes, one-path PDF export & the AI agent tools

Demonstrated live on the Fincept Terminal Report Builder tab.

Contents

11 SECTIONS

GETTING ORIENTED

Overview — what the Report Builder tab is 3

1 The workspace at a glance 4

2 The Home view — three ways to start 6

3 The template library 7

BUILDING THE REPORT

4 Editing on the canvas 8

5 The block catalogue 10

6 Data blocks & charts 11

7 Document setup — metadata, branding & themes 13

DELIVERING

8 Export & the PDF 14

9 The AI & MCP agent tools 15

10 Storage, autosave & cloud sync 16

11 Quick reference & honest caveats 17

ABOUT THIS GUIDE

Every screenshot is a real, live capture of the Report Builder running inside the Fincept Terminal — including the branded PDF that the tab actually produces. Control names, menus and dialogs are reproduced exactly as they appear on screen.

What the Report Builder tab is

OVERVIEW

The **Report Builder** is the terminal's document-authoring desk. It turns headings, prose, tables, live quotes and charts into a paginated, brand-styled PDF — the research notes, market briefs, factsheets, deal memos and risk reports an institutional desk sends to clients.

Unlike a word processor, the report you edit is a **structured stack of typed blocks** owned by a background service, not a stream of text. That single design choice is what makes the tab powerful: the very same live document can be edited **by hand** on the page and **by the AI agent** through its tools, both flowing through one shared undo history and one autosave. And because there is a single rendering path, **what you see in Preview is exactly what exports** — cover page, contents, running header/footer and all.

The workspace has three parts — a component rail on the left, a live WYSIWYG page in the middle, and a contextual property panel on the right — plus a command bar across the top for file, undo, refresh and export actions. When a document is empty, a full-screen **Home** view offers three ways to begin: generate with AI, pick a template, or reopen a recent report.

At a glance, in Report Builder you can…

- Start from **12 finance templates**, an AI prompt, or a blank page
- Drop **15 block types** from a rail or a `/` slash menu
- Edit text, headings and lists **inline on the page**
- Chart from CSV or a live symbol in **10 chart styles**
- Bind blocks to a ticker and **fetch** quotes & price history
- Add a **branded cover, contents & disclaimer**
- Recolour the whole report with **4 themes**
- Export a print-ready PDF where **preview == export**
- Have the **AI agent build or edit** the same live document

WHERE TO FIND IT

Report Builder is reached from the **Navigate** menu, or by typing `report builder` into the command bar. It is not plan-gated. Work autosaves continuously, so you can leave and return without losing a draft.

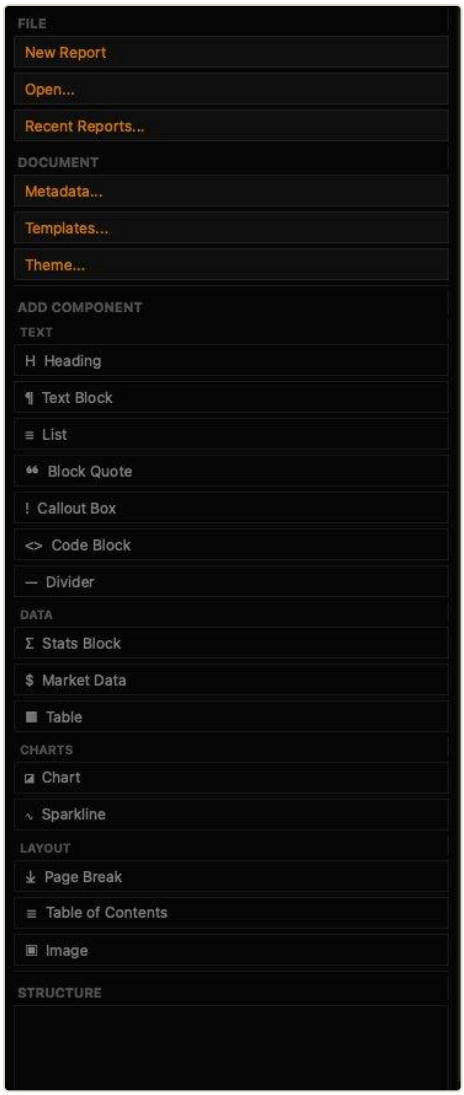
1 The workspace at a glance

ANATOMY · 4 REGIONS

Report Builder is a three-panel workbench under a fixed command bar. The middle **canvas** is the report itself — a continuous white page you edit directly. The left rail inserts blocks; the right panel configures whichever data block is selected.

The full workspace. Left: the component rail (FILE · DOCUMENT · ADD COMPONENT · STRUCTURE). Centre: the live document page, here showing a title, a Portfolio Snapshot stats block and a fetched price chart. Right: the contextual property panel for the selected chart. Top: the command bar.

Region	What it holds	What you do there
① **Command bar**	Top strip: `‹` · `⌂` Home · title · ✦ **AI** · **Data as of…** · ⟳ **Refresh All** · Undo · Redo · Open · Save · **Export PDF** · Preview · `›`	File actions, undo/redo, refresh all data, export & preview, and collapse either side panel (**Ctrl+B** / **Ctrl+Shift+B**).
② **Component rail**	Left. Four groups: **FILE** (New / Open / Recent), **DOCUMENT** (Metadata / Templates / Theme), **ADD COMPONENT** (15 blocks), **STRUCTURE** (the outline).	Insert any block, open the document dialogs, and drag-reorder or delete blocks in the outline.
③ **Canvas**	Centre. The WYSIWYG page: an editable title header plus the stack of blocks.	Type and format inline, use the `/` menu and hover handles, drag to reorder.
④ **Property panel**	Right. A two-tab **SETUP** | **STYLE** editor — appears only for data blocks (chart, table, stats, market data, sparkline, image).	Bind a symbol and fetch, or enter data by hand; set chart type, size, axes and legend.



The component rail. Every insert is one click; the STRUCTURE outline mirrors the document.

The rail, top to bottom

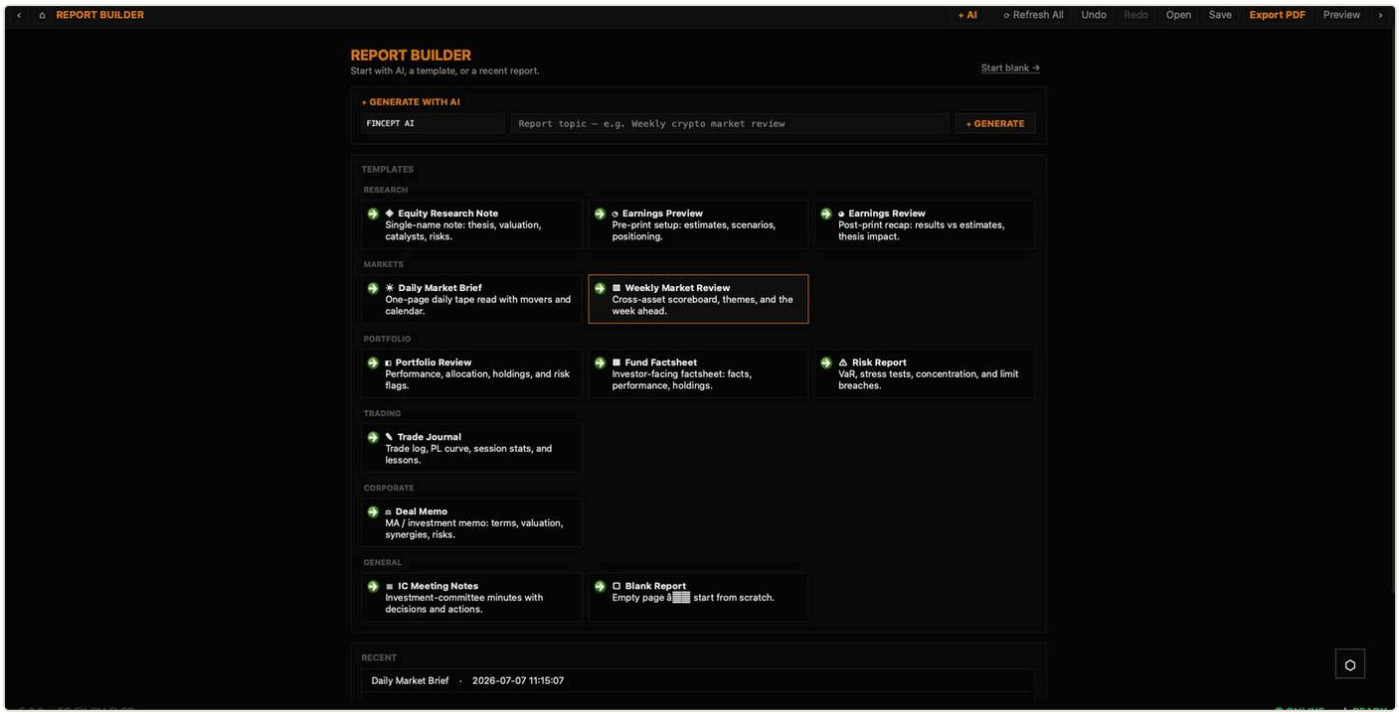
FILE — New Report (confirms first), Open… (`.fincept` / `.json`), Recent Reports… (your stored library).
DOCUMENT — Metadata… (title, header/footer, branding), Templates… (jumps to the Home gallery), Theme… (four colour presets, live preview).
ADD COMPONENT — the same 15 blocks the slash menu offers, grouped **TEXT · DATA · CHARTS · LAYOUT**. A new block lands just after the selected one.
STRUCTURE — a live outline of the document. Click a row to select its block; drag rows to reorder; **Delete Block** removes the current row.

> **CONTEXTUAL BY DESIGN**
> The right panel is **hidden for text blocks** — those are edited straight on the page — and slides in only when you select a data block. The **Export PDF** and **✦ AI** buttons are amber to mark them as the primary actions.

2 The Home view — three ways to start

HOME · 3 DOORS

When a document is empty — or whenever you press the ⌂ or ✦ **AI** buttons — the canvas is replaced by a full-width **Home** screen. It offers three doors into a new report, plus a **Start blank** → escape hatch in the top-right.



The Home view. Top: **Generate with AI** (an analyst persona + a topic box). Middle: the **template gallery**, grouped RESEARCH · MARKETS · PORTFOLIO · TRADING · CORPORATE · GENERAL. Bottom (not shown): **Recent**, your stored reports.

Door	What it does
✦ **Generate with AI**	Pick an analyst **persona**, type a **topic** (e.g. "Weekly crypto market review"), and press ✦ **GENERATE**. The persona composes an entire report from real terminal data, building it live on the canvas — the command bar shows a running status while it works. *This is the only door that can spend model credits.*
Templates	A 3-column gallery of **12 finance templates**. Click a card: templates with no placeholders apply instantly; those with `{ticker}` / `{portfolio}` / `{period}` open a small inline form first (below).
Recent	Your stored reports, newest first (title · last-updated). Click to reopen. This list is **open-only** — no rename or delete here. Empty state: `(No saved reports yet)`.

Inline template parameters. Fill `portfolio` and `period` (both optional) and press **Apply Template**. Blank fields are honest — the raw `{token}` is kept in the document so you can fill it later on the page.

> **GETTING BACK**
> The ⌂ button toggles between Home and the editor *without unloading* your document, so you can browse templates and return to your draft. **Start blank** → drops you onto an empty page titled "Untitled Report".

3 The template library

12 TEMPLATES

Each template is a ready-made scaffold — headings, the right tables, callouts, stats blocks and **chart shells** already laid out for a specific deliverable. A template **replaces** the current document; you then fill it in on the page or let the AI populate it. Templates seed the *structure*, so freshly-applied charts show an "add data" placeholder until you fetch or type values.

Template	Group	What it is	Parameters
Equity Research Note	Research	Single-name note: quote tile, thesis callout, rating & valuation tables, 1-yr price chart, revenue bars, catalysts, risks, disclaimer.	`ticker`
Earnings Preview	Research	Pre-print setup: street-estimates table, what-to-watch list, scenario table, positioning.	`ticker`, `period`
Earnings Review	Research	Post-print recap: results vs estimates, segment bars, guidance, management quote, revised estimates.	`ticker`, `period`
Daily Market Brief	Markets	One-page tape read: four quote tiles (S&P / Nasdaq / VIX / DXY), sector bars, movers table, calendar.	—
Weekly Market Review	Markets	Cross-asset scoreboard with a table of contents, indices chart, winners/losers, week-ahead.	`period`
Portfolio Review	Portfolio	Performance & risk: stats snapshot, benchmark line chart, allocation donut, holdings & contributors tables, actions.	`portfolio`, `period`
Fund Factsheet	Portfolio	Investor-facing: key facts, growth chart, returns table, allocation pie, top-10 holdings, risk metrics.	`portfolio`, `period`
Risk Report	Portfolio	VaR bars, stress-test table, concentration pie, liquidity table, breach callout, recommendations.	`portfolio`
Trade Journal	Trading	Trade log, cumulative P&L curve, session stats, win/loss donut, rules checklist.	`period`
Deal Memo	Corporate	M&A memo: contents, recommendation callout, deal terms, valuation bridge & synergy bars, risks, timeline.	`ticker`
IC Meeting Notes	General	Investment-committee minutes: attendees, agenda, discussion, decisions callout, action items.	—
Blank Report	General	An empty page with a title and a starter paragraph.	—

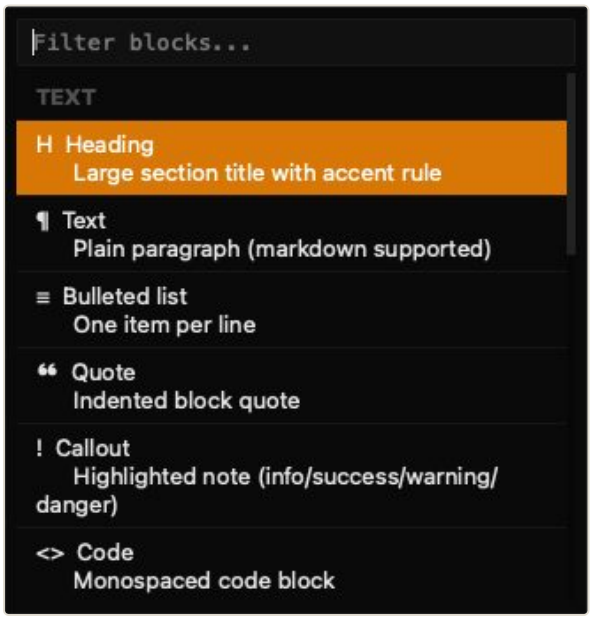
LEGACY NAMES STILL WORK

Beyond the 12 gallery templates, **23 legacy names** (Stock Research, Investment Memo, Macro Economic Summary, Crypto Research Report, Yield Curve Analysis, and more) resolve to the closest modern template — so older reports and AI calls that reference them keep working. That is **35 accepted template names** in total; an unknown name falls back to Blank Report.

4 Editing on the canvas

INLINE · SLASH · FORMAT

The canvas is a real editor, not a preview. Text-family blocks — text, heading, quote, list, callout, code — are typed **directly on the page**; every keystroke auto-saves ~400 ms after you stop, through the same service the AI edits flow through. The title, author, company and date in the page header are editable too.



The slash menu. Type `/` in a block to open a filterable, grouped palette of every block type.

Inserting blocks

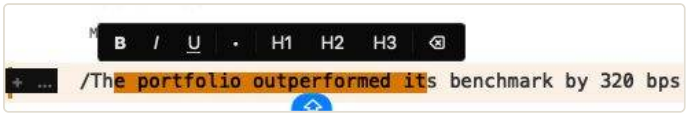
Type `/` to open the **slash menu** — start typing to filter ("chart", "call"...). Or hover a block's left edge and click the `[+]` handle to insert below. The rail's ADD COMPONENT group does the same thing by category.

Markdown shortcuts

At the start of an empty text block: `#` → heading, `##` → sub-heading, `-` or `1.` → list, `>` → quote. Inside text, `**bold**` and `*italic*` render on export.

Formatting a selection

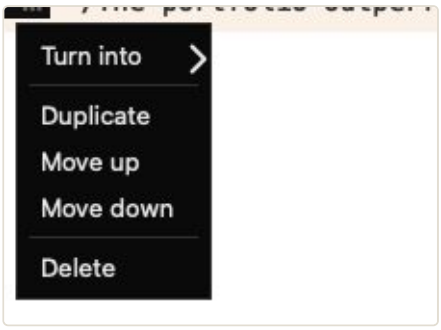
Select text in any editable block and a floating toolbar appears: **B** bold, **I** italic, **U** underline, • turn into a bulleted list, **H1/H2/H3** turn into a heading, and ⌫ to strip all formatting marks.



The floating format bar over a live text selection.

Re-shaping a block

Hover a block and click the ⠿ grip for its menu, or **drag the grip to reorder**. The menu lets you **Turn into** another text type, recolour a callout's tone, **Duplicate**, **Move up/down**, or **Delete**.



The block grip menu.

Keyboard (block-select mode)	Action
`Esc`	Leave the text editor and select the block
`↑ / ↓`	Move the selection between blocks
`Ctrl+Shift+↑ / ↓`	Reorder the selected block
`Ctrl+D` · `Delete`	Duplicate · delete the selected block
`Enter` (at end of a text/heading)	Start a new text block below

GOOD TO KNOW

While editing, markdown marks such as `**` or `<u>` show as **literal characters** — the styled result appears in the read-only render and the exported PDF. The **H1/H2/H3** buttons record a heading level, but every heading renders at one size on the page and in this export path.

5 The block catalogue

17 SLASH BLOCKS

The `/` menu offers 17 block types across four groups (the rail lists the 15 most-used). Text blocks are edited on the page; data and chart blocks are configured in the right panel.

Block	Group	What it renders
Heading	Text	Bold section title with an accent rule underneath.
Text	Text	A paragraph; supports markdown bold/italic/underline and lists.
Bulleted list	Text	Disc-bullet list, one item per line.
Quote	Text	Indented, italic block quote on a tinted background.
Callout	Text	A coloured note box in one of four tones — **info · success · warning · danger** — each with its own icon.
Code	Text	Monospaced block on a shaded background.
Divider	Text	A full-width horizontal rule.
Stats block	Data	A titled grid of key/value pairs (two per row). Auto-fills from a ticker or takes "Label: Value" lines.
Market data	Data	A live quote tile: symbol, name, price and coloured ± change, with a high/low/volume line.
Table	Data	A bordered grid; row 1 is the header, numeric cells right-align. Edited in a grid.
Ticker table	Data	A table pre-seeded with a Symbol / Price / Change % header row.
Chart	Charts	A rasterised chart (10 styles — see §6), rendered at 2× density for print.
Sparkline	Charts	A tiny inline trend line with an optional value and ± change.
Two columns	Layout	Inserts a pair of text blocks for side-by-side notes.
Page break	Layout	Forces a new page in the exported PDF.
Table of contents	Layout	An auto-numbered list built from every Heading block.
Image	Layout	An image (browse or paste) with left/centre/right alignment, width and an optional caption.

The table grid pop-over (double-click a table): edit cells directly, with **+ Row / + Col / – Row / – Col**. Row 1 is the header; changes apply live.

TWO WAYS TO EDIT A TABLE

Double-click a table on the page for the in-place grid pop-over shown here, or select it and use the same grid in the right panel's SETUP tab. Both store the table as CSV under the hood, so there is no comma-wrangling.

Callout tones you can pick:

ℹ info ✓ success ⚠ warning ✗ danger

6 Data blocks & charts

FETCH OR TYPE · 10 CHARTS

Data blocks are configured in the right **SETUP | STYLE** panel. Each can pull live data from a **ticker** or take values you enter by hand. Charts, stats blocks and market-data tiles all bind to a symbol; a **↻ as of HH:mm** chip on the block (and a **Data as of...** stamp in the command bar) record the last fetch.

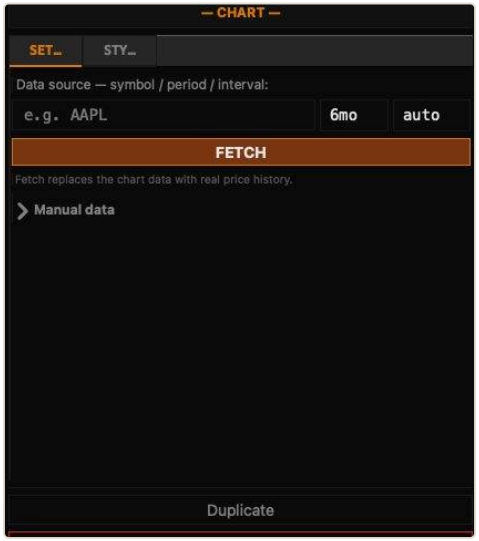


Chart · SETUP. Bind a symbol, choose a period & interval, and **FETCH** real price history — or open **Manual data** to enter CSV.

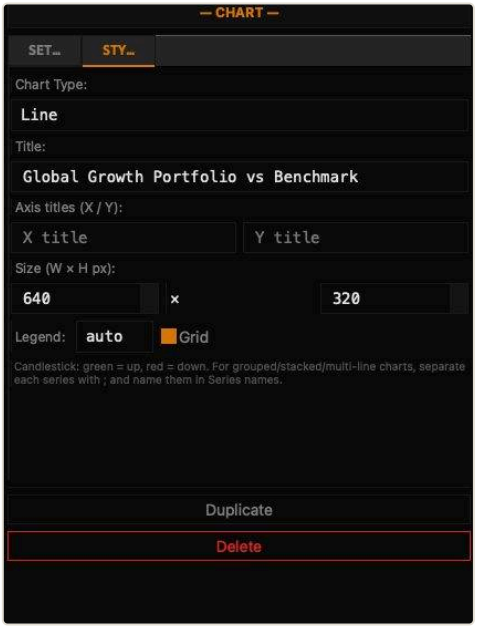


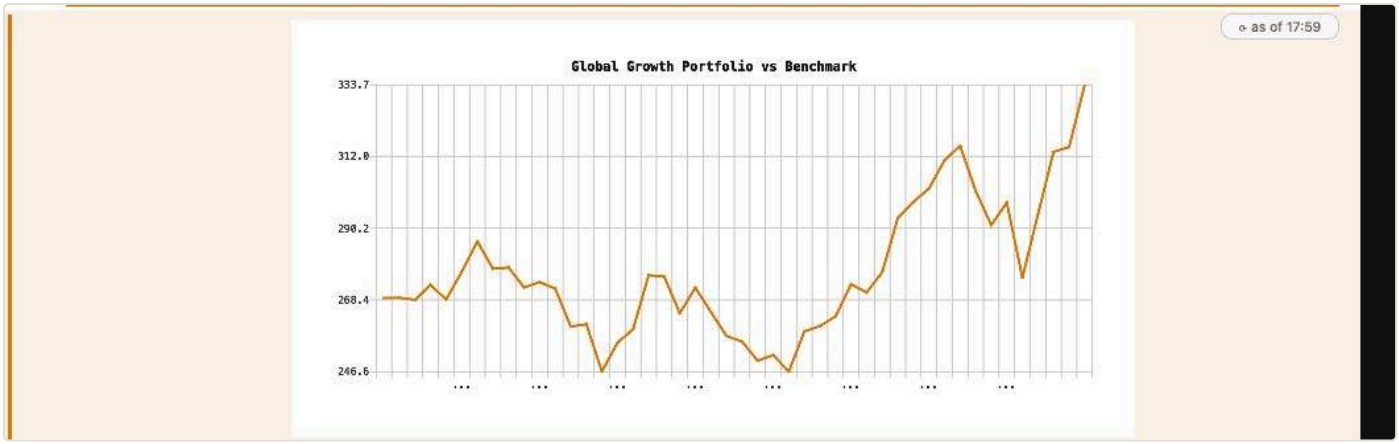
Chart · STYLE. Pick one of 10 chart types, set the title, X/Y axis labels, width × height, legend mode and grid.

Fetch vs. manual

FETCH pulls real price history for the symbol and replaces the chart's data (auto-switching to weekly bars for long ranges). **Manual data** takes CSV: one series, or several separated by `;`, with X-axis labels and legend names. The result rasterises at 2× so it stays crisp in print.

Stats & quotes

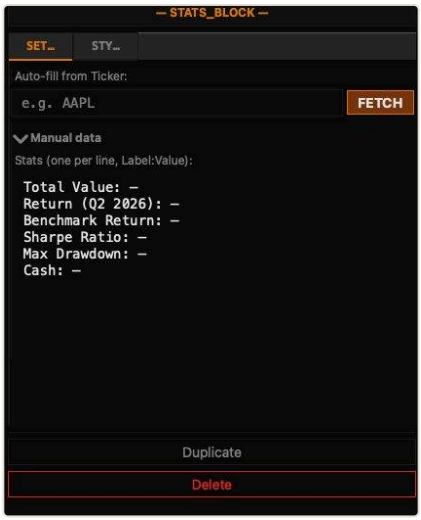
A **Stats block** auto-fills a fixed key-statistics panel from a ticker, or takes "Label: Value" lines. A **Market data** tile fills price and change from just a symbol.



A fetched price chart on the page, with its live "↻ as of 17:59" refresh chip. The chart renders identically here and in the exported PDF.

Chart type	Data format	Notes
Line · Spline · Area	CSV of y-values; multiple series split by `;`	Category X axis from labels; legend auto-shows for >1 series.
Scatter	CSV of y-values	Point markers along the index.
Bar · Stacked · Horizontal bar	One or more CSV series	Grouped, stacked, or category-on-Y bars.
Pie · Donut	Single CSV row of slice values + labels	All slice labels shown; donut adds a centre hole.
Candlestick	`O,H,L,C` per row, rows split by `;`	Green up / red down, axis fitted to the true range.

A donut built from five slice values and labels.



Stats · SETUP — auto-fill from a ticker or type lines.

Table · SETUP — a live cell grid.

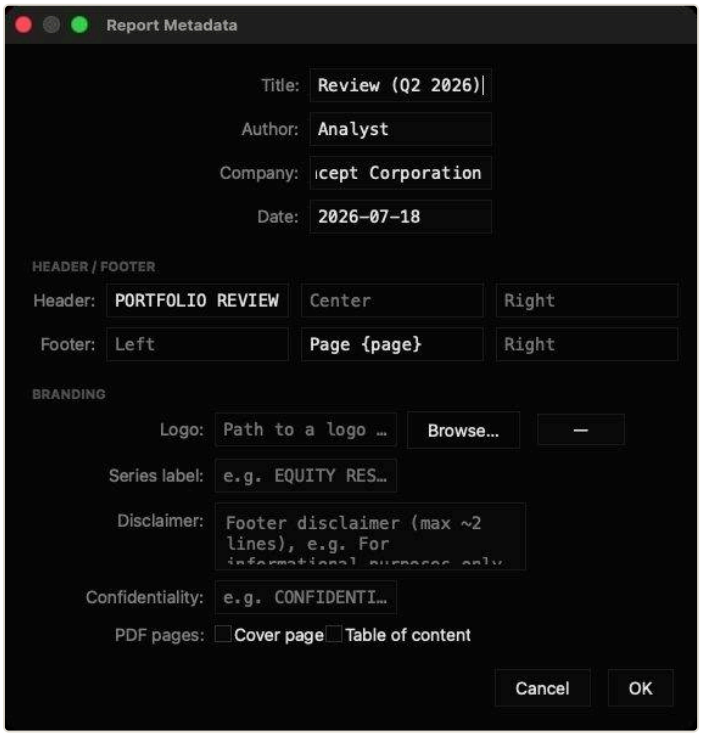
HONEST NOTES ON DATA

Typed tickers on a **chart** or **stats** block commit only when you press **FETCH** — click away without fetching and the symbol is not kept (a Market-data tile does keep its symbol). A sparkline's preview refreshes only when you re-select the block. The two **STYLE** tabs for tables and market-data tiles are informational only — those blocks always follow the report theme.

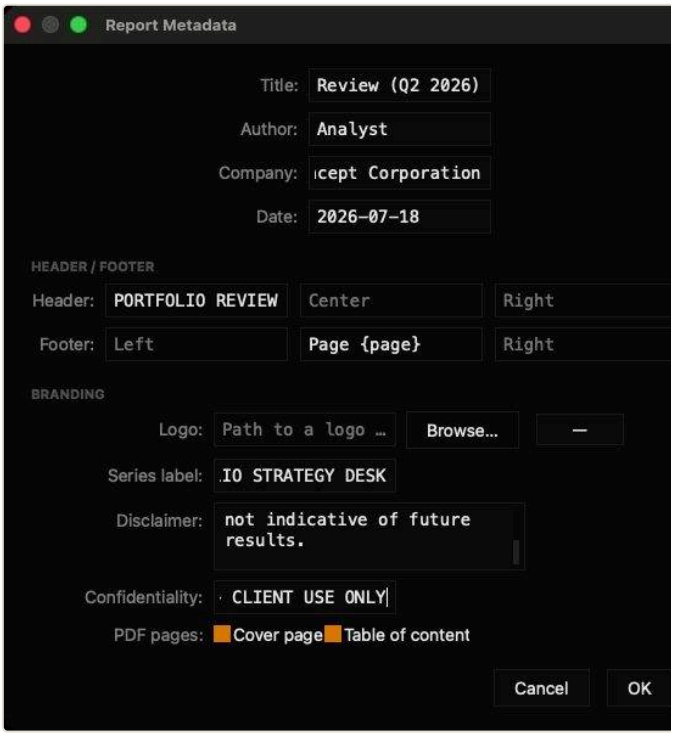
7 Document setup — metadata, branding & themes

BRANDING · 4 THEMES

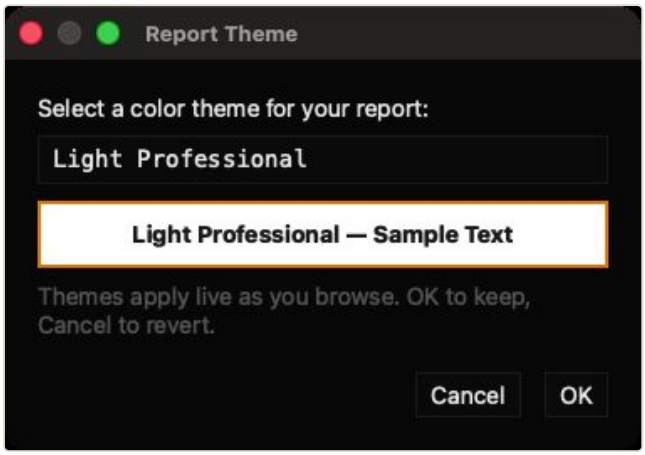
Two small dialogs control how the finished PDF looks. **Metadata…** sets the title block, running header/footer and the institutional furniture — logo, series label, disclaimer, confidentiality line, and whether to add a cover page and contents. **Theme…** recolours the whole report.



Report Metadata. Title / Author / Company / Date, a 3×2 header & footer grid, and a **BRANDING** block: logo, series label, disclaimer, confidentiality, and **Cover page** / **Table of content** toggles.

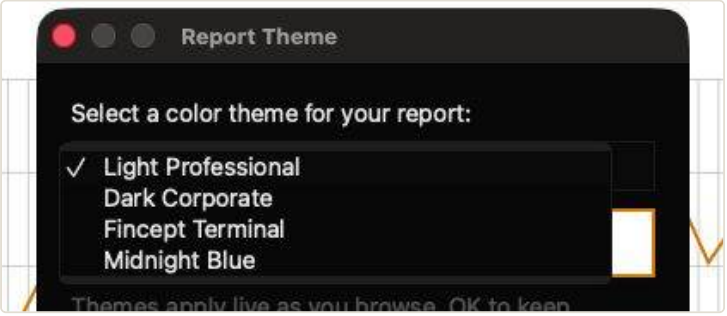


Filled in. A series label, disclaimer and confidentiality line are set, and both cover and contents are switched on — these drive the PDF's front matter and every page's header/footer.



Report Theme with a live "Sample Text" swatch. The theme applies as you browse; **OK** keeps it, **Cancel** reverts.

The four themes



Light Professional · Dark Corporate · Fincept Terminal · Midnight Blue.

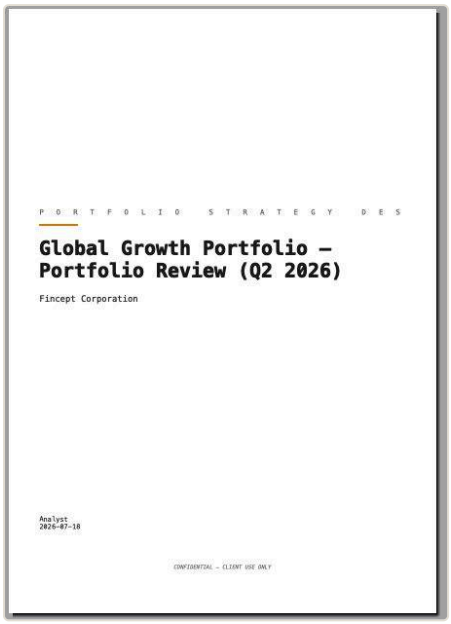
Light Professional is the default and the right choice for a printed client document. The others recolour headings, rules, table headers and callouts to a house palette.

> **ONE THING TO REMEMBER**
>
> A theme is stored by **name**, so the four presets always reproduce exactly; there are no per-report custom colours to lose. The Date field is free text — leave it blank to auto-stamp today's date.

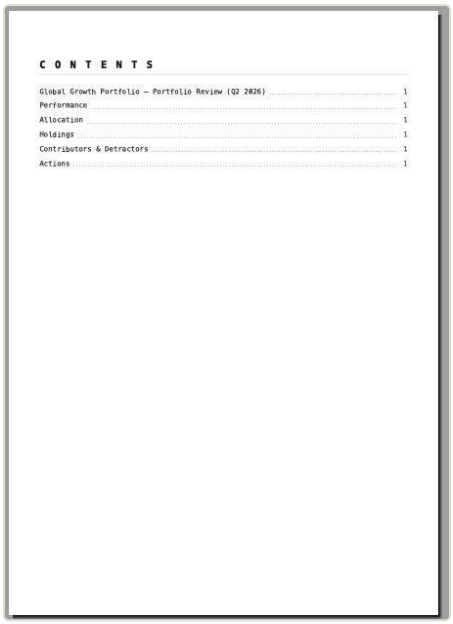
8 Export & the PDF

PREVIEW == EXPORT

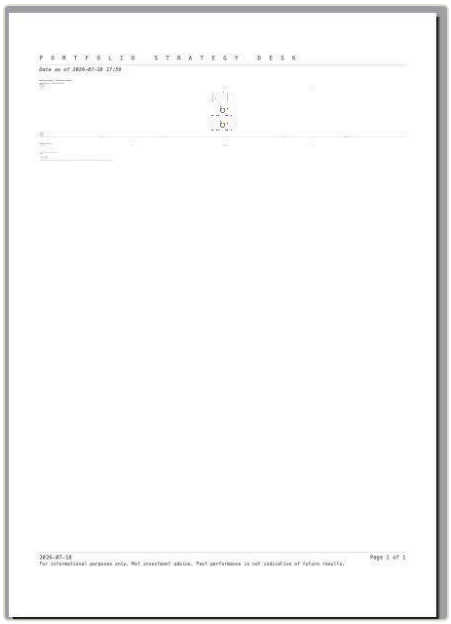
Report Builder has a single rendering path, so **Preview** shows exactly what **Export PDF** writes — cover, contents, running header and footer included. With branding set (§7), the tab produces a genuinely client-ready A4 document.



Cover page. Series label in letter-spaced caps, the report title, company, author & date, and the confidentiality line at the foot.

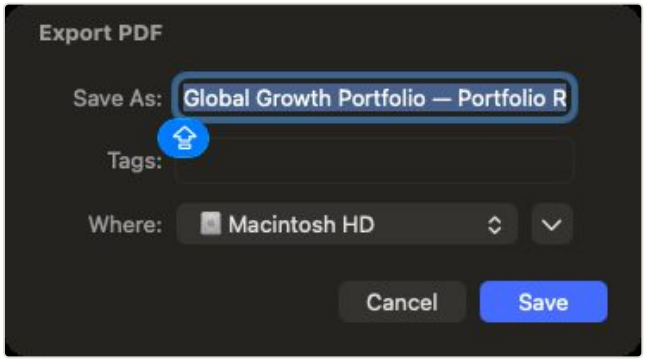


Contents. Auto-built from every Heading, with dotted leaders and page numbers.



A content page. Running header (series label), a **Data as of** stamp, and a footer with the date, disclaimer and **Page X of Y**.

PDF element	Where it comes from & when it appears
Cover page	Shown when **Cover page** is ticked in Metadata. Uses the logo, series label, title, company, author, date and confidentiality line.
Contents	Shown when **Table of content** is ticked and the document has headings. Built automatically from Heading blocks.
Running header/footer	On every content page: series label + logo in the header; date, disclaimer and **Page X of Y** in the footer.
"Data as of" stamp	Appears when any block carries a fetch timestamp — the newest fetch across the report.
Charts	Embedded at 2× density for crisp print — the same rasters you see on the page.



Export PDF opens a normal Save dialog; **Preview** opens a paged print-preview. Both call the identical compiler.

PREVIEW IS THE TRUTH

Because preview and export share one compiler, there is no "looked fine on screen, broke in the PDF" gap. Page through the preview before you send — the cover, contents and every header/footer you see are exactly what your client receives.

Reports also save as portable `.fincept` / `.json` files via **Save**, and to your durable library (see §10).

9 The AI & MCP agent tools

18 TOOLS · report-builder

The document lives in a background service, so the **AI agent can build and edit the very report you have open** — the same undo history, the same autosave. Institutional desks often drive the whole thing from chat: "draft an equity note on NVDA", "add a risk callout", "export it". Eighteen tools (category `report-builder`) expose the document; the first edit in a burst even auto-navigates you to the tab so you can watch it build.

Tool	What it does
report_get_state	Read the whole document — every block with its stable id, plus metadata & theme. Called first, so edits target the right block.
report_session_context	Report whether this chat already owns the on-canvas report (continue) or inherited another's (clear first).
report_add_component	Insert one block of any of the 15 types, with content & config; returns its id.
report_bulk_add_components	Append many blocks in one atomic step — how the agent lays down a whole report.
report_update_component	Replace a block's content and merge config keys, by id.
report_move_component	Reorder a block to a new position.
report_remove_component	Delete a block by id.
report_clear	Clear all blocks and reset metadata.
report_set_metadata	Set title, author, company, date, header/footer and page numbers.
report_set_theme	Switch among the four colour themes.
report_apply_template	Drop in a built-in scaffold by name (quick-start only).
report_list_component_types	Enumerate the 15 block types with their config keys & examples.
report_list_templates	List every accepted template name.
report_save · report_load	Save to / load from a `.fincept` file on disk.
report_export_pdf	Render the report to a PDF path (needs the tab open — prior edits open it automatically).
report_undo · report_redo	Step the shared undo stack; a whole agent burst counts as one step.

In practice, the agent can…

- Build a full report from a one-line brief
- Add, edit, move or remove any block by id
- Set metadata, switch themes, apply templates
- Insert live market-data tiles, charts and tables
- Save, and export a finished PDF
- Undo/redo its own work alongside yours

10 Storage, autosave & cloud sync

NEVER LOSE A DRAFT

Report Builder is built so you never lose work. Three mechanisms run in parallel: continuous autosave, a durable library, and cloud sync.

Mechanism	How it works
Autosave	Every ~60 seconds the current document is written to a local autosave file and restored on relaunch — the tab reopens on your last draft. (You saw the title bar read "autosave" throughout this guide.) Empty scaffolds are skipped.
Recent Reports (the library)	Saving — or the AI finishing a report — upserts the document into your durable library, the list you see under **Recent** on the Home view and in **Recent Reports...**. Newest first, opened with one click.
Cloud sync	Library reports mirror to your Fincept account, so a report authored on one machine appears in Recent on another after a sync. Opening the Recent list requests a fresh pull.
Portable files	**Save** also writes a self-contained `.fincept` / `.json` file you can archive or share; **Open...** reads it back.

TWO "RECENT" LISTS

The **Recent Reports** library (stored documents, cloud-synced) is separate from the file-open *recent files* list (paths to `.fincept` files on disk). The Home view and the Recent Reports dialog both show the library.

HONEST NOTE

Applying a template, opening a file, or a bulk clear **resets the undo history** — you cannot Ctrl+Z a template application. Save or duplicate a draft you care about before applying a new template over it.

11 Quick reference & honest caveats

HOW DO I… + CAVEATS

How do I…

Goal	Do this
Start a report fast	Home → pick a template card → fill any parameters → Apply Template.
Let AI write it	Home → ✦ Generate with AI → pick a persona, type a topic → ✦ GENERATE.
Add a block	Type `/` on the page, or click a block in the ADD COMPONENT rail.
Insert a live chart	Add a Chart → SETUP → type a symbol → FETCH. Change type in STYLE.
Make a two-line chart	Chart → Manual data → two CSV series split by `;` → name them in Series names.
Edit a table	Double-click it, or use the grid in the right panel. Row 1 = header.
Add a cover & contents	Metadata… → tick **Cover page** and **Table of content**.
Brand the pages	Metadata… → BRANDING → logo, series label, disclaimer, confidentiality.
Recolour the report	Theme… → pick a preset (live preview) → OK.
Check the output	Preview (identical to Export). Then Export PDF.
Reopen a report	Home → Recent, or Recent Reports… in the rail.
Refresh all data	Command bar → ⟳ Refresh All; read the "Data as of" stamp.

Honest caveats worth knowing

- **Freshly-applied template charts show a placeholder** ("add data in the Properties panel") until you fetch or type values — templates seed structure, not data.
- **Applying a template clears the undo stack** — you can't Ctrl+Z it. Save first.
- **Typed chart/stats tickers aren't kept until you FETCH**; click away first and the symbol is lost (Market-data tiles do keep theirs).
- **Heading levels look identical** on the page and in this export path, though H1/H2/H3 are recorded.
- Editing markdown shows **literal marks** (`**`, `<u>`) until render/export.
- The **PDF contents page fits on one page** and finds headings by their on-page style; very long reports may not list every heading.
- Table and market-data **STYLE tabs are informational** — those blocks always follow the theme.
- Live quotes and price history come from the terminal's market feed; non-US symbols may need an exchange suffix (e.g. `.NS`).

> **THE ONE-LINE SUMMARY**
> Report Builder is a block-based authoring desk where **you and the AI edit the same live document**, data blocks bind to real tickers, and a single compiler makes the **preview identical to the branded PDF** your client receives.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Excel Spreadsheet

A complete, screen-by-screen walkthrough of the Excel tab — the terminal's built-in multi-sheet spreadsheet for scratch models, imported workbooks and CSV hand-off.

`>_ multi-sheet grid · 6 formula functions · xlsx in/out · 18 agent tools`

Covers · **toolbar, grid and formula bar** · **formulas and their limits** · **sheets** import & export · the right-click menu · what persists · the 18 agent tools

Demonstrated live on the Fincept Terminal Excel tab.

Contents

12 SECTIONS

GETTING ORIENTED

01 What the Excel tab is 3

02 Anatomy of the tab 4

03 The toolbar 5

WORKING IN THE SHEET

04 Entering data: the grid and the fx bar 6

05 Formulas — functions and arithmetic 7

06 Formula limits you should know 8

07 Sheets — add, rename, delete 9

08 The right-click menu 10

GETTING DATA IN AND OUT

09 Import and export 11

10 What is saved, and what is not 12

REFERENCE

11 Driving the sheet from AI Chat 13

12 Quick reference and honest notes 14

> **About this guide.** Every screenshot in this document is a live capture of the Excel tab in the shipping build — including the worked formula example, which was typed into the grid, photographed, and then removed.

01 What the Excel tab is

ORIENTATION · 01

The Excel tab is a **spreadsheet that lives inside the terminal**. It gives you a familiar lettered grid with a formula bar, several sheets in one workbook, a small set of spreadsheet functions, and two-way `.xlsx` / `.csv` file exchange — so a quick margin calculation, a scratch model or a counterparty's workbook never has to leave the research desk.

It is deliberately a *scratch pad*, not a replacement for Microsoft Excel. There is no charting, no pivot table, no conditional formatting and no cell styling. What it does give you is the shortest path from "I need to check this arithmetic against the numbers on screen" to an answer — plus a clean export that other Fincept screens and your colleagues can read.

Where to find it

Route	How
Navigate menu	**Navigate ‣ Tools ‣ Excel**
Command bar	Type `excel`, `spreadsheet` or `xls`, then pick the row
From a file	In **File Manager**, press **EXCEL** on any spreadsheet or CSV row

At a glance, in the Excel tab you can…

- Type numbers and text into a 100-row × 26-column grid
- Write formulas — `=SUM`, `=AVG`, `=MIN`, `=MAX`, `=COUNT` and arithmetic
- Keep several sheets in one workbook and reorder their tabs
- Import an `.xlsx` workbook, every sheet at once
- Export the whole workbook back to `.xlsx`
- Export the active sheet as a `.csv`
- Insert and delete rows and columns from the right-click menu
- Have every export land automatically in File Manager
- Drive all of it from AI Chat through 18 agent tools

> **One thing to know before you start.** Cell contents live in memory for the length of your terminal session — they are never written to disk on their own. Section 10 sets out exactly what survives and what does not, and the habit that keeps you safe.

02 Anatomy of the tab

Five regions, top to bottom. Everything in this guide refers back to them.

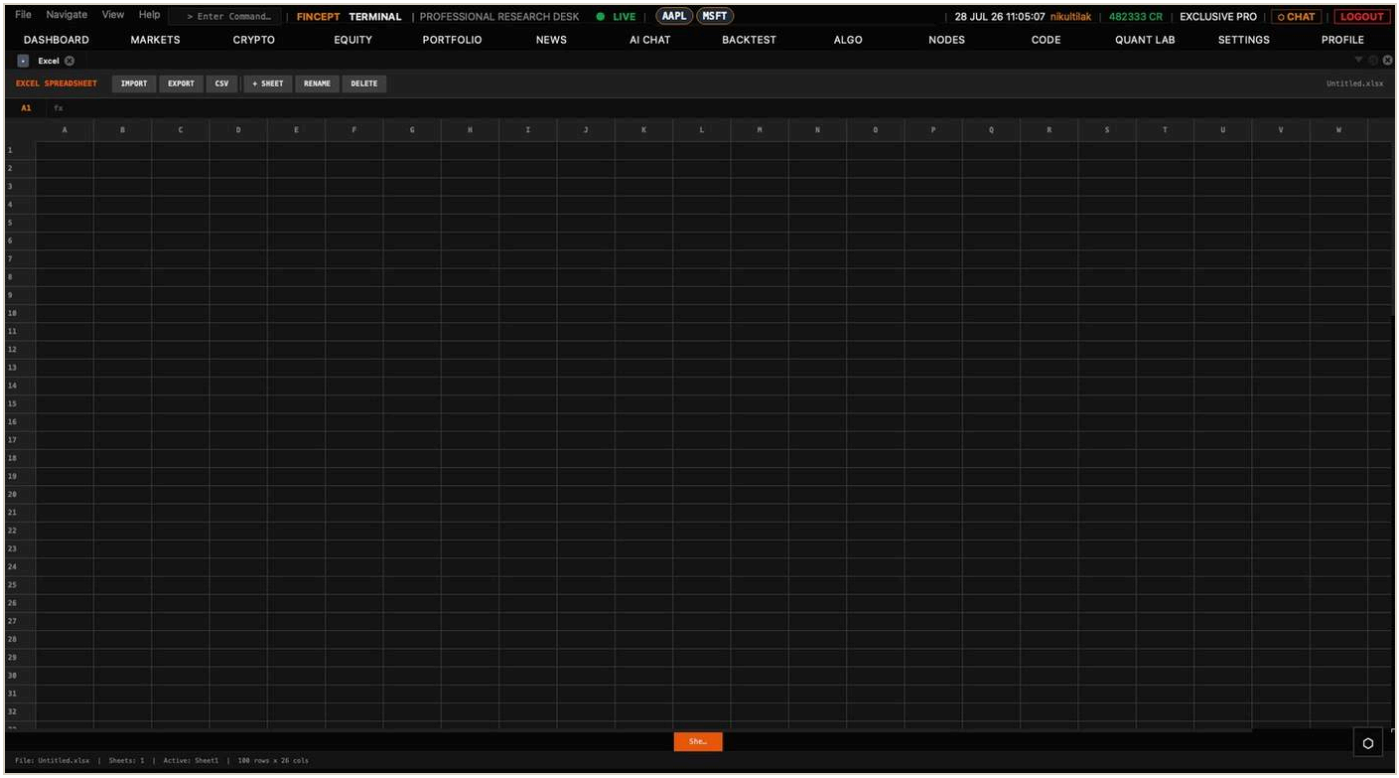


Fig 1 — The Excel tab on open, with all five regions visible.

#	Region	What it is for
1	Toolbar	The amber **EXCEL SPREADSHEET** strip: three file buttons (IMPORT / EXPORT / CSV), three sheet buttons (+ SHEET / RENAME / DELETE), and the current file name on the right. Section 03.
2	Cell reference + **fx** bar	Shows which cell is selected and the raw contents of that cell — the formula, not the result. You can type here instead of into the grid. Section 04.
3	The grid	100 rows × 26 columns (`A` – `Z`) on a new sheet. Columns and rows resize by dragging their header dividers. Section 04.
4	Sheet tabs	One tab per sheet along the bottom, active tab in amber, draggable to reorder. Section 07.
5	Status bar	A live summary: file name, sheet count, active sheet, and its dimensions.

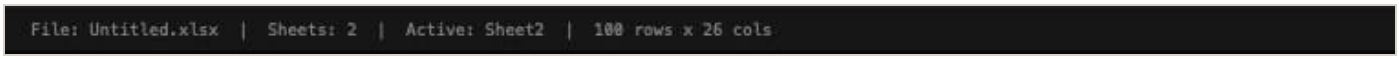


Fig 2 — The status bar answers four questions at a glance, and updates on every sheet add, delete, rename and switch.

03 The toolbar

Six buttons and a file name. Everything the tab can do to a *file* or a *sheet* is here; everything it can do to a *cell* is in the right-click menu (section 08).

Fig 3 — The toolbar. Left: the tab title. Middle: three file buttons, a divider, then three sheet buttons. Right: the current file name.

Control	Type	What it does
`IMPORT`	button	Opens a file picker and loads an `.xlsx` workbook — **replacing** every sheet currently open. Section 09.
`EXPORT`	button	Saves the whole workbook (all sheets) as `.xlsx` and registers it in File Manager.
`CSV`	button	Saves **only the active sheet** as a comma-separated file, trimmed to the last used cell.
`+ SHEET`	button	Adds an empty 100 × 26 sheet named `SheetN` and switches to it.
`RENAME`	button	Renames the active sheet through a small text dialog.
`DELETE`	button	Deletes the active sheet after a confirmation. Turns red on hover — it cannot be undone.
`Untitled.xlsx`	label	The current file name. It changes when you import or export; it is not a save target.

> The toolbar has **no Save button**, and that is deliberate — the workbook only becomes a file when you press **EXPORT** or **CSV**. Both also register the result in File Manager, so an export is filed for you rather than lost in a downloads folder.

04 Entering data: the grid and the fx bar

A new sheet is **100 rows by 26 columns** — `A` to `Z`, rows `1` to `100`. Rows are 22 px tall and columns 80 px wide; both drag wider or narrower from the header dividers, and inserting columns past `Z` continues `AA`, `AB`, and so on.

To do this	Do this
Start typing into a cell	Select it and simply type — the first keystroke opens the editor
Edit an existing value	Double-click the cell, or select it and press **Enter**
Commit what you typed	Press **Enter**, or click another cell
Type through the formula bar	Select the cell, click the **fx** field, type, press **Enter**
Select a block	Click and drag, or shift-click — the right-click menu acts on the whole block
Empty a block	Select it, then right-click ▸ **Clear Cell**

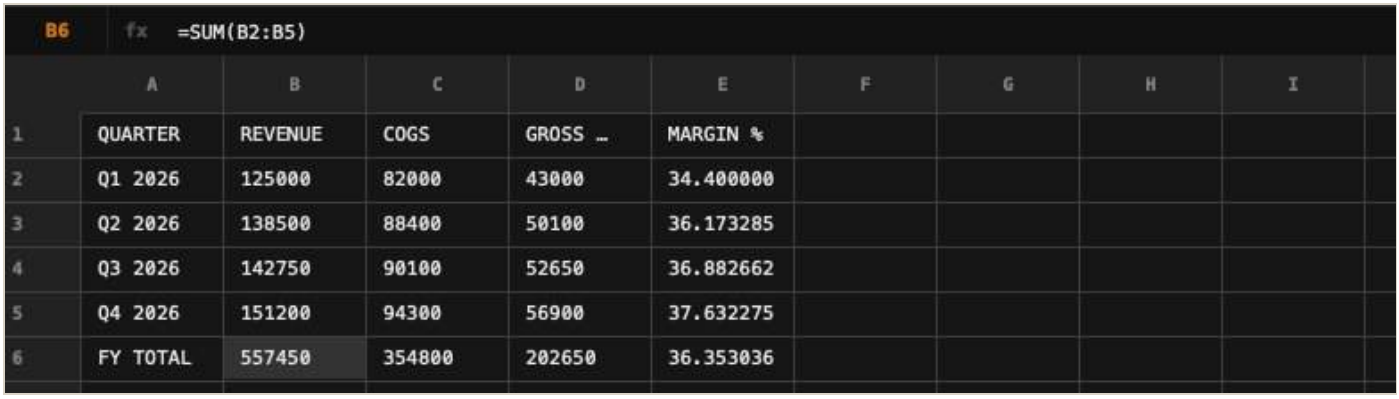


Fig 4 — The reference box reads **B6**, the **fx** field shows the raw formula `=SUM(B2:B5)`, and the cell itself shows the result, 557450.

> **Two Enter keys, two behaviours.** Pressing **Enter** in the **fx** formula bar commits the cell *and moves down one row*, so you can type a column of numbers without touching the mouse. Pressing **Enter** while editing *inside* a cell commits the value but leaves the selection where it is — use the arrow keys to move on.

> **The fx bar refreshes when you change cells, not while you type.** Type a formula straight into a cell and the **fx** field keeps showing what was there before, until you select another cell and come back. The cell's own value is always correct — this is a display quirk in the bar, not a calculation error.

05 Formulas — functions and arithmetic

Any cell that starts with `=` is a formula: the sheet remembers what you typed and shows the calculated result. There are six functions and full four-operator arithmetic.

The six functions

Formula	Result
`=SUM(B2:B5)`	Total of the range
`=AVG(B2:B5)` `or` `=AVERAGE(B2:B5)`	Mean of the range
`=MIN(B2:B5)`	Smallest value in the range
`=MAX(B2:B5)`	Largest value in the range
`=COUNT(B2:B5)`	How many cells the range spans — see section 06
`=B6`	A plain reference: mirrors another cell's value

A function takes **exactly one rectangular range**, written `=FUNC(first:last)`. Ranges may run across rows, down columns, or both — `=SUM(B2:D5)` sums the whole block. Text and empty cells inside a range count as zero.

Arithmetic

Outside those six you get `+`, `-`, `*`, `/` and parentheses over cell references and numbers, with normal precedence:

- `=B2-C2` — difference of two cells
- `=(B2-C2)/B2*100` — a percentage margin
- `=B2*1.18` — a cell against a constant
- `=$B$2+$C$2` — dollar signs are accepted and ignored

	A	B	C	D	E	F	G	H	I
1	QUARTER	REVENUE	COGS	GROSS …	MARGIN %				
2	Q1 2026	125000	82000	43000	34.400000				
3	Q2 2026	138500	88400	50100	36.173285				
4	Q3 2026	142750	90100	52650	36.882662				
5	Q4 2026	151200	94300	56900	37.632275				
6	FY TOTAL	557450	354800	202650	36.353036				

Fig 5 — A worked example. Column B is typed revenue, column C typed cost; row 6 is `=SUM(…)`, column D is `=B-C`, column E is `=(B-C)/B*100`. Whole numbers print plainly; fractional results print to six decimal places.

06 Formula limits you should know

The engine is small and predictable. These are its exact edges — worth reading once before you lean on a number.

Rule	What actually happens
`Functions do not nest or mix`	`=SUM(B2:B5)*2` returns `#ERR`. Put the SUM in its own cell and multiply that cell instead.
`One range per function`	`=SUM(B2:B5,D2:D5)` is not supported. Total each range separately, then add the two cells.
`COUNT counts cells, not values`	`=COUNT(B2:B9)` over four filled and four empty cells returns 8, not 4. For a true count of entries, keep the range tight.
`Division by zero returns 0`	There is no `#DIV/0!`. An empty denominator quietly yields zero, so sanity-check ratios built on cells that may be blank.
`Number formatting is fixed`	Integers print as integers; anything else prints to six decimal places. There is no currency, percent or decimal formatting.
`Anything unparseable is #ERR`	Text inside arithmetic, an unknown function name, or a malformed reference all produce `#ERR`.
`No circular-reference guard`	Never point a formula at its own cell — `=A1` typed into `A1` re-enters the evaluator instead of reporting an error.
`Row and column edits do not rewrite ranges`	References are literal text. Insert a row inside a `=SUM(B2:B5)` range and the formula still reads B2:B5 — widen it yourself.
`No fill handle`	Formulas are not copied down a column automatically. Type each one, or have an agent write the column (section 11).

> **Rule of thumb.** One idea per cell. The engine is happiest with a column of typed numbers, one `=SUM` beneath it, and a separate cell for each derived ratio — which is also the layout that exports most cleanly to `.xlsx`.

07 Sheets — add, rename, delete

One workbook holds as many sheets as you like. Their tabs sit along the bottom edge of the grid; the active tab is amber. Tabs can be **dragged left and right** to reorder them, and that order is what the `.xlsx` export writes.

Fig 6 — Two sheet tabs, the active one amber. At the default width the labels clip to a few characters, so keep sheet names short and read the full name from the status bar.

Action	Route	Notes
`Add a sheet`	**+ SHEET**	Creates an empty 100 × 26 sheet named `Sheet2`, `Sheet3`… and switches to it
`Switch sheet`	Click its tab	The status bar updates; the **fx** bar follows the new sheet's selection
`Reorder sheets`	Drag a tab	Export order follows tab order
`Rename a sheet`	**RENAME**	Pre-filled with the current name; a blank name is rejected
`Delete a sheet`	**DELETE**	Always asks first. The last remaining sheet cannot be deleted

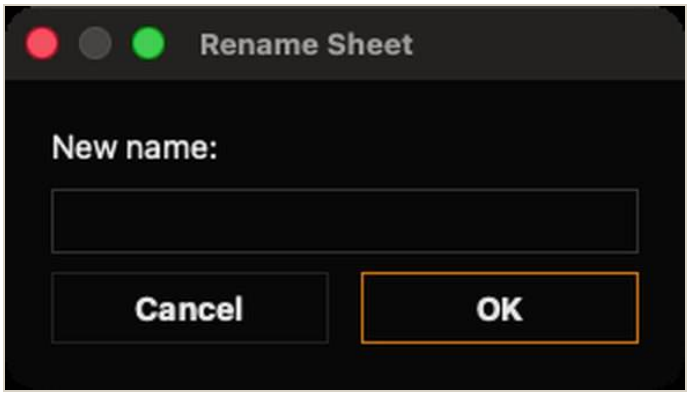


Fig 7 — **RENAME** opens a small dialog pre-filled with the current name.

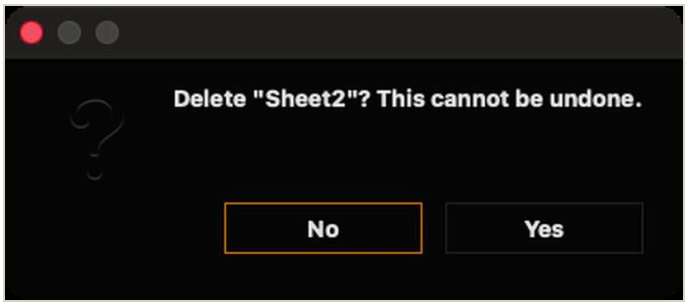


Fig 8 — **DELETE** always confirms, and defaults to **No**.

> Deleting a sheet discards its cells immediately — there is no undo. If a sheet holds anything you might want later, press **CSV** or **EXPORT** first. Try to delete the only sheet in the workbook and the tab refuses with *"Cannot delete the last sheet."*

08 The right-click menu

Right-clicking anywhere in the grid opens the cell menu — ten actions in four groups. This is where row and column editing lives.

Fig 9 — The full cell context menu.

Action	Acts on	What it does
`Cut`	current cell	Copies the **displayed** value to the clipboard, then empties the cell
`Copy`	current cell	Copies the displayed value — a formula copies as its result, not as the formula
`Paste`	current cell	Pastes clipboard text into that one cell
`Insert Row Above / Below`	current row	Inserts one empty row and pushes the rest down
`Insert Column Left / Right`	current column	Inserts one empty column and re-letters the headers
`Delete Selected Rows`	selection	Removes every row the selection touches
`Delete Selected Columns`	selection	Removes every column the selection touches
`Clear Cell`	whole selection	Empties every selected cell but keeps the rows and columns

Cut, Copy and Paste are single-cell operations. They move one value at a time through the system clipboard, so you cannot duplicate a block of cells inside the sheet, and pasting multi-line clipboard text drops all of it into one cell. **Clear Cell** is the one action that honours a multi-cell selection.

Because **Copy** takes the displayed value, it is the quickest way to lift a calculated result out of the terminal and into an email or a chat message — you get the number, not the formula behind it.

09 Import and export

The tab reads and writes real Excel workbooks, and everything it writes is registered with **File Manager** automatically.

IMPORT — reading a workbook

- Opens a system file picker and loads an `.xlsx` workbook.
- **Every sheet** in the file becomes a tab, carrying the file's own sheet names.
- Sheets are sized to their data, but never smaller than 100 × 26.
- The toolbar file name changes to the imported file, and the file is added to File Manager under the **EXCEL** source.

> **Import replaces the whole workbook** — every sheet currently open is closed first. And although the picker offers `.xlsx`, `.xls` and `.csv`, **only `.xlsx` is actually parsed**: choosing an old `.xls` file or a `.csv` leaves the grid unchanged with no error. To bring a CSV in, re-save it as `.xlsx` first, or paste the values by hand.

EXPORT — writing .xlsx

- Writes **all sheets**, in tab order, with their names.
- Anything that looks like a number is written as a number, so Excel can total and chart it.
- Formula text is written as a formula and Excel recalculates it on open.
- Empty cells are skipped, so the file stays small.

CSV — writing the active sheet

- Only the **active** sheet is written; run it once per sheet if you need several.
- The sheet is trimmed to the last cell that actually holds something.
- Values containing a comma, a quote or a line break are quoted and escaped properly; the file is UTF-8.

> **CSV export writes formulas as text.** A cell holding `=SUM(B2:B5)` exports as the literal string, not as 557450. Spreadsheet applications recalculate it on open, but a data pipeline reading the CSV will see the formula. When a downstream system needs plain numbers, export to `.xlsx` instead, or copy the displayed results into a clean sheet first.

10 What is saved, and what is not

This is the most important page in the guide.

Item	Saved?	Where
`Which sheet tab was active`	Yes	Terminal screen state — restored when the tab reopens
`Cell contents`	**No**	Held in memory for the life of the terminal session only
`Sheet names and sheet count`	**No**	A reopened tab starts again with one empty `Sheet1`
`The file name in the toolbar`	**No**	Resets to `Untitled.xlsx`
`Files you exported`	Yes	On disk, and indexed in File Manager

> **Quit the terminal and the grid is empty next time.** The workbook is a session scratch pad: never written to disk on its own, never synced to your Fincept account, never part of a saved layout. Anything you want tomorrow must be exported today — **EXPORT** for the whole workbook, **CSV** for one sheet.

Switching to another tab and coming back is safe: the screen stays alive inside the window for as long as the terminal runs, so your cells are still there. What clears the grid is quitting the terminal, closing the Excel tab itself, or importing another workbook over the top.

> **A habit that works.** Treat each session's sheet as disposable and export at the end of every piece of work. Because exports register in File Manager automatically, the export *is* your filing system — you never have to choose a folder or remember a path. Use **File Manager ▸ SAVE** to pull a managed copy back out to any location later.

11 Driving the sheet from AI Chat

The Excel tab publishes **18 agent tools** in the `excel` category. With the tab open you can ask the assistant in AI Chat to build or read a sheet in plain language — "put my holdings in a sheet and total the market value" — and it will use these:

Group	Tools	What they cover
`Sheets (6)`	list_excel_sheets, get_excel_active_sheet, set_excel_active_sheet, add_excel_sheet, delete_excel_sheet, rename_excel_sheet	Everything the three sheet buttons do
`Cells (3)`	get_excel_cell, set_excel_cell, clear_excel_cell	Read or write one cell, including writing a formula
`Sheet data (4)`	get_excel_sheet_data, set_excel_sheet_data, get_excel_sheet_dimensions, recalculate_excel_sheet	Bulk read and write of a whole sheet, plus a forced recalculation
`Rows & columns (4)`	insert_excel_row, insert_excel_col, delete_excel_rows, delete_excel_cols	The structural half of the right-click menu
`Export (1)`	export_excel_sheet_csv	Writes the active sheet to a CSV path

> **The tools drive the live tab, so the Excel tab must be open.** If it is not, every call returns *"Excel screen not open"*. Open it once — Navigate ▸ Tools ▸ Excel — and leave it in the window while you work with the assistant.

Two things worth asking for

- **Bulk fill.** `set_excel_sheet_data` writes an entire grid in one call, far faster than dictating cells one at a time — ideal for dropping a table from another screen into the sheet.
- **Formulas, not just values.** `set_excel_cell` accepts `=SUM(B2:B20)` exactly as if you had typed it, so an agent can leave a *live* model behind rather than a static snapshot.

> **xlsx import and export are deliberately not exposed as tools**, because both are bound to a system file dialog only a person can answer. CSV export *is* programmatic, so an agent can build a sheet and write it out end to end.

12 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Tools ▸ Excel, or type `excel` in the command bar
…total a column?	Put `=SUM(B2:B20)` in the cell below it
…work out a percentage?	`=(B2-C2)/B2*100` — the result prints to six decimals
…see a cell's formula?	Select the cell and read the **fx** bar (select another cell and back if you just typed it)
…add or rename a sheet?	**+ SHEET**, then **RENAME**
…insert a row?	Right-click a row ▸ **Insert Row Above / Below**
…empty a block of cells?	Select it, right-click ▸ **Clear Cell**
…keep my work?	**EXPORT** (all sheets, .xlsx) or **CSV** (active sheet) — nothing else persists
…find what I exported?	File Manager, under the **EXCEL** source filter
…reopen a CSV already in the terminal?	File Manager ▸ the **EXCEL** button on that row
…bring in a CSV from outside?	Re-save it as `.xlsx` first — the CSV import path does not parse
…let the assistant fill the sheet?	Leave the tab open and ask in AI Chat; it uses the 18 `excel` tools

Honest notes on this build

- **Cells are session-only.** Export before you quit; there is no autosave (section 10).
- **The import filter over-promises** — `.xls` and `.csv` can be chosen but are not parsed, and the failure is silent.
- **Functions do not compose**, and take one range each.
- **COUNT returns the size of the range**, not the number of filled cells.
- **Division by zero yields 0** rather than an error.
- **No circular-reference detection** — never point a formula at its own cell.
- **Row and column edits do not rewrite formula ranges**; re-check totals after inserting.
- **Copy/Cut/Paste are one cell at a time**, and copy yields the value, not the formula.
- **CSV export writes formula text**, not calculated values.
- **Sheet tab labels clip** at the default width; the status bar shows the full name.
- **No formatting, charts or pivots.** For presentation output, export and finish in Report Builder or a desktop spreadsheet.

> Nothing here was inferred from a manual: each behaviour was read out of the shipping source and then confirmed on screen — including the ones that surprised us.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Research Notes

A complete, screen-by-screen walkthrough of the Notes tab — the terminal's research journal: categorised, tickered, searchable notes that live next to the data they were written about.

`>_ 8 categories · priority · sentiment · tickers · markdown export`

Covers · **the three panels** · **writing and reading a note** · **the metadata model**
markdown export · archive vs delete · storage and sync · the 4 agent tools

Demonstrated live on the Fincept Terminal Notes tab.

Contents

12 SECTIONS

GETTING ORIENTED

01 What the Notes tab is 3
02 Anatomy of the tab 4
03 The category sidebar 4
04 The notes list 6

WRITING AND READING

05 Reading a note 7
06 Writing a note 8
07 The metadata model 9

GETTING NOTES OUT, AND AWAY

08 Exporting a note as markdown 10
09 Favourite, archive and delete 11
10 Where notes are stored 12

REFERENCE

11 Driving notes from AI Chat 13
12 Quick reference and honest notes 14

About this guide. Every screenshot is a live capture of the Notes tab in the shipping build. The worked example note was written for this guide, exported, photographed and then deleted — the journal was left exactly as it was found.

01 What the Notes tab is

ORIENTATION · 01

Notes is the terminal's **research journal**. Each note carries a title, free text, a category, a priority, a directional view, free-form tags and a list of tickers — so the thesis you wrote before an earnings call is still findable, and still attached to the names it was about, three months later.

It is a three-panel workspace: categories on the left, the notes in that category in the middle, and the note itself on the right, either being read or being written. Nothing about it is clever — that is the point. It is a fast, quiet place to write down what you think, inside the same window as the data that made you think it.

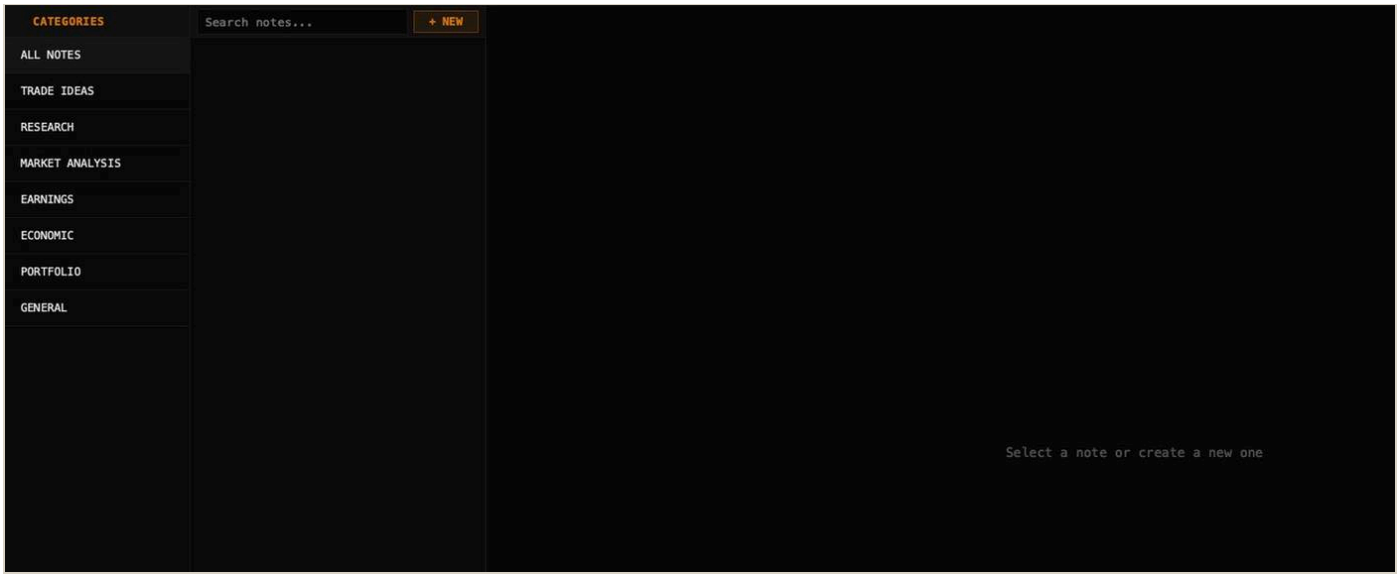


Fig 1 — A fresh journal. The eight categories are always present; the middle and right panels fill in as you write.

Where to find it

Route	How
Navigate menu	**Navigate ‣ Tools ‣ Notes**
Command bar	Type `notes` or `note`, then pick the row

At a glance, in the Notes tab you can…

- Write a note with a title, body and metadata
- File it under one of eight categories
- Flag it HIGH, MEDIUM or LOW priority
- Record a BULLISH, BEARISH or NEUTRAL view
- Attach tickers so the note is findable by name
- Add free-form tags
- Search titles, bodies, tags and tickers at once
- Star the notes you return to
- Export any note as a markdown file
- Have an assistant write and update notes for you

> **The command palette lists a shortcut for this tab that does not apply.** Notes appears with **F11** beside it, but F11 is the terminal's fullscreen key — it will not open the journal. Use the menu or type the name into the command bar.

02 Anatomy of the tab

Three panels, divided by draggable splitters.

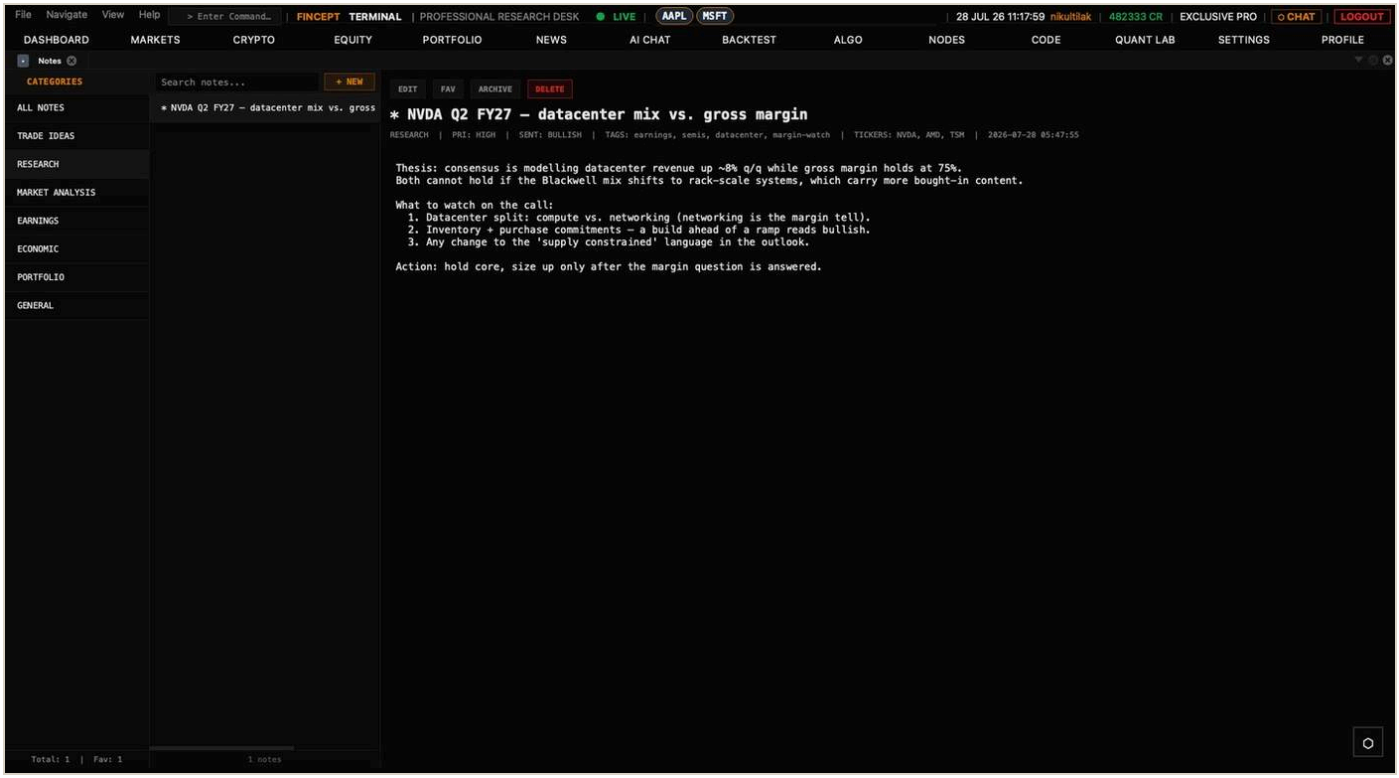


Fig 2 — The Notes tab with one note selected. Left: categories. Middle: the notes in the selected category. Right: the note in view mode.

#	Region	What it is for
1	Category sidebar	**CATEGORIES** header, the eight fixed categories, and a footer counting total and starred notes. Section 03.
2	Search + NEW	Free-text search across the journal, and the amber **+ NEW** button that starts a blank note. Section 04.
3	Notes list	One row per note in the current category, coloured by priority, starred if favourite, with its tickers in brackets. Section 04.
4	Note panel	Three states: an empty prompt, **view mode** (section 05) and **edit mode** (section 06).
5	List footer	*N notes* — the number of rows currently showing, after category and search filtering.

> The right-hand panel is a stack of three pages, not a scrolling document. Selecting a note shows it read-only; pressing **EDIT** or **+ NEW** swaps in the editor. That is why there is no cursor in the text you are reading — you have to ask for the editor.

03 The category sidebar

Eight fixed categories. **ALL NOTES** shows the whole journal; the other seven are the buckets you can file a note into. The list is not editable — you cannot add, rename or remove a category — which keeps a shared journal consistent.

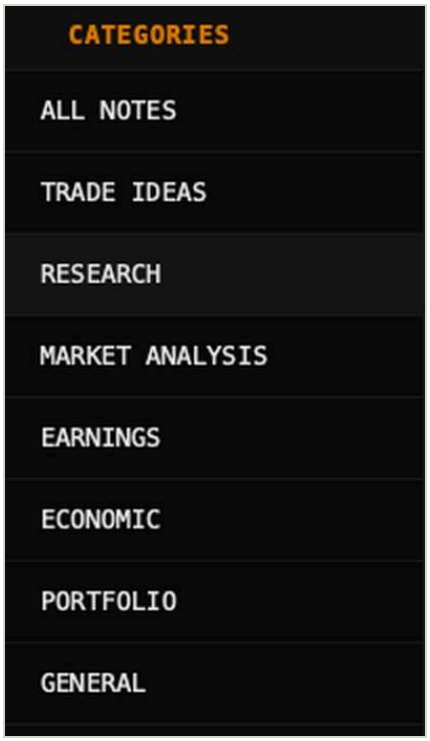


Fig 3 — The category sidebar.

Category	What it is for
ALL NOTES	Every note in the journal, newest first
TRADE IDEAS	A specific trade you are considering or have taken
RESEARCH	Company or sector work — the deep reading
MARKET ANALYSIS	Cross-asset and market-structure observations
EARNINGS	Call previews, results reactions, guidance changes
ECONOMIC	Macro prints, central-bank meetings, policy
PORTFOLIO	Position reviews, sizing decisions, risk notes
GENERAL	Everything else — and the default for a new note

Fig 4 — The sidebar footer: total notes in the journal, and how many are starred. It counts the whole journal, not the current category.

Behaviour worth knowing

- **Selecting a category filters the middle panel** and clears the note panel, so you always start from a clean list.
- **Your category choice is remembered** between sessions, along with the note you had open.
- **A new note inherits the category you are standing in** — start from RESEARCH and the editor opens pre-filed as RESEARCH. From **ALL NOTES**, the default is GENERAL.

04 The notes list

The middle panel lists the notes in the current category, most recently updated first. Each row packs four pieces of information into one line.

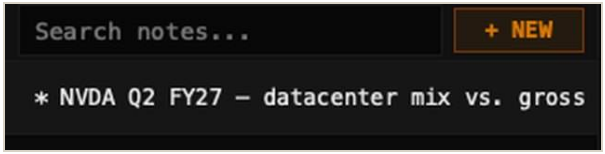


Fig 5 — A list row: the * marks a favourite, the title follows, and the note's tickers appear in brackets. The row's colour is its priority.

Element	Meaning
`* prefix`	The note is starred as a favourite
`Title`	The note's title, as typed
`[TICKERS]`	The note's tickers, appended in brackets — this is what makes a note findable by name in a long list
`Row colour`	Priority: **red** for HIGH, **amber** for MEDIUM, **green** for LOW

Search

The search field filters the list as you type, and it is broader than it looks: it matches the **title**, the **body text**, the **tags** and the **tickers** of every note — so `nvda` finds notes that mention NVDA anywhere, not just in the title.

Search behaviour	Detail
`Fields searched`	Title, body, tags, tickers
`Not searched`	Category — use the sidebar for that
`Combined with category`	Yes. Search narrows within the selected category; pick **ALL NOTES** to search the whole journal
`Case`	Insensitive

> **+ NEW starts a blank note immediately.** There is no dialog: the right-hand panel switches to the editor with the category pre-filled from the sidebar. If you change your mind, press **CANCEL** — nothing is written until you press **SAVE**.

05 Reading a note

VIEW MODE · 05

Selecting a row opens the note read-only, with a four-button toolbar above it and a single metadata line under the title.

```
EDIT   FAV   ARCHIVE   DELETE

NVDA Q2 FY27 — datacenter mix vs. gross margin

RESEARCH | PRI: HIGH | SENT: BULLISH | TAGS: earnings, semis, datacenter, margin-watch | TICKERS: NVDA, AMD, TSM | 2026-07-28 05:47:31

 Thesis: consensus is modelling datacenter revenue up ~8% q/q while gross margin holds at 75%.
 Both cannot hold if the Blackwell mix shifts to rack-scale systems, which carry more bought-in content.

 What to watch on the call:
   1. Datacenter split: compute vs. networking (networking is the margin tell).
   2. Inventory + purchase commitments — a build ahead of a ramp reads bullish.
   3. Any change to the 'supply constrained' language in the outlook.

 Action: hold core, size up only after the margin question is answered.
```

Fig 6 — View mode. The * in the title marks a favourite; the metadata line carries category, priority, sentiment, tags, tickers and the last-updated timestamp.

Button	What it does
`EDIT`	Swaps in the editor with every field populated — section 06
`FAV`	Toggles the favourite star. The row gains a * and the sidebar's *Fav* count goes up
`ARCHIVE`	Hides the note from the journal. **Read section 09 before using this**
`DELETE`	Removes the note permanently, after a confirmation

The metadata line

Everything the note knows about itself, in one row, separated by pipes:

- **Category** — as filed, in its stored form (`RESEARCH`, `TRADE_IDEA`).
- **PRI:** priority, and **SENT:** the directional view.
- **TAGS:** and **TICKERS:** — shown only if the note has them.
- **The last-updated timestamp**, written when the note was last saved.

> **The timestamp is in UTC, not your local time.** A note saved at 11:17 in India shows `05:47` here. File Manager, by contrast, converts to local time — so the same note exported to markdown will appear to carry two different times. Both are correct; only the zone differs.

> **The body is shown as plain text.** Markdown you type — `#` headings, `**bold**`, bullets — is displayed literally rather than rendered. Write for legibility as plain text, and let the markdown pay off on export (section 08).

06 Writing a note

The editor is one screen: a toolbar, the title, two rows of metadata, and the body.

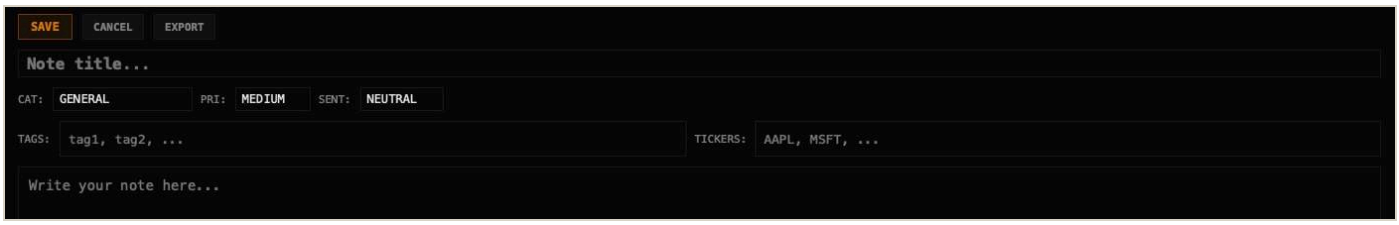


Fig 7 — A blank editor from **+ NEW**. Priority defaults to MEDIUM, sentiment to NEUTRAL, and the category to whichever one you were standing in.

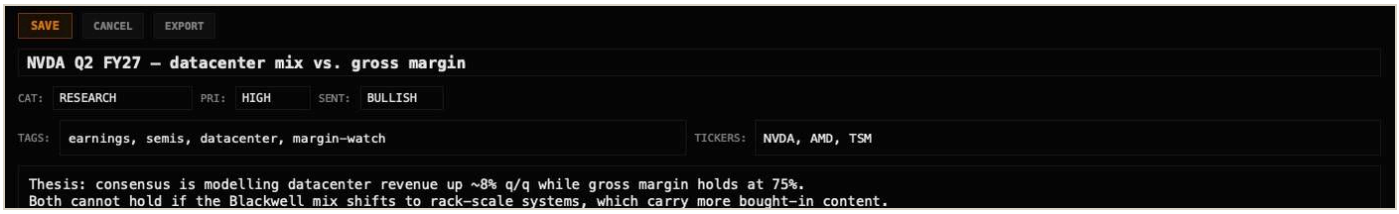


Fig 8 — The same editor filled in: a RESEARCH note, HIGH priority, BULLISH, tagged, and tickered NVDA, AMD, TSM.

Field	Type	Notes
`Title`	text	**Required.** Press SAVE with it empty and nothing happens except the cursor jumping back here
`CAT:`	7-option list	The seven filing categories — **ALL NOTES** is not offered, because it is a view, not a bucket
`PRI:`	HIGH / MEDIUM / LOW	Sets the note's colour in the list. Defaults to MEDIUM
`SENT:`	BULLISH / BEARISH / NEUTRAL	Your directional view. Defaults to NEUTRAL
`TAGS:`	text	Comma-separated and free-form — stored as typed and matched by search
`TICKERS:`	text	Comma-separated. **Upper-cased on save** and shown in brackets in the list
`Body`	multi-line text	The note itself. Enter inserts a line break; nothing is committed until SAVE

> **Drafts survive a restart.** If you are part-way through writing when the terminal closes, the title, body, tags, tickers and all three selectors come back when you return — even for a note that was never saved.

> **SAVE returns you to view mode** with the note selected in the list, so you can confirm the metadata line reads as you intended. **CANCEL** on an existing note goes back to view mode; on a brand-new note it returns to the empty prompt and discards the draft.

07 The metadata model

Six pieces of metadata, and each one earns its place by making notes findable later. This page is the whole model on one sheet.

Field	Values	What it drives
`Category`	TRADE_IDEA, RESEARCH, MARKET_ANALYSIS, EARNINGS, ECONOMIC, PORTFOLIO, GENERAL	Which sidebar bucket the note appears in
`Priority`	HIGH, MEDIUM, LOW	The row colour in the list — red, amber, green
`Sentiment`	BULLISH, BEARISH, NEUTRAL	Recorded on the note and shown in the metadata line
`Favourite`	on / off	A * in the list and the title, plus the sidebar's *Fav* count
`Tags`	free text, comma-separated	Searchable — the flexible axis, for themes that cut across categories
`Tickers`	free text, comma-separated, upper-cased	Searchable, and shown in brackets in the list

Using the model well

- **Category is the shelf; tags are the cross-reference.** A note on margin pressure at a chip maker belongs in RESEARCH, tagged `margins, semis` — so it turns up whether you come at it by sector or by theme.
- **Tickers are the highest-value field** — the one thing that lets you pull up every note on a name, from any category, in one search.
- **Priority is for triage, not importance.** Because it colours the row, HIGH works best as "come back to this before the print".
- **Sentiment is a record, not a filter.** No view lists all your bullish notes, but it makes reading an old note honest about what you thought at the time.
- **Three stored fields never surface here** — a colour code, a reminder date and a word count — so do not plan a workflow around note colours or reminders.

08 Exporting a note as markdown

EXPORT · 08

The editor's third button, **EXPORT**, writes the note out as a markdown file and files it in **File Manager**. It is the way a note leaves the terminal — into an email, a document, or a repository.

Aspect	Behaviour
Where the button is	**Edit mode only.** To export a note you are reading, press EDIT first
What is written	`# Title`, a blank line, then the body exactly as typed
What is not written	Category, priority, sentiment, tags and tickers are **not** included in the file
File name	The title with anything unusual replaced by underscores, plus `.md`; a name clash gets a timestamp suffix
Where it lands	Managed storage, registered in File Manager under the **NOTES** source
Saving required?	No — export uses what is in the editor, so you can export a draft

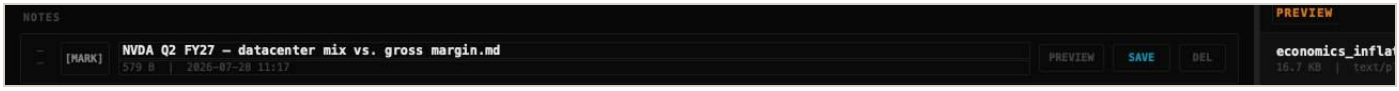


Fig 9 — The exported note in File Manager moments later: badge `[MARK]` for `text/markdown`, 579 bytes, filed under NOTES.

> **Export is silent.** No dialog, no toast, no change on screen — press it once and it is easy to believe nothing happened. It did: open File Manager and look under the **NOTES** chip. Pressing it twice writes two files, the second with a timestamp in its name.

> **Because the metadata is not exported**, put anything that must travel with the note into the body — a first line such as *RESEARCH · HIGH · BULLISH · NVDA, AMD, TSM* costs nothing and makes the markdown file stand on its own.

> **From File Manager the file behaves like any other export:** preview it in place, **SAVE** a copy anywhere on your machine, or hand it to an assistant to summarise. Deleting it there does not touch the note in the journal.

09 Favourite, archive and delete

Three buttons change a note's life. One is harmless, one is permanent, and one behaves in a way worth knowing before you press it.

FAV — harmless and reversible

Toggles the star. The list row gains a *, the title gains one too, and the sidebar footer's *Fav* count moves. Press it again to unstar. Nothing else changes.

DELETE — permanent, but it asks

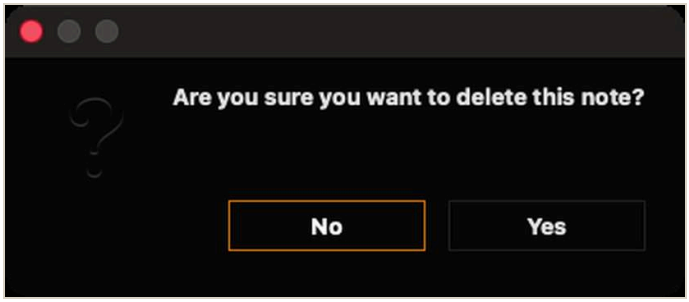


Fig 10 — The delete confirmation defaults to **No**.

Fig 11 — After deleting, the footer counts fall.

The note is removed from the journal and cannot be recovered from the interface. Export it first if there is any doubt (section 08).

> **ARCHIVE is effectively "hide for good" in this build.** Archiving removes the note from the list and clears the panel — and because every view of the journal excludes archived notes, and there is no *Archived* category to switch to, the note cannot be reached again afterwards. The assistant cannot see it either: the agent tools read the same filtered view. The note is still in your database and still counted nowhere, but nothing in the terminal will show it to you.

> **What to do instead.** If you want a note out of the way but recoverable, **EXPORT** it to markdown first — then archive or delete freely, knowing the content is filed in File Manager. If you simply want to de-emphasise a note, re-file it under GENERAL and take the star off; it stays reachable.

Action	Confirms?	Reversible in the interface?
`FAV`	No	Yes — press it again
`ARCHIVE`	No	**No** — the note leaves every view
`DELETE`	Yes	**No** — the note is gone

10 Where notes are stored

Notes are one of the domains that still keeps a local database, with a cloud mirror on top.

Aspect	Detail
`Local store`	A `financial_notes` table in the terminal's local database, under your profile
`Cloud mirror`	Notes are synced through the terminal's cloud-sync engine as the `note` entity
`When it pulls`	Every time you open the Notes tab, rate-limited so that flicking between tabs does not hammer the service
`What arrives`	A cloud change refreshes the list in place — you do not have to reload the tab
`Live updates from agents`	When an assistant creates or deletes a note, the tab is notified and the list refreshes itself
`Screen state`	The selected category, the open note, and any unsaved draft

> **What this means in practice.** Notes are the most durable thing in the terminal that you type: unlike the Excel scratch pad, they survive a restart without any action from you, and unlike a layout, they are content rather than configuration. Treat the journal as the place where thinking accumulates.

> **Sync is a mirror, not a merge.** The journal is designed around one desk writing its own notes. If the same account is open on two machines, the safest habit is to write in one place at a time and let the other pull the change when you next open the tab.

What is not stored

- **No attachments.** There is no way to attach a file or an image to a note; use File Manager and reference the file name in the body.
- **No version history.** Saving overwrites the previous text and updates the timestamp.
- **No links between notes.** Tickers and tags are the way to relate notes to one another.

11 Driving notes from AI Chat

Notes publishes **four agent tools** in the `notes` category — a small set that covers the whole life of a note, and one of the most natural things to hand to an assistant.

Tool	What it does
`create_note`	Writes a new note, with optional category, priority, tickers and sentiment
`update_note`	Edits an existing note in place by its id, keeping the same note and id
`get_notes`	Lists the journal, or searches it with a query
`delete_note`	Deletes a note by id

Why this is worth using

- **Dictated notes.** "Note this as a HIGH priority EARNINGS note on NVDA: ..." produces a properly filed, tickered note without you touching the editor.
- **End-of-day summaries.** An assistant can read what it just analysed on another screen and leave the conclusion in the journal, tagged so you find it later.
- **Amendments, not duplicates.** `update_note` edits in place, so asking to "add the actual result to my NVDA preview note" keeps one note with a history-free clean body rather than two competing ones.
- **The list keeps up.** Notes created or deleted by an assistant appear in — or vanish from — the middle panel immediately, without a manual refresh.

> **Archived notes are invisible to the tools too.** `get_notes` reads the same filtered view as the interface, so an assistant asked to “find my note about X” will not see an archived one. Another reason to prefer export-then-delete over archive (section 09).

> **Tickers are the handle to give an assistant.** Because `get_notes` searches titles, bodies, tags and tickers, a consistent ticker convention turns the journal into something an agent can actually navigate — “what did I write about TSM before the last print?” works when the notes carry `TSM`.

12 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Tools ▸ Notes, or type `notes` in the command bar
…write a note?	+ **NEW**, type a title, then **SAVE** — the title is the only required field
…edit one?	Select it, then **EDIT**
…find every note about a name?	Pick **ALL NOTES** and search the ticker
…file a note under a category?	Set **CAT:** in the editor, or start the note from that category
…star a note?	**FAV** in view mode
…get a note out of the terminal?	**EDIT ▸ EXPORT**, then find it in File Manager under **NOTES**
…include the metadata in an export?	Type it into the body — export writes the title and body only
…hide a note but keep it?	Export it, then delete. **ARCHIVE** cannot be undone
…recover an archived note?	Not from the terminal — see section 09
…dictate a note to the assistant?	Ask in AI Chat; it uses `create_note`

Honest notes on this build

- **ARCHIVE is one-way.** Archived notes are excluded from every view and from the agent tools, and there is no archived category to bring them back.
- **EXPORT lives in edit mode only**, and gives no confirmation when it writes.
- **Export omits the metadata** — title and body only.
- **Timestamps are shown in UTC** in the metadata line, while File Manager shows local time.
- **The body is not rendered.** Markdown appears as typed, in both view and edit modes.
- **The command palette advertises F11** for this tab; F11 is fullscreen.
- **Categories are fixed** — no adding, renaming or reordering.
- **Colour code, reminder date and word count exist in storage but not in the interface.**
- **No attachments, no version history, no note-to-note links.**
- **DELETE confirms; FAV and ARCHIVE do not.**

> Everything above was read from the shipping source and confirmed on screen, on a note written for this guide and then removed.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The File Manager

A complete, screen-by-screen walkthrough of the File Manager — the terminal's own index of everything it has exported, saved or generated, with preview, search and one-click hand-off back into the screens that made it.

`>_ one index · 11 producing screens · preview · 15 agent tools`

Covers · **search, sort and the source chips** · **the file card** · **preview**
bulk delete · where files come from · storage and quota · the 15 agent tools

Demonstrated live on a working terminal holding 18 managed files.

Contents

11 SECTIONS

GETTING ORIENTED

01 What the File Manager is 3

02 Anatomy of the tab 4

03 The header and the storage bar 5

FINDING AND USING FILES

04 Finding a file: search, sort, source chips 6

05 The file card and its four buttons 7

06 The preview panel 8

07 Selecting and deleting 9

WHERE FILES COME FROM

08 The eleven producing screens 10

09 Storage, quota and what is on disk 11

REFERENCE

10 Driving it from AI Chat 12

11 Quick reference and honest notes 13

About this guide. Every screenshot is a live capture of the File Manager in the shipping build, on a real file index. The Notes-to-File-Manager hand-off in section 08 was performed for this guide and then reversed, leaving the index exactly as it was found.

01 What the File Manager is

ORIENTATION · 01

The File Manager is **one index of every file the terminal has produced for you** — exported reports, saved notebooks, spreadsheet and CSV exports, portfolio dumps, backtest results, research statements, exported notes — plus anything you upload into it by hand. One list, grouped by the screen that made each file, with search, preview, download and delete.

The point is that you never have to remember where a screen put something. Every Fincept screen that writes a file registers it here automatically at the moment it is written, so the answer to "where did that export go?" is always the same: **File Manager**.

Fig 1 — The header: the tab title, a live file count and total size, **REFRESH**, and the amber **UPLOAD FILES** button.

Where to find it

Route	How
File menu	**File ▸ File Manager** — the quickest route, and the reason it sits in that menu
Navigate menu	**Navigate ▸ Tools ▸ File Manager**
Command bar	**Not available.** The File Manager has no command-bar alias — use either menu above

At a glance, in the File Manager you can…

- See every file the terminal has produced, newest first
- Filter to one producing screen with a single chip
- Search by file name, type or source screen
- Sort by date, name, size or type
- Preview text, CSV and JSON without leaving the tab
- Open a file back into the screen that understands it
- Save a copy anywhere on your machine
- Delete one file, or tick several and delete them together
- Upload your own files into the same index
- Watch storage against a 500 MB budget

02 Anatomy of the tab

Six regions stacked above a two-panel body.

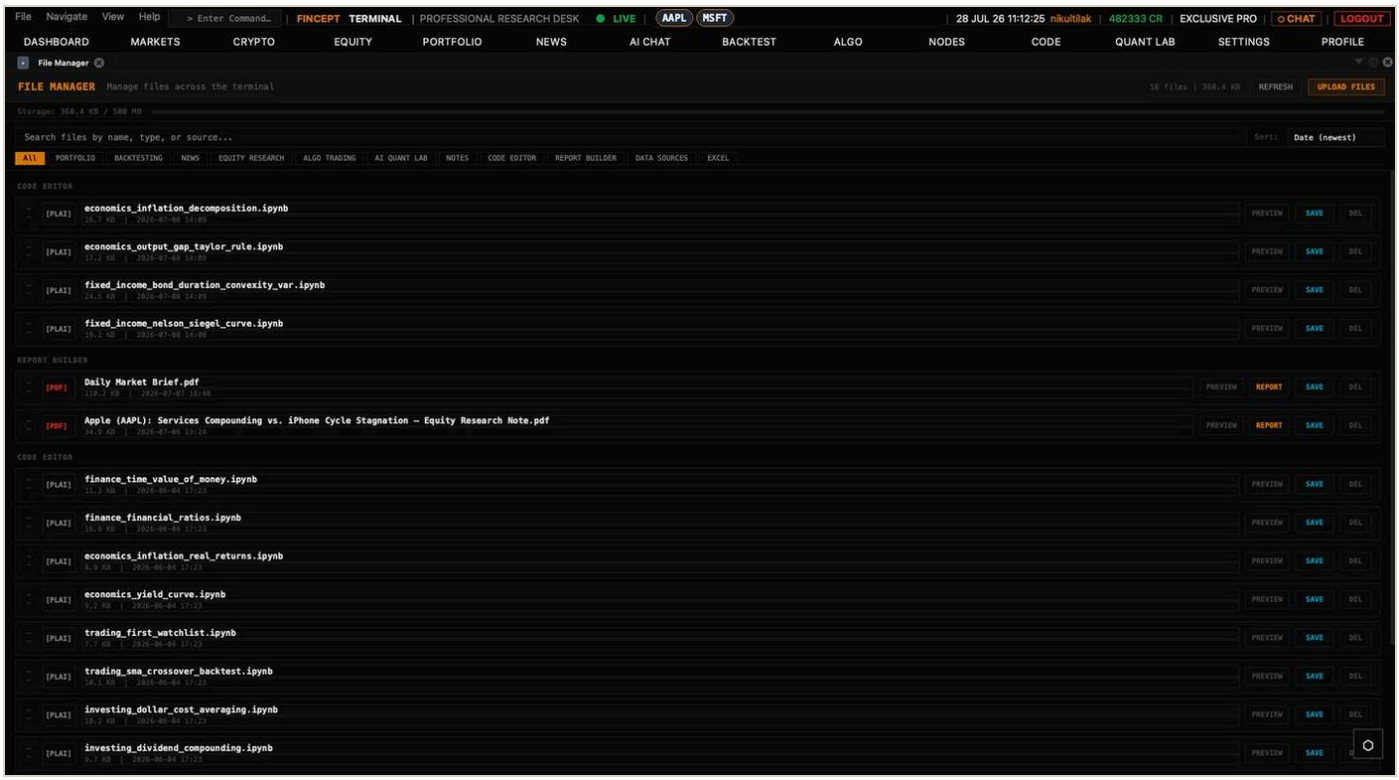


Fig 2 — The File Manager on a working index of 18 files, grouped by the screen that produced each one.

#	Region	What it is for
1	Header	Title, live *N files* | *total size* counter, **REFRESH**, **UPLOAD FILES**. Section 03.
2	Storage bar	Total managed size against a 500 MB budget, amber until 80% then red. Section 09.
3	Search + Sort	Free-text search over name, type and source; a seven-option sort. Section 04.
4	Source chips	**All** plus one rounded chip per producing screen — the fastest filter on the tab. Section 04.
5	Bulk bar	Hidden until you tick a checkbox, then offers **DELETE N SELECTED** and **CLEAR SELECTION**. Section 07.
6	File list	One card per file, grouped under the source screen's name. Section 05.
7	Preview panel	Opens on the right when you press **PREVIEW**; closes with its ✕. Section 06.

> The preview panel is hidden until you ask for it, so the list gets the full width while you are browsing. When it opens, the split is roughly 60 / 40 — drag the divider if you want more of the document, and press ✕ in the preview header to give the list all the room back.

03 The header and the storage bar

HEADER · 03

Control	What it does
`N files | size`	A live counter of every managed file and their total size. It updates the instant any screen registers or removes a file — you do not have to refresh to see a new export appear.
`REFRESH`	Re-reads the index and redraws the list. Rarely needed, because the tab listens for changes, but useful if a file was written by something outside the terminal.
`UPLOAD FILES`	Opens a multi-select file picker and **copies** each chosen file into managed storage. Your originals are left exactly where they are.

Fig 3 — The header. The counter on the left of REFRESH is the fastest way to confirm an export actually landed.

The storage bar

Directly under the header, a thin progress bar shows total managed size against **500 MB**. It is amber up to 80% of the budget and turns red past it, next to a *Storage: 360.4 KB / 500 MB* caption.

> **The 500 MB figure is a budget, not a wall.** Nothing blocks an upload or an export when the bar fills — it is there so a desk that generates hundreds of notebook and PDF exports notices before the profile folder grows unreasonably. When it turns red, sort by **Size (largest)** and clear out what you no longer need.

> **Uploads are not tagged with a source screen.** A file you add with **UPLOAD FILES** has no producing screen, so it groups under a blank heading at the top of the list and no source chip matches it — only **All** shows it. If you want a file filed under a screen's chip, produce it from that screen, or ask an agent to re-tag it (section 10).

04 Finding a file: search, sort, source chips

FILTERS · 04

Three filters, applied together. In practice the chips do most of the work.

Fig 4 — The search field, the sort selector, and the source chips. **All** is selected here; picking any other chip narrows the list to that screen's output.

Search

Type into *Search files by name, type, or source...* and the list narrows as you type. It matches on three things: the file's display name, its type, and its source screen. So `csv` finds every CSV, `portfolio` finds everything the Portfolio screen wrote, and `nvda` finds a file with NVDA in its name.

> **Search does not look inside files.** It matches names, types and sources only. To find a phrase inside a document, preview the file (section 06), or ask an assistant to read it with the `search_files` and `read_file_contents` tools (section 10).

Sort

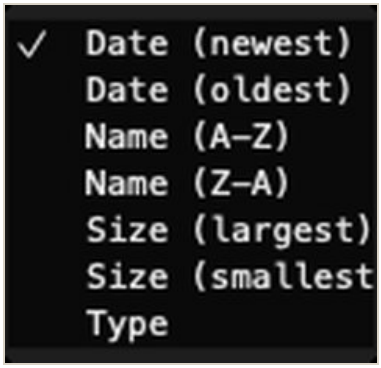


Fig 5 — All seven sort options.

Option	Use it when
`Date (newest)`	The default — the export you just made is at the top
`Date (oldest)`	Finding the earliest version of a recurring report
`Name (A–Z) / (Z–A)`	You know the file name and want it alphabetically
`Size (largest)`	Clearing space — pair it with the storage bar
`Size (smallest)`	Spotting truncated or empty exports
`Type`	Grouping all PDFs, then all notebooks, then all CSVs together

Source chips and grouping

The chips are **All** plus eleven producing screens: Portfolio, Backtesting, News, Equity Research, Algo Trading, AI Quant Lab, Notes, Code Editor, Report Builder, Data Sources and Excel. Exactly one chip is active at a time, and the choice is remembered between sessions.

With **All** selected the list is **grouped**: a small grey heading names the producing screen above each run of cards. Pick a single chip and the headings disappear, because every card below belongs to that one screen.

> **Why the same heading can appear twice.** Grouping labels a run of cards, and the list is sorted before it is grouped — so a date-sorted list that returns to a screen later in time prints that screen's heading again. It is not a duplicate group; it is the same screen, further down the timeline.

05 The file card and its four buttons

One card per file. Left to right: a selection checkbox, a coloured type badge, the file name with its size and timestamp, then the action buttons.

Fig 6 — Four notebook cards. The `[PLAI]` badge is the file's MIME subtype abbreviated — these notebooks are registered as `text/plain`.

Fig 7 — Two PDF cards. Because their type maps to a screen, they carry an extra amber **REPORT** button that opens them in Report Builder.

Element	What it means
☐ / ☑	Selection checkbox. Ticking any card reveals the bulk bar (section 07).
`Type badge`	The first four characters of the MIME subtype, in a colour keyed to the family: green for spreadsheets and CSV, red for PDF, cyan for JSON and notebooks, blue for images, amber for archives.
`File name`	The display name the producing screen chose.
`Size | timestamp`	Human-readable size, and the registration time in your local time zone.
`PREVIEW`	Opens the file in the right-hand panel. Section 06.
`NOTEBOOK / EXCEL / REPORT`	The *open-with* button. Appears only when the file's type maps to a screen: notebooks to Code Editor, spreadsheets and CSV to Excel, PDFs to Report Builder.
`SAVE`	Copies the managed file to any location you choose — this is the download button.
`DEL`	Deletes the file after a confirmation. Section 07.

Only the PREVIEW button opens the preview. Clicking the card body or the file name does nothing in this build — including on the empty-state screen, whose tip suggests clicking a card. Use the button.

Why some notebooks have no NOTEBOOK button. The open-with button is chosen from the file's registered MIME type, not its extension. A screen that registers a notebook as `text/plain` gets no route, so the card shows **PREVIEW / SAVE / DEL** only — as in Fig 6. The file is perfectly readable: preview it, or **SAVE** it and open it in the Code Editor yourself.

06 The preview panel

Press **PREVIEW** on any card and the right-hand panel opens with the file's name, a metadata line — size, MIME type, source screen, registration time — and the content itself.

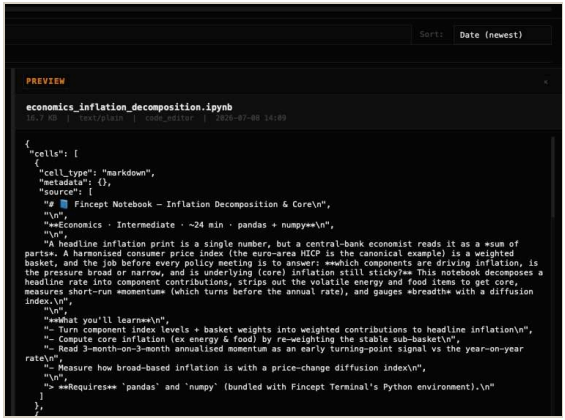


Fig 8 — Previewing a notebook. The metadata line reads *16.7 KB | text/plain | code_editor | 2026-07-08 14:09*.

File kind	How it previews
`CSV`	A real table, first row as headers — quoted fields and embedded commas handled correctly
`JSON`	Re-indented for reading
`Text, markdown, notebooks, code`	Shown as plain text
`PDF, images, audio, video, archives, xlsx`	*Binary file — preview not available.* Use **SAVE** or the open-with button

Limits worth knowing

- **Text stops at 64,000 characters**, marked *[… truncated at 64K characters …]*; CSV shows the first **200 data rows**. The files themselves are complete.
- **The preview is read-only** — to edit, open the file in Excel, Code Editor or Report Builder.
- **The panel remembers the last file** for the session. Press ✕ to close it and give the width back to the list.

> **A practical use.** Preview is the fastest sanity check on an export: open the CSV you just wrote and confirm the headers and first rows are what you expected before sending it on.

07 Selecting and deleting

There are two delete paths — one file at a time, or several at once — and both always confirm.

One file

Press **DEL** on the card. The confirmation names the file so you can be sure of what you are about to lose.

Several files

Tick the checkbox on each card you want. The bulk bar appears at the top of the list, its button counting your selection.

Fig 9 — The bulk bar with one file selected. **CLEAR SELECTION** on the right unticks everything.

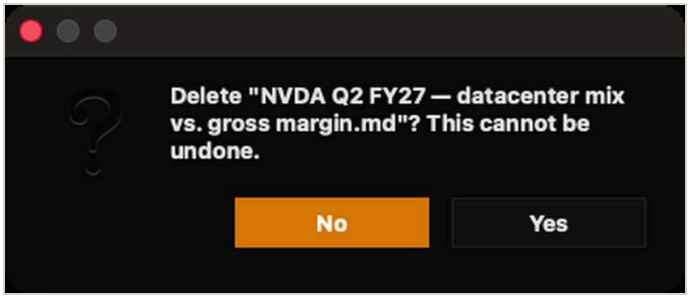


Fig 10 — The single-file confirmation names the file.

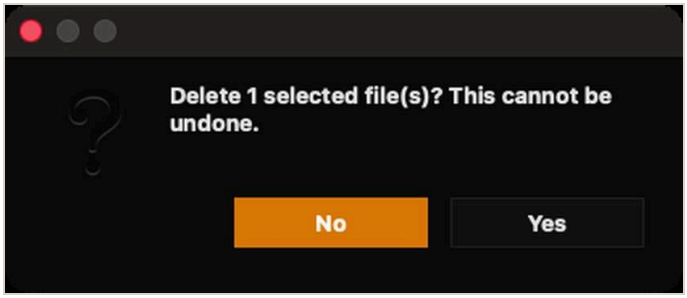


Fig 11 — The bulk confirmation counts them. Both default to **No**.

> **Deletion is permanent.** There is no trash, no undo and no recycle bin: the managed copy is removed from disk and from the index. If a file might matter later, press **SAVE** first and keep a copy somewhere of your own.

Action	Where	Confirmation
`Delete one file`	**DEL** on the card	Names the file
`Delete several`	Tick cards ▸ **DELETE N SELECTED**	Counts the files
`Clear a selection`	**CLEAR SELECTION**	None needed — nothing is deleted

> **Selections are cleared whenever the list redraws** — after a delete, a search, a sort change or a chip change. Tick and delete in one pass rather than building a selection while filtering.

08 The eleven producing screens

Files arrive here on their own. Whenever one of these screens writes something, it registers the file with the manager in the same breath — which is why the index is a complete record of your output rather than a folder you have to maintain.

Chip	Screen	What it registers
REPORT BUILDER	Report Builder	Saved reports and exported PDFs
CODE EDITOR	Code Editor	Saved Jupyter notebooks (`.ipynb`)
EXCEL	Excel	Imported workbooks, and every `.xlsx` / `.csv` export
NOTES	Notes	Notes exported as markdown
PORTFOLIO	Portfolio	Portfolio CSV and JSON exports
DATA SOURCES	Data Sources	Connector configuration JSON
ALGO TRADING	Algo Trading	Saved strategy definitions
BACKTESTING	Backtesting	Backtest result JSON exports
AI QUANT LAB	AI Quant Lab	Module result exports
NEWS	News	Saved article text
EQUITY RESEARCH	Equity Research	Financial-statement CSV exports

The hand-off, end to end

A worked example, performed live for this guide: a research note was written in the **Notes** tab and its **EXPORT** button pressed. Nothing appeared to happen in Notes — the export is silent — but the file was already here, filed under **NOTES**:

Fig 12 — The exported note in the index seconds later: badge `[MARK]` for `text/markdown`, 579 B, timestamped, under the NOTES group heading.

> **The pattern to remember.** Screens do not ask you where to put things; they register them. If a screen's export button seems to do nothing, come here and look — the file is almost always sitting under that screen's chip.

> **Round trips work too.** A CSV registered by Portfolio can be opened in Excel with the **EXCEL** button, edited, and exported again — arriving back in the index under the Excel chip as a new file. Nothing is overwritten in place.

09 Storage, quota and what is on disk

Where the files actually are

Managed files live in a `files` folder inside your terminal profile directory, alongside a small `metadata.json` index that records each file's display name, size, type, source screen and registration time. Two consequences worth knowing:

- **Stored names are not display names.** On disk each file carries a generated identifier; the friendly name you see in the list comes from the index. This is what allows two exports with the same name to coexist without clobbering each other.
- **The index and the folder travel together.** Everything is under your profile, so a second profile on the same machine has its own separate file index.

Property	Value
`Location`	`<terminal profile>/files/`
`Index`	`metadata.json` in the same folder
`Storage budget`	500 MB — displayed, not enforced
`Bar turns red at`	80% of the budget
`Sync`	**Local only.** Managed files are not uploaded to your Fincept account
`Deletion`	Removes the managed copy and its index entry, permanently

> **Managed files are local.** They are not backed up to the Fincept service and they do not follow you to another machine. For anything you cannot regenerate, press **SAVE** and keep the copy in your own backed-up location.

> **Uploads copy, never move. UPLOAD FILES** makes a managed copy and leaves your original untouched, so bringing a counterparty's CSV into the terminal cannot damage the file you were sent. The reverse is also true: deleting the managed copy here never touches the original.

10 Driving it from AI Chat

The File Manager publishes **15 agent tools** in the `file_manager` category. This is the most useful tool set on the whole tab, because it lets the assistant work with the contents of your files, not just their names — "read my latest backtest export and summarise the drawdowns".

Group	Tools	What they do
`Browse (3)`	list_files, search_files, get_file_info	Enumerate the index, search it, and inspect one entry's metadata
`Read (1)`	read_file_contents	Read a managed file's text — the tool behind “summarise this export”
`Write (3)`	write_managed_text_file, import_file_into_manager, register_external_file	Create a text file in managed storage, copy an external file in, or index a file already there
`Organise (2)`	rename_managed_file, move_managed_file	Change a file's display name, or re-tag which screen owns it
`Remove (2)`	delete_managed_file, bulk_delete_managed_files	Delete one file or several by id
`Retrieve (2)`	download_managed_file, get_managed_file_path	Copy a file out to a path, or resolve its location on disk
`Inspect (2)`	get_file_manager_stats, get_file_manager_storage_dir	Totals for the index, and the storage folder itself

> **Two things the tools can do that the tab cannot.** `rename_managed_file` changes a file's display name and `move_managed_file` re-tags its source screen — there is no button for either in the interface. If an export landed with an unhelpful name, or an upload needs to sit under a particular chip, ask the assistant.

> **The delete tools are as permanent as the buttons.** `delete_managed_file` and `bulk_delete_managed_files` remove files without a confirmation dialog, because there is no one to confirm. Be specific in what you ask for, and prefer naming files explicitly over “delete the old ones”.

A pattern that works well

Ask for a summary that ends in a file: *“read the three newest backtest exports, compare their Sharpe ratios, and write the comparison to a managed markdown file.”* That uses `list_files`, `read_file_contents` and `write_managed_text_file` in one turn, and the result appears in this tab like any other export.

11 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	**File ▸ File Manager**, or Navigate ▸ Tools ▸ File Manager. There is no command-bar alias
…find the export I just made?	It is at the top of the list under **Date (newest)**, and under its screen's chip
…see only one screen's output?	Click that screen's chip
…look inside a file?	Press **PREVIEW** on the card — clicking the card body does nothing
…get a file out to my desktop?	**SAVE**, then choose a location
…open a CSV or notebook?	The **EXCEL** or **NOTEBOOK** button when present; otherwise **SAVE** it and open it yourself
…delete several files?	Tick their checkboxes, then **DELETE N SELECTED**
…bring in a file from outside?	**UPLOAD FILES** — it copies, leaving your original alone
…rename a file?	Ask the assistant: `rename_managed_file`. The interface has no rename
…have an agent read a file?	Ask in AI Chat — it uses `read_file_contents`

Honest notes on this build

- **Card bodies are not clickable** — only the **PREVIEW** button opens the preview, even though the empty-state tip suggests otherwise.
- **Type badges abbreviate the MIME subtype** — `text/plain` shows as `[PLAI]`, `text/markdown` as `[MARK]`.
- **The open-with button follows the registered MIME type**, not the extension: a notebook registered as `text/plain` gets no **NOTEBOOK** button.
- **A source heading can appear more than once** in a date-sorted list — the list is sorted before it is grouped.
- **Uploaded files have no source tag**, so they group under a blank heading and match no chip but **All**.
- **The chip filter is restored on restart, but the chip highlight is not** — the list is correctly filtered while **All** appears selected. Click the intended chip to resynchronise.
- **Preview truncates** text at 64,000 characters and CSV at 200 rows, and the longest sort label clips in the dropdown (*Size (smallest)*).
- **The 500 MB quota is advisory** and does not block writes.
- **No rename, move or folders in the interface** — grouping is by producing screen, and renaming is agent-only.
- **Deletion is permanent** from both the buttons and the tools, and **managed files are local to the profile** — they are not synced to your account.

> Everything above was read from the shipping source and confirmed on a live index of 18 files.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Terminal Settings

A control-by-control walkthrough of every Settings sub-tab — how the shell behaves, what is stored where, and which switches change the way the rest of the terminal runs.

`>_ 7 shell · 7 profile · 14 sub-tabs · 26 credential slots`

Covers · **General, Appearance, Notifications, Keybindings** · **Logging, Usage Analytics, Developer**
Profiles · Credentials · Security · Data Sources · Python Env
Storage & Cache · Cloud Sync

Every sub-tab captured live from the shipping build.

Contents

23 SECTIONS

GETTING ORIENTED

01 What the Settings screen is 3
02 Anatomy — the rail and the two scopes 4

SHELL — HOW THE TERMINAL BEHAVES

03 General 5
04 Appearance 6
05 Notifications — the fifteen providers 7
06 Notifications — alert triggers 8
07 Keybindings 9
08 Logging — levels and the log file 10
09 Logging — shipping logs to the backend 11
10 Usage Analytics — what is recorded 12
11 Usage Analytics — control, status and erasure 13
12 Developer 14

PROFILE — YOUR ACCOUNT AND CONNECTIONS

13 Profiles 15
14 Credentials 16
15 Security 17
16 Data Sources 18
17 Python Env 19
18 Storage — disk usage and categories 20
19 Storage — files, state and the SQL console 21
20 Cloud Sync — the three tiers 22
21 Cloud Sync — per-domain control and first sync 23

REFERENCE

22 Where each setting is stored 24
23 Quick reference and honest notes 25

About this guide. Every screenshot is a live capture of the Settings screen. Labels, options, ranges and warning text are transcribed from the shipping source. Account identifiers in the captures are placeholders.

01 What the Settings screen is

ORIENTATION · 01

Settings configures the terminal itself — not one form, but fourteen sub-tabs in two scopes. **SHELL** is how the application looks and behaves on this machine. **PROFILE** is what your account is connected to: API keys, data sources, the Python environment and what syncs to the cloud.

The split is not cosmetic. Shell settings are device-local and follow the installation. Profile settings belong to the account, and a few of them — broker credentials above all — are deliberately kept out of every file the terminal writes.

What you can do here

- Set language, currency and window behaviour
- Change font, density and spacing everywhere
- Wire alerts to 15 notification providers
- Rebind any keyboard shortcut
- Tune log levels per subsystem
- Ship five log streams to the backend
- Choose what usage analytics records, or switch it off
- Erase analytics collected but not yet sent
- Hold 26 third-party API keys
- Set a PIN, auto-lock and lock-on-minimize
- Manage every saved data connection
- Repair the embedded Python environment
- See disk usage and clear cached data
- Run SQL against the local databases
- See all 24 data domains and where each one lives

> **Changes apply immediately.** Almost nothing here needs a restart — the screen writes as you change a control and the rest of the terminal re-reads it. The one exception is called out where it occurs.

02 Anatomy — the rail and the two scopes

A fixed 200-pixel navigation rail on the left, one panel on the right. The rail is headed `SETTINGS` and split by two scope headers. Clicking an entry swaps the panel; the selection is remembered, so re-opening Settings returns you where you left off.

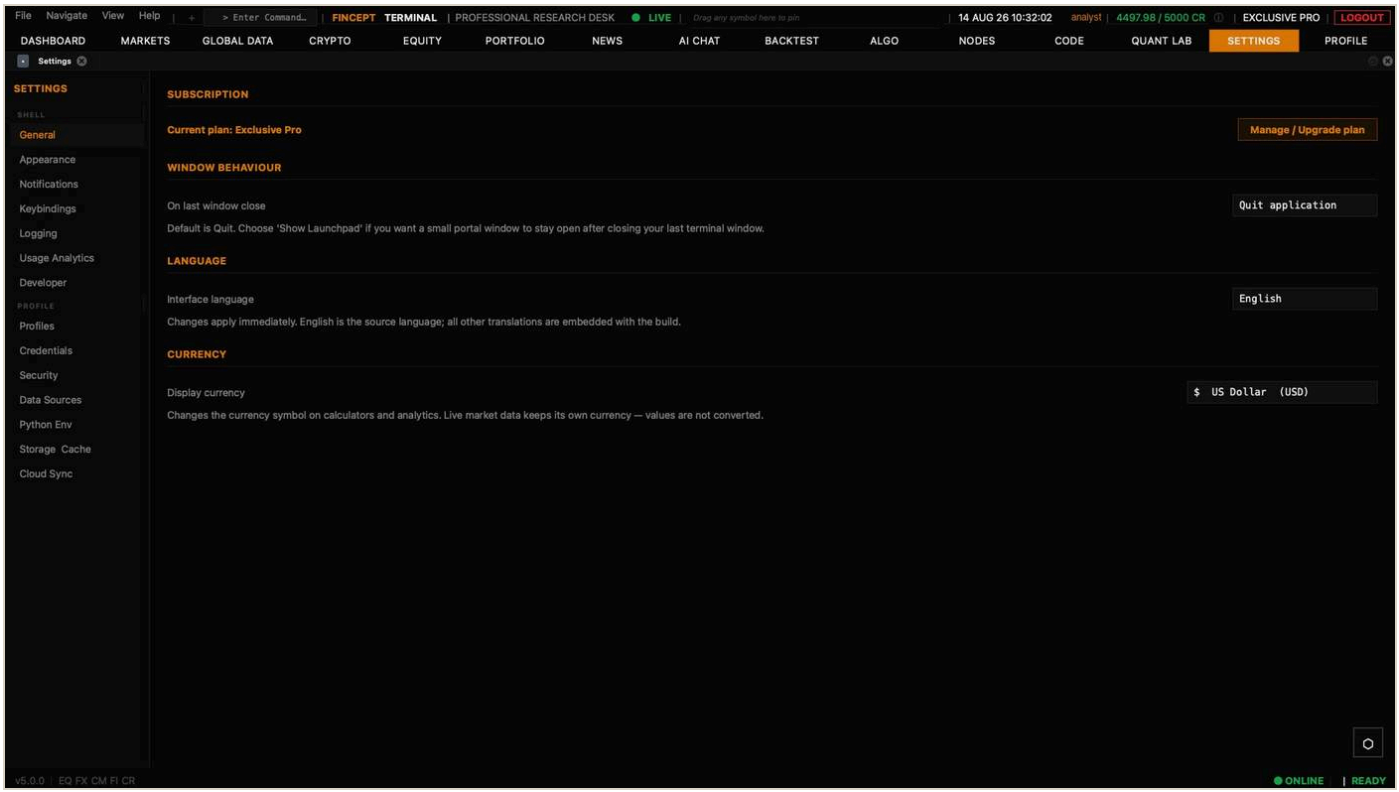


Fig 1 — The Settings screen on open. The rail lists SHELL (7 entries) then PROFILE (7); General is selected by default.

Region	What it does
`Rail header`	Reads SETTINGS. Fixed width, always visible.
`SHELL`	General, Appearance, Notifications, Keybindings, Logging, Usage Analytics, Developer.
`PROFILE`	Profiles, Credentials, Security, Data Sources, Python Env, Storage & Cache, Cloud Sync.
`Panel`	The selected sub-tab — a scrolling stack of titled cards.

> **Three sub-tabs were retired in v5, and none of them lost you a feature. Voice** went because speech now runs on the platform's own account — there is no provider, model or key left to choose; the one genuinely local switch, clap-to-open-mic, moved to the chat bubble's History ▾ menu beside the microphone it opens. **MCP Servers** went because the dedicated MCP Servers tab is a superset of it. **LLM config** went because chat, agents and teams run on the gateway-managed model.

03 General

Four cards: subscription, window behaviour, interface language and display currency.

Control	Options	What it does
`Current plan`	read-only, with **Manage / Upgrade plan**	The tier the account resolves to; the button opens the billing flow.
`On last window close`	Quit application *(default)* · Show Launchpad	Whether closing the final window exits, or leaves a small portal window open.
`Interface language`	every shipped translation, in its own native name	Applies immediately — the UI is rebuilt in place. English is the source language; the rest are embedded with the build.
`Display currency`	symbol, name and ISO code per entry	Changes the currency symbol on calculators and analytics.

Display currency does not convert anything. It changes the symbol shown on calculators and analytics. Live market data keeps its own currency and values are *not* converted — a US quote stays in dollars whatever you pick here.

Language names are deliberately left untranslated. Each language is always listed in its own script, so you can find your way back if the current UI language is unfamiliar.

04 Appearance

Three cards — TYPOGRAPHY, THEME, INTERFACE. Every change re-renders the running screens immediately; you do not need to revisit tabs you already had open.

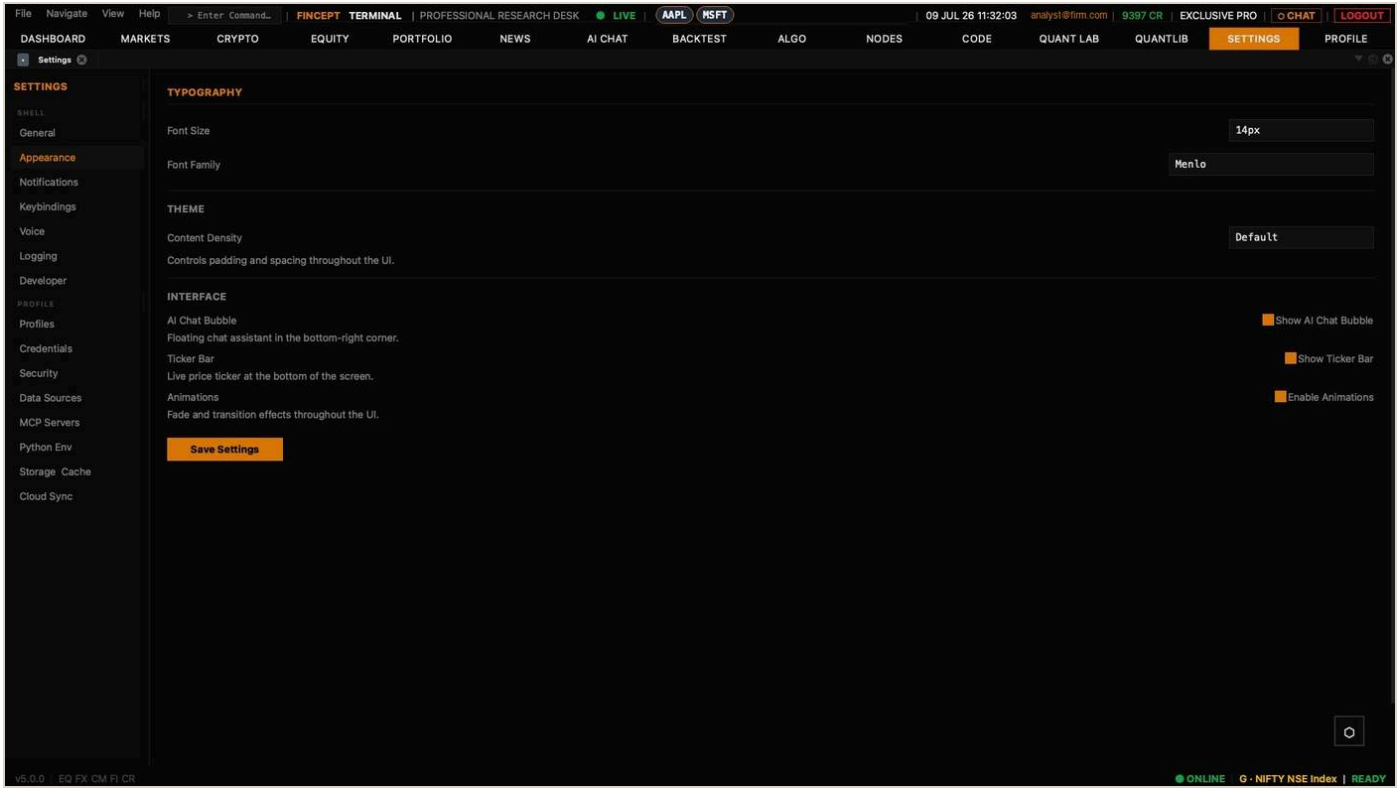


Fig 2 — Appearance. Each card is a stack of labelled rows with the control on the right.

Control	Range / options	Notes
`Font Size`	8px – 24px	Applies to the entire interface, not just this screen.
`Font Family`	installed families	Private and emoji families are filtered out — see below.
`Content Density`	Compact · Default · Comfortable	Controls padding and spacing throughout the UI.
`Show AI Chat Bubble`	on / off	The floating chat assistant in the bottom-right corner.
`Show Ticker Bar`	on / off	The live price ticker at the bottom of the screen.
`Enable Animations`	on / off	Fade and transition effects throughout the UI.

> **Why the font list is filtered.** On macOS the system font database includes hidden families whose digit glyphs are emoji keycap bases. Picking one renders every number in the terminal grossly letter-spaced, so dot-prefixed and emoji families are excluded from the picker on purpose.

05 Notifications — the fifteen providers

The upper half of the sub-tab. Each provider is a collapsible row carrying its own credential fields, an **ENABLED** switch — *turn this provider on/off, applies immediately* — a **How to set this up** link and a **Test Send** button.

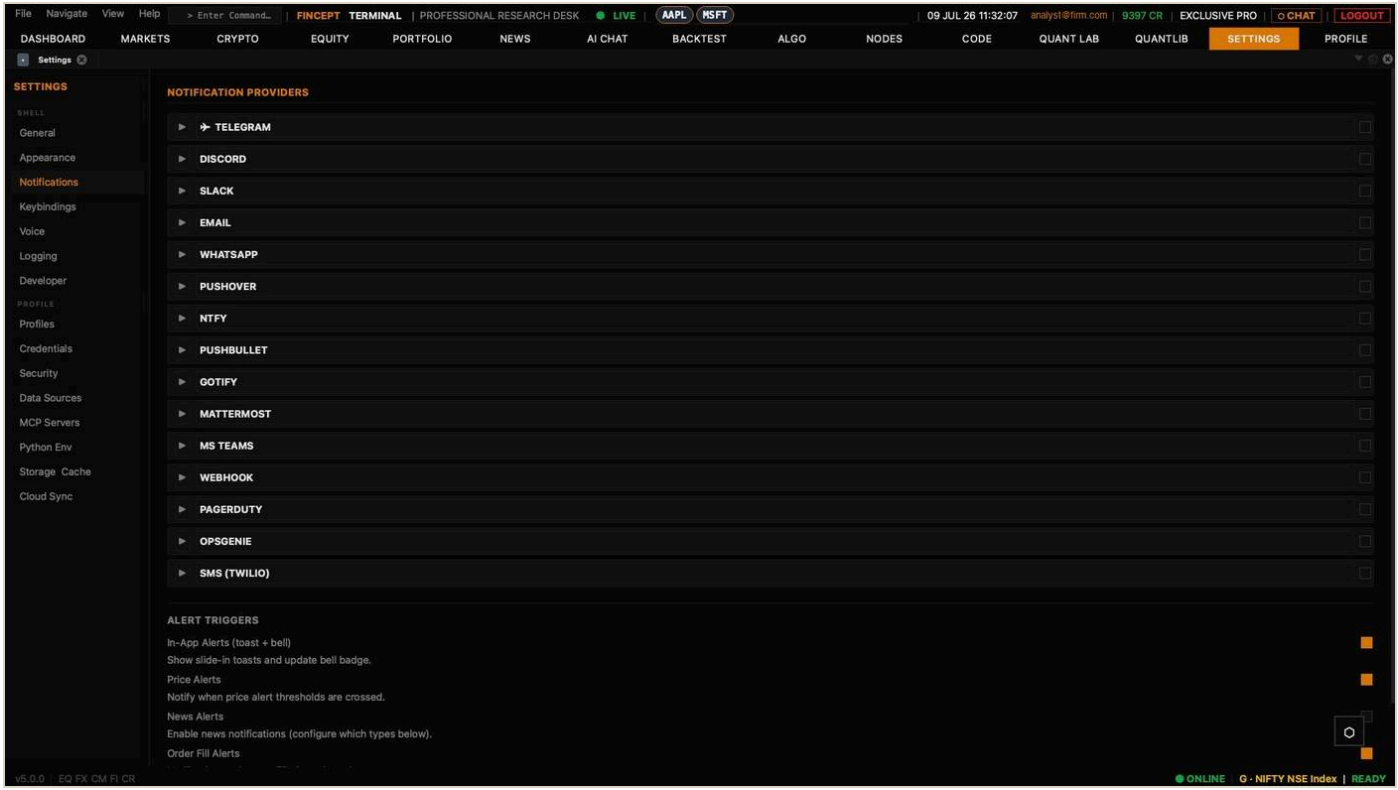


Fig 3 — Notifications. Providers expand to reveal only the fields that provider needs.

Provider	What it asks for
`Telegram`	Bot Token (from @BotFather) and Chat ID
`Discord`	Webhook URL
`Slack`	Webhook URL, optional Channel
`Email`	SMTP host, port, username, password, To and optional From
`WhatsApp · SMS`	Twilio Account SID, Auth Token, From and To numbers
`Pushover · Pushbullet · Gotify`	Provider token / key
`ntfy`	Topic and optional server
`Mattermost · MS Teams · Webhook`	Webhook URL (Webhook also takes a method)
`PagerDuty · Opsgenie`	Integration / API key

> **Telegram is one click.** Paste the bot token, press **Detect Chat ID**, then send your bot any message — the terminal fills the chat ID in and fires a test itself. **Test Send** delivers a real message titled *Fincept Test*, so you can prove the wiring before you rely on it. Press **Save All Providers** when you are done.

06 Notifications — alert triggers

The lower half. Providers decide *where* a notification goes; triggers decide *whether* there is one. Each is an independent switch.

Trigger	Fires when
`In-App Alerts (toast + bell)`	Always available — slide-in toasts plus the bell badge.
`Price Alerts`	A price-alert threshold you set is crossed.
`News Alerts`	Master switch for the four news triggers below it.
`Breaking News`	A FLASH / BREAKING / URGENT priority cluster appears.
`Monitor Keyword Matches`	A news monitor watch list gets new matches.
`Category Volume Spikes`	A category has abnormally high article volume (z-score ≥ 3).
`FLASH + High-Impact Articles`	An individual article is both FLASH priority and high market impact.
`Order Fill Alerts`	An order is filled or rejected.

> **News Alerts is a gate.** Turning it off silences Breaking News, Monitor Keyword Matches, Category Volume Spikes and FLASH + High-Impact together, whatever their own switches say.

> In-app alerts are independent of providers — you will still see toasts and the bell badge with no provider configured at all. If you want alerts to reach you when the terminal is closed, you need at least one provider enabled and tested.

07 Keybindings

Every bindable action in one searchable list with its current shortcut alongside. Filter with **Search actions...**, then press **Rebind** on the row you want.

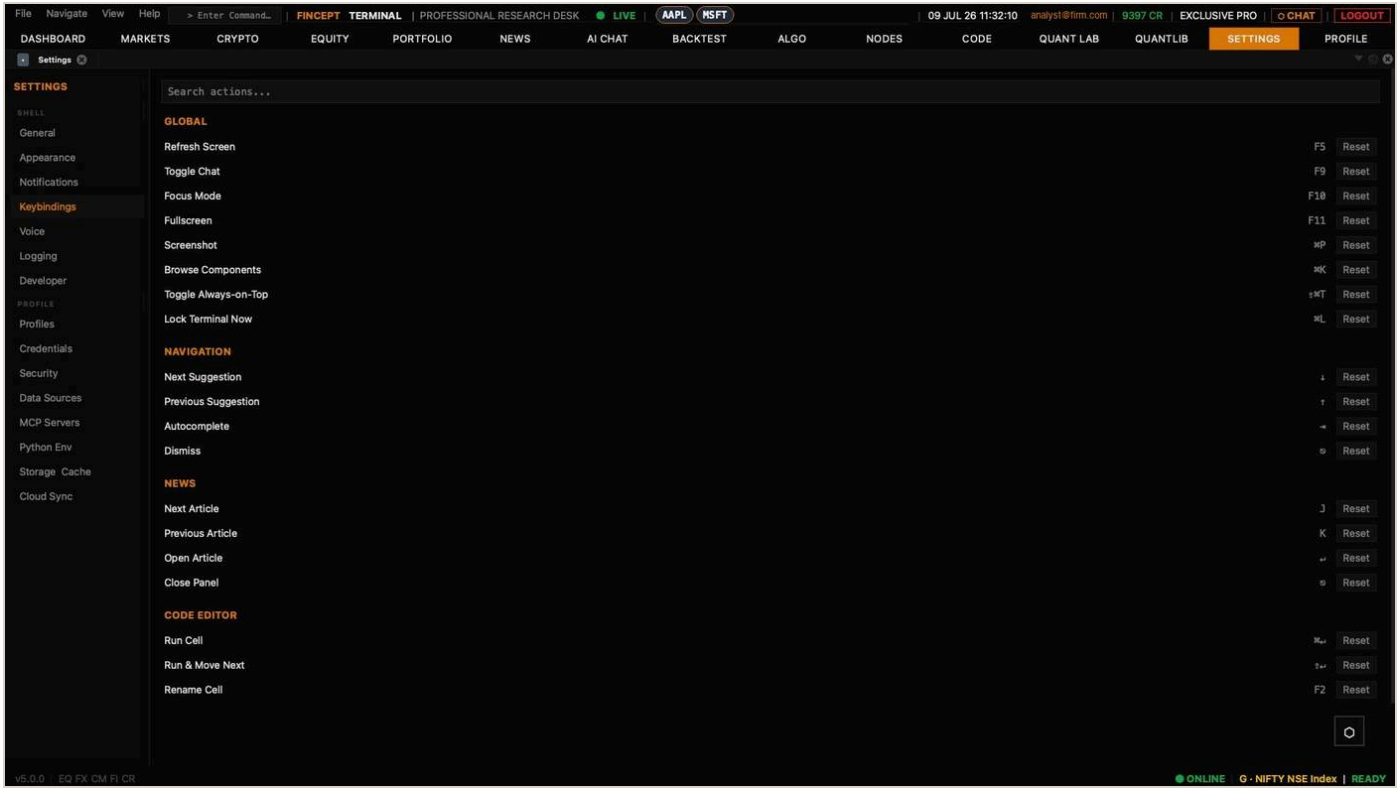


Fig 4 — Keybindings. One row per action; the current combination sits on the right with a Reset beside it.

Control	What it does
`Search actions...`	Filters the list as you type.
`Rebind`	Opens a capture dialog headed *Rebind: <action>*, showing *Current: <keys>* and *Press new key combination...*
`Apply / Cancel`	Commit or abandon the captured combination.
`Reset`	Returns that one action to its shipped default.
`Reset All to Defaults`	Returns every binding to its shipped default.

> **Conflicts are surfaced, not blocked.** If the combination you press is already bound, the dialog warns *Warning: already used by ...* and names the other action. You can still apply it — the warning is there so the collision is your choice, not a surprise.

08 Logging — levels and the log file

Six levels throughout: `Trace`, `Debug`, `Info`, `Warn`, `Error`, `Fatal`.

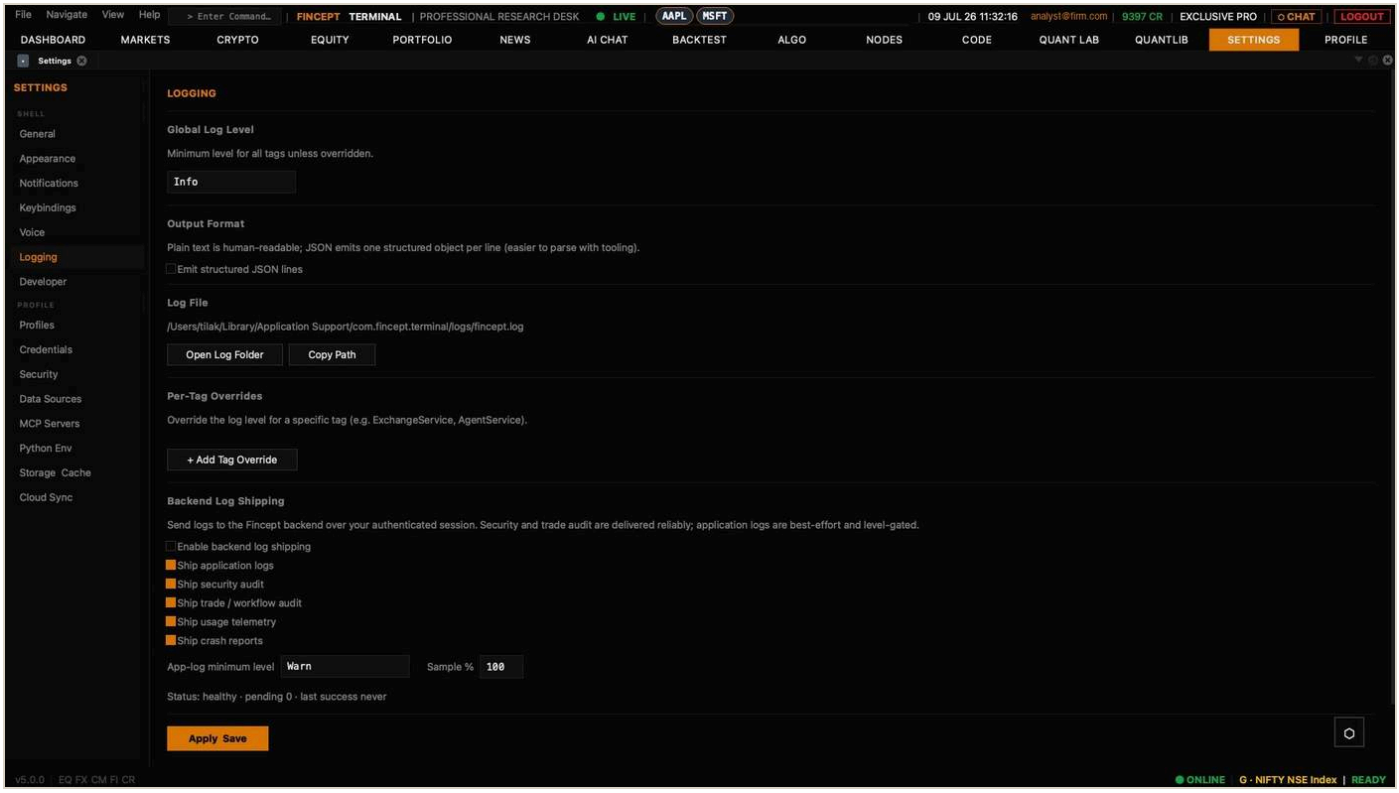


Fig 5 — Logging. Global level and format at the top, per-tag overrides in the middle, backend shipping at the foot.

Control	What it does
`Global Log Level`	Minimum level for all tags unless overridden.
`Output Format`	**Emit structured JSON lines** instead of plain text — use this when something else will parse the log.
`Log File`	Shows the active path, with **Open Log Folder** and **Copy Path** beside it.
`Per-Tag Overrides`	Override the level for one tag — e.g. `ExchangeService`, `AgentService`. **+ Add Tag Override** adds a row, **Remove** deletes one.

> **Per-tag overrides are the tool for a noisy subsystem.** Rather than dropping the whole terminal to Debug, name the one tag you are investigating and leave everything else at Info — the log stays readable.

09 Logging — shipping logs to the backend

The second half of the sub-tab sends logs to Fincept over your authenticated session. It is off until you enable it, and each stream is opted into separately.

Control	What it does
`Enable backend log shipping`	Master switch. Nothing leaves the machine until this is on.
`Ship application logs`	The general application log.
`Ship security audit`	Security-relevant events.
`Ship trade / workflow audit`	Trade and workflow actions.
`Ship usage telemetry`	Feature-usage telemetry.
`Ship crash reports`	Crash diagnostics.
`App-log minimum level`	A floor level for the application stream, independent of the global level.
`Sample %`	Ship a percentage of eligible records rather than all of them.

Reading the status line

The card carries a live line: `Status: healthy · pending 0 · last success 12:04`. **healthy** or **degraded** reports the last attempt, **pending** counts records waiting, and **last success** reads *never* until the first batch lands.

> **Apply & Save** commits this page. A *degraded* status with a rising pending count means the backend is unreachable, not that logging is misconfigured — records queue rather than being dropped.

> This is the **diagnostic** plane — crashes, errors and audit trails. Product usage capture is a separate switch with its own consent surface, two pages on.

10 Usage Analytics — what is recorded

New in v5, and the only sub-tab that exists to tell you what the terminal does rather than to change it. It states its own purpose: *records how the terminal is used so we can find the parts that are hard to reach and the features nobody can find.*

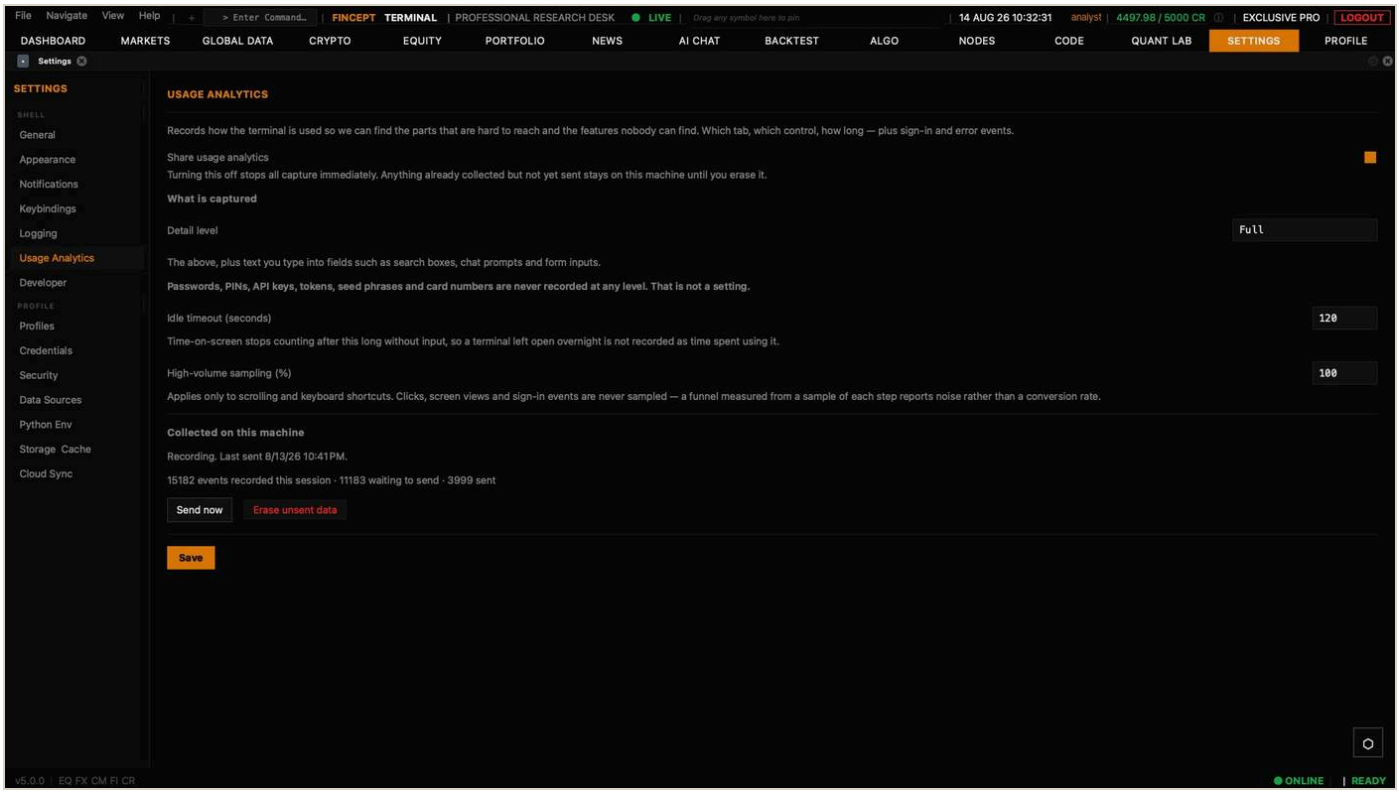


Fig 6 — Usage Analytics. The consent switch, the detail level with its plain-English description, then what has been collected on this machine.

Detail level has four settings, described on screen as you select them. Each includes the ones above it.

Detail level	What that records
`Off`	*Nothing is recorded.*
`Structural`	*Which screen and which control, how long you spent, and whether an action succeeded. No content of any kind.*
`Content`	*The above, plus short selections such as symbols, data providers and chart timeframes. Free text is not recorded.*
`Full`	*The above, plus text you type into fields such as search boxes, chat prompts and form inputs.*

> **The shipped default is on, at Full** — which includes text you type into search boxes, chat prompts and form inputs. *Content* and *Structural* are the two steps down; the master switch is the step off.

> **Credentials are never in scope, at any level.** The screen says so in bold: *passwords, PINs, API keys, tokens, seed phrases and card numbers are never recorded at any level. That is not a setting*. The exclusion is structural — masked and credential-named fields are dropped before the detail level is consulted.

11 Usage Analytics — control, status and erasure

The rest of the sub-tab is the switch, two tuning fields, and a live account of what is sitting on this machine right now.

Control	Range	What it does
`Share usage analytics`	on / off	*Turning this off stops all capture immediately. Anything already collected but not yet sent stays on this machine until you erase it.*
`Idle timeout (seconds)`	15 – 3600, default 120	*Time-on-screen stops counting after this long without input, so a terminal left open overnight is not recorded as time spent using it.*
`High-volume sampling (%)`	0 – 100, default 100	Applies *only* to scrolling and keyboard shortcuts.

> **Clicks, screen views and sign-in events are never sampled**, whatever you set the percentage to. The screen gives the reason: *a funnel measured from a sample of each step reports noise rather than a conversion rate*. Sampling exists to thin out scroll and shortcut spam, nothing else.

Collected on this machine

A status line that names the states which would otherwise look identical to healthy, followed by counters — `15182 events recorded this session · 11183 waiting to send · 3999 sent`, plus refusals, drops and the last error when there are any.

Status reads	Meaning
`Off — nothing is being recorded.`	The master switch is off.
`Recording. Not signed in…`	Events are held on this machine until you sign in — nothing has been sent.
`Recording. The last send failed (retrying in Ns).`	Reachability, not consent. It backs off and retries.
`Recording. Last sent <time>.`	Healthy.
`Recording. Nothing sent yet.`	Enabled, signed in, first batch not yet away.

Send now flushes immediately, **Erase unsent data** destroys everything still held locally, and **Save** commits the settings above.

> **Erase only reaches what has not left yet.** It purges the local spool; records already accepted by the server are not recalled by it. Erase before you sign in, or straight after switching the master switch off, if that is what you intend.

12 Developer

The smallest sub-tab, and the only one whose switches are labelled experimental.

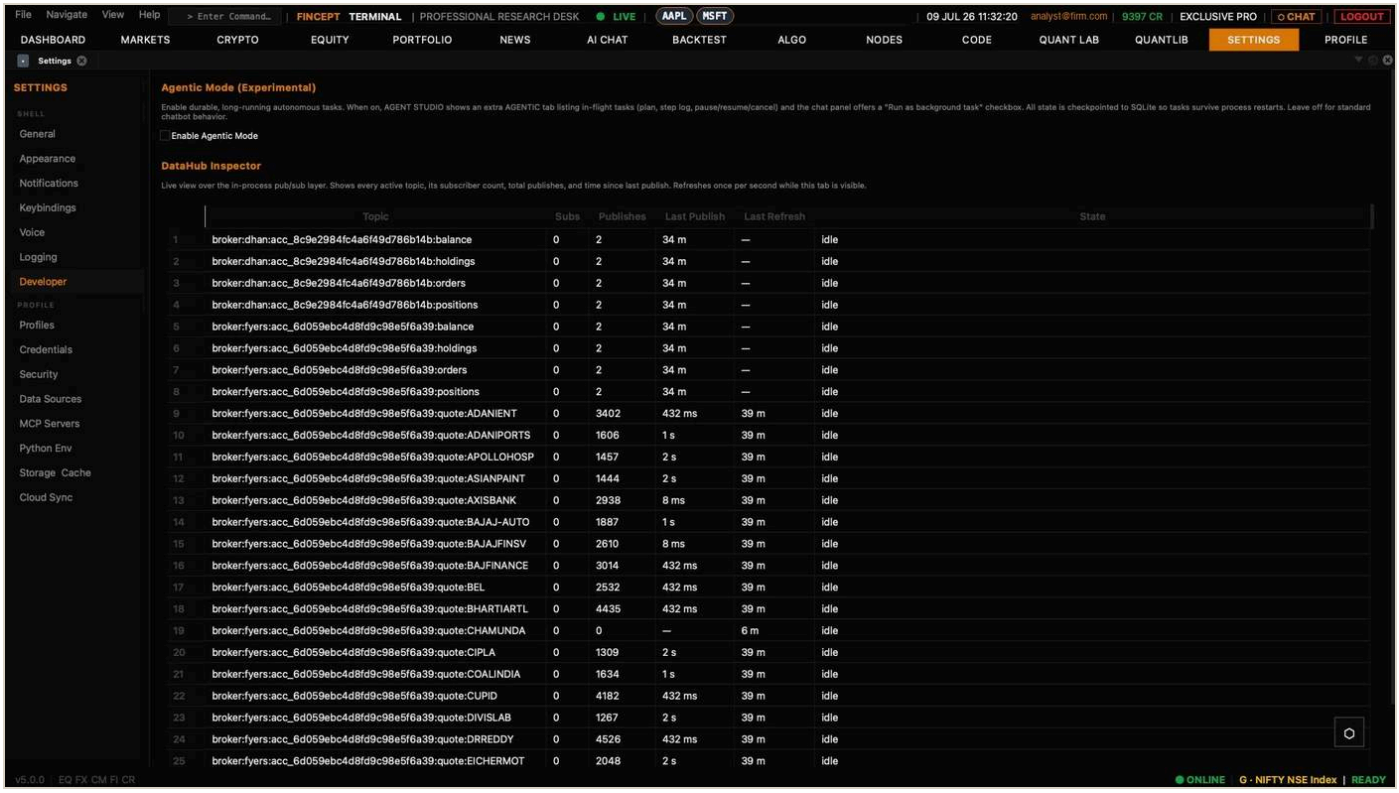


Fig 7 — Developer. Two entries: the Agentic Mode switch and the DataHub Inspector.

Control	What it does
`Agentic Mode (Experimental)`	**Enable Agentic Mode** turns on durable, long-running autonomous tasks. AGENT STUDIO gains an extra **AGENTIC** tab listing in-flight tasks with their plan, step log and pause / resume / cancel controls, and the chat panel offers a **Run as background task** checkbox.
`DataHub Inspector`	A live view over the in-process pub/sub layer: every active topic, its subscriber count, total publishes, and time since last publish. Refreshes once per second while the tab is visible.

> **Agentic tasks survive a restart.** All state is checkpointed to SQLite, so a long-running task is still there — and still advancing — after the process comes back. Leave the switch off for standard chatbot behaviour.

> **The Inspector is the fastest triage for an empty panel.** If the topic is there and publishing, the data is arriving and the bug is in the rendering; if publishes are frozen or the topic is missing, stop looking at the widget.

13 Profiles

A profile is a separate workspace inside one installation — its own layouts, preferences and local state. The card names the active profile in bold, lists them all, and creates new ones.

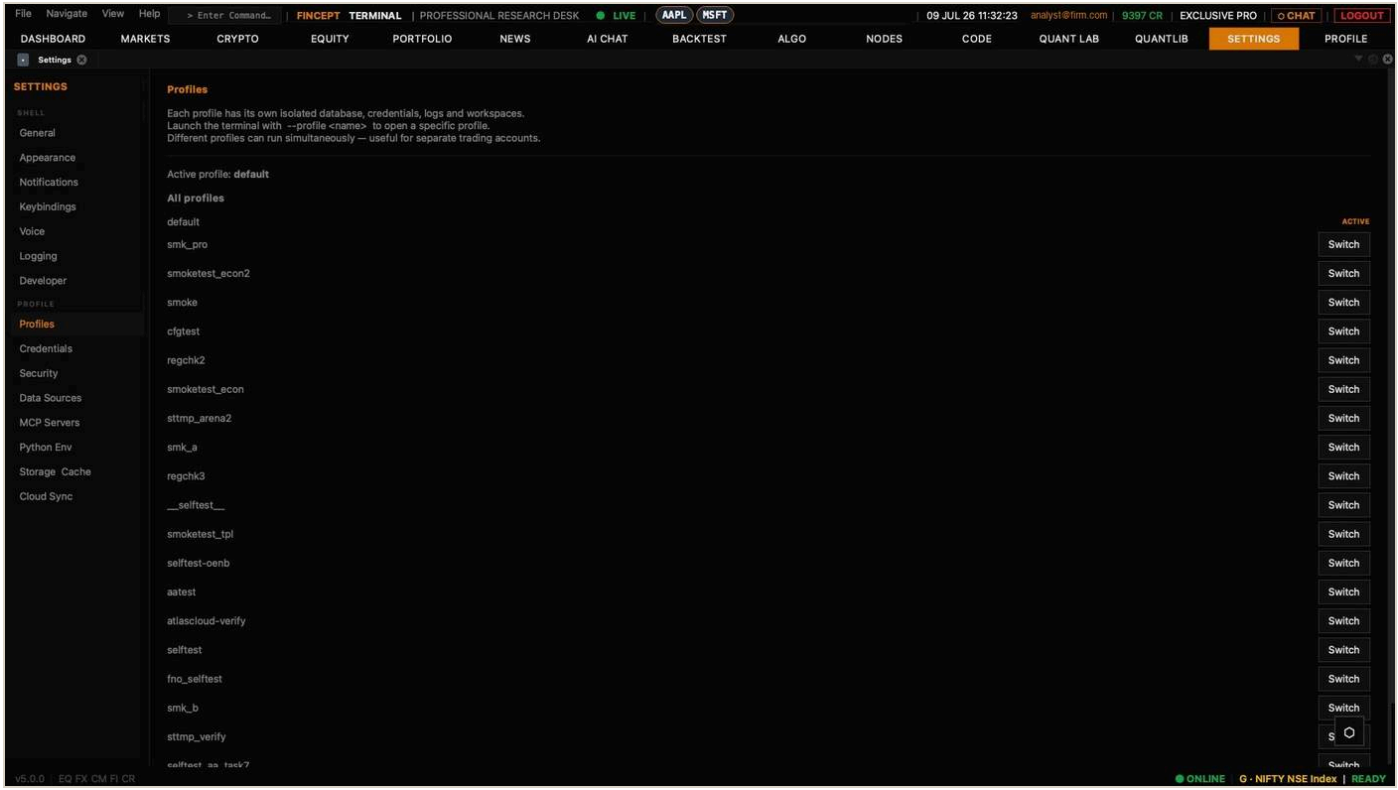


Fig 8 — Profiles. The active profile is badged ACTIVE; every other row carries a Switch button.

Control	What it does
`Active profile`	The profile currently in use, shown in bold.
`All profiles`	Every profile on this machine — the active one badged ACTIVE, the rest offering **Switch**.
`Create new profile`	Name it (alphanumeric, hyphen and underscore only), then **Create & Switch**.

> Profiles partition this installation's state. They are not separate accounts and they do not change who you are signed in as — switching profile keeps your login and swaps the workspace around it.

14 Credentials

Twenty-six slots for the third-party API keys the data tabs and embedded Python use. Each row is a labelled field with its own **Save**; an unset row reads *Not set*, and a stored one reads `••••••••  (saved)` rather than echoing the value back.

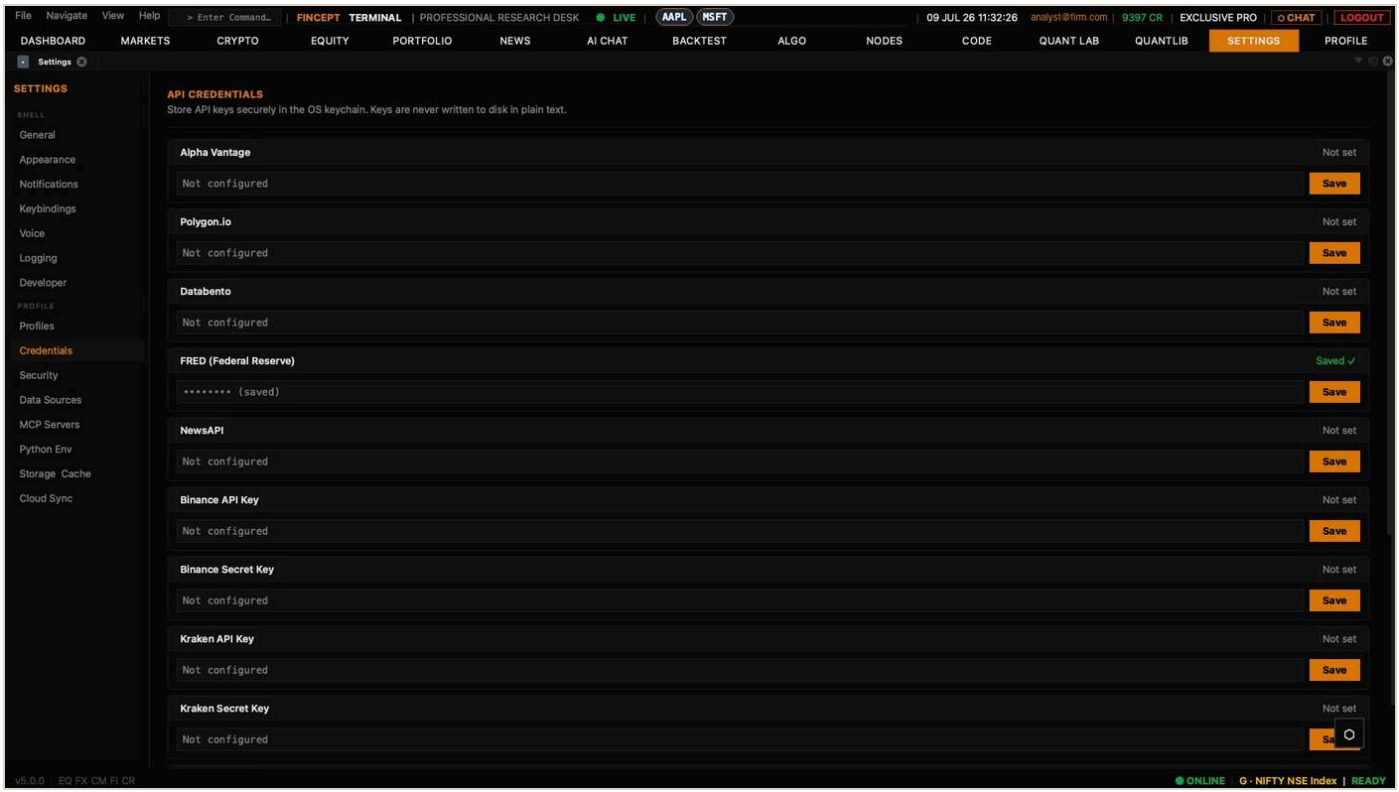


Fig 9 — Credentials. One row per provider, saved values masked.

Group	Keys
`Market data (8)`	Alpha Vantage · Polygon.io · Databento · IEX Cloud Token · Finnhub · Tiingo · Quandl · Nasdaq Data Link (Quandl)
`Economics & government (9)`	FRED (Federal Reserve) · BEA · BLS · US Census Bureau · EIA · UN Comtrade · WTO · Trading Economics · Congress.gov
`News (1)`	NewsAPI
`Crypto venues (4)`	Binance — API key and secret · Kraken — API key and secret
`Prediction markets (4)`	Polymarket — API key, secret, passphrase and wallet address

> **Broker credentials are not in this list.** Broker API keys live in the operating system keychain and are entered from the trading screens' ACCOUNTS dialog — by design, never in a file the terminal writes and never synced.

> A failed save says so on the row rather than silently dropping the value. A persistent failure normally means the secrets service is unreachable, not that the key was wrong. Keys hydrate asynchronously after login, so a panel opened very early can show *Not set* for a moment before the saved values land.

15 Security

A six-digit PIN guards the terminal; an idle timer and a minimize hook can re-lock it.

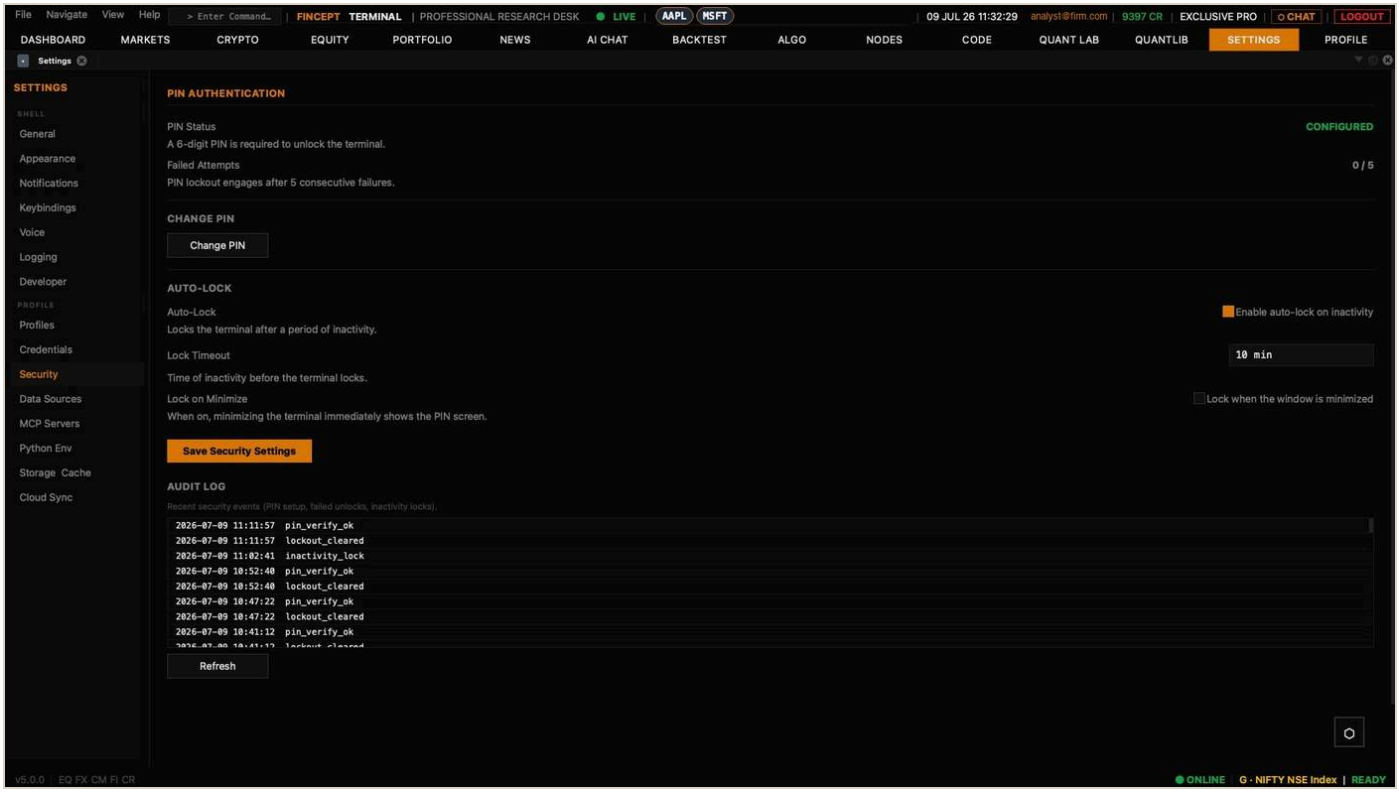


Fig 10 — Security. PIN status and failed attempts, the change form, then auto-lock.

Control	Behaviour
`PIN Status`	Whether a PIN is set. *A 6-digit PIN is required to unlock the terminal.*
`Failed Attempts`	Running count. *PIN lockout engages after 5 consecutive failures.*
`Change PIN`	Current PIN, New PIN, Confirm PIN, then **Update PIN**. A mismatch is rejected before anything is written.
`Auto-Lock`	*Locks the terminal after a period of inactivity.*
`Lock Timeout`	1 · 2 · 5 · 10 · 15 · 30 · 60 min (default 10). Enabled only while Auto-Lock is on.
`Lock on Minimize`	When on, minimizing the terminal immediately shows the PIN screen.

> **Lockout is enforced, not advisory.** After five wrong PINs the screen refuses further attempts and counts down — *Too many failed attempts — locked for N s*. Waiting is the only way through; there is no override in the UI.

> **Lock on Minimize** is the switch worth knowing about on a shared desk: it does not wait for the idle timer, so hiding the window locks it at once.

16 Data Sources

A compact console over every saved connection — four counters, a table and bulk actions. It is for *managing* connections; creating them happens in the full tab.

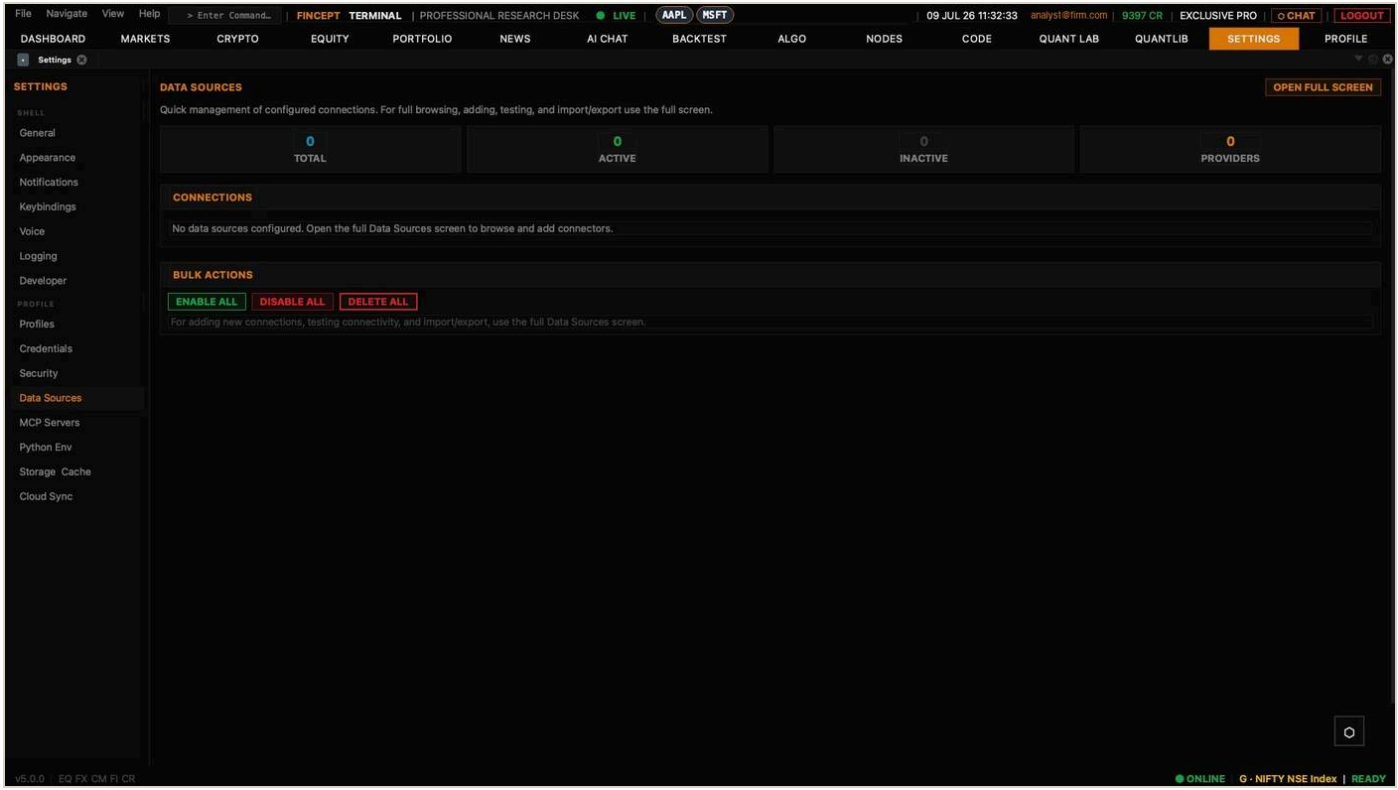


Fig 11 — Data Sources. Counters across the top, the connections table below, bulk actions at the foot.

Control	What it does
`TOTAL · ACTIVE · INACTIVE · PROVIDERS`	Live counts across every saved connection.
`CONNECTIONS table`	SOURCE, TYPE and CATEGORY per row, with an on/off toggle and a **DEL** button.
`ENABLE ALL / DISABLE ALL`	Flip every connection at once. Disabling confirms first.
`DELETE ALL`	Removes every saved connection, after a confirmation.
`OPEN FULL SCREEN`	Hands off to the Data Sources tab — where new connections are added, connectivity is tested, and import/export lives.

> **Disable is reversible, delete is not.** A disabled connection keeps its credentials and mapping and simply stops being queried; a deleted one has to be built again.

17 Python Env

The analytics tabs run on embedded Python — two environments, a trading one and an analytics one. This sub-tab inspects both, reports what is missing, and repairs them.

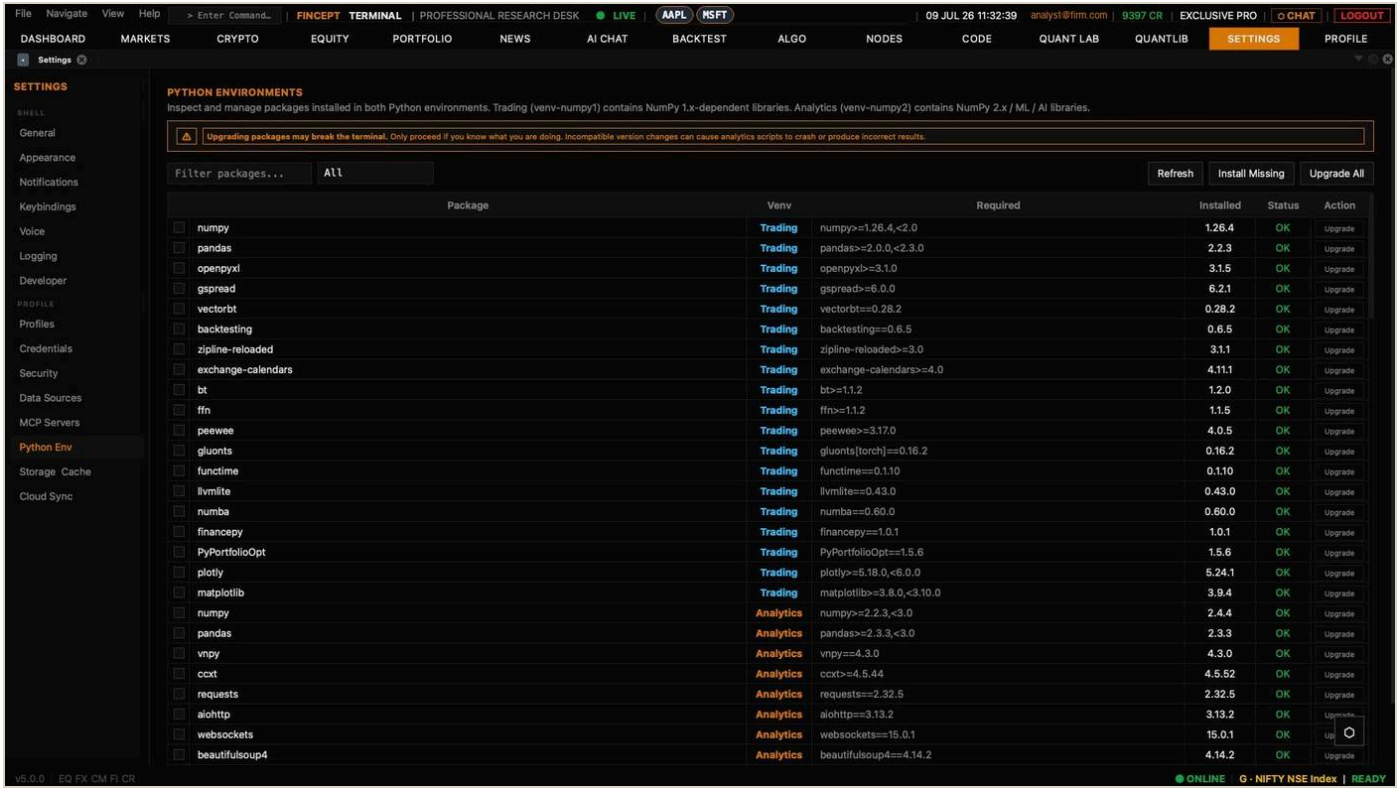


Fig 12 — Python Env. Required against installed versions, with a status and an action per row.

Control	What it does
`Filter packages... / All · Trading · Analytics`	Free-text filter plus a tab per environment.
`Package table`	Package, Venv, Required, Installed, Status, Action. Status reads **OK** or **MISSING**.
`Install Missing`	Installs everything currently marked MISSING.
`Upgrade All`	Brings every package to its required version.
`Install / Upgrade Selected`	Acts only on the rows you ticked.
`Refresh`	Re-reads both environments.

> **Upgrading packages may break the terminal.** That warning is on the screen and it is meant literally — the analytics code is pinned to the required versions. Prefer **Install Missing**. If the panel reads *Python environment not set up — run Setup first*, every Python-backed analytic fails until setup is run: start there rather than debugging the tab downstream.

18 Storage — disk usage and categories

Where the disk went, and how to get it back. **Refresh** re-measures on demand — the figures are read when the panel opens, not continuously.

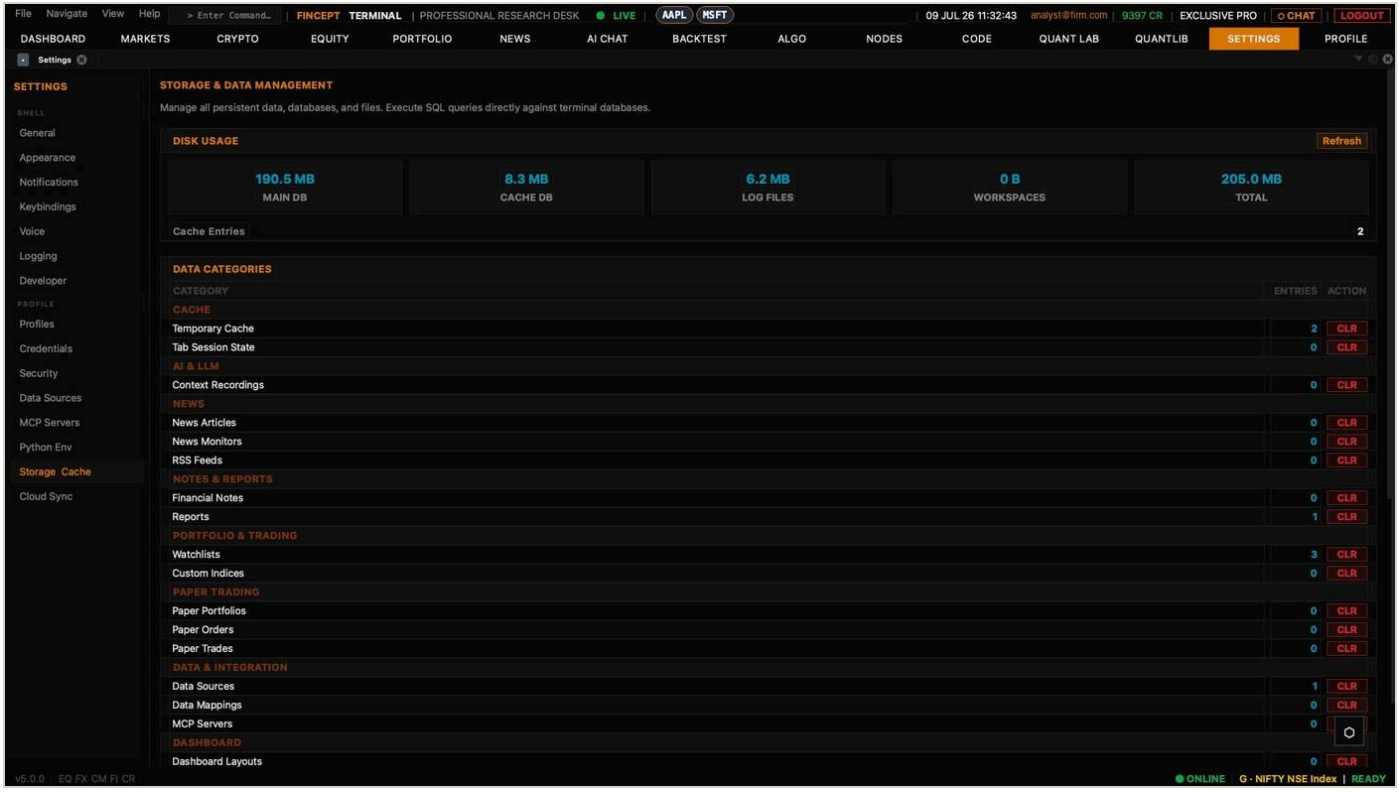


Fig 13 — Storage & Cache. DISK USAGE at the top, per-category counts with a CLR button on each row.

DISK USAGE	Reports
`MAIN DB`	The primary database file.
`CACHE DB`	The cache database.
`LOG FILES`	Everything the logger has written.
`WORKSPACES`	Saved workspace files.
`TOTAL`	The sum of the above.
`Cache Entries`	A live count of rows held in cache.

DATA CATEGORIES lists each category with its entry count and a **CLR** button, grouped under headers. Clearing one asks first, naming the category.

> **Clearing cannot be undone.** The confirmation reads *Permanently delete all <category>? This cannot be undone.* and it is the only guard. Clearing cache is safe — it will be rebuilt; clearing a data category removes the data itself.

19 Storage — files, state and the SQL console

The lower half of the sub-tab: the files on disk, the window state, and a direct SQL console over the terminal's own databases.

Store	Action	Effect
`Log Files`	Clear Logs	*Clear all application log files? Current log data will be lost.*
`Workspace Files (.fwsp)`	Delete Workspaces	*Delete all saved workspace files? This cannot be undone.*
`Window & UI State`	Reset UI State	Resets window positions, dock layouts and perspectives. **Takes effect on next restart.**
`Registry`	—	Listed with its size for reference.

SQL console

Executes SQL directly against the terminal databases. Type a query and press **EXEC**; **Quick:** offers ready-made queries. Results report as *OK — N row(s) returned*, and a read is capped at **100 rows** (*limited to 100*).

> **Writes are confirmed, then permanent.** A write query prompts *This will modify the database: … Continue?* and answers *OK — N row(s) affected*. There is no transaction to roll back from the console — treat it as a maintenance tool, not a scratchpad.

> **Reset UI State is the one control here that needs a restart.** Everything else in Settings applies immediately; window geometry is re-read at launch.

20 Cloud Sync — the three tiers

Rebuilt in v5. No longer a flat list of things to tick — an inventory of your data sorted by *where it actually lives*. Twenty-four domains in three groups, and the group a domain sits in tells you what you can and cannot do to it.

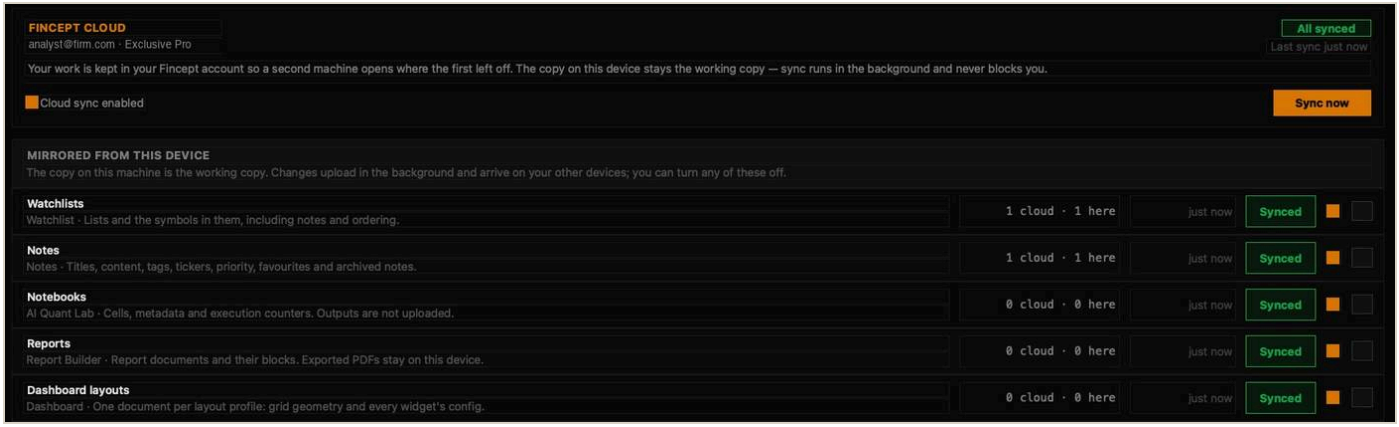


Fig 14 — Cloud Sync. The FINCEPT CLOUD header with the overall state, then MIRRORED FROM THIS DEVICE — the only tier with switches.

Tier	N	What the screen says
`MIRRORED FROM THIS DEVICE`	9	*The copy on this machine is the working copy… you can turn any of these off.*
`MANAGED BY YOUR ACCOUNT`	11	*Stored in your account and read live — no local copy to fall behind, nothing here to switch off.*
`NEVER LEAVES THIS DEVICE`	4	*Excluded by design. These stay on this machine even with sync on.*

The nine mirrored domains — the only ones with a switch	Caveat
`Watchlists · Notes · Dashboard layouts`	Uploaded whole.
`Notebooks`	Cells and metadata; **outputs are not uploaded**.
`Reports`	Documents and blocks; exported PDFs stay here.
`News monitors · RSS feeds`	Your own only — shipped feeds are not uploaded.
`Saved scans`	Conditions and universe; run history stays local.
`Custom indices`	Constituents, weights and method; values recalculated.

> The header carries the account, an overall pill (**All synced**), a last-sync line, the **Cloud sync enabled** master switch and **Sync now**. Turning the master switch off stops the mirroring; it does not reach the managed tier, which is where that data already lives.

21 Cloud Sync — per-domain control and first sync

Fig 15 — The lower two tiers. Managed rows read live and carry no switch; device-only rows are badged This device.

Tier	Domains
`Managed by your account (11)`	Preferences · Portfolios · AI chat history · Agent run history · Agent configs · Workflows · Data sources · Paper trading · Broker accounts · Algo strategies · Files
`Never leaves this device (4)`	Broker credentials · MCP server registry · Window and layout state · PIN and encryption key

Every row shows a count — `1 cloud · 1 here` for mirrored, `9 in your account` for managed, or the settings key it lives under — a relative time and a status pill. Mirrored rows add a tick and a ⋯ menu offering **Sync this now**, **Upload this device's copy**, **Replace this device's copy from cloud…** and, only when something is queued, **Discard N queued changes…**

Pill	Means
`Synced · Syncing · N queued`	Healthy: up to date, working, or waiting to upload.
`Retrying · Offline`	Transport. Changes queue here and go up when the connection returns.
`Paused · Plan limit`	Stopped deliberately — out of credits, or a plan cap. *Your data is safe on this device.*
`Off · Live · This device`	You unticked it; or a managed / device-only row with nothing to sync.

> **Replace from cloud is destructive and says so first:** *anything in this domain that exists only on this device will be removed*. Enabling a domain with data on both sides offers the same choice as a dialog — **Upload & merge** (recommended, keeps everything here) or **Use cloud copy**, which backs up the local database first.

22 Where each setting is stored

Three storage tiers sit behind this screen, and Cloud Sync now names all three on one page. Knowing which one a setting uses tells you whether it follows the machine, the account, or neither.

Tier	What lives there	Follows
`Device-local`	Window and layout state, the active profile, the PIN and its encryption key, the MCP server registry, the instrument cache.	The installation.
`OS keychain`	Broker API credentials, by design.	The machine's keychain.
`Account (server)`	Preferences, portfolios, chat history, agent configs, workflows, data sources, files — plus any mirrored domain you have ticked.	Your login.

What this means in practice

- **Signing in on a new machine** brings the account tier with you. Shell settings start at their defaults again — font, density, log levels and keybindings are per-installation, unless *Preferences* is mirroring, which it does by default.
- **Broker credentials never travel.** They are re-entered per machine on purpose; only account *metadata* leaves the device.
- **The MCP server registry is excluded on purpose**, and the screen gives the reason: server entries carry pasted keys and tokens.
- **An unticked mirrored domain stays local.** The tier decides where it *can* live; the tick decides whether it does. A managed domain has no tick because there is no local copy to keep back.

> The Settings screen is API-backed: preference changes are written through to the account rather than a local database, which is why they survive a reinstall.

23 Quick reference and honest notes

I want to…	Go to
`Make everything bigger`	Appearance › Font Size, or Content Density › Comfortable
`Get alerted on my phone`	Notifications › Telegram › Detect Chat ID › Save All Providers
`Stop news notifications entirely`	Notifications › turn off News Alerts
`Change a shortcut`	Keybindings › search the action › Rebind
`Debug one noisy subsystem`	Logging › Per-Tag Overrides › + Add Tag Override
`Share less usage data`	Usage Analytics › Detail level › Structural
`Stop usage capture completely`	Usage Analytics › untick Share usage analytics › Erase unsent data
`Add a data provider key`	Credentials › the provider's row › Save
`Add a broker key`	*Not here* — the trading screens' ACCOUNTS dialog
`Lock up when I walk away`	Security › Auto-Lock, or Lock on Minimize
`Fix a dead analytics panel`	Python Env › Install Missing
`Reclaim disk`	Storage › the category's CLR
`Reset a broken window layout`	Storage › Reset UI State, then restart
`Work across two machines`	Cloud Sync › Cloud sync enabled › tick the mirrored domains
`Find out where one kind of data lives`	Cloud Sync › read which of the three tiers it is in

Honest notes

- **Display currency does not convert.** It changes the symbol on calculators and analytics; live market data keeps its own currency.
- **Usage capture ships on, at Full detail** — which includes typed text. Credentials are excluded structurally at every level, but the rest is yours to narrow. **Erase** reaches only unsent records, not what the server already has.
- **Agentic Mode is experimental** — that is why it lives under Developer.
- **Broker keys are not in Credentials**, and are never synced.
- **Upgrading Python packages may break the terminal**; the versions are pinned.
- **Clearing a data category cannot be undone**, and the confirmation is the only guard.
- **SQL writes are permanent** — the console has no rollback.
- **Replace-from-cloud deletes device-only rows** in that domain; Upload & merge is the safe half of that dialog. **Cloud sync spends credits**, and pauses rather than drops when the balance empties.
- **PIN lockout is time-based**: five failures locks you out for a countdown, with no override in the UI.
- **Font families are filtered on purpose** — hidden and emoji families would mangle every number on screen.
- **Log shipping is off until enabled**, and each of the five streams is opted into separately.
- **Reset UI State needs a restart.** It is the only setting on this screen that does.

Guide ends here

Thank you for reading.

That is all fourteen Settings sub-tabs, from General through Cloud Sync.

FINCEPT RESEARCH DESK · EDITION 5.0

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Profile & Account

A complete, screen-by-screen walkthrough of the Profile tab — your identity, your credit balance and every charge behind it, your subscription, and the seats your licence is spread across.

`>_ identity · credits · itemised ledger · billing · seats · 15 agent tools`

Covers · **account identity** · **credits, usage and the ledger** · **profile & password**
subscription & billing · devices & seats · logout · the 15 agent tools

Demonstrated live on a working Fincept account.

Contents

14 SECTIONS

GETTING ORIENTED

01 What the Profile tab is 3
02 Anatomy of the tab 4
03 The header bar 5

IDENTITY AND SPEND

04 The ACCOUNT panel 6
05 Credits and usage 7
06 The two usage breakdowns 8
07 The credit ledger 9

CHANGING THINGS

08 Edit profile 10
09 Changing your password 11
10 Subscription and billing 12
11 Devices and seats 13
12 Logout, refresh and what is stored 14

REFERENCE

13 Driving it from AI Chat 15
14 Quick reference and honest notes 16

About this guide. Every screenshot is a live capture of the Profile tab in the shipping build, on a real account — the credit, usage, ledger and billing figures are genuine. Personal identifiers (email address, account id, display name, bio and machine name) have been replaced with neutral placeholders.

01 What the Profile tab is

ORIENTATION · 01

Profile is **your account, in full**: who you are signed in as, how many credits you hold, every credit that has been added or spent and what it was spent on, which plan you are on and what you have paid, which machines your licence is activated on, and the two things you can change about yourself — your display details and your password.

It is the only screen in the terminal where money and identity meet, and it is deliberately answerable: the balance at the top can be reconciled line by line against the ledger further down, so no figure has to be taken on trust.

Fig 1 — The header bar: the tab title, your username, the live credit balance, your plan, and REFRESH.

Where to find it

Route	How
Tab bar	**PROFILE** — the last tab on the main bar
Command bar	Type `profile`, `prof` or `account`

At a glance, in the Profile tab you can…

- Confirm which account and plan you are signed in on
- See your live credit balance
- Reconcile credits added against credits used
- See what each product cost you over 30 days
- Read an itemised ledger of every credit movement
- Change your display name and bio
- Change your password
- Review your plan, renewal date and payments
- See which machines hold a seat, and release one
- Buy another seat, or upgrade the plan
- Log out of the terminal

> **Everything on this page comes from the Fincept service**, not from your machine — the tab re-reads account, profile, usage, ledger and billing every time you open it. Nothing here is cached between sessions, and nothing you see is computed locally.

02 Anatomy of the tab

One long scrolling page — no sub-tabs. Two panels sit side by side at the top, two more below them, then three full-width panels and the logout row. The pairing is deliberate: identity beside spend, profile edits beside security, and the three table-bearing panels given the full width.

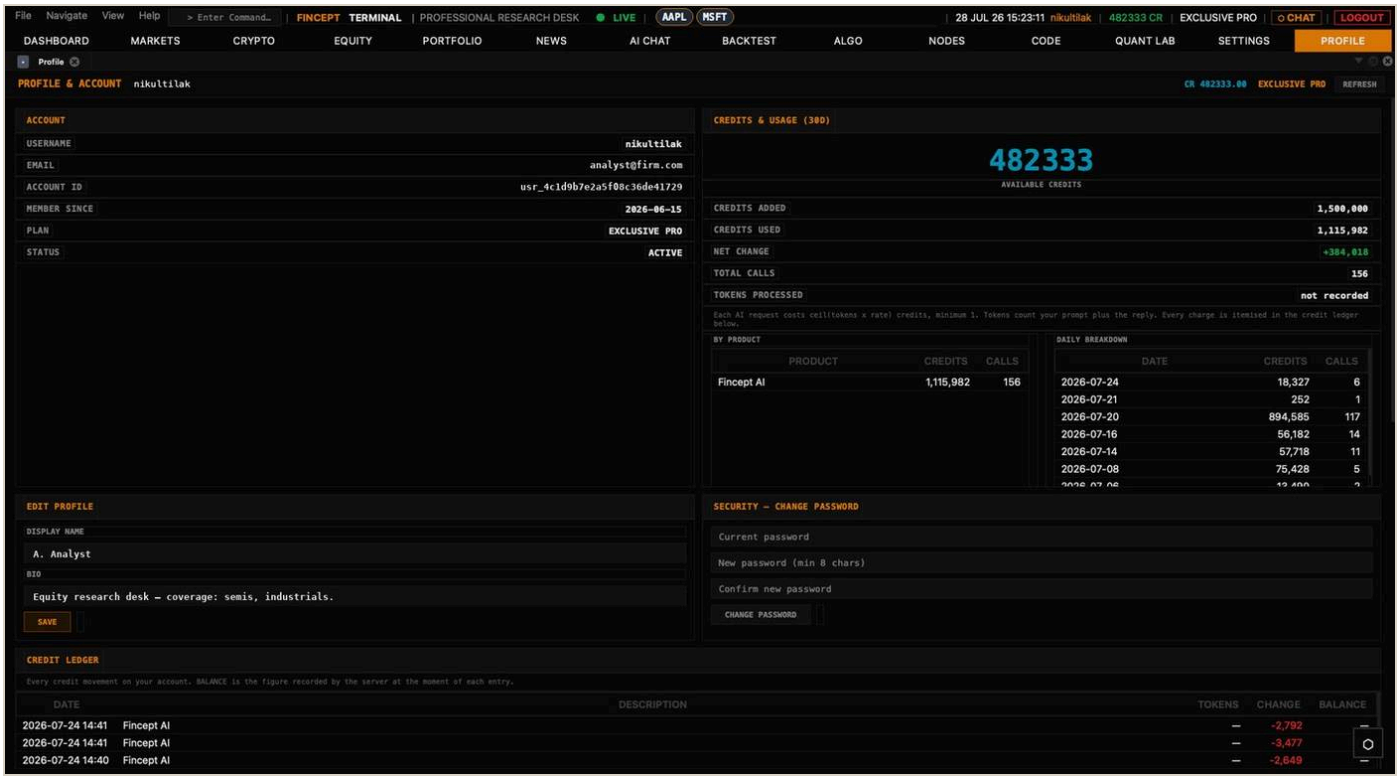


Fig 2 — The top of the page: ACCOUNT and CREDITS & USAGE side by side, then EDIT PROFILE and SECURITY, with the CREDIT LEDGER beginning underneath.

#	Panel	What it is for
1	Header bar	Identity, live balance, plan, and the REFRESH button. Section 03.
2	ACCOUNT	Who you are: username, email, account id, member since, plan, status. Section 04.
3	CREDITS & USAGE (30D)	The balance, the 30-day reconciliation, the pricing rule, and two breakdown tables. Sections 05–06.
4	EDIT PROFILE	Display name and bio — the only profile fields you can change here. Section 08.
5	SECURITY — CHANGE PASSWORD	Current, new and confirm, with client-side rules. Section 09.
6	CREDIT LEDGER	Every credit movement, newest first, paged 50 at a time. Section 07.
7	SUBSCRIPTION & BILLING	Plan, balance, renewal date, upgrade, and payment history. Section 10.
8	DEVICES & SEATS	Seats in use, the machines holding them, release and add-a-seat. Section 11.
9	Logout row	A single red **LOGOUT** button at the foot of the page. Section 12.

03 The header bar

A 38-pixel strip that stays put while the page scrolls beneath it.

Fig 3 — Left to right: the tab title, the signed-in username, the credit balance in cyan, the plan in amber, and REFRESH.

Element	What it shows
`PROFILE & ACCOUNT`	The tab title
`Username`	Who you are signed in as. Falls back to your email address if no username is set
`CR` *n*	Your live credit balance, to two decimal places
`Plan badge`	Your subscription tier, in capitals — e.g. **EXCLUSIVE PRO**
`REFRESH`	Re-reads your identity, profile fields, usage, ledger and billing from the service

> **The balance updates itself.** The terminal keeps an account-event connection open, so when credits are spent on another screen — an AI chat turn, an agent run, an Alpha Arena round — the header figure and the big number below it move without you pressing anything.

> **REFRESH does not re-read the devices table.** It covers identity, profile, usage, ledger and billing. The seat roster is re-read when you *enter* the tab, so if you have just released a seat on another machine, leave the tab and come back rather than pressing REFRESH.

The same figures elsewhere

Your credit balance also appears in the terminal's top toolbar, on every screen. The two are the same number from the same source; this page is where it can be explained.

04 The ACCOUNT panel

Six rows of identity, read straight from the service.

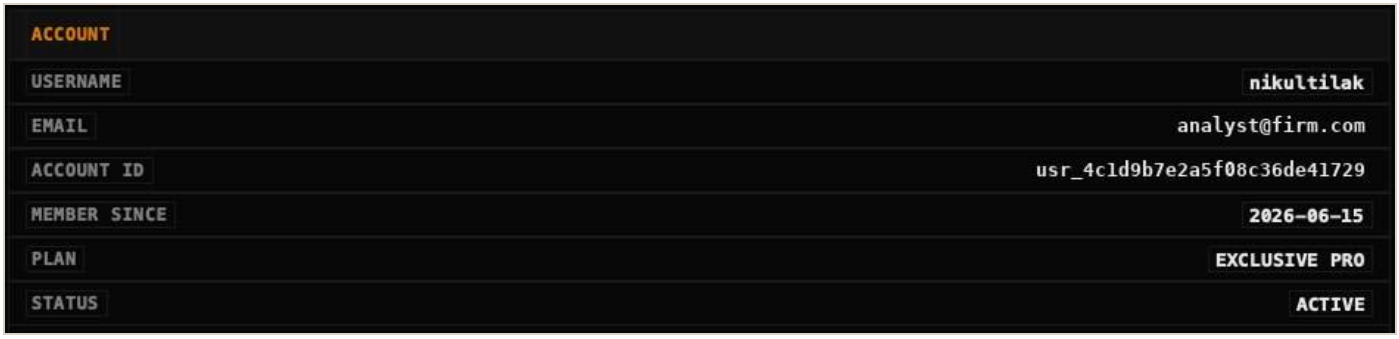


Fig 4 — The ACCOUNT panel. (Email and account id are replaced with placeholders in this guide.)

Row	What it is	Use it for
`USERNAME`	Your Fincept username	Confirming which account a terminal is signed in as
`EMAIL`	The address on the account	Checking where service mail is going
`ACCOUNT ID`	The service's identifier for you (`usr_…`)	**Quote this in a support ticket** — it identifies you unambiguously
`MEMBER SINCE`	The date the account was created	Tenure, and sanity-checking which account you are on
`PLAN`	Subscription tier, in capitals	What you are entitled to
`STATUS`	**ACTIVE** or **INACTIVE**	Whether the subscription is currently in force

> **STATUS is about the subscription, not the connection. INACTIVE** means the account has no paid plan in force — it does not mean the terminal is offline. Connectivity is shown by the status bar at the foot of the window.

> **Two of these rows are the ones support will ask for:** ACCOUNT ID and PLAN. Screenshot this panel with the version panel from the About tab and a ticket has everything it needs.

05 Credits and usage

Credits are what the terminal's AI features consume. This panel shows the balance, then both sides of the 30-day window so the balance can actually be reconciled.

CREDITS ADDED	1,500,000
CREDITS USED	1,115,982
NET CHANGE	+384,018
TOTAL CALLS	156
TOKENS PROCESSED	not recorded

Each AI request costs ceil(tokens x rate) credits, minimum 1. Tokens count your prompt plus the reply. Every charge is itemised in the credit ledger below.

Fig 5 — The reconciliation rows and the pricing rule, under the big balance figure.

Row	Meaning
big cyan number	Your available credits right now, rounded down to a whole credit
CREDITS ADDED	Everything granted to the account in the last 30 days — plan grants, top-ups
CREDITS USED	Everything spent in the same window
NET CHANGE	Added minus used. **Green** when you are ahead over the window, **red** when you have spent more than arrived
TOTAL CALLS	How many metered requests were made
TOKENS PROCESSED	The tokens those requests were billed on — or *not recorded* (see below)

How a charge is worked out

The panel states the rule on screen rather than leaving it in the service: **each AI request costs `ceil(tokens × rate)` credits, with a minimum of 1**, where tokens count your prompt *plus* the reply. Every individual charge is itemised in the ledger (section 07).

> **Why "not recorded" appears.** Token counts were added to the billing record part-way through the product's life. When no entry in the 30-day window carries a token count, the row says *not recorded* rather than showing a bare **0**, which would read as "this was free". Newer activity shows a real figure.

> **The window is fixed at 30 days.** There is no date picker on this panel — for anything older, page back through the ledger, which is not windowed.

06 The two usage breakdowns

Underneath the summary sit two tables, side by side: what you spent it on, and when.

BY PRODUCT

PRODUCT	CREDITS	CALLS
Fincept AI	1,115,982	156

DAILY BREAKDOWN

DATE	CREDITS	CALLS
2026-07-24	18,327	6
2026-07-21	252	1
2026-07-20	894,585	117
2026-07-16	56,182	14
2026-07-14	57,718	11
2026-07-08	75,428	5
2026-07-06	13,490	2

Fig 6 — **BY PRODUCT** on the left, **DAILY BREAKDOWN** on the right. Numeric columns are right-aligned and thousands-separated to match the rows above.

Table	Columns	Ordering
`BY PRODUCT`	PRODUCT · CREDITS · CALLS	One row per metered product — the managed **Fincept AI** model is normally the only entry
`DAILY BREAKDOWN`	DATE · CREDITS · CALLS	Newest day first, one row per day that had activity

Reading them together

- **A single heavy day** in the daily table usually explains a surprising monthly total — look for the day whose credit figure dwarfs the others, then find that day in the ledger to see which runs did it.
- **Calls versus credits** is the useful ratio: many calls with few credits means short prompts; few calls with many credits means long context or long replies, which is where cost accumulates.
- **Days with no activity are absent**, not zero rows — the table is a list of days that had spend.

> **Both tables are read-only** and cannot be sorted, filtered or exported from this screen. For a figure you can put in a spreadsheet, ask an assistant for `profile_get_usage` (section 13), or read the ledger below.

07 The credit ledger

The itemised statement behind every aggregate above: one row per credit movement, newest first. This is the panel that makes the balance answerable.

Fig 7 — The credit ledger, showing 50 of 167 entries, with LOAD OLDER at the foot.

Column	What it holds
DATE	When the entry was written, in your local time, to the minute
DESCRIPTION	What it was — the product or the reason for a grant
TOKENS	The tokens the charge was computed from, or — for grants and for entries written before token counts were recorded
CHANGE	The movement itself: **green** + for credits arriving, **red** – for credits spent
BALANCE	The balance the service recorded at the moment of that entry, or — where none was stored

Paging

- The ledger loads **50 entries at a time**, newest first.
- The footer reads *Showing 50 of 167 entries* so you always know how much is behind you.
- **LOAD OLDER** appends the next 50 and disappears once everything is loaded.
- Re-entering the tab reloads the newest page from scratch.

> **An empty ledger is never silent.** If the read fails, the footer says so explicitly — because a blank table would otherwise be indistinguishable from “nothing was ever charged”, which is the opposite of what this panel is for. A genuinely new account reads *No credit activity yet*.

> **Reconciling.** Sum the CHANGE column over the last 30 days and you get the CREDITS ADDED and CREDITS USED figures from section 05; the newest BALANCE should equal the number at the top of the page.

08 Edit profile

Two fields, one button. These are the only pieces of your profile that this screen can change — username, email and plan are all managed by the service.

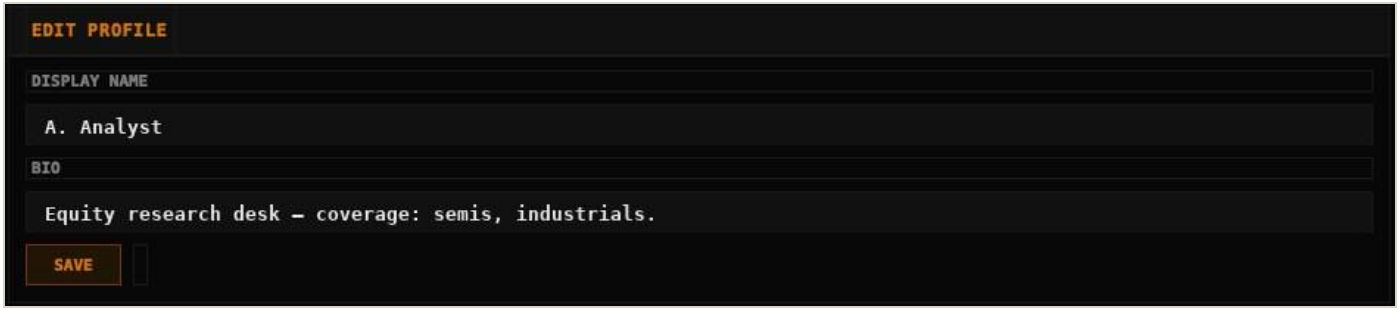


Fig 8 — The EDIT PROFILE panel. (The values shown are placeholders.)

Field	Limit	Notes
`DISPLAY NAME`	80 characters	How you are shown in the product. Blank is allowed
`BIO`	500 characters	A short description. It is a **single-line** field — long text scrolls sideways rather than wrapping
`SAVE`	—	Writes both fields, then refreshes your session so the rest of the terminal picks up the change

What you will see

Status	Meaning
`Saving…`	The request is in flight; the button is disabled meanwhile
`Saved ✓`	Both fields were accepted, in green
`an error message`	The service rejected the change, in red, with its reason

> **Your typing is never overwritten.** The tab re-reads these fields whenever it refreshes, but it skips any field you currently have the cursor in — so a background refresh cannot clobber a half-typed bio.

> **There is no cancel.** Editing a field and navigating away simply leaves it unsaved; the original values come back the next time the tab loads. Nothing is written until you press **SAVE**.

09 Changing your password

Three masked fields and one button. The checks run in the terminal before anything is sent, so mistakes are caught instantly.

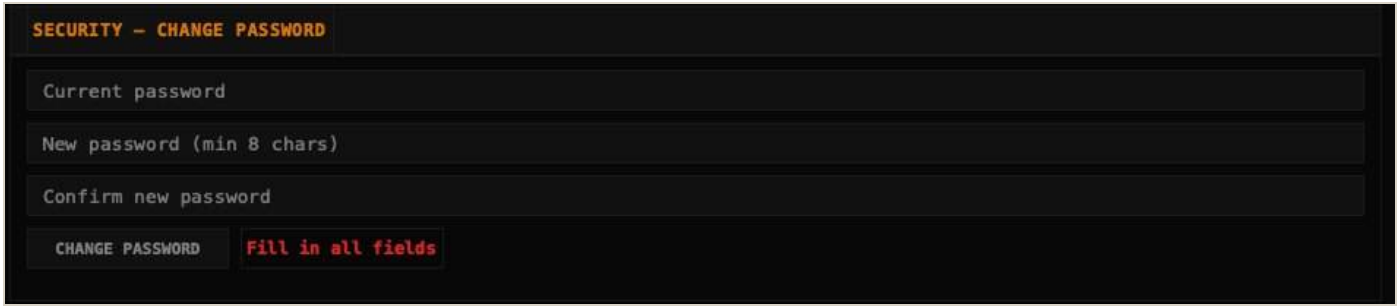


Fig 9 — The SECURITY panel, showing the validation message that appears when the fields are empty.

Rule	Message when it fails
`All three fields must be filled`	*Fill in all fields*
`The new password must be at least 8 characters`	*New password must be ≥ 8 characters*
`The new password and its confirmation must match`	*Passwords do not match*

What happens on success

- The status reads **Password changed ✓** in green.
- **All three fields are cleared automatically**, so nothing is left on screen.
- You stay signed in on this machine — changing the password does not log you out here.

> **If the current password is wrong**, the service says so and nothing changes. The check happens server-side, so an incorrect current password is reported after the round trip rather than instantly like the three rules above.

> **Two-factor authentication is not on this screen.** The account supports it and an assistant can turn it on or off through the agent tools (section 13), but there is no control for it in this build's Profile tab.

10 Subscription and billing

What you are on, when it renews, and what you have paid.

Fig 10 — SUBSCRIPTION & BILLING: three summary rows, the upgrade button, and the payment history table.

Element	What it shows
PLAN	Your current tier
CREDIT BALANCE	The same balance as the header, to two decimals
RENEWS	The next billing date, or — if the plan does not renew
MANAGE / UPGRADE PLAN	Opens the pricing view, where plans are compared and checkout starts
PAYMENT HISTORY	**DATE · AMOUNT · STATUS · REFERENCE** — the 20 most recent payments

> **Amounts are shown in the currency you were charged in**, not in your display-currency preference. Fincept plans are priced in US dollars, so a payment normally reads `$299.00` even if the terminal is set to display another currency elsewhere — the preference changes symbols, it does not convert money you have already paid.

> **REFERENCE is the payment gateway's own identifier** for the transaction. Quote it when querying a specific charge; it is the fastest way for support to find one payment among many. It can be blank on older or manually-applied entries.

> **STATUS** reflects the gateway's verdict — **SUCCESS** for a completed payment. A failed or pending charge appears here too, which is where to look first if a renewal did not go through.

11 Devices and seats

Each subscription includes one seat — one machine. This panel is the only place a licence can be moved between computers, so it is worth understanding before you need it on a deadline.

Fig 11 — DEVICES & SEATS: the seat count, the roster, and the two actions. (The machine name is a placeholder in this guide.)

Element	What it shows or does
`SEATS IN USE`	*n of m* — seats occupied against seats owned
`DEVICE`	The machine's name. The one you are sitting at is marked **(this machine)**
`ACTIVATED`	When that machine first took a seat
`LAST SEEN`	When it last checked in
`RELEASE SELECTED DEVICE`	Frees that machine's seat. Disabled until you select a row
`ADD A SEAT`	Opens the pricing view to buy an additional seat

Releasing a machine

Select the row, then press **RELEASE SELECTED DEVICE**. The confirmation names the machine, and it is worded differently depending on which one you picked:

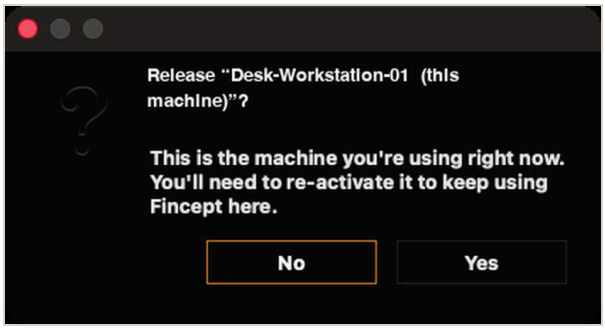


Fig 12 — Releasing the machine you are using right now says so plainly. Releasing any other machine warns that it will need to re-activate.

- The freed seat becomes available **immediately** — you can activate elsewhere straight away.
- A released machine is not wiped; it simply needs to re-activate before it can be used again.
- Releasing *this* machine un-licenses the terminal you are looking at, which is legal but self-defeating — hence the separate wording.
- **The DEVICE column is narrow**, so a long name — and the **(this machine)** marker — is usually clipped in the table. The dialog always shows the full label: read it there before answering **Yes**.

12 Logout, refresh and what is stored

Logging out

A single red **LOGOUT** button sits at the foot of the page, well away from everything else. It always confirms.

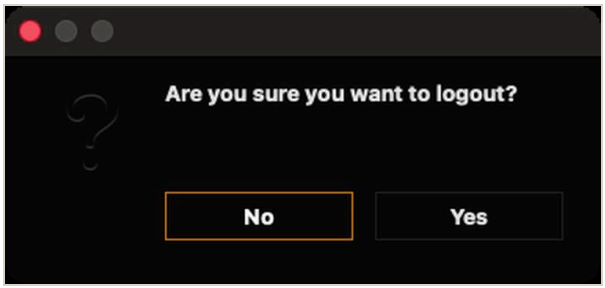


Fig 13 — The logout confirmation, defaulting to **No**.

Logging out ends the session on this machine and returns the terminal to its sign-in screen. It does **not** release your seat — the machine stays activated, so signing back in does not consume another seat. To actually free the seat, release the device (section 11).

When the tab refreshes

Trigger	What is re-read
`Opening the tab`	Everything — identity, profile fields, usage, ledger, billing **and** the seat roster
`The REFRESH button`	Identity, profile fields, usage, ledger, billing — **not** the seat roster
`A credit being spent anywhere`	The balance figures, pushed live over the account connection
`Signing in or out`	The whole page

What the tab stores

Nothing. The Profile tab keeps no local state: it does not remember your scroll position, how many ledger pages you had loaded, or anything you typed but did not save. Every value is fetched fresh from the service each time the tab is shown — which is exactly what you want from a page whose job is to tell you the truth about your account.

> **One practical consequence.** If you have paged deep into the ledger and switch away, you will come back to the newest 50 entries again. Do your ledger reading in one pass.

13 Driving it from AI Chat

The Profile tab publishes **15 agent tools** in the `profile` category — and notably, several of them reach account features that have no control on this screen.

Group	Tools	What they do
`Profile (3)`	profile_get, profile_update, profile_get_session	Read the profile, change display name and bio, inspect the signed-in session
`Credentials (2)`	profile_get_api_key, profile_regenerate_api_key	**Read or replace your API key** — no equivalent control exists on this screen
`Security (3)`	profile_enable_mfa, profile_disable_mfa, profile_get_login_history	Turn two-factor authentication on or off, and review sign-ins — none of it in the UI
`Money (4)`	profile_get_credits, profile_get_usage, profile_get_subscription, profile_get_payment_history	The same figures as sections 05, 06 and 10, in a form you can put in a document
`Notifications (3)`	profile_get_notifications, profile_mark_notification_read, profile_mark_all_notifications_read	Account notifications — also not surfaced on this screen

> **Treat `profile_get_api_key` with care.** It returns your account's API key in plain text to the assistant, and therefore into the conversation transcript. Ask for it only when you actually need it, and prefer `profile_regenerate_api_key` if you believe a key has been exposed.

> **`profile_disable_mfa` weakens your account.** Because there is no UI switch, an assistant is the only way to toggle two-factor authentication — in both directions. Be deliberate about what you ask for.

What these are good for

- **Cost reporting.** "Summarise my credit usage by day for the last month" turns the daily table into prose or a file you can send on.
- **Reconciliation.** An assistant can pull the ledger and the usage totals together and point at the day that explains a spike.
- **Account hygiene.** Reviewing login history and rotating the API key are both one sentence.

14 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	**PROFILE** on the tab bar, or type `profile` in the command bar
…find my account id for support?	The ACCOUNT panel — quote it with your build version from the About tab
…see my credit balance?	The header, or the big cyan number in CREDITS & USAGE
…find out what used my credits?	BY PRODUCT and DAILY BREAKDOWN, then the ledger for the individual charges
…see older credit history?	**LOAD OLDER** at the foot of the ledger, 50 entries at a time
…change my display name?	EDIT PROFILE ▸ type ▸ **SAVE**
…change my password?	SECURITY — fill all three fields; the new one needs 8+ characters
…check when my plan renews?	The **RENEWS** row in SUBSCRIPTION & BILLING
…find a payment for an invoice query?	PAYMENT HISTORY — quote the **REFERENCE**
…move my licence to another machine?	DEVICES & SEATS ▸ select the old machine ▸ **RELEASE SELECTED DEVICE**
…use Fincept on two machines at once?	**ADD A SEAT** — one seat is one machine
…log out?	The red **LOGOUT** button at the foot of the page

Honest notes on this build

- **REFRESH does not re-read the devices table** — leave the tab and return to refresh seats.
- **The usage window is fixed at 30 days** with no date picker; the ledger is the way further back.
- **TOKENS PROCESSED can read "not recorded"**, and ledger TOKENS/BALANCE cells can read **—**, where the underlying entries predate those columns.
- **BIO is a single-line field** despite allowing 500 characters.
- **There is no cancel button on the edit form**; unsaved edits simply disappear on reload.
- **Payment amounts ignore your display-currency preference** — they are shown as charged.
- **The DEVICE column clips long machine names**, including the **(this machine)** marker.
- **Logging out does not free a seat**; only releasing the device does.
- **API key, two-factor authentication, login history and notifications are agent-only** — the tools exist, the controls do not.
- **The tab stores nothing locally**: scroll position and loaded ledger pages reset each visit.
- **Payment history is capped at 20 entries** and cannot be paged from the screen.

> Everything above was read from the shipping source and confirmed on a live account. No profile field was changed, no password was set, no seat was released and no logout was performed while this guide was captured.

FINCEPTTERMINAL

A professional research desk in a single window — markets, analytics, automation and the data behind them.

46	635	864	7
DOCUMENTS	SECTIONS	PAGES	PARTS

Part 0 The Prospectus — 5 documents
What the terminal is, where it stands, and what it costs.

Part I Getting Started — 6 guides
The terminal, end to end — what it is and how it is laid out.

Part II Markets & Trading — 5 guides
Live prices, order entry and automated execution.

Part III Research & Analytics — 10 guides
The analyst desks — valuation, risk, portfolios and strategy.

Part IV Data & Intelligence — 9 guides
Where the numbers come from — macro, sovereign, market and news feeds.

Part V Automation & AI — 6 guides
Agents, workflows and the tooling that drives the terminal for you.

Part VI Workspace & System — 5 guides
The everyday tools, your account and how the terminal is configured.

NAVIGATING THIS VOLUME

Contents Every row of the contents and the detailed contents is a link — click a guide or a section to jump straight to its page.

Bookmarks The sidebar carries the whole volume three levels deep: part, guide, then section. It works from any page.

Back The tag at the top right of a guide page returns you to that guide's own contents.

Fincept Research Desk · Professional Edition v5.0 · Issued August 2026

Every screenshot in this volume is a live capture of the shipping build. Control labels, field defaults and behaviour are transcribed from the running application.

Confidential — prepared for institutional clients. Not for redistribution.