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.



41	472	700	6
GUIDES	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 · JULY 2026

Contents

41 GUIDES · 6 PARTS

PART I Getting Started

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

Dashboard 9 sections · 25 pp 17

Dashboard Widgets 6 sections · 17 pp 42

Docs 8 sections · 11 pp 59

About 9 sections · 11 pp 70

PART II Markets & Trading

Live prices, order entry and automated execution.

Markets 8 sections · 12 pp 81

Equity Trading 17 sections · 24 pp 93

Crypto Trading 11 sections · 15 pp 117

Algo Trading 21 sections · 23 pp 132

Watchlist 7 sections · 9 pp 155

Trade Viz 8 sections · 10 pp 164

PART III Research & Analytics

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

Equity Research 12 sections · 25 pp 174

Portfolio 18 sections · 25 pp 199

Backtest 11 sections · 21 pp 224

Derivatives 14 sections · 24 pp 245

Alternative Investments 15 sections · 26 pp 269

M&A Analytics 13 sections · 18 pp 295

Surface Analytics 9 sections · 18 pp 313

AI Quant Lab 9 sections · 16 pp 331

Alpha Arena 8 sections · 12 pp 347

PART IV Data & Intelligence

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

Economics 7 sections · 12 pp 359

Government Data 11 sections · 15 pp 371

DBnomics 11 sections · 13 pp 386

AkShare 10 sections · 12 pp 399

Asia Markets 8 sections · 10 pp 411

Data Sources 10 sections · 14 pp 421
Data Mapping 12 sections · 15 pp 435
News 11 sections · 13 pp 450
RSS Feeds 9 sections · 13 pp 463
Geopolitics 12 sections · 18 pp 476
Maritime 11 sections · 17 pp 494

PART V Automation & AI

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

AI Chat 8 sections · 10 pp 511
Agent Studio 15 sections · 27 pp 521
Node Editor 23 sections · 28 pp 548
MCP Servers 10 sections · 14 pp 576
Code 7 sections · 10 pp 590
Report Builder 12 sections · 17 pp 600

PART VI Workspace & System

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

Excel 12 sections · 14 pp 617
Notes 12 sections · 14 pp 631
File Manager 11 sections · 13 pp 645
Settings 23 sections · 26 pp 658
Profile 14 sections · 16 pp 684

> **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 I · GETTING STARTED

Dashboard 17

01 What the Dashboard is 19
02 Anatomy of the screen 20
03 The top bar & navigation 21
04 Board controls (the toolbar) 22
05 Working with widgets 23
06 Widget catalog — all 36 widgets 24
07 The Market Pulse rail 39
08 Ticker tape 40
09 Quick reference — “How do I…” 41

Dashboard Widgets 42

01 Markets — indices, crypto, heatmap, sentiment, Polymarket… 44
02 Research — candle chart, economic calendar, news… 49
03 Portfolio — summary, holdings, positions… 52
04 Trading — margin usage, orders, trade tape… 54
05 Tools — web scraper, notes, RSS, agent errors… 56
06 Geopolitics — event map & feeds 58

Docs 59

01 What the Docs tab is 61
02 Anatomy of the tab 62
03 All forty topics 63
04 What a topic page contains 64
05 How pages load and re-style 66
06 What is covered — and what is not 67
07 Using it well 68
08 Quick reference and honest notes 69

About 70

01 What the About tab is 72
02 Anatomy of the tab 73
03 Version information 74
04 Check for Updates 75
05 How updates are delivered 76
06 Licence 77
07 Diagnostics and crash dumps 78
08 Resources and contacts 79
09 Quick reference and honest notes 80

PART II · MARKETS & TRADING

Markets 81

01 What the Markets tab is 83
02 Anatomy of the screen — the five regions 84
03 The header bar 85
04 Reading a market panel 86
05 The nine default panels 87
06 Building your own board (+ the panel editor) 88
07 Refresh, states & good to know 90
08 Quick reference — “How do I…” 91

Equity Trading 93

01 What the Equity tab is 95
02 Anatomy of the workstation 96
03 The command bar 97
04 Broker accounts 98
05 Paper, live, and the approval gate 100
06 The watchlist 101
07 Charts, compare and chart trading 102
08 Market depth 104
09 The order ticket: the core form 105
10 Advanced, GTT and option strategies 106
11 Sending one order to several accounts 108
12 Blotter: Positions and Holdings 109
13 Blotter: Orders, Funds and Stats 111
14 US brokers, feeds and the unified monitor 113
15 Data, persistence and agent tools 114
16 Quick reference 115
17 Honest notes on this build 116

Crypto Trading 117

01 What the Crypto Trading tab is 119
02 Anatomy — the six regions 120
03 Command bar, exchanges & modes 121
04 The watchlist 122
05 The chart 123
06 Order book, depth & tape 125
07 The order ticket 126
08 The blotter 128
09 Paper, live & connecting an exchange 129
10 AI tools & the data engine 130
11 Quick reference & honest notes 131

Algo Trading 132

01 What the ALGO tab is 134
02 Anatomy: the frame and the seven tabs 135
03 Builder — the four-step flow 136
04 Builder — writing rules in sentences 137
05 Builder — the safety net and F&O legs 138
06 Builder — the AI strategy assistant 139
07 Strategies — your library 140
08 Backtest Lab — one symbol, one report 141
09 Backtest Lab — baskets, and what it does not model 142
10 Scan — the one-time check 143
11 Scan — the live universe sweep 144
12 Alerts — background watches 145
13 The deploy dialog, field by field 146
14 Baskets and F&O options baskets 147
15 Dashboard — the fleet 148
16 Dashboard — positions, fills and controls 149
17 History — the permanent ledger 150
18 How a deployment actually runs 151
19 The safety rails, in numbers 152
20 Driving the desk from AI Chat 153
21 Quick reference and honest notes 154

Watchlist 155

01 What the Watchlist tab is 157
02 Anatomy — the four regions 158
03 Your watchlists — the sidebar 159
04 Adding symbols & the two views 160
05 Live prices, CSV & backtest 161
06 AI tools & where lists appear 162
07 Quick reference & honest notes 163

Trade Viz 164

01 What the Trade Viz tab is 166
02 Anatomy of the tab 167
03 The chord diagram, line by line 168
04 The partner ranking table 169
05 The filter bar 170
06 The dataset behind the picture 171
07 Using it well — and where live trade data lives 172
08 Quick reference and honest notes 173

PART III · RESEARCH & ANALYTICS

Equity Research 174

· Overview — what the Equity Research workstation is 176
1 Anatomy of the workstation (title bar, quote bar, rail, dock) 177
2 OVERVIEW — the one-screen snapshot 179
3 FUNDAMENTALS — Financials · Valuation · Analysis · Quality 181
4 MARKET — Technicals · Peers · Ownership · Cross-Asset 185
5 CONTEXT & RESEARCH — Macro · Quant Lab · Scenario · Filings 189
6 FLOW — News · Sentiment 192
7 The research dock — Notes · Copilot · Alerts 194
8 Toolbar actions — Buy/Sell · Backtest · CSV export 194
9 Where the numbers come from (data & provenance) 196
10 AI agent tools (45 MCP tools) 196
11 Quick reference & important notes 198

Portfolio 199

01 What the Portfolio tab is 201
02 Workspace anatomy & the command bar 202
03 The live dashboard — holdings, chart, sectors, positions 204
04 Sectors & concentration 206
05 Performance & risk 207
06 Portfolio optimization 208
07 QuantStats report 211
08 Reports & attribution 212
09 Custom indices 213
10 Risk management 214
11 Financial planning 215
12 Economics & macro factors 216
13 Trading — order entry, buy / sell / dividend 217
14 Managing portfolios — create, import, export 218
15 AI & Agent insights 219
16 Analytics engine, live data & metrics 220
17 Agent automation tools 221
18 Quick reference & honest notes 222

Backtest 224

· Overview — what the Backtest tab is 226
1 Anatomy of the workbench (top bar, three panels, status bar) 227
2 Providers — the 6 backtest engines 229
3 Commands — the 11 operations & how the UI morphs 230
4 Configuring a run — market data, execution, advanced, benchmark, strategy 231
5 The five result windows (Summary · Equity · Metrics · Details · Raw JSON) 232
6 Core commands — Backtest, Optimize, Walk-Forward 236
7 Analysis commands — Indicators, Signals, ML Labels, CV Splits, Returns, Sweep 238

8 Strategy catalog (per engine) + the Custom Combo builder 242
9 Engine, data & AI agent tools 243
10 Quick reference — "How do I…?" & important notes 244

Derivatives 245

· Overview — what the Derivatives tab is 247
1 Anatomy of the workbench (shell, rail + canvas, charts) 248
2 CHAIN — the option chain 250
3 VOL SURFACE — SVI / SSVI calibration 252
4 VOL ANALYSIS — realized vs implied, VRP, VIX strip 254
5 STRATEGY LAB — 23 templates, payoff & greeks 256
6 RISK — greeks, scenarios & VaR 258
7 POSITIONING — dealer gamma (GEX) 260
8 EVENTS — earnings / event vol & density 262
9 PRICERS — 16 multi-asset calculators 264
10 Data sources & the seeded demo 266
11 Analytics engine — capability catalog 266
12 AI agent tools (35 MCP tools) 267
13 Quick reference — "How do I…?" & important notes 268

Alternative Investments 269

· Overview — what the Alt Investments tab is 271
1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 272
2 PROGRAM — set up & edit the LP program 274
3 PERFORMANCE — IRR / TVPI / PME & J-curves 276
4 PACING — Takahashi-Alexander projection & commitment solver 278
5 LIQUIDITY — calls vs distributions, coverage & redemption ladder 280
6 EXPOSURE — concentration, treemap, matrix & de-smoothing 282
7 SCENARIOS — historical & custom stress, denominator effect 284
8 DEAL LAB — waterfalls, fees, secondaries, RE/infra underwriting 286
9 INSTRUMENT LAB — 24 instrument calculators 288
10 MARKET — live proxies & embedded benchmarks 290
11 The seeded demo program (reference numbers) 292
12 Analytics engine — capability catalog 293
13 AI agent tools (30 MCP tools) 293
14 Quick reference — "How do I…?" & important notes 294

M&A Analytics 295

· Overview — what the M&A Analytics tab is 297
1 Anatomy of the workbench (shell, rail + canvas, shared widgets) 298
2 DEAL DESK — set up the deal 300
3 VALUATION — DCF, comps, precedents, LBO & the football field 302
4 MERGER MODEL — accretion/dilution, sources & uses, synergies 304
5 STRUCTURE — payment mix, exchange ratio, collar, earnout, CVR 306

6 MARKET INTEL — deal database, live EDGAR scan, cross-deal analytics 307
7 SPECIALIZED — startup valuation & industry lens 309
8 RISK & FAIRNESS — Monte Carlo, fairness opinion, premium, process 310
9 The seeded sample deal & deal database (reference) 311
10 Analytics engine — capability catalog 311
11 AI agent tools (47 MCP tools) 311
12 Quick reference — “How do I…?” & important notes 312

Surface Analytics 313

· Overview — what the Surface Analytics tab is 315
1 Anatomy of the workbench 316
2 The 7 categories & 35 surfaces — the catalog 317
3 The Control Panel — inputs & parameters 324
4 The three views — 3D, Table, Line 325
5 The Data Inspector — raw data, lineage, status 327
6 Data & connectivity — the four tiers, Databento, CSV 327
7 AI agent tools (15 MCP tools) 329
8 Quick reference & important notes 329

AI Quant Lab 331

· Overview — what the AI Quant Lab is 333
1 Anatomy — the 24-module workbench 334
2 CORE — factors, models, backtests, signals 336
3 AI / ML — agents, RL, online & meta learning 338
4 ADVANCED — HFT, optimization, feature engineering 340
5 ANALYTICS — GS Quant, statsmodels, CFA, forecasting 342
6 The engine — Python, credits & dependencies 344
7 Driving Quant Lab by AI — the MCP tool suite 345
8 Quick reference & honest caveats 346

Alpha Arena 347

01 What Alpha Arena is 349
02 Anatomy — the live dashboard 350
03 Setting up — the competition wizard 351
04 Running & controlling a competition 352
05 Reading the arena — the six panels 353
06 Under the hood — the trading engine 355
07 Live mode & its beta status 356
08 Cost, AI tools & honest notes 358

PART IV · DATA & INTELLIGENCE

Economics 359

01 What the Economics tab is 361

02 Anatomy — rail, workspace & panel 362
03 Reading a series — chart & table 364
04 Picking your data — the panel patterns 365
05 The 48 data sources — full catalog 367
06 Providers, API keys & the fetch engine 369
07 Quick reference & honest notes 370

Government Data 371

· Overview — what the GOVT Data tab is 373
1 Anatomy — the explorer shell 374
2 US Treasury — securities, auctions & summary 375
3 US Congress — bills & legislative activity 376
4 France — data.gouv.fr 377
5 Hong Kong — data.gov.hk 378
6 Australia — data.gov.au 379
7 The open-data (CKAN) panel — six portals in one 380
8 How it works — the live fetch engine 383
9 The eleven sovereign portals — catalog 384
10 Quick reference & honest caveats 385

DBnomics 386

01 What the DBnomics tab is 388
02 Anatomy of the tab 389
03 The selection panel 390
04 Global search 391
05 Reading the chart and the table 392
06 SINGLE view — several series, one chart 393
07 COMPARE view — the slot system 394
08 Exporting to CSV 395
09 Where the data comes from 396
10 Driving it from AI Chat 397
11 Quick reference and honest notes 398

AkShare 399

01 What the AkShare tab is 401
02 Anatomy of the tab 402
03 The twenty-six sources 403
04 The endpoint list 404
05 The parameter form 405
06 Running a query and reading results 406
07 When a query fails — the hints 407
08 Caching, state and cost 408
09 Driving it from AI Chat 409
10 Quick reference and honest notes 410

Asia Markets 411

01 What the Asia Markets tab is 413
02 Anatomy of the tab 414
03 The eight categories 415
04 Finding and running an endpoint 416
05 Reading the results 417
06 When a query fails 418
07 Regions, caching and what is remembered 419
08 Quick reference and honest notes 420

Data Sources 421

· Overview — what the Data Sources tab is 423
1 Anatomy of the screen (header, stats, tabs, two views) 424
2 BROWSE — search & filter the catalog 425
3 The connector Inspector 426
4 The Connection Config window (add / edit / clone) 427
5 Testing connections & live status 429
6 CONNECTIONS — manage saved connections 430
7 Import / Export / Template 430
8 The connector catalog — 118 connectors by category 432
9 Security & data handling · Quick reference · Notes 434

Data Mapping 435

· Overview — what the Data Mapping tab is 437
1 Anatomy of the screen (chrome, panels, status) 438
2 View · MAPPINGS — the saved-mappings list 439
3 View · TEMPLATES — 14 ready-made APIs 440
4 Wizard Step 1 · API Config 442
5 Wizard Step 2 · Schema 443
6 Wizard Step 3 · Field Mapping 444
7 Wizard Step 4 · Cache & Security 445
8 Wizard Step 5 · Test & Save 446
9 Reference — the 7 schemas, parsers, transforms, templates 447
10 Under the hood — network, storage, security 448
11 Quick reference & notes 449

News 450

01 What the News tab is 452
02 Anatomy of the screen 453
03 Filtering & searching the wire 454
04 Counts, sentiment & board controls 455
05 Reading the wire 456
06 Clusters, the banner & the ticker 457

07 The article detail panel 458
08 The Intelligence drawer 459
09 Managing news sources 460
10 Live feed, refresh & states 461
11 Quick reference — “How do I…” 462

RSS Feeds 463

01 What the RSS Feeds tab is 465
02 Anatomy — the board & toolbar 466
03 Feed boxes — cards & tables 468
04 Adding a feed — the Add Feed dialog 469
05 Field mapping & one-click Discover 470
06 Managing feeds — edit, refresh, remove 472
07 Under the hood — the feed engine 473
08 AI feed tools & cross-app reuse 474
09 Quick reference & honest notes 475

Geopolitics 476

1 What this tab is 478
2 Anatomy of the screen 479
3 Filters & legend 480
4 The conflict map 481
5 The events table 483
6 The intelligence sidebar 483
7 HDX Data 486
8 Relations 488
9 Trade 489
10 Credits — what costs what 491
11 Status & message reference 492
12 Quick reference — how do I… 493

Maritime 494

1 What this tab is 496
2 Anatomy of the screen 497
3 Top bar 497
4 Left panel — load, draw, intelligence 498
5 Map toolbar & the live vessel map 500
6 The vessel table 502
7 Right panel — search 503
8 Voyage history 506
9 Credits — what costs what 507
10 Status & message reference 508
11 Quick reference — how do I… 510

AI Chat 511

01 What the AI Chat tab is 513
02 Anatomy of the screen 514
03 Getting started 515
04 Reading a conversation 516
05 The agentic engine 517
06 Conversations & history 518
07 The Quick-Chat bubble 519
08 Quick reference — "How do I…" 520

Agent Studio 521

· Overview — what the Agent Studio tab is 523
1 Anatomy — the Agent Studio shell 524
2 AGENTS — browse & configure the persona catalog 526
3 CREATE — author a custom agent 528
4 TEAMS — multi-agent teams & the six modes 530
5 WORKFLOWS — one-click analyses 532
6 PLANNER — plan, edit, then execute 534
7 DESKS — standing, scheduled agent desks 536
8 TOOLS — the tool catalog & assignment console 538
9 CHAT — talk to an agent 540
10 SYSTEM — the capabilities dashboard 542
11 The agent engine — how a run executes 543
12 Driving Agent Studio by AI — the MCP tool suite 544
13 Advanced — Agentic Mode (developer) 545
14 Quick reference & honest caveats 546

Node Editor 548

01 What the Node Editor is 550
02 Anatomy of the workbench 551
03 The toolbar, button by button 553
04 The node palette 554
05 The canvas: move, zoom, connect 556
06 A node up close 557
07 The properties panel 558
08 How a workflow actually runs 559
09 The execution results panel 560
10 Triggers, Core, Control Flow 562
11 Data Transform and Data Format 563
12 Market Data and Analytics 564
13 Utilities, Files and Data Sources 565
14 Trading, Safety, Agents and MCP 566

15 What costs money, what touches the network 567
16 Trading safely: paper, approval, audit 568
17 Expressions and data flow 569
18 The fifteen templates 570
19 Saving, loading, import and export 571
20 Deploy — and what it does not do 572
21 Driving workflows from AI Chat 573
22 A worked example: Economic Dashboard 574
23 Quick reference and honest notes 575

MCP Servers 576

· Overview — what the MCP Servers tab is 578
1 Anatomy — the header, three views & status bar 579
2 MARKETPLACE — the curated catalog 580
3 INSTALLED — manage your servers (cards, logs, remove) 582
4 TOOLS — enable / disable individual tools 584
5 Adding a server — step by step (with dialogs) 586
6 How servers connect — model & transport 587
7 Persistence & security 588
8 Driving this tab from the AI agent (14 tools) 588
9 Quick reference & important notes 589

Code 590

01 What the Code tab is 592
02 Anatomy & the two views 593
03 The Notebook Library 594
04 The Editor 595
05 Cells & running code 596
06 Notebooks, saving & files 598
07 Quick reference — “How do I…” 599

Report Builder 600

· Overview — what the Report Builder tab is 602
1 The workspace at a glance 603
2 The Home view — three ways to start 605
3 The template library 606
4 Editing on the canvas 607
5 The block catalogue 609
6 Data blocks & charts 610
7 Document setup — metadata, branding & themes 612
8 Export & the PDF 613
9 The AI & MCP agent tools 614
10 Storage, autosave & cloud sync 615
11 Quick reference & honest caveats 616

Excel 617

01 What the Excel tab is 619
02 Anatomy of the tab 620
03 The toolbar 621
04 Entering data: the grid and the fx bar 622
05 Formulas — functions and arithmetic 623
06 Formula limits you should know 624
07 Sheets — add, rename, delete 625
08 The right-click menu 626
09 Import and export 627
10 What is saved, and what is not 628
11 Driving the sheet from AI Chat 629
12 Quick reference and honest notes 630

Notes 631

01 What the Notes tab is 633
02 Anatomy of the tab 634
03 The category sidebar 635
04 The notes list 636
05 Reading a note 637
06 Writing a note 638
07 The metadata model 639
08 Exporting a note as markdown 640
09 Favourite, archive and delete 641
10 Where notes are stored 642
11 Driving notes from AI Chat 643
12 Quick reference and honest notes 644

File Manager 645

01 What the File Manager is 647
02 Anatomy of the tab 648
03 The header and the storage bar 649
04 Finding a file: search, sort, source chips 650
05 The file card and its four buttons 651
06 The preview panel 652
07 Selecting and deleting 653
08 The eleven producing screens 654
09 Storage, quota and what is on disk 655
10 Driving it from AI Chat 656
11 Quick reference and honest notes 657

Settings 658

01 What the Settings screen is 660
02 Anatomy — the rail and the two scopes 661
03 General 662
04 Appearance 663
05 Notifications — the fifteen providers 664
06 Notifications — alert triggers 665
07 Keybindings 666
08 Voice — speech-to-text 667
09 Voice — speech, keys and the wake trigger 668
10 Logging — levels and the log file 669
11 Logging — shipping logs to the backend 670
12 Developer 671
13 Profiles 672
14 Credentials 673
15 Security 674
16 Data Sources 675
17 MCP Servers 676
18 Python Env 677
19 Storage — disk usage and categories 678
20 Storage — files, state and the SQL console 679
21 Cloud Sync 680
22 Where each setting is stored 681
23 Quick reference and honest notes 682

Profile 684

01 What the Profile tab is 686
02 Anatomy of the tab 687
03 The header bar 688
04 The ACCOUNT panel 689
05 Credits and usage 690
06 The two usage breakdowns 691
07 The credit ledger 692
08 Edit profile 693
09 Changing your password 694
10 Subscription and billing 695
11 Devices and seats 696
12 Logout, refresh and what is stored 697
13 Driving it from AI Chat 698
14 Quick reference and honest notes 699

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Dashboard

A complete, screen-by-screen walkthrough of the Dashboard — your at-a-glance command center of live market widgets, a market-pulse rail and a streaming ticker, all on one screen.

`>_ 36 widgets · market pulse · ticker tape`

Covers · **the 7 screen regions** · **working with widgets** · **all 36 widgets** **the Market Pulse rail** · the ticker tape · board controls & states

Demonstrated live on the Fincept Terminal Dashboard.

Contents

9 SECTIONS

GETTING ORIENTED

01 What the Dashboard is 3

02 Anatomy of the screen 4

03 The top bar & navigation 5

04 Board controls (the toolbar) 6

WIDGETS & THE BOARD

05 Working with widgets 7

06 Widget catalog — all 36 widgets 8

07 The Market Pulse rail 23

08 Ticker tape 24

REFERENCE

09 Quick reference — “How do I…” 25

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 and tiles shown reflect a single live moment.

OVERVIEW

What the Dashboard is

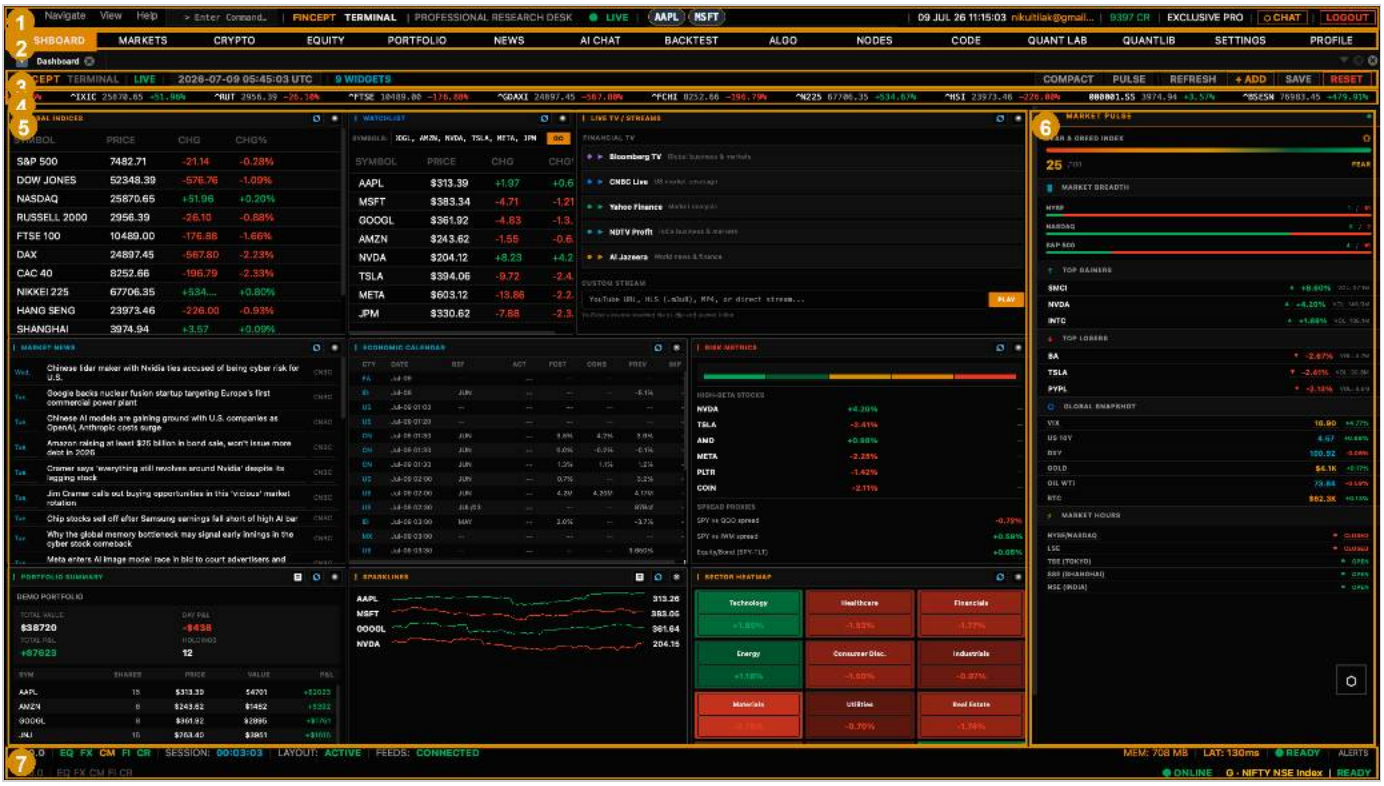
The Dashboard is the terminal's home screen. It is a **grid you build yourself**: each tile is a live "widget" — quotes, news, a chart, your portfolio, a TV stream — that updates on its own. Alongside the grid sits a fixed **Market Pulse rail** summarizing the whole market, a **ticker tape** scrolling your symbols, and a **status bar** reporting feed health.

Nothing about the layout is fixed. You can add, remove, move, resize and configure tiles, then save the arrangement — the terminal remembers it for next time.

At a glance, you can:

- **Build a custom layout** — add from 36 widget types, drag to rearrange, resize on a grid.
- **Track live markets** — indices, watchlists, sparklines, a sector heatmap and risk metrics, all streaming.
- **Follow the news & the economy** — a market-news feed and an economic-events calendar.
- **Watch financial TV** — Bloomberg, CNBC, Yahoo Finance and more, or paste your own stream.
- **Monitor your portfolio** — value, day and total P&L, and positions for any portfolio you own.
- **Read the whole market** — sentiment, breadth, top movers, global snapshot and exchange hours in the Pulse rail.
- **Save, reset and switch layouts** — keep your board or start from a template; toggle a compact density.

Anatomy of the screen



#	REGION	WHAT IT IS FOR
1	**Top bar**	App menus, command bar, pinned symbols, and your account, credits, plan & sign-out.
2	**Tab bar**	The 15 top-level workspaces; ~40 more screens open from the Navigate menu.
3	**Dashboard toolbar**	Controls for the board itself — density, pulse rail, refresh, add, save, reset.
4	**Ticker tape**	A scrolling marquee of your symbols with live price and % change.
5	**Widget grid**	Your customizable tiles — the heart of the Dashboard.
6	**Market Pulse rail**	A fixed, whole-market overview on the right.
7	**Status bar**	Feed lights, session timer, memory, latency and connectivity.

REGION 1 & 2

The top bar & navigation

The persistent bar across the top of every screen — identity, navigation and your account.



ELEMENT	WHAT IT DOES
File / Navigate / View / Help	App menus. **Navigate** jumps to any of the terminal's screens (a lock icon marks features your plan doesn't include). **View** has fullscreen, focus mode, float-panel and screenshot.
Command bar	Type a screen name or command to jump straight there — e.g. `markets`, `economics`.
Pinned symbols	Drag any ticker here to pin it. **Click** a chip to broadcast that symbol to linked panels; **drag** it onto a panel; **right-click** to copy or remove.
Account & credits	Your signed-in email and metered API credit balance (`CR`) — green with credits, red at zero.
Plan badge	Your subscription tier. Click to open Plans & Pricing.
CHAT	Opens the in-app AI assistant over the current screen (also `F9`).
Tab bar	Dashboard, Markets, Crypto, Equity, Portfolio, News, AI Chat, Backtest, Algo, Nodes, Code, Quant Lab, QuantLib, Settings, Profile.

REGION 3

Board controls (the toolbar)

The row of buttons that manage the board as a whole.



Left: live status line. Right: the board action buttons.

The left of the toolbar shows a live status line — **LIVE / OFFLINE** feed state, a clock (click it to switch UTC ⇄ local time), and a running count of how many widgets are on the board. The buttons on the right act on the whole layout:

BUTTON	WHAT IT DOES
COMPACT	Toggles a denser grid (shorter tile rows) so you can fit more on screen.
PULSE	Shows or hides the Market Pulse rail on the right.
REFRESH	Force-refreshes every live source on the board at once — ticker, rail and all tiles.
+ ADD	Opens the widget picker to add a new tile (see the Widget catalog).
SAVE	Saves the current layout. (The board also auto-saves shortly after any change.)
RESET	Clears the board and lets you start from a ready-made template.

> **Your layout is remembered.** Positions, sizes and each widget's settings are saved on this device and restored the next time you open the Dashboard.

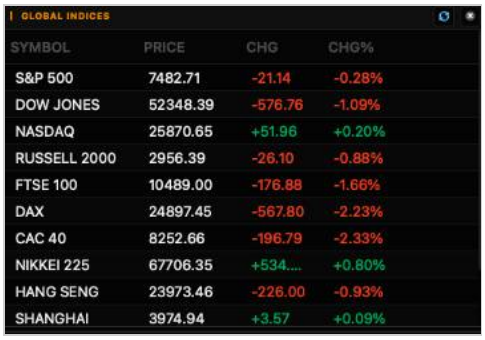
REGION 5 · CORE SKILL

Working with widgets

Every tile behaves the same way — learn it once and it applies to all 36 widgets.

Each widget sits in a tile with a title bar and a few icons in its top-right corner. From there you can:

- **Move it** — drag the title bar; other tiles slide out of the way.
- **Resize it** — drag the bottom-right corner; it snaps to the grid.
- **Refresh it** — the ⟳ icon reloads just that tile.
- **Configure it** — the gear icon (on configurable widgets) opens its settings, e.g. which symbols or which portfolio.
- **Remove it** — the ✕ icon takes it off the board.



A tile: title bar (drag to move) and its ⟳ / ✕ icons.

Adding a widget

Click **+ ADD** to open the widget picker. It shows every widget your plan includes, organized into category tabs (Markets, Research, Portfolio, Trading, Tools, Geopolitics) with a **search box** at the top. Pick a card and click **Add Widget** — it drops onto the board, where you can move and size it.

Saving, resetting & density

- **SAVE** keeps your exact arrangement; it also auto-saves after edits.
- **RESET** lets you swap to a starter template (the default is *Portfolio Manager*, a 7-tile layout).
- **COMPACT** shrinks row height for a denser board; press again to restore.

REGION 5 · THE BUILDING BLOCKS

Widget catalog

All 36 widgets the **+ ADD** picker offers — what each shows, how you use it, and its data source.

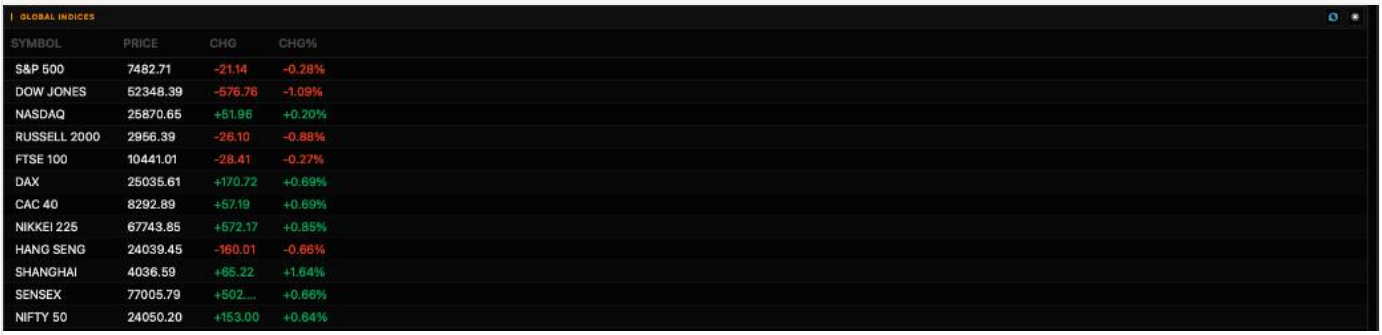
Widgets that need a symbol, a connected broker, or a URL are shown in their default state with a **Setup** note. Everything else is captured live.

Markets

Live prices across every asset class. · 12 widgets

Market Indices

MARKETS



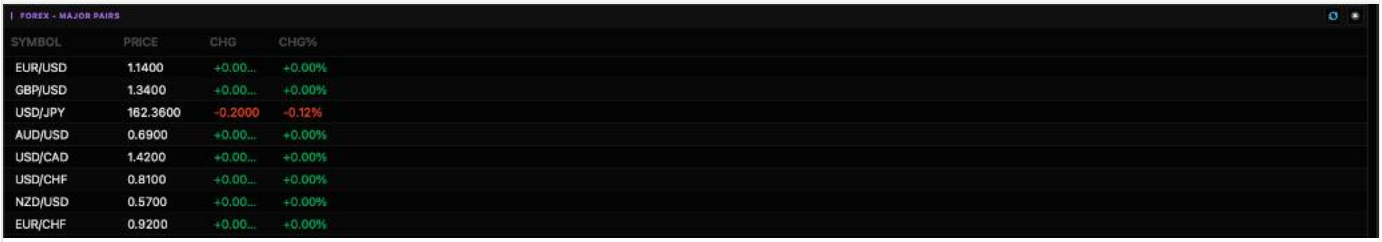
12 major world indices — S&P 500, Dow, Nasdaq, Russell, FTSE, DAX, CAC, Nikkei, Hang Seng, Shanghai, Sensex, Nifty.

Columns: symbol · price · change · % change (green/red).

SOURCE Live market quotes

Forex Pairs

MARKETS

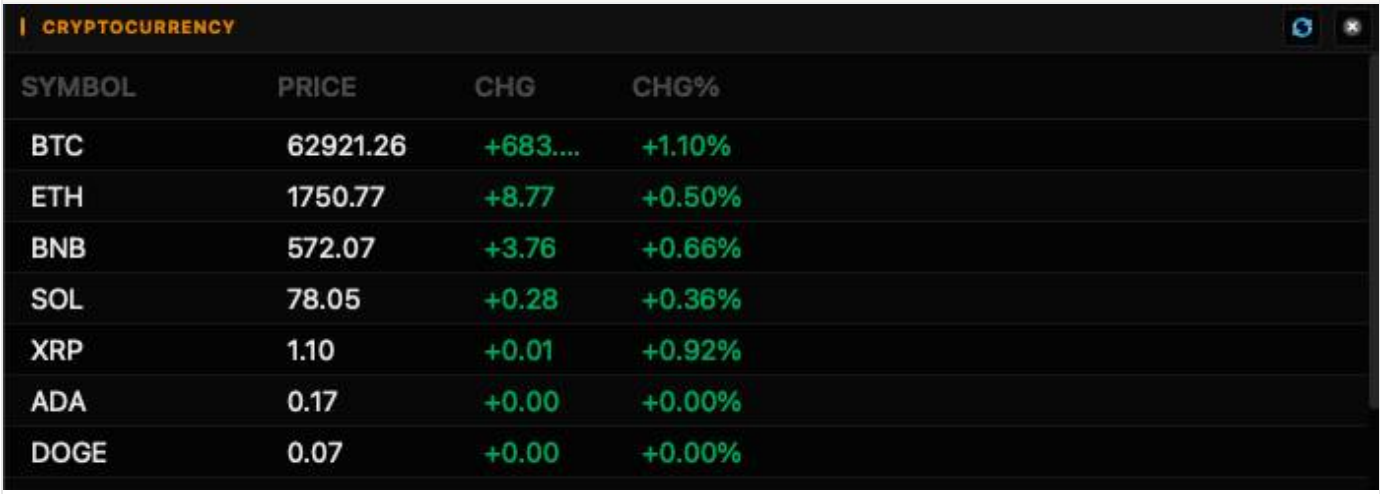


The major FX pairs (EUR/USD, GBP/USD, USD/JPY, AUD/USD, USD/CAD, USD/CHF, NZD/USD, EUR/CHF) with live price and change.

SOURCE Live market quotes

Crypto Markets

MARKETS

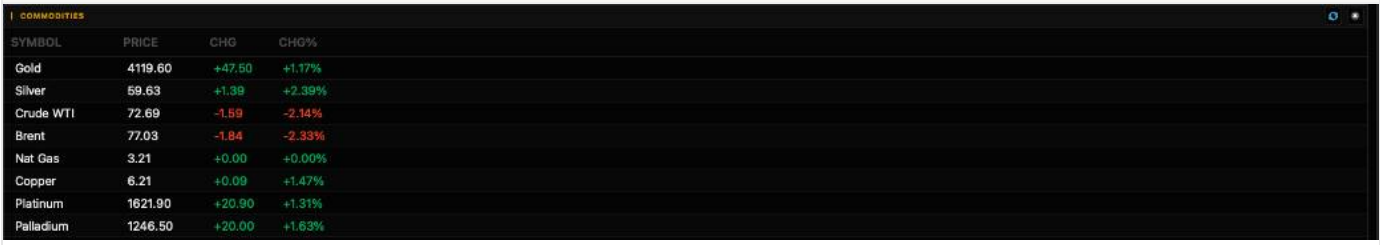


Top cryptocurrencies — BTC, ETH, BNB, SOL, XRP, ADA, DOGE, LTC — with price and % change.

SOURCE Live market quotes

Commodities

MARKETS



Gold, Silver, Crude WTI, Brent, Nat Gas, Copper, Platinum and Palladium with price and change.

SOURCE Live market quotes

Sector Heatmap

MARKETS



The 11 S&P sectors (plus Semis) as tiles, coloured green/red and shaded more strongly for bigger moves — a one-glance read of what's leading and lagging.

SOURCE Sector ETF quotes

Top Movers

MARKETS



The day's biggest gainers and losers with price and % change — toggle between the two.

SOURCE Live market quotes

Market Sentiment

MARKETS



A bull/bear gauge with a scored reading, advancer/decliner counts and the VIX — a quick market-mood snapshot.

SOURCE Computed from a large-cap basket + VIX

Quote Strip

MARKETS



A compact strip of quotes for a symbol list you choose.

SOURCE Live market quotes

> **Setup:** Pick your symbols with the gear icon.

Crypto Ticker

MARKETS



A live crypto ticker strip sourced from Kraken / HyperLiquid, for a configurable list of pairs.

SOURCE Kraken / HyperLiquid

> **Setup:** Configure the pair list (gear) to populate.

Polymarket

MARKETS



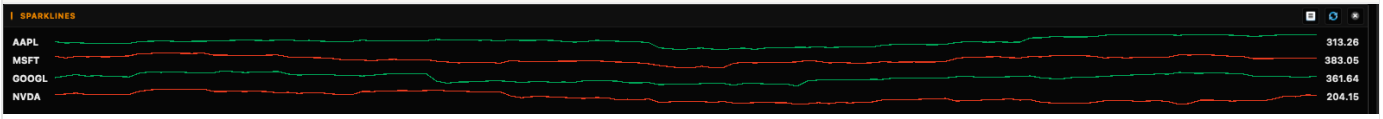
Live prices from Polymarket prediction markets.

SOURCE Polymarket

> **Setup:** Add the markets you want to track via the gear.

Sparklines

MARKETS



A mini 5-day trend line per symbol with its last price — green if up, red if down.

SOURCE 5-day hourly closes

> **Setup:** Choose which symbols via the gear icon.

Trade Tape

MARKETS



A streaming feed of individual trade prints as they happen.

SOURCE Live trade feed

> **Setup:** Configure a symbol/source to stream prints.

Research

News, the economy, and single-name analysis. · 7 widgets

News Feed

RESEARCH

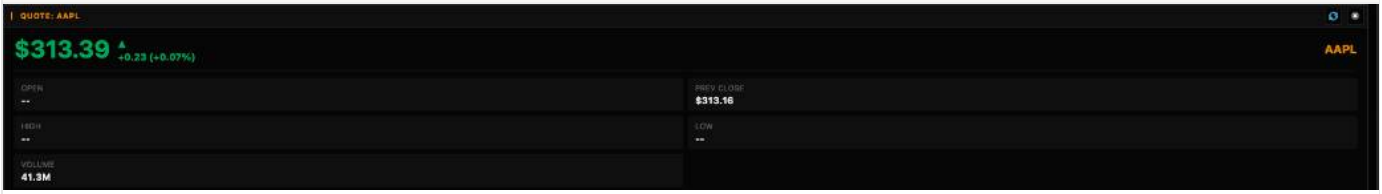


Aggregated market headlines from major sources (Bloomberg, MarketWatch, CNBC, BBC, FT, WSJ, AP…), newest first, each tagged with its source.

SOURCE RSS news feeds

Stock Quote

RESEARCH



One symbol in detail — last price with change, open, previous close, high, low and volume.

SOURCE Live market quotes

> **Setup:** Set the symbol via the gear icon.

Candle Chart

RESEARCH



An interactive candlestick price chart for a chosen symbol.

SOURCE Historical candles

> **Setup:** Set the symbol via the gear icon.

Stock Screener

RESEARCH



Rank and filter a universe of stocks by change, price, volume and more — sortable columns.

SOURCE Live market quotes

Economic Calendar

RESEARCH



Upcoming macro releases with country, date, and actual / forecast / consensus / previous values plus an importance rating.

SOURCE Macro-events API

News — Category

RESEARCH



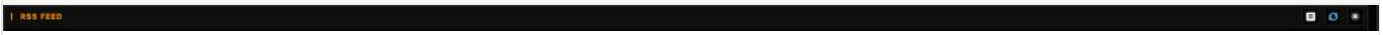
Headlines narrowed to a single topic/category you choose.

SOURCE RSS news feeds

> **Setup:** Pick a category to load its headlines.

RSS Feed

RESEARCH



Headlines from any RSS source you point it at.

SOURCE Any RSS URL

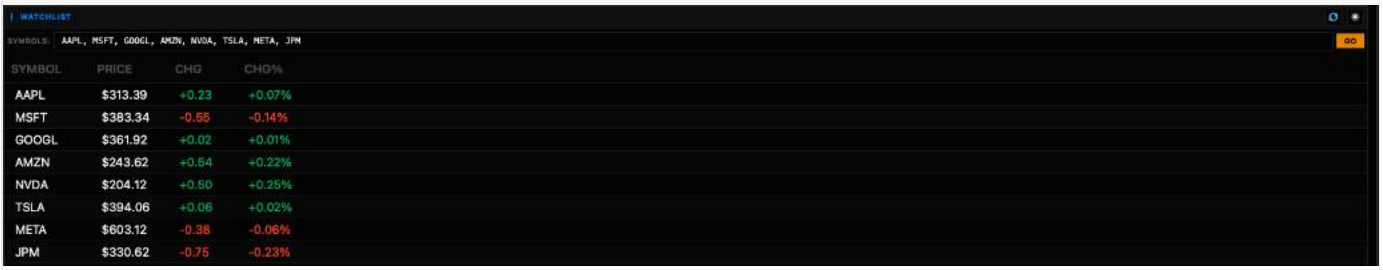
> **Setup:** Add a feed URL via the gear icon.

Portfolio

Your holdings, performance and risk. · 5 widgets

Watchlist

PORTFOLIO

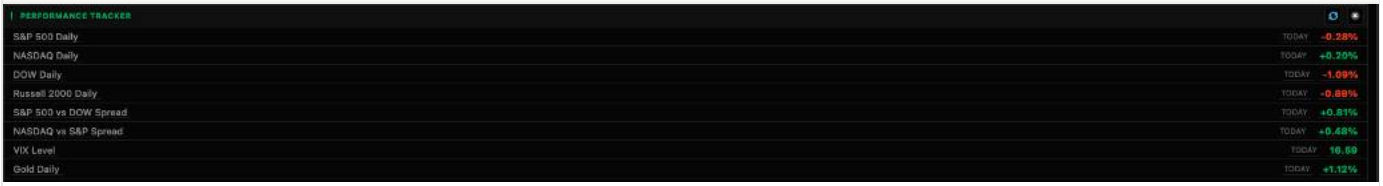


An editable quote table — type comma-separated symbols and press GO to track your own list.

SOURCE Live market quotes

Performance

PORTFOLIO

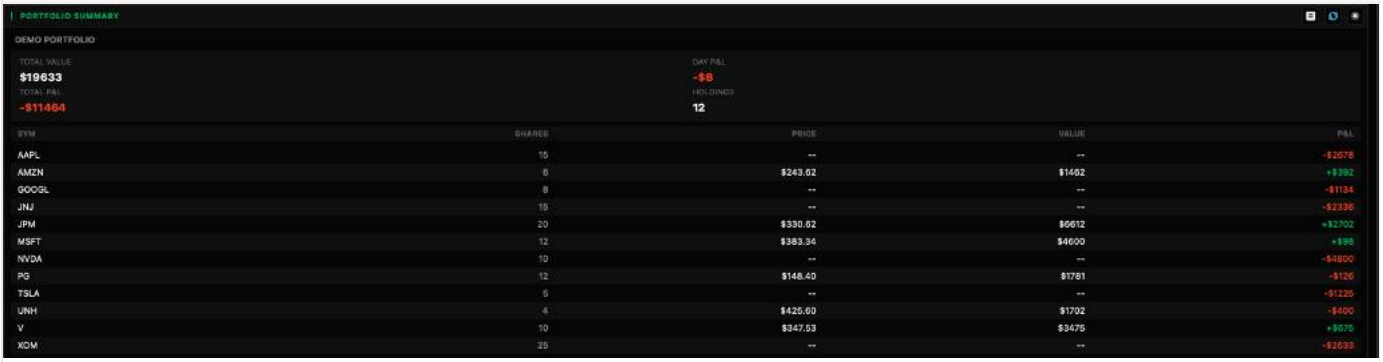


Performance readings across benchmarks — S&P/Nasdaq/Dow daily, index spreads, the VIX level and gold.

SOURCE Live market quotes

Portfolio Summary

PORTFOLIO



One portfolio's snapshot — total value, day P&L, total P&L, holdings count and a positions table.

SOURCE Your portfolio + live prices

> **Setup:** Pick which portfolio via the gear icon.

Risk Metrics

PORTFOLIO



A VIX fear gauge, high-beta stocks with their moves, and key spread proxies (SPY-QQQ, SPY-IWM, equity vs bonds).

SOURCE Live market quotes

Holdings

PORTFOLIO



Your long-term broker holdings with average cost, last price and P&L.

SOURCE Connected broker

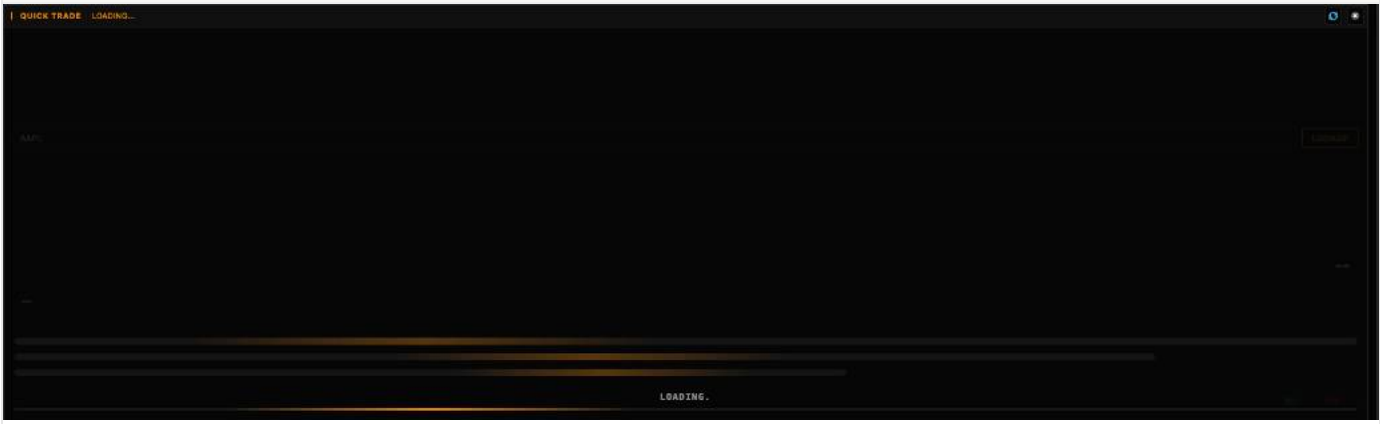
> **Setup:** Connect a broker (Settings → Credentials) to populate.

Trading

Order entry and live trading state. · 5 widgets

Quick Trade

TRADING



A fast order ticket to place buy/sell orders without leaving the board.

SOURCE Connected broker

Open Positions

TRADING



Your live open positions for a broker, with quantity, price and P&L.

SOURCE Connected broker

> **Setup:** Connect a broker to populate.

Working Orders

TRADING



Resting / unfilled orders for a broker.

SOURCE Connected broker

> **Setup:** Connect a broker to populate.

Margin Usage

TRADING

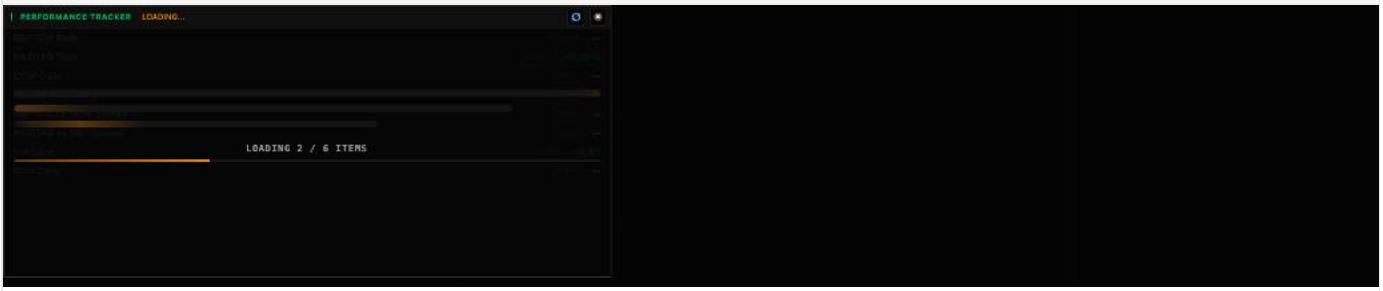


Margin used versus available for a broker account.

SOURCE Connected broker

Today P&L

TRADING



Realized and unrealized profit & loss for today.

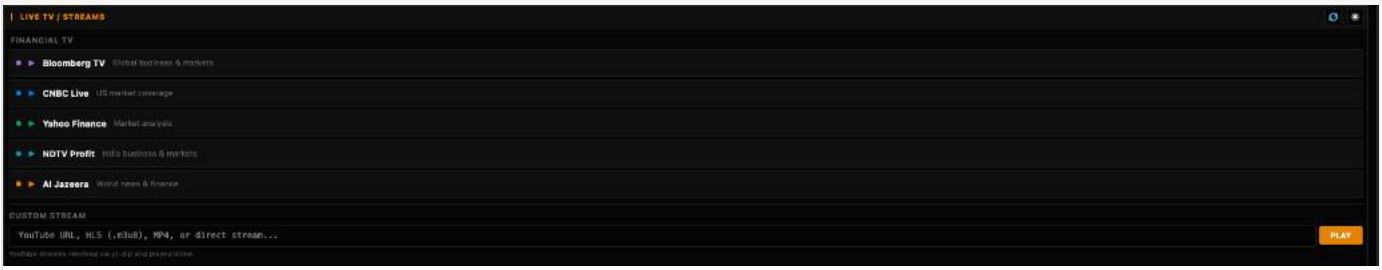
SOURCE Connected broker / portfolio

Tools

Media, files and utilities. · 5 widgets

Live TV / Streams

TOOLS



Financial TV played inside the terminal — Bloomberg, CNBC, Yahoo Finance, NDTV Profit, Al Jazeera — plus a box to paste any YouTube, HLS or MP4 stream.

SOURCE YouTube / HLS (via yt-dlp)

Recent Files

TOOLS



Your recently opened files and notebooks, with timestamps — click to reopen.

SOURCE Local file history

Agent Errors

TOOLS



A running log of AI-agent execution errors. An empty list means everything is healthy.

SOURCE AI agent runtime

Web Scraper

TOOLS



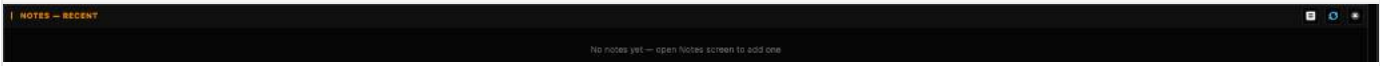
Pull structured data from a web page you point it at.

SOURCE Any web URL

> **Setup:** Configure the target URL via the gear icon.

Notes

TOOLS



A free-text scratchpad pinned to your board for quick jottings.

SOURCE Local

> **Setup:** Click in and start typing.

Geopolitics

World events and maritime tracking. · 2 widgets

Geopolitics Events

GEOPOLITICS



Live geopolitical and conflict events with category, region and date.

SOURCE Geopolitics feed

Maritime Vessels

GEOPOLITICS



Vessel tracking — names, type and position/status.

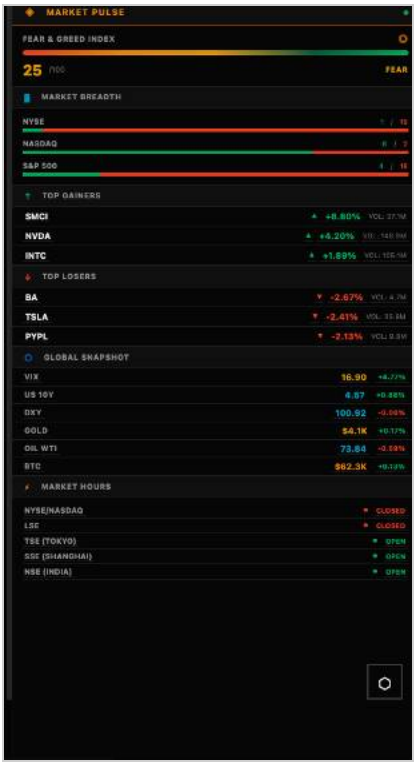
SOURCE Maritime feed

> **Setup:** Configurable via the gear icon.

REGION 6

The Market Pulse rail

A fixed, whole-market overview — not a widget. Toggle it with the toolbar's PULSE button.



SECTION	WHAT IT SHOWS
Fear & Greed	A 0–100 sentiment score and label.
Market Breadth	Advancers vs decliners for NYSE, NASDAQ and the S&P.
Top Gainers	The day's three strongest movers, with volume.
Top Losers	The day's three weakest movers.
Global Snapshot	VIX, US 10-year, the dollar (DXY), gold, WTI oil and Bitcoin.
Market Hours	Open / closed status for NYSE, London, Tokyo, Shanghai and India.

> **Good to know:** Fear & Greed, Market Breadth and Market Hours are **calculated inside the terminal** from live quotes and the clock — quick indicators, not official exchange figures.

REGION 4 & 7

Ticker tape



A continuously scrolling band of your symbols, each with live price and % change (green up, red down). By default it shows major indices and movers. **Right-click the tape and choose "Edit Symbols…"** to set your own comma-separated list — it's saved for next time.

Status bar



SEGMENT	WHAT IT MEANS
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 widgets, EMPTY when it doesn't.
FEEDS	CONNECTED / DISCONNECTED data-feed 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	Connection to Fincept — ONLINE, RECONNECTING or OFFLINE.
G · symbol	The symbol currently shared across linked panels (set by clicking a pinned chip).

QUICK REFERENCE

How do I…

I WANT TO…	DO THIS
Add a new widget	Click **+ ADD**, pick a widget, click **Add Widget**.
Move or resize a tile	Drag its **title bar** to move; drag its **bottom-right corner** to resize.
Change a widget's symbols/settings	Click the tile's **gear** icon.
Remove a tile	Click the tile's ✕ icon.
Refresh one tile / everything	The tile's ⟳, or the toolbar's **REFRESH** for the whole board.
Keep my layout	Click **SAVE** (it also auto-saves).
Start over from a template	Click **RESET** and choose a template.
Fit more on screen	Click **COMPACT**.
Hide/show the Pulse rail	Click **PULSE**.
Change the ticker symbols	Right-click the ticker → **Edit Symbols…**
Track my own watchlist	Type symbols in the Watchlist tile and press **GO**.
Watch a specific stream	Paste a URL into **Custom Stream** and press PLAY.
See a specific portfolio	Use the Portfolio Summary tile's **gear** to select it.

About the data

Market prices stream continuously and are briefly cached to stay responsive. Some Pulse-rail figures (Fear & Greed, Breadth, Market Hours) are computed locally as quick indicators rather than vendor feeds. Your Dashboard layout is stored on this device and restored automatically.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Dashboard Widgets

A complete reference to all 36 dashboard widgets across six categories — what each one displays, how you use it, and where its data comes from.

`>_ 36 widgets · 6 categories · live tiles`

Covers · **Markets** · **Research** · **Portfolio**
Trading · **Tools** · **Geopolitics** · every tile documented

Demonstrated live on the Fincept Terminal Dashboard.

Contents

6 CATEGORIES · 36 WIDGETS

WIDGET CATEGORIES

01 Markets — indices, crypto, heatmap, sentiment, Polymarket… 3

02 Research — candle chart, economic calendar, news… 8

03 Portfolio — summary, holdings, positions… 11

04 Trading — margin usage, orders, trade tape… 13

05 Tools — web scraper, notes, RSS, agent errors… 15

06 Geopolitics — event map & feeds 17

ABOUT THIS GUIDE

Every widget shown is a real capture of the running terminal — added, sized and populated live. Control labels and data sources are described exactly as they behave.

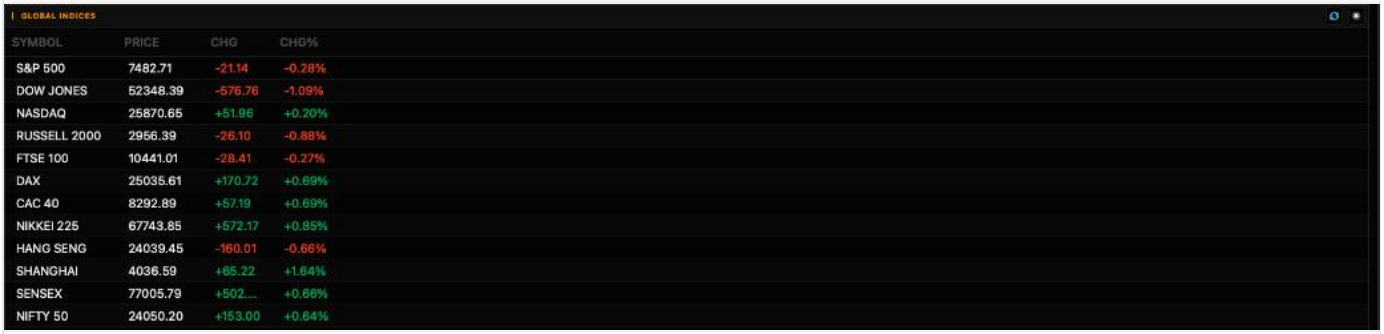
CATEGORY

Markets

Live prices across every asset class. · 12 widgets

Market Indices

MARKETS



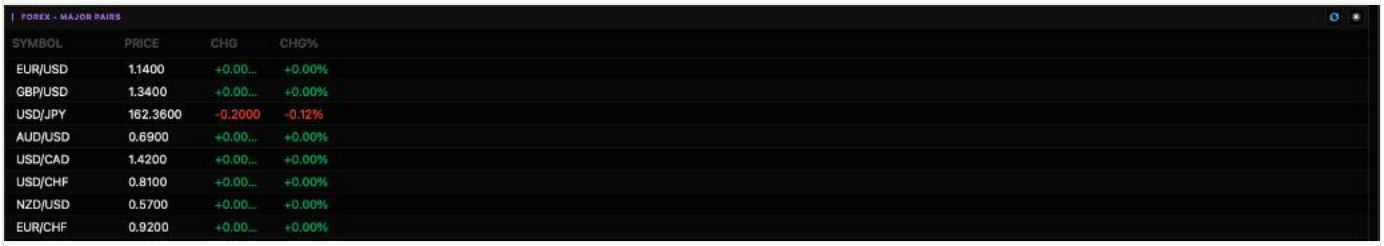
12 major world indices — S&P 500, Dow, Nasdaq, Russell, FTSE, DAX, CAC, Nikkei, Hang Seng, Shanghai, Sensex, Nifty.

Columns: symbol · price · change · % change (green/red).

SOURCE Live market quotes

Forex Pairs

MARKETS

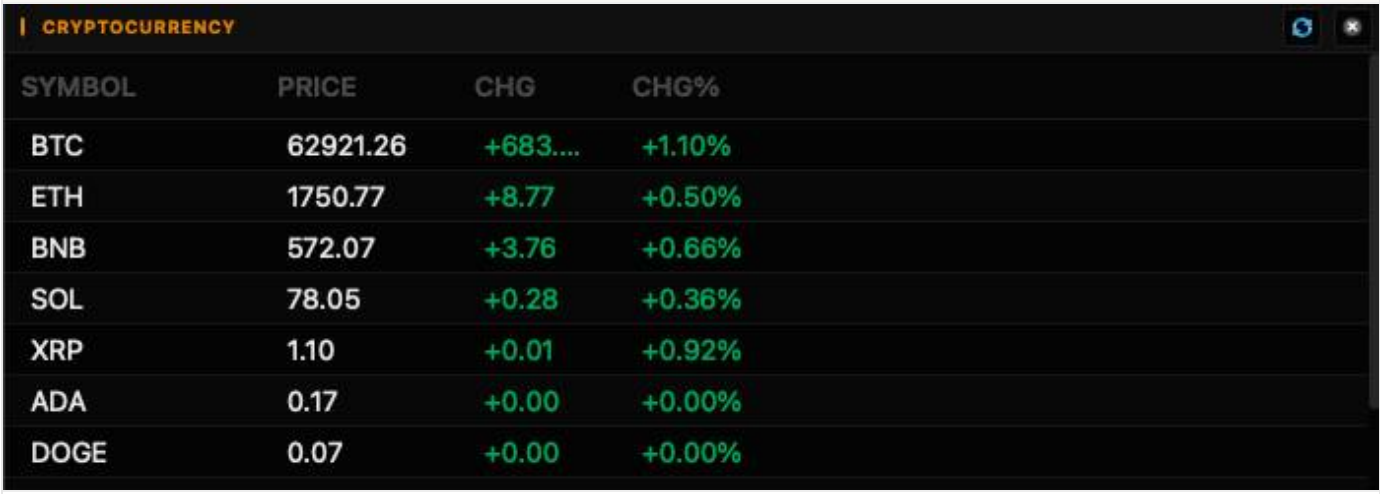


The major FX pairs (EUR/USD, GBP/USD, USD/JPY, AUD/USD, USD/CAD, USD/CHF, NZD/USD, EUR/CHF) with live price and change.

SOURCE Live market quotes

Crypto Markets

MARKETS

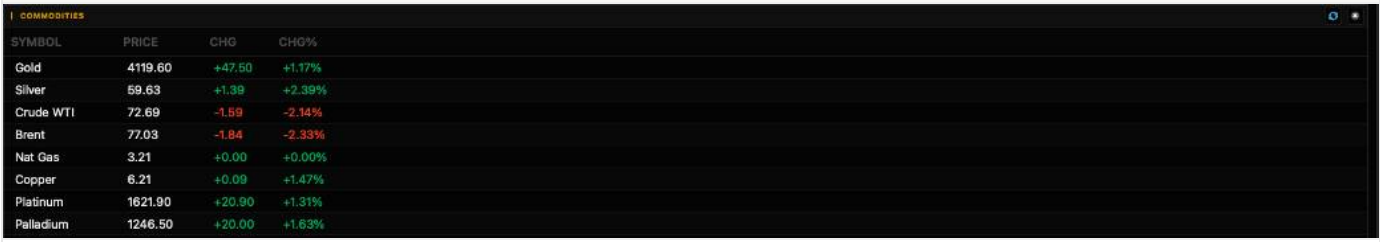


Top cryptocurrencies — BTC, ETH, BNB, SOL, XRP, ADA, DOGE, LTC — with price and % change.

SOURCE Live market quotes

Commodities

MARKETS



Gold, Silver, Crude WTI, Brent, Nat Gas, Copper, Platinum and Palladium with price and change.

SOURCE Live market quotes

Sector Heatmap

MARKETS



The 11 S&P sectors (plus Semis) as tiles, coloured green/red and shaded more strongly for bigger moves — a one-glance read of what's leading and lagging.

SOURCE Sector ETF quotes

Top Movers

MARKETS



The day's biggest gainers and losers with price and % change — toggle between the two.

SOURCE Live market quotes

Market Sentiment

MARKETS



A bull/bear gauge with a 0-scored reading, advancer/decliner counts and the VIX — a quick market-mood snapshot.

SOURCE Computed from a large-cap basket + VIX

Quote Strip

MARKETS



A compact strip of quotes for a symbol list you choose.

SOURCE Live market quotes

> **Setup:** Pick your symbols with the gear icon.

Crypto Ticker

MARKETS



A live crypto ticker strip sourced from Kraken / HyperLiquid, for a configurable list of pairs.

SOURCE Kraken / HyperLiquid

> **Setup:** Configure the pair list (gear) to populate.

Polymarket

MARKETS



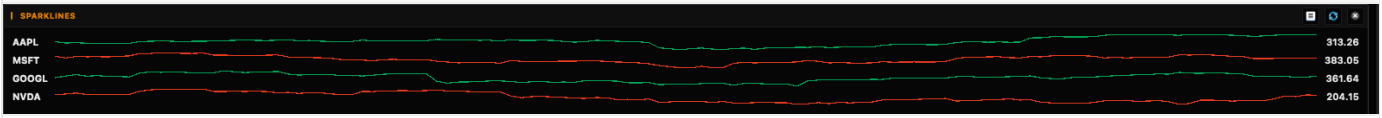
Live prices from Polymarket prediction markets.

SOURCE Polymarket

> **Setup:** Add the markets you want to track via the gear.

Sparklines

MARKETS



A mini 5-day trend line per symbol with its last price — green if up, red if down.

SOURCE 5-day hourly closes

> **Setup:** Choose which symbols via the gear icon.

Trade Tape

MARKETS



A streaming feed of individual trade prints as they happen.

SOURCE Live trade feed

> **Setup:** Configure a symbol/source to stream prints.

CATEGORY

Research

News, the economy, and single-name analysis. · 7 widgets

News Feed

RESEARCH

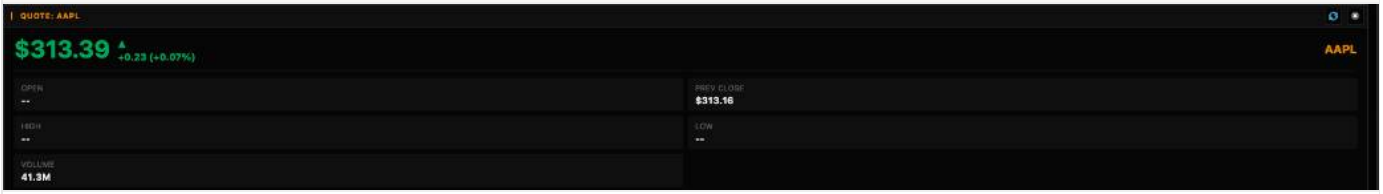


Aggregated market headlines from major sources (Bloomberg, MarketWatch, CNBC, BBC, FT, WSJ, AP…), newest first, each tagged with its source.

SOURCE RSS news feeds

Stock Quote

RESEARCH



One symbol in detail — last price with change, open, previous close, high, low and volume.

SOURCE Live market quotes

> **Setup:** Set the symbol via the gear icon.

Candle Chart

RESEARCH



An interactive candlestick price chart for a chosen symbol.

SOURCE Historical candles

> **Setup:** Set the symbol via the gear icon.

Stock Screener

RESEARCH



Rank and filter a universe of stocks by change, price, volume and more — sortable columns.

SOURCE Live market quotes

Economic Calendar

RESEARCH



Upcoming macro releases with country, date, and actual / forecast / consensus / previous values plus an importance rating.

SOURCE Macro-events API

News — Category

RESEARCH



Headlines narrowed to a single topic/category you choose.

SOURCE RSS news feeds

> **Setup:** Pick a category to load its headlines.

RSS Feed

RESEARCH



Headlines from any RSS source you point it at.

SOURCE Any RSS URL

> **Setup:** Add a feed URL via the gear icon.

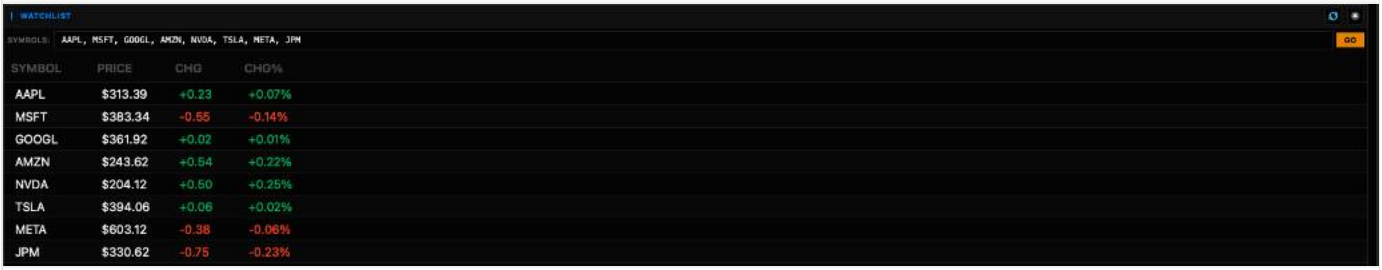
CATEGORY

Portfolio

Your holdings, performance and risk. · 5 widgets

Watchlist

PORTFOLIO

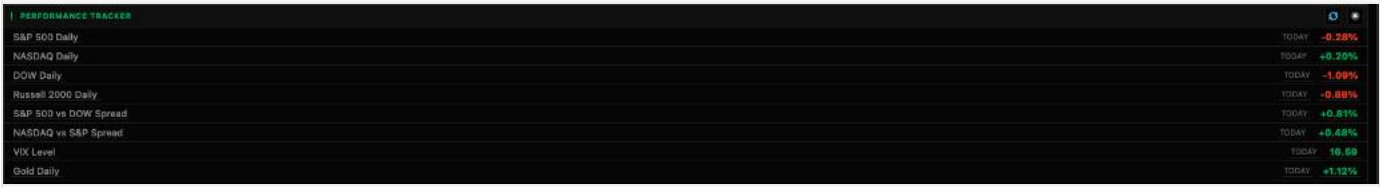


An editable quote table — type comma-separated symbols and press GO to track your own list.

SOURCE Live market quotes

Performance

PORTFOLIO

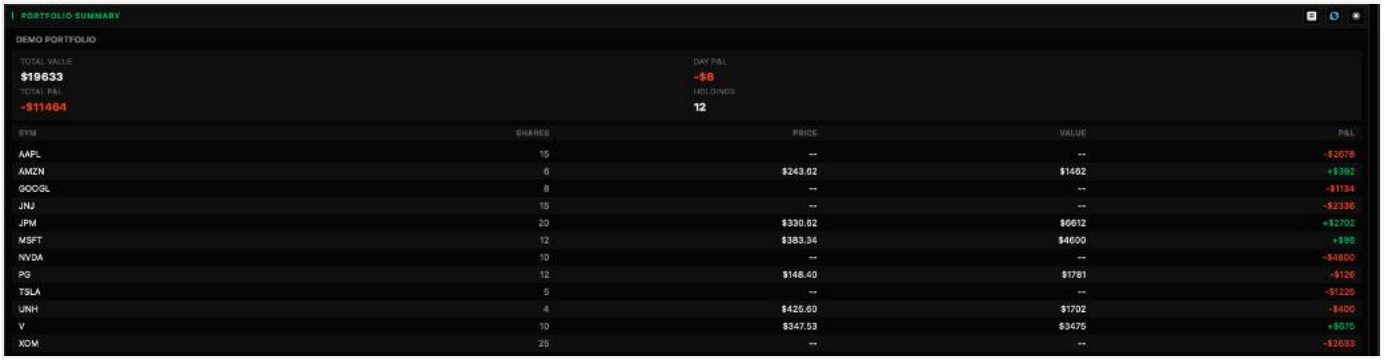


Performance readings across benchmarks — S&P/Nasdaq/Dow daily, index spreads, the VIX level and gold.

SOURCE Live market quotes

Portfolio Summary

PORTFOLIO



One portfolio's snapshot — total value, day P&L, total P&L, holdings count and a positions table.

SOURCE Your portfolio + live prices

> **Setup:** Pick which portfolio via the gear icon.

Risk Metrics

PORTFOLIO



A VIX fear gauge, high-beta stocks with their moves, and key spread proxies (SPY-QQQ, SPY-IWM, equity vs bonds).

SOURCE Live market quotes

Holdings

PORTFOLIO



Your long-term broker holdings with average cost, last price and P&L.

SOURCE Connected broker

> **Setup:** Connect a broker (Settings → Credentials) to populate.

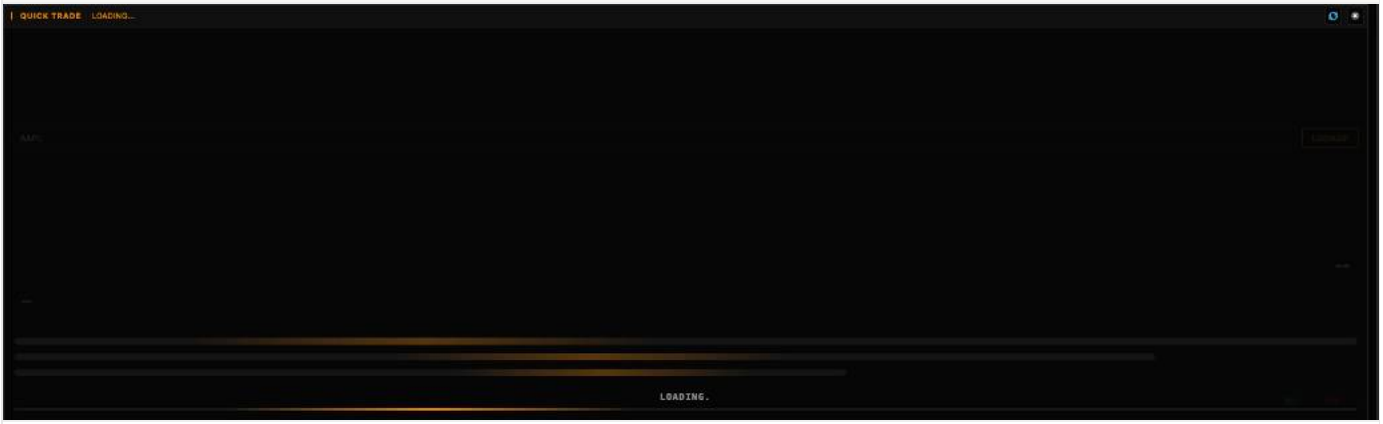
CATEGORY

Trading

Order entry and live trading state. · 5 widgets

Quick Trade

TRADING



A fast order ticket to place buy/sell orders without leaving the board.

SOURCE Connected broker

Open Positions

TRADING



Your live open positions for a broker, with quantity, price and P&L.

SOURCE Connected broker

> **Setup:** Connect a broker to populate.

Working Orders

TRADING



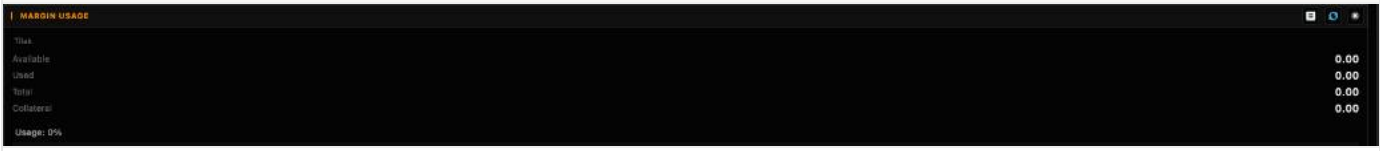
Resting / unfilled orders for a broker.

SOURCE Connected broker

> **Setup:** Connect a broker to populate.

Margin Usage

TRADING

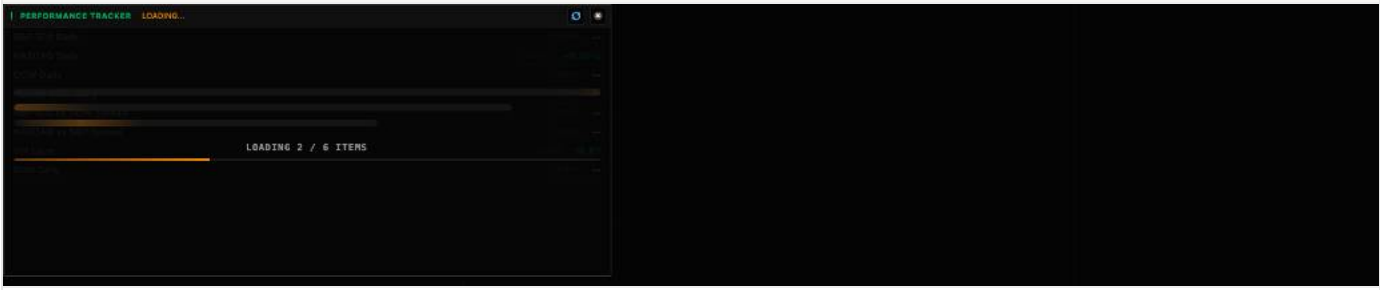


Margin used versus available for a broker account.

SOURCE Connected broker

Today P&L

TRADING



Realized and unrealized profit & loss for today.

SOURCE Connected broker / portfolio

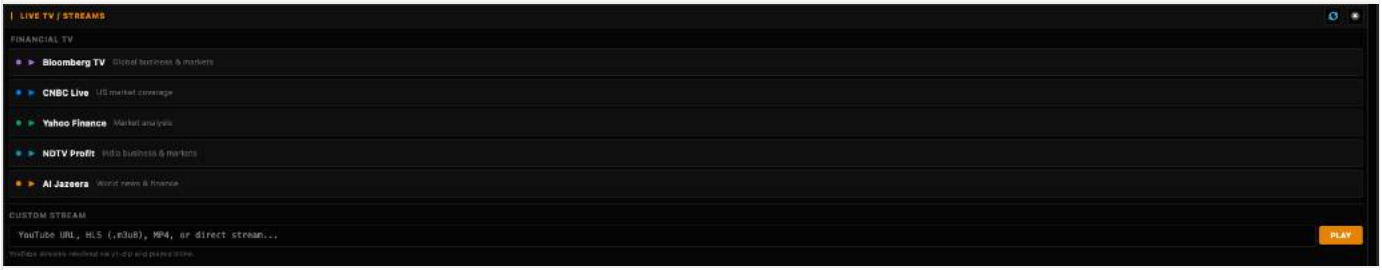
CATEGORY

Tools

Media, files and utilities. · 5 widgets

Live TV / Streams

TOOLS



Financial TV played inside the terminal — Bloomberg, CNBC, Yahoo Finance, NDTV Profit, Al Jazeera — plus a box to paste any YouTube, HLS or MP4 stream.

SOURCE YouTube / HLS (via yt-dlp)

Recent Files

TOOLS



Your recently opened files and notebooks, with timestamps — click to reopen.

SOURCE Local file history

Agent Errors

TOOLS



A running log of AI-agent execution errors. An empty list means everything is healthy.

SOURCE AI agent runtime

Web Scraper

TOOLS



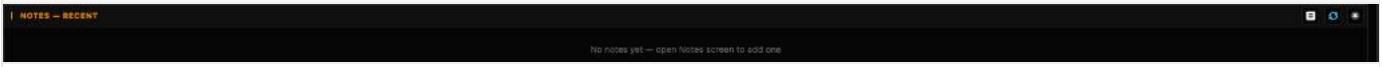
Pull structured data from a web page you point it at.

SOURCE Any web URL

> **Setup:** Configure the target URL via the gear icon.

Notes

TOOLS



A free-text scratchpad pinned to your board for quick jottings.

SOURCE Local

> **Setup:** Click in and start typing.

CATEGORY

Geopolitics

World events and maritime tracking. · 2 widgets

Geopolitics Events

GEOPOLITICS



Live geopolitical and conflict events with category, region and date.

SOURCE Geopolitics feed

Maritime Vessels

GEOPOLITICS



Vessel tracking — names, type and position/status.

SOURCE Maritime feed

> **Setup:** Configurable via the gear icon.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The In-App Documentation

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

`>_ 40 topics · 9 categories · ~200 screenshots · offline`

Covers · **the sidebar tree** · **all 40 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 forty 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 forty 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: **40 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 forty 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.

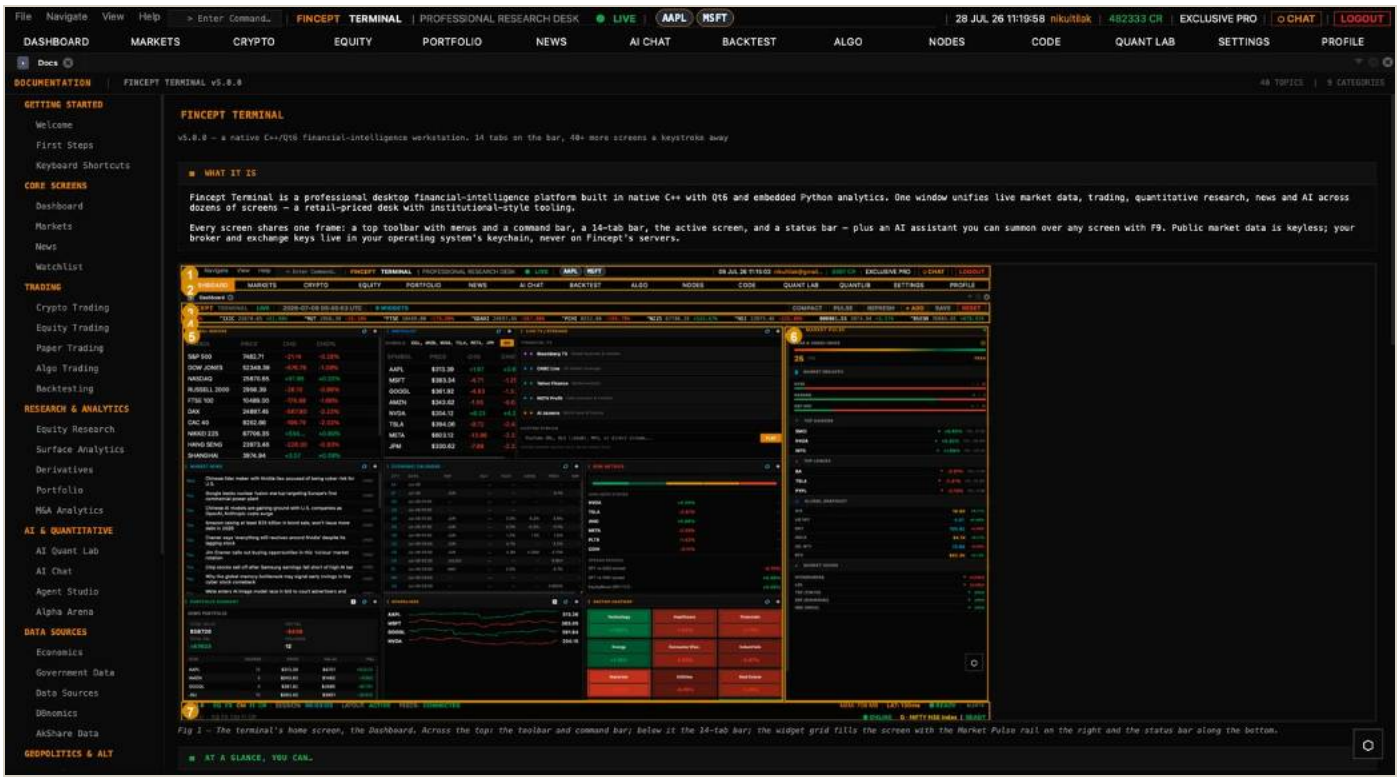


Fig 2 — The Docs tab on open, showing the Welcome topic. The tree is on the left; the page fills the rest of the width.

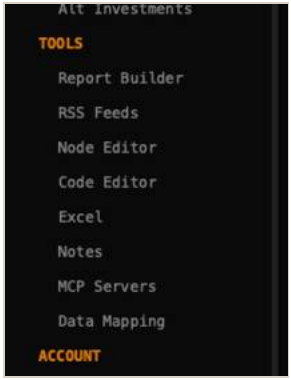


Fig 3 — Scrolling the tree reveals the lower categories: **TOOLS** carries eight topics, including Excel and Notes.

#	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 forty topics

TOPIC TREE · 03

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 (5)	Economics · Government Data · Data Sources · DBnomics · AkShare Data
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

PAGE ANATOMY · 04

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

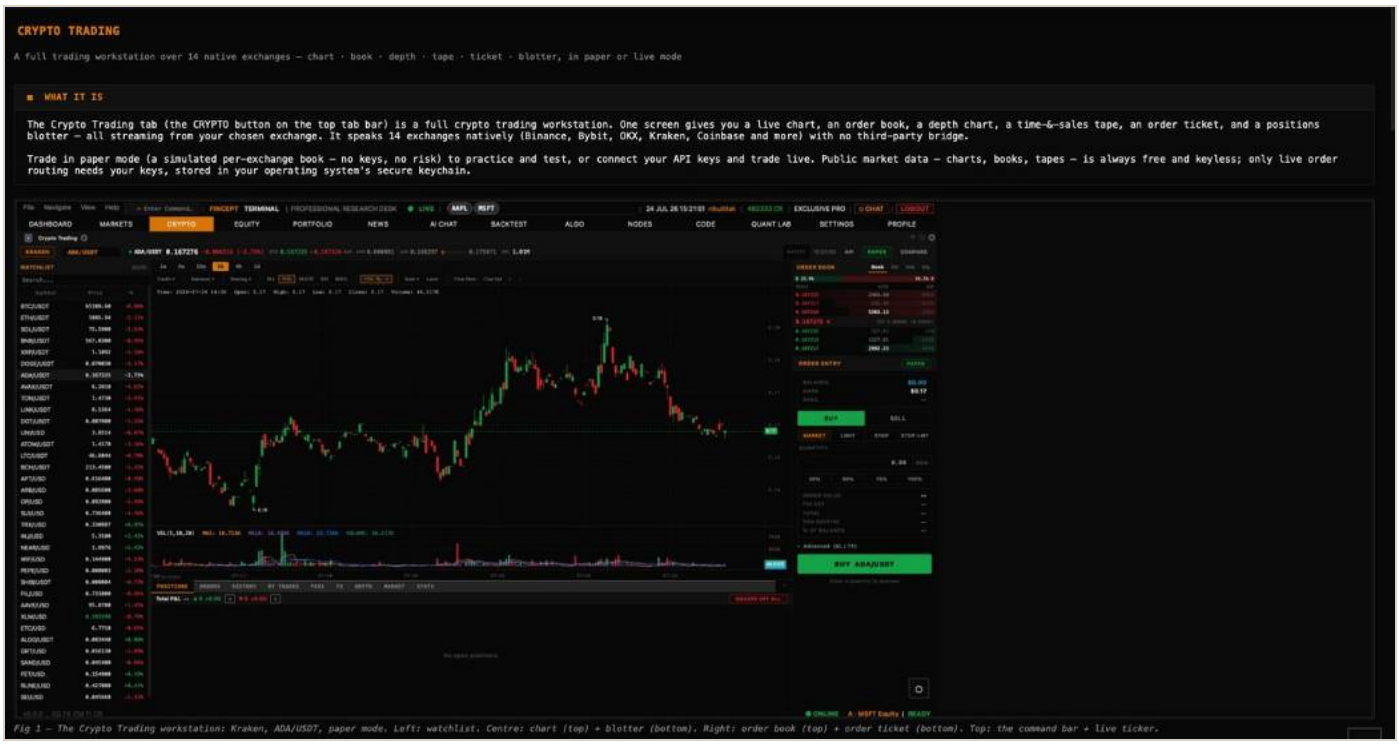
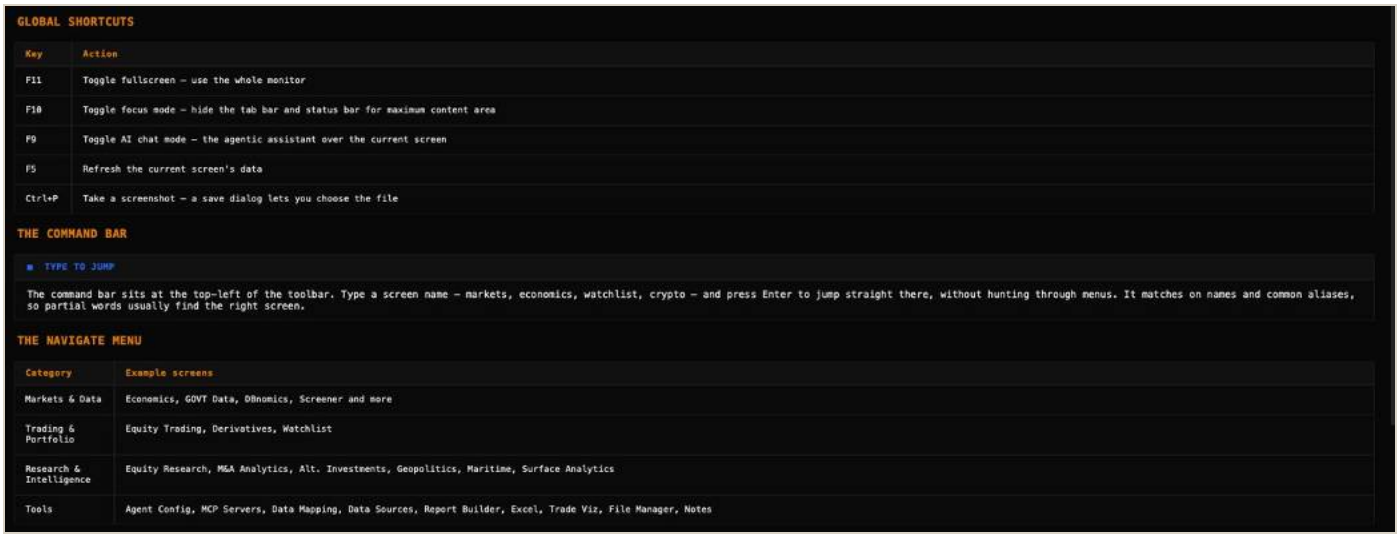


Fig 4 — The Crypto Trading topic: amber title, one-line subtitle, a **WHAT IT IS** panel, then a full-width annotated screenshot with a numbered caption.

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, OBnomics, 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

LOADING · 05

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 forty 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

Forty topics cover thirty-seven screens plus the three orientation pages. 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
`Trade Viz`	**Fincept-Trade-Viz-Guide.pdf**
`Docs`	This guide
`About`	**Fincept-About-Guide.pdf**
`Relationship Map`	No guide yet — the screen is self-describing
`Asia Markets`	No guide yet
`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 forty 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 Polymarket, Government Data is the GOVT Data tab, 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 four tabs the manual does not cover** — File Manager, Trade Viz, 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, Trade Viz, 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**)

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, Trade Viz, Docs, About, Relationship Map, Asia Markets and F&O.
- **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.

> The counts in this guide — forty 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.

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 · licence · crash dumps · contacts`

Covers · **version information** · **Check for Updates** · **the licence** crash dumps · resources and contacts · the sibling legal pages

Demonstrated live on the Fincept Terminal About tab.

Contents

9 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 Licence 8

07 Diagnostics and crash dumps 9

08 Resources and contacts 10

REFERENCE

09 Quick reference and honest notes 11

> **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 05, 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.

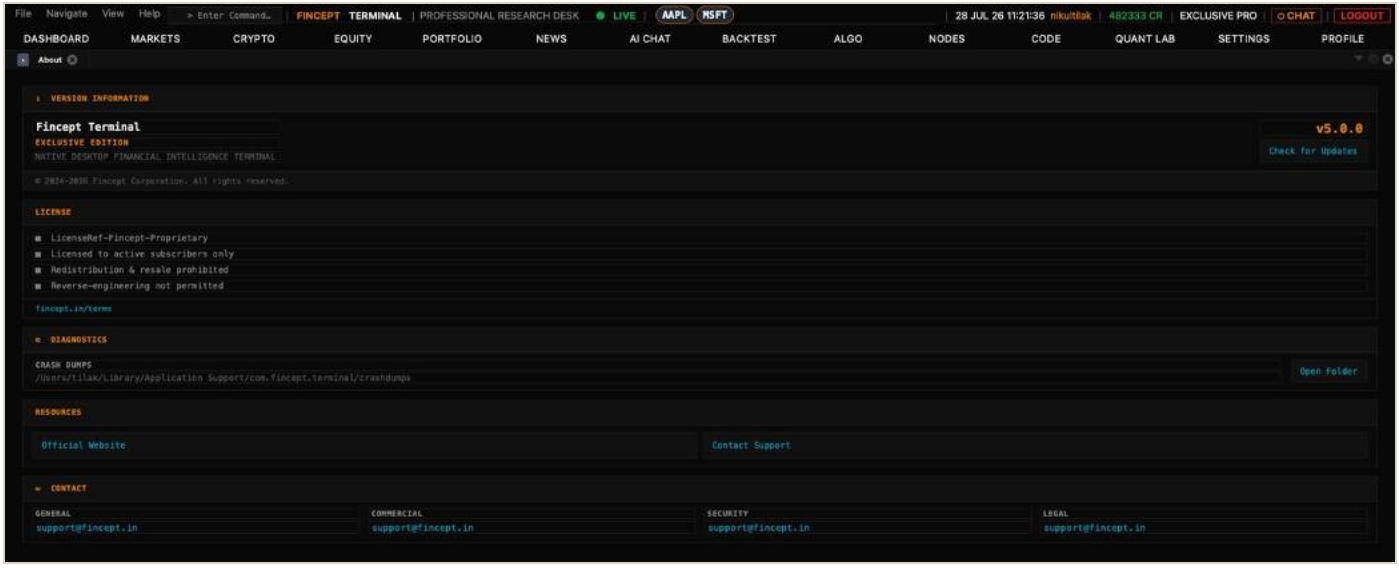


Fig 2 — The whole About tab: version, licence, diagnostics, resources and contact.

#	Panel	What it holds
1	VERSION INFORMATION	Product name, edition badge, one-line description, the build version, **Check for Updates**, and the copyright line. Sections 03–05.
2	LICENCE	The licence identifier and its three practical terms, plus the terms-of-service address. Section 06.
3	DIAGNOSTICS	The crash-dump folder path and an **Open Folder** button. Section 07.
4	RESOURCES	**Official Website** and **Contact Support**. Section 08.
5	CONTACT	Four addressed routes — general, commercial, security and legal. Section 08.

> 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
`v5.0.0`	The build version, in amber. 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
`© 2024–2026 Fincept Corporation. All rights reserved.`	The copyright line, in the panel footer



Fig 3 — The version panel as it appears on a current installation.

> **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 07. 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.

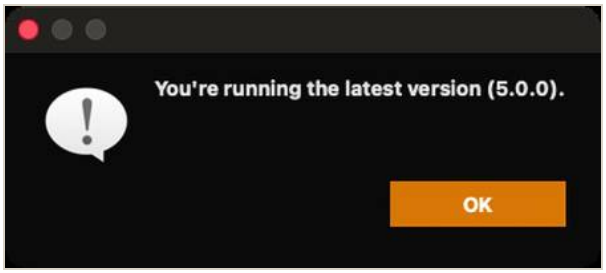


Fig 4 — The result on an up-to-date installation: *You're running the latest version (5.0.0).*

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 Licence

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



Fig 5 — 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.

07 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 6 — 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.

08 Resources and contacts

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



Fig 7 — 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 8 — 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

09 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

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.

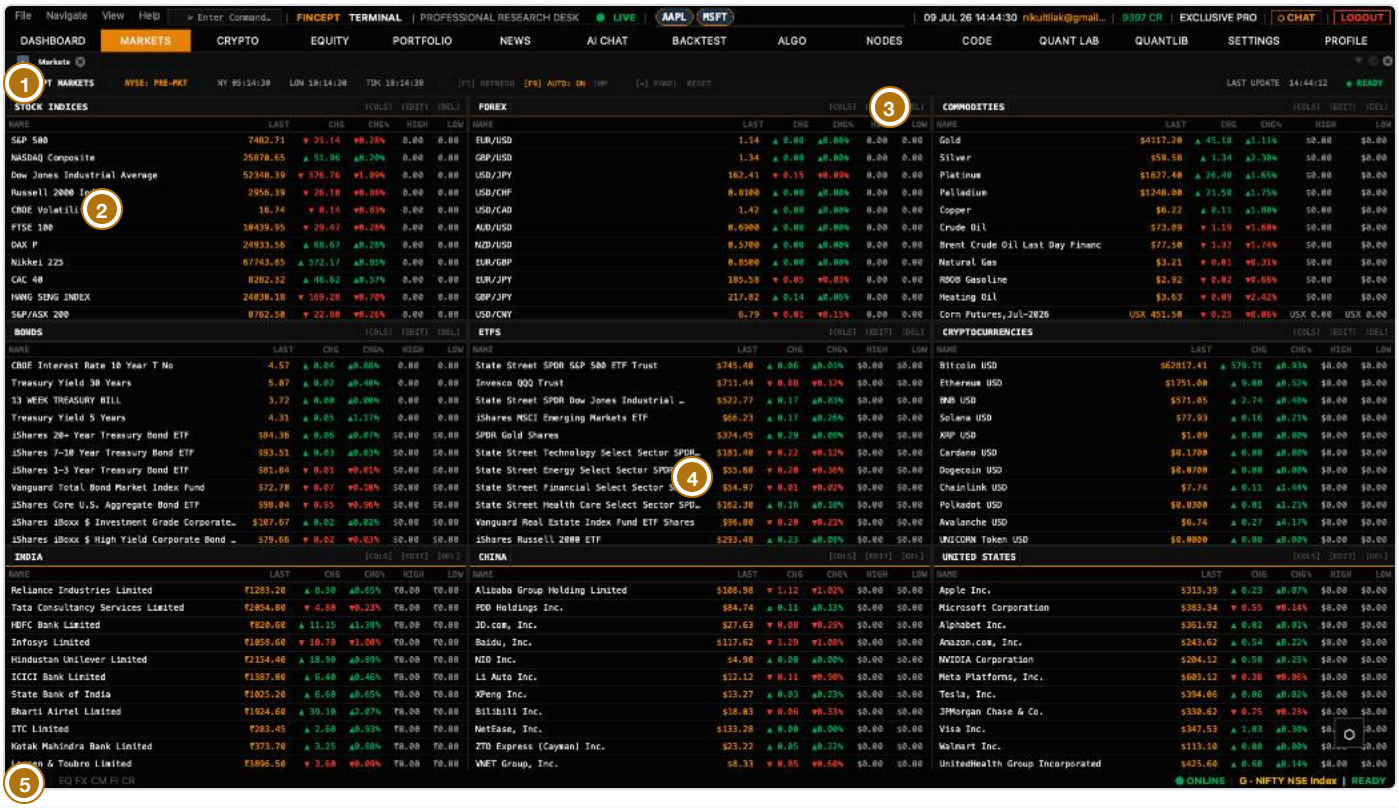
At a glance, you can…

- 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.



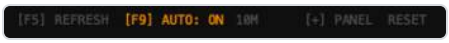
#	Region	What it is
1	**Header bar**	Brand label, NYSE session status, world clocks, and all board-wide controls (refresh, auto-refresh, interval, add panel, reset, last-update time, status badge).
2	**Market panel**	One tile = one basket. A titled table of live quotes. The board ships with nine of them.
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

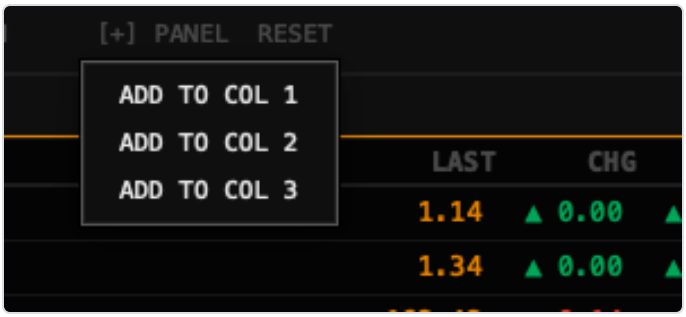


Last update · status

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.

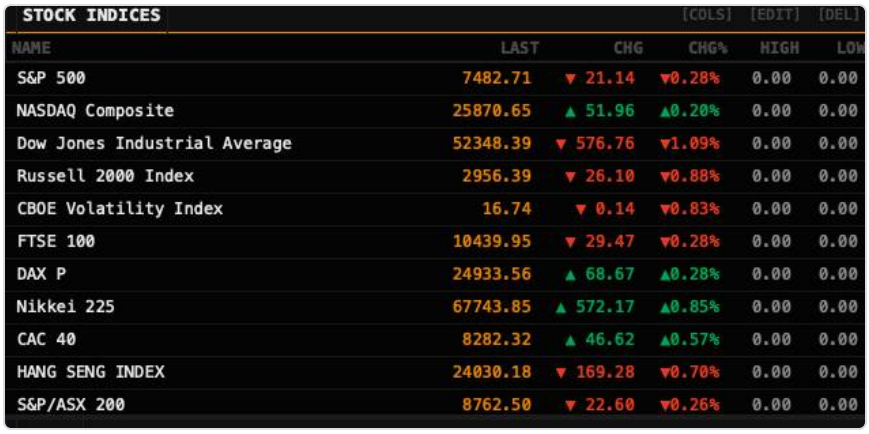


`[+] PANEL` → choose the target column.

> **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

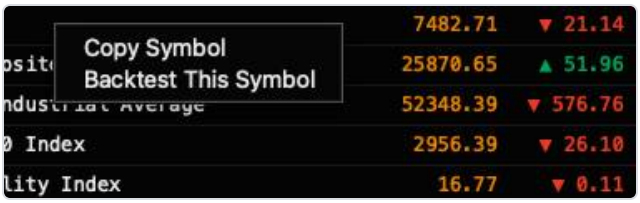
Instruments are listed by **friendly name** (the raw ticker such as `^GSPC` resolves to "S&P 500"; hover a name for its full name and ticker). Prices carry the right currency symbol ($, €, ₹, ¥…).

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…



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.

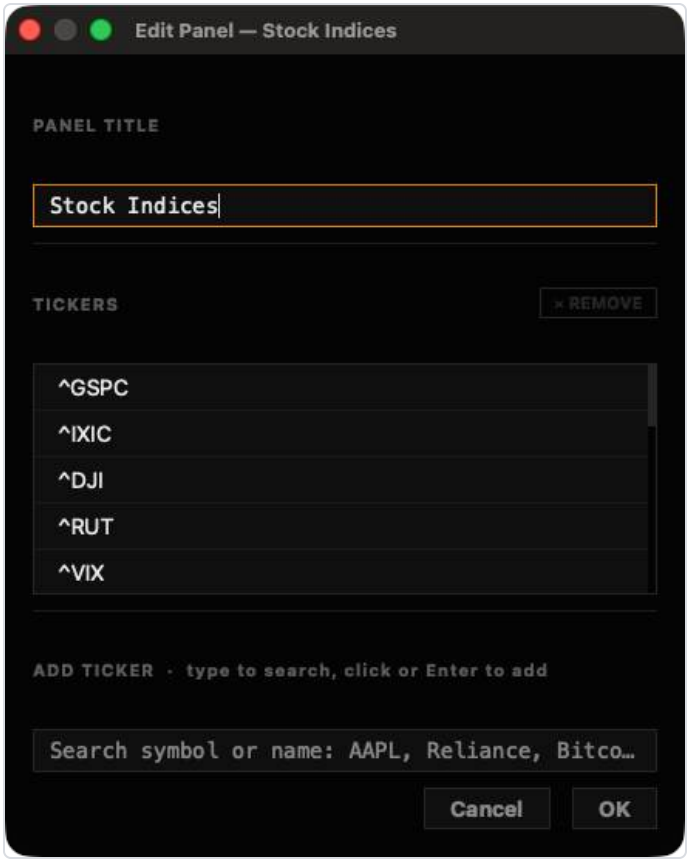
Column picker

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.

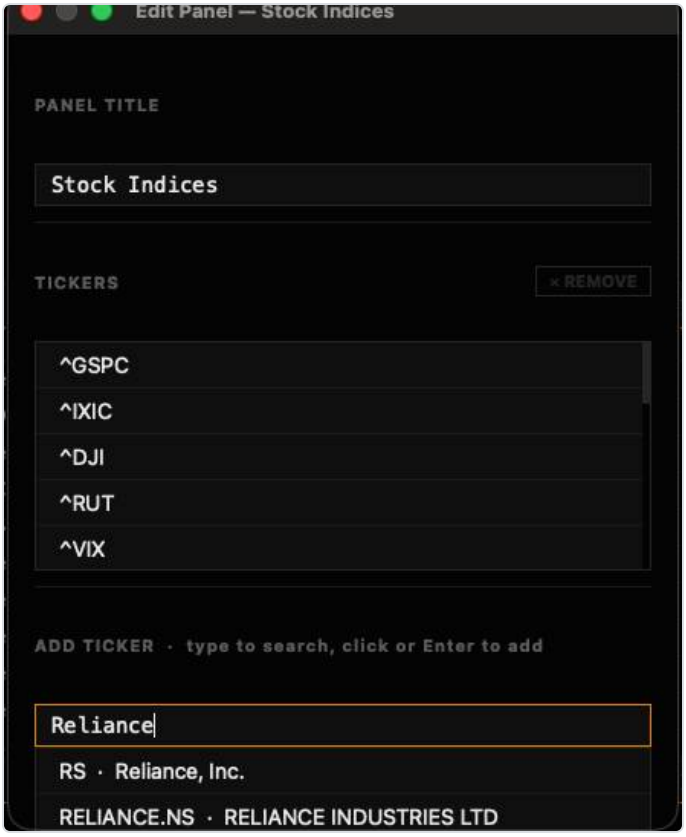


The panel editor

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

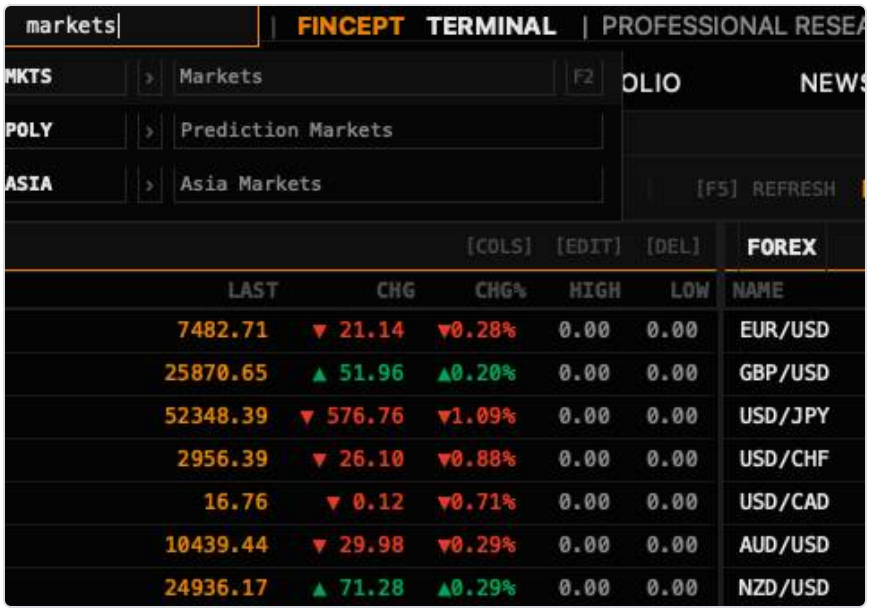
You see…	It means…
`⠿ LOADING` **(in a panel)**	That panel is fetching its first data. It clears as quotes arrive.
`● READY` **(badge)**	The last full refresh completed successfully.
`● LOADING` **(badge)**	A refresh is in progress across the board.
`● TIMEOUT` **(badge)**	Not every panel answered within 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.

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.

`>_ 22 brokers · paper + live · depth · GTT · baskets · 28 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

Demonstrated live on a working desk with four broker accounts.

Contents

17 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

REFERENCE

15 Data, persistence and agent tools 22

16 Quick reference 23

17 Honest notes on this build 24

About this guide. Every screenshot is a live capture of the Equity tab in the shipping build, on a desk with four configured brokers. **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.

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
- 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

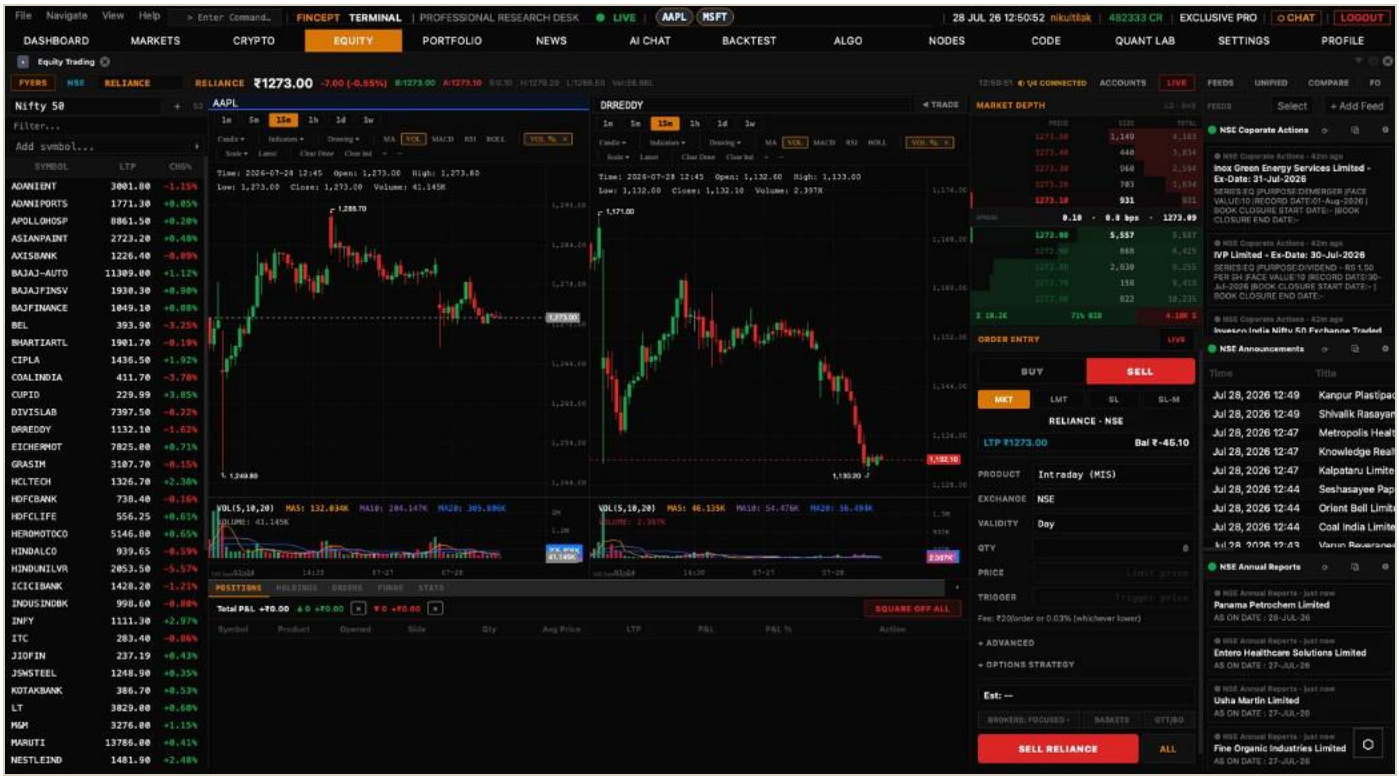


Fig 1 — The Equity tab on a live desk: Fyers focused, RELIANCE selected, compare chart and the feed column both open.

#	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 15 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.

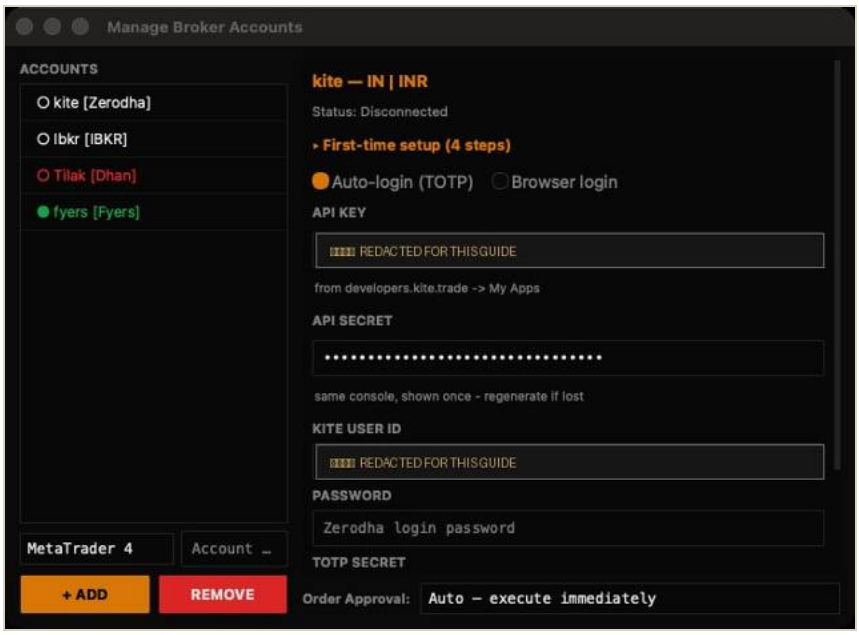


Fig 4 — The account manager. Left: the configured accounts with a status dot each. Right: the selected broker's credential form, its connection status, a first-time-setup walkthrough and the **Order Approval** mode. *Two credential values are masked in this figure.*

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

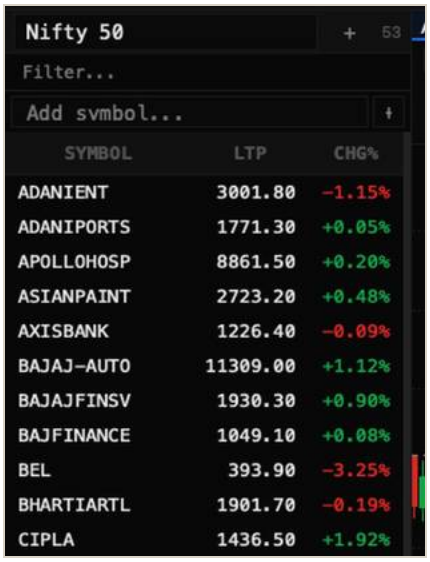


Fig 7 — The watchlist: list switcher, filter box, add box, then **SYMBOL / LTP / CHG%** streaming live.

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

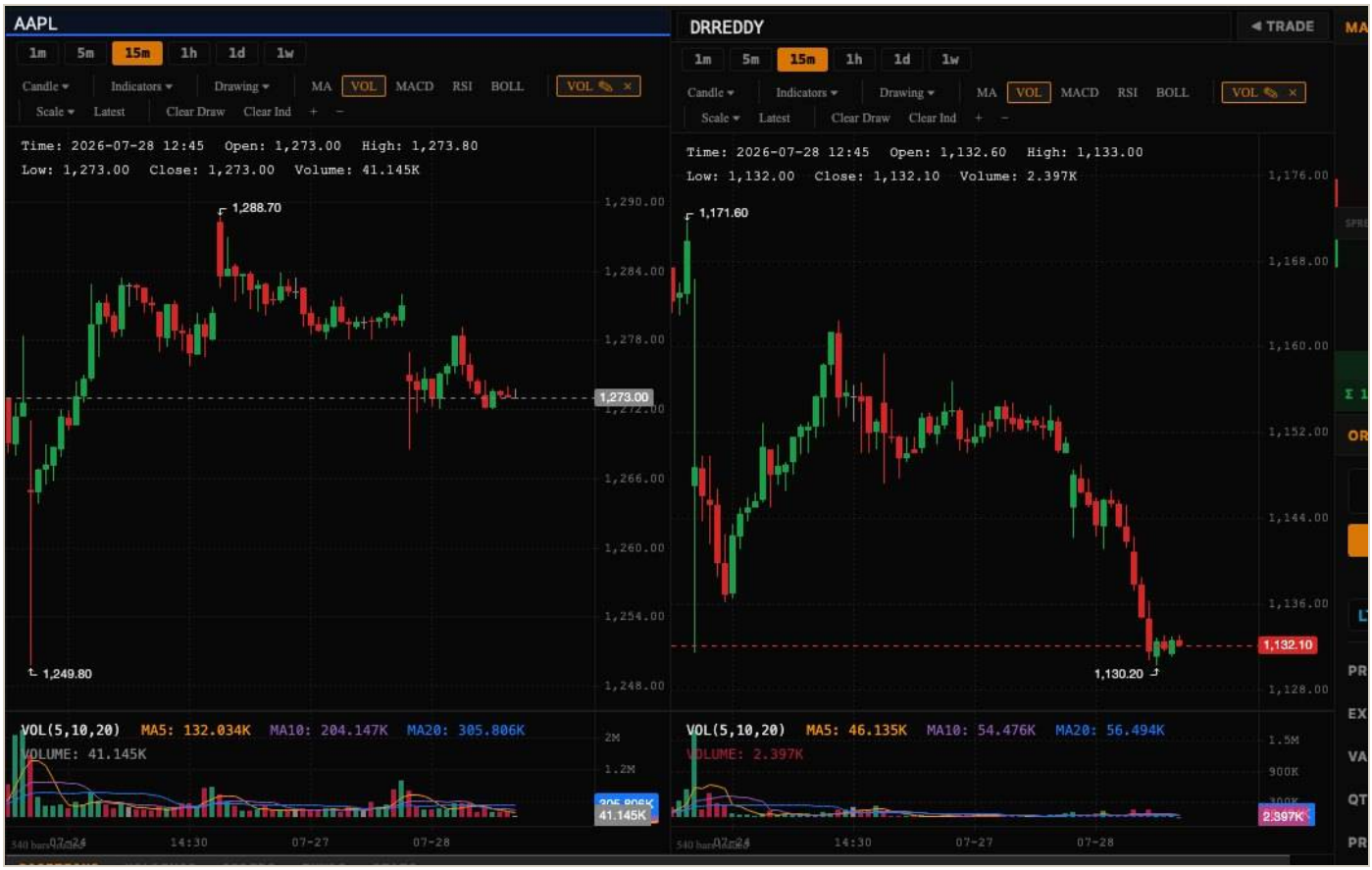


Fig 8 — Two live charts. The left pane is the primary (the traded symbol); the right is the compare pane, with a ◀ **TRADE** button to promote it.



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

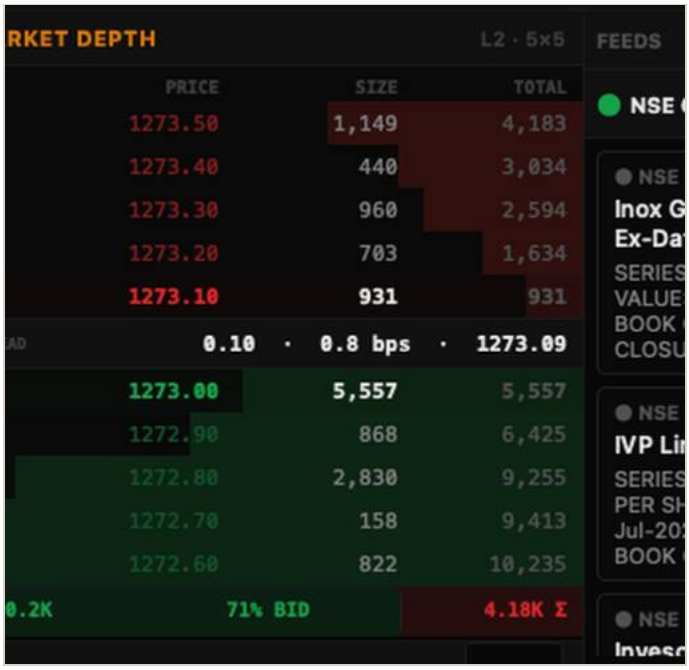


Fig 12 — The depth ladder: asks above, bids below, cumulative bars behind each level, the spread band across the middle and the imbalance footer at the bottom.

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

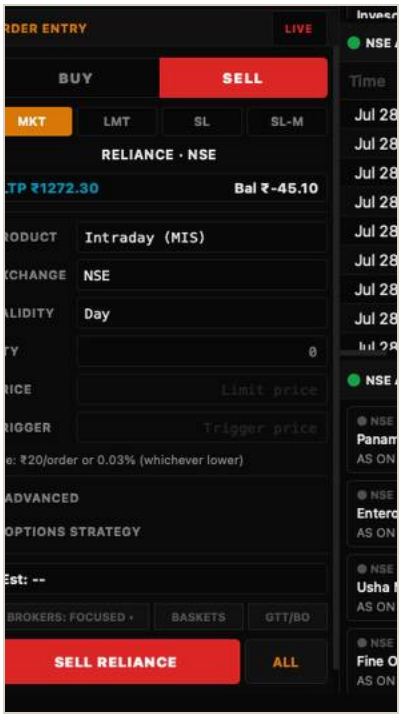


Fig 13 — The ticket with both collapsible sections closed. The chip beside **ORDER ENTRY** reads **LIVE**.

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

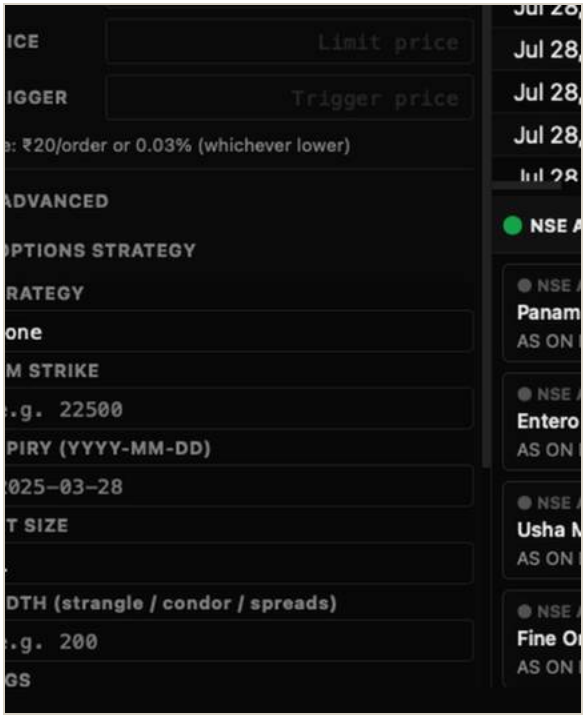


Fig 15 — The strategy builder.

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

MULTI ACCOUNT · 11

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

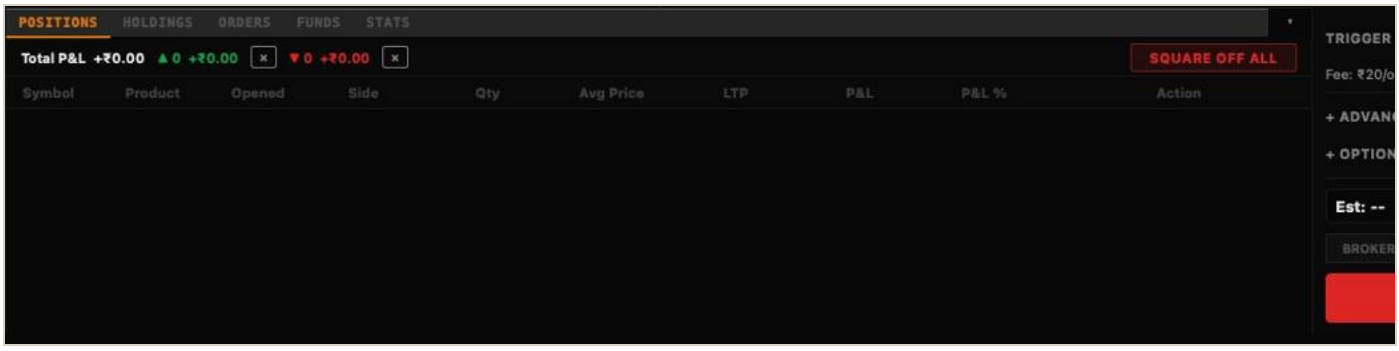


Fig 16 — The Positions tab: the action bar with net P&L, winner and loser chips and **SQUARE OFF ALL**, above the position rows.

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

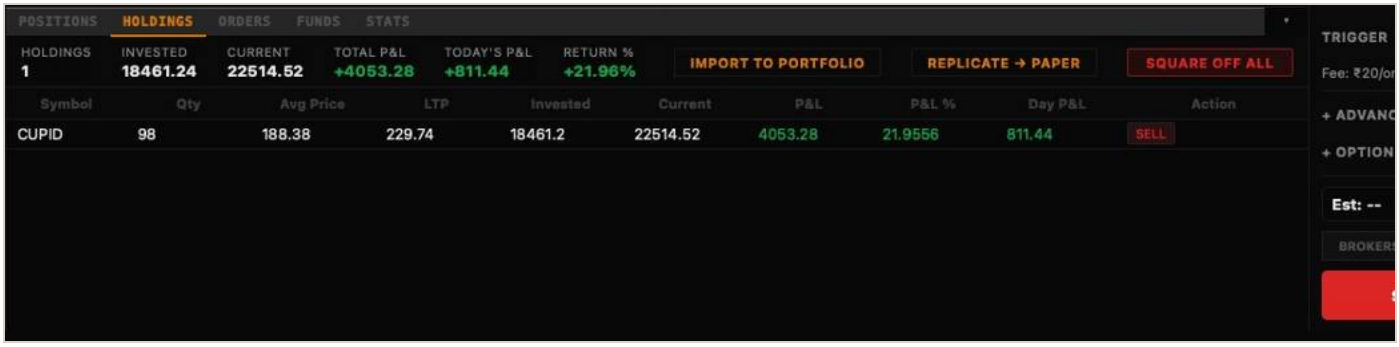


Fig 17 — The Holdings tab, with its own summary strip and bulk actions.

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

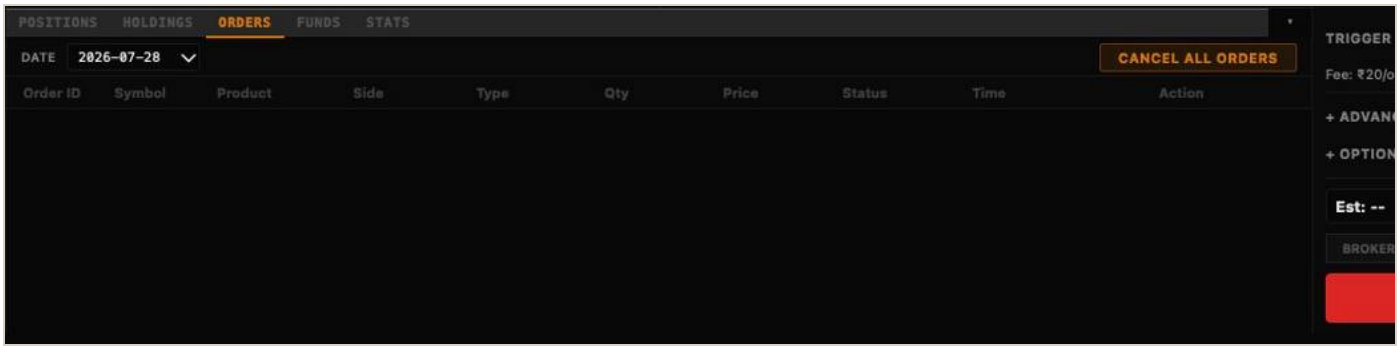


Fig 18 — The Orders tab with its day selector and bulk cancel.

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

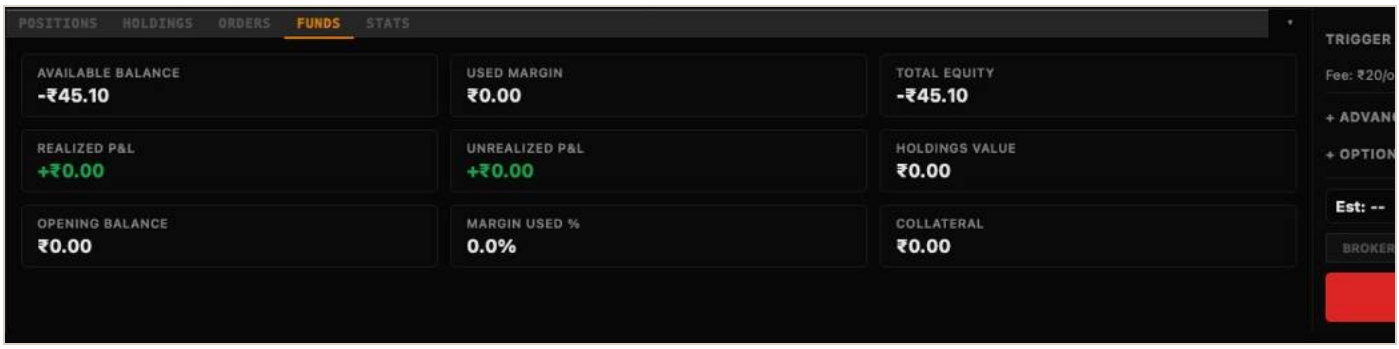


Fig 19 — The Funds tab. Live accounts fill the subset the broker reports; paper accounts fill the full set from the paper engine.

AVAILABLE BALANCE · USED MARGIN · TOTAL EQUITY · REALIZED P&L · UNREALIZED P&L · HOLDINGS VALUE · OPENING BALANCE · MARGIN USED % · COLLATERAL

Stats — twelve cards

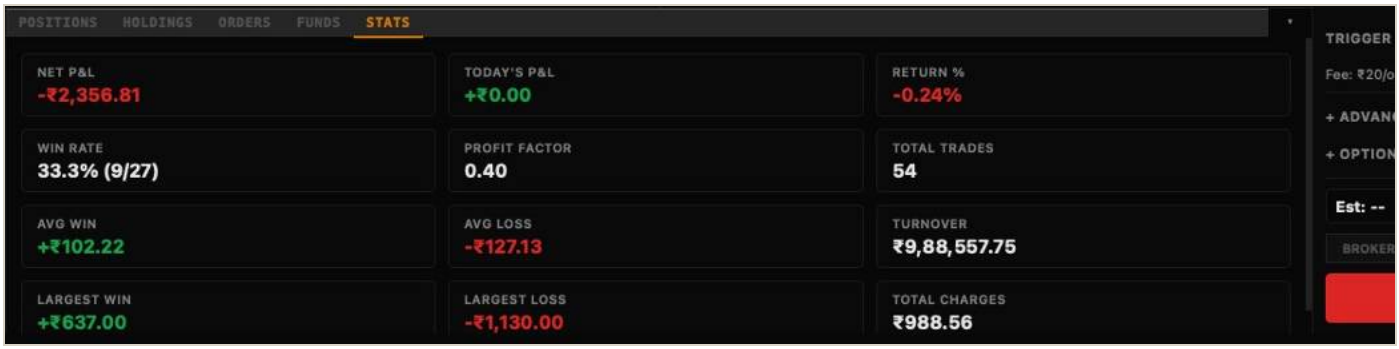


Fig 20 — The Stats tab: the trading record for this account.

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

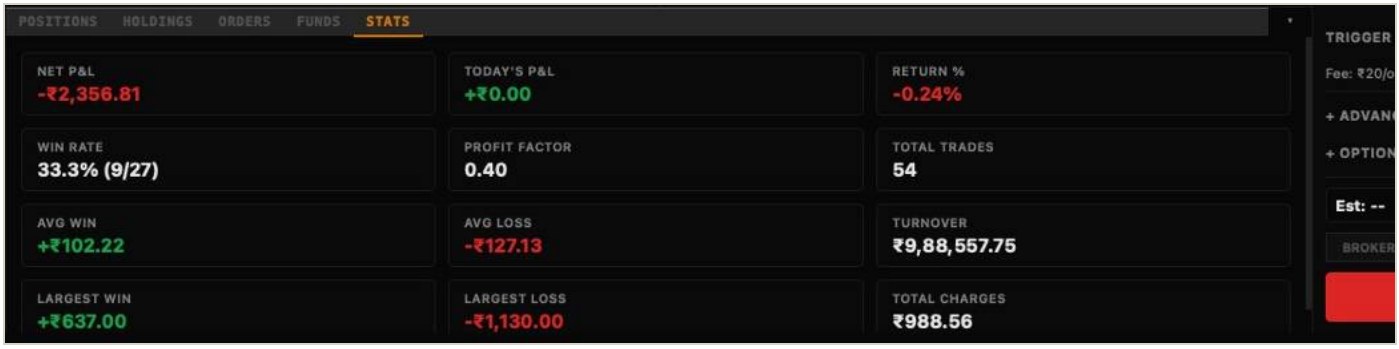


Fig 23 — The Time & Sales tab.

The two side columns

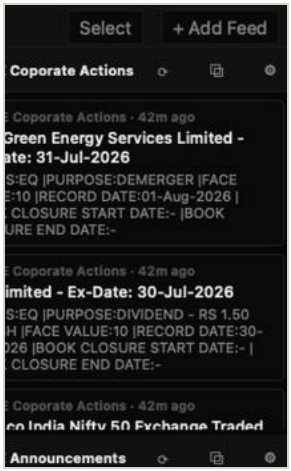


Fig 24 — The feed column.

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 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.

16 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**, or the roomier **F&O** tab
…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

17 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**.

> 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.

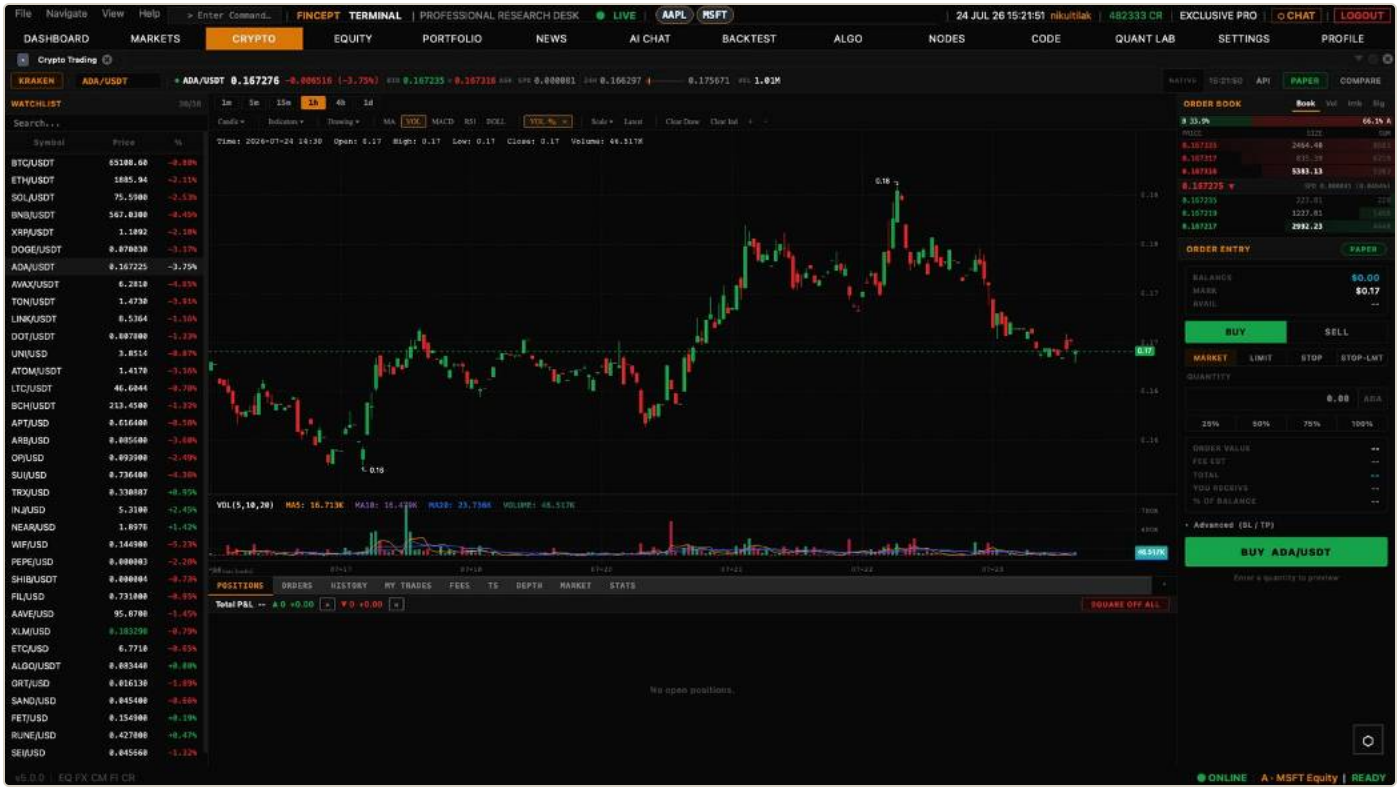


Fig 1. The Crypto Trading workstation — Kraken, ADA/USDT, paper mode. Left: the watchlist. Center: the chart (top) and blotter (bottom). Right: the order book (top) and order ticket (bottom). Across the top: the command bar with the live ticker.

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.

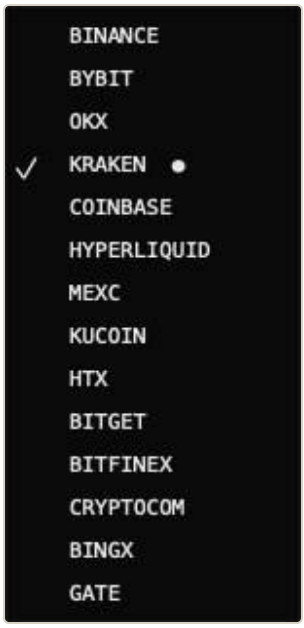
03 Command bar, exchanges & modes

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

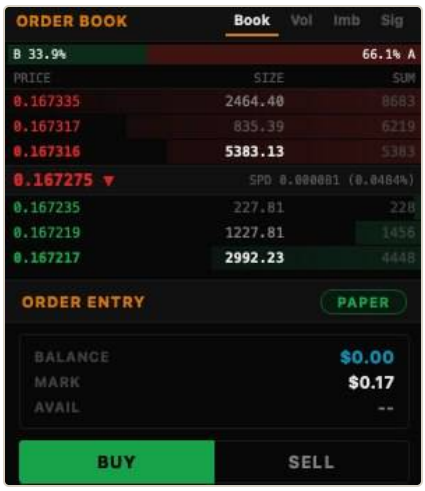


Fig 4. COMPARE shows a second live chart beside the first. Click a pane to focus it (the next symbol pick lands there); ◀ **TRADE** promotes the compared pair into the primary, traded slot.

> **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



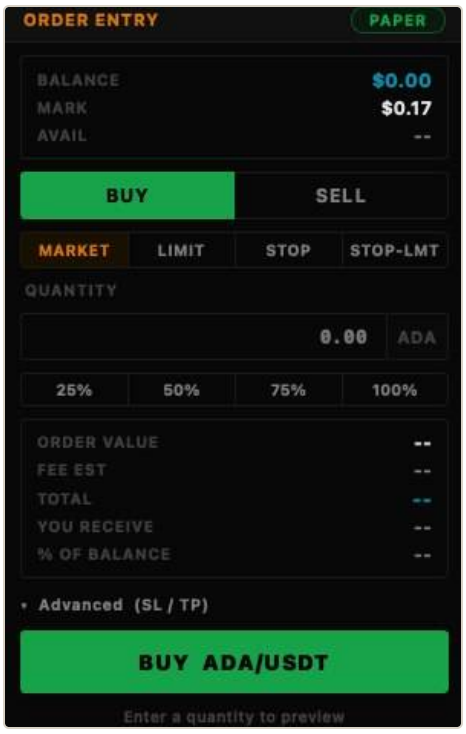
Fig 5. The **DEPTH** blotter tab: cumulative bid (green) / ask (red) depth with an amber mid line; hover for the cumulative size at any price.

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.

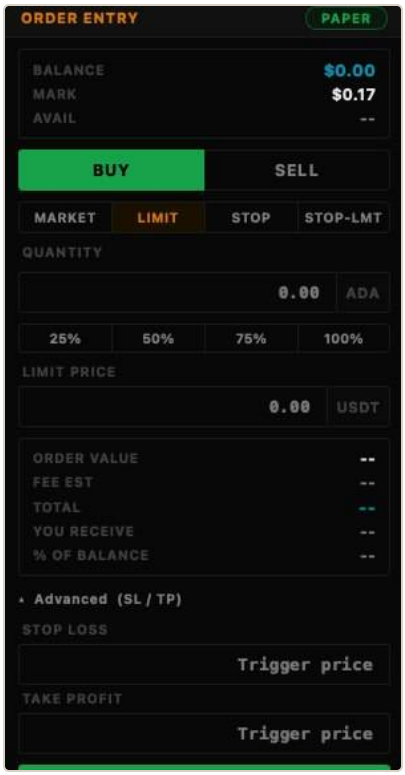
Order types

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

ANATOMY · 02

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

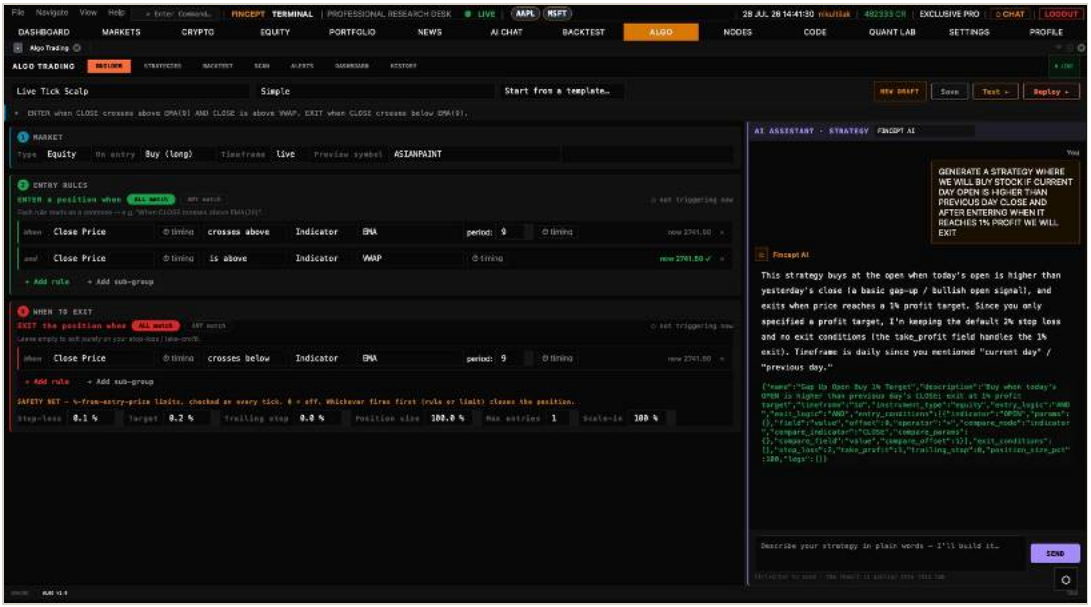


Fig 2 — The ALGO tab on the Builder, with a real draft loaded: the numbered authoring flow on the left, the AI strategy assistant on the right.

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

AI ASSISTANT · 06

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.

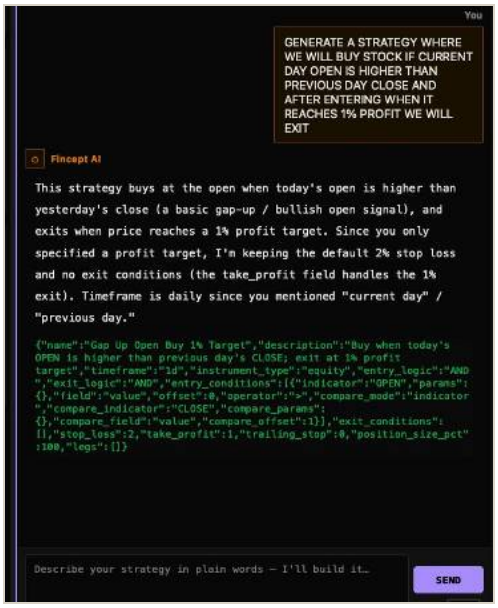


Fig 8 — The assistant turning *"buy stock if current day's open is higher than previous day's close and after entering when it reaches 1% profit we will exit"* into a complete configuration, and explaining the choices it made.

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

LIBRARY · 07

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.

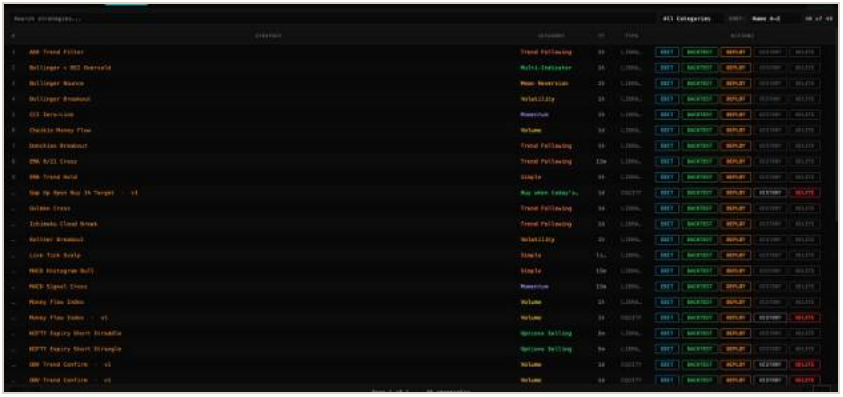


Fig 9 — The strategy grid. Built-in library rows and your own saved strategies sit in one list; the count on the right reads *40 of 40*.

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

BACKTEST · 08

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

BASKET TESTS · 09

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.

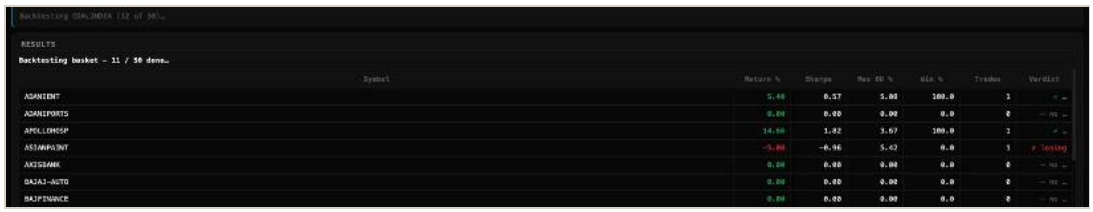


Fig 12 — A 50-symbol basket mid-run: *Backtesting basket — 11 / 50 done…* with results landing as each symbol finishes.

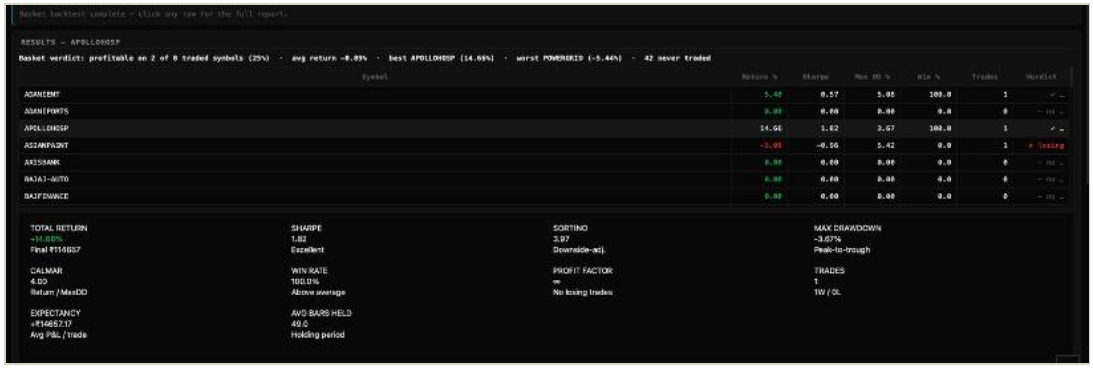


Fig 13 — The finished basket. The verdict line is the most honest sentence in the tab: *profitable on 2 of 8 traded symbols (25%) · avg return −0.89% · best APOLLOHOSP (14.66%) · worst POWERGRID (−5.44%) ·* ***42 never traded***. Click any row for that symbol's full report.

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.

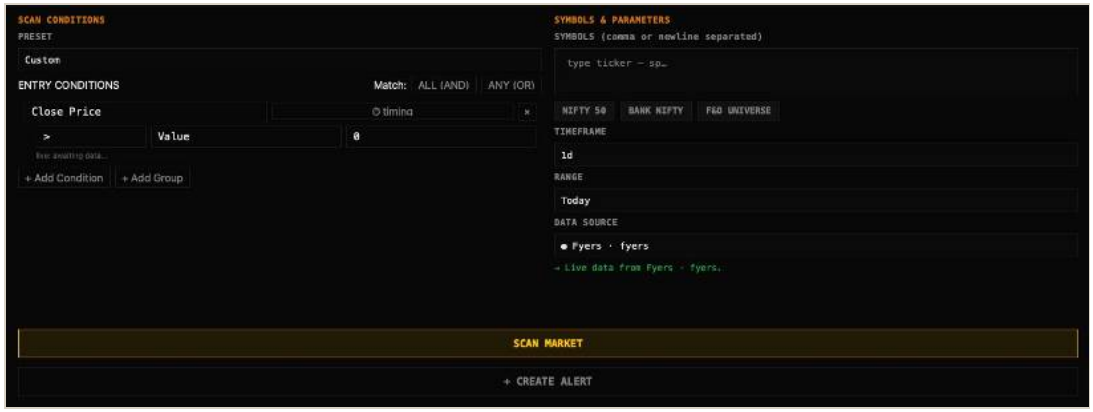


Fig 16 — One-time mode: the same condition builder as the Builder, a symbol list with universe shortcuts, and the data-source picker.

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.

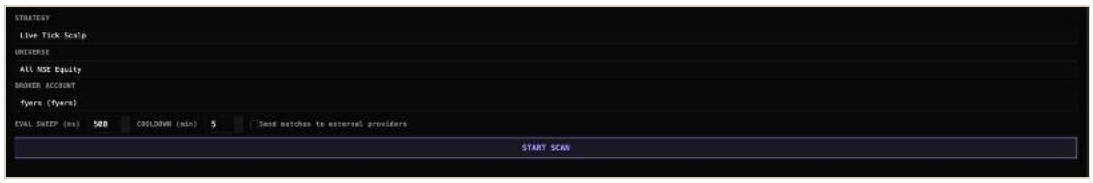


Fig 17 — Live mode: strategy, universe, broker account, sweep interval, cooldown, and the match table with a **DEPLOY** button per row.

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.

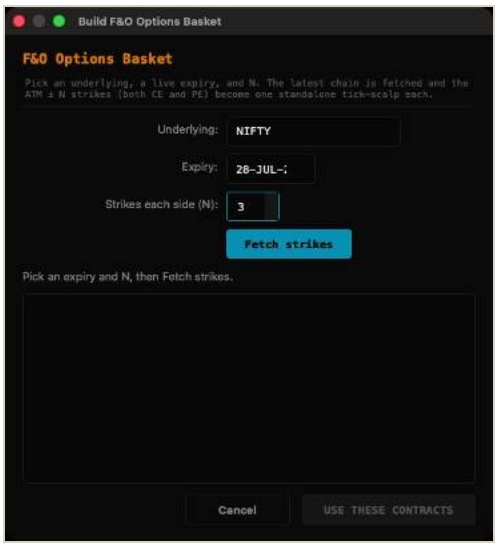


Fig 21 — The F&O options basket builder.

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

FLEET · 15

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.

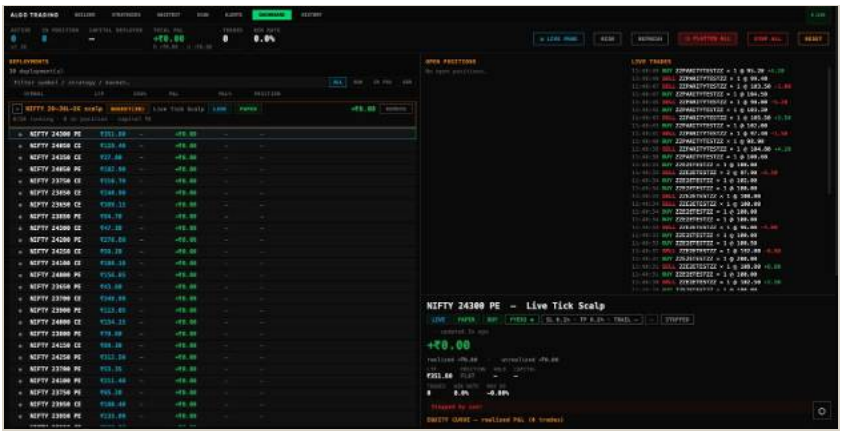


Fig 22 — The Dashboard on a real 30-member F&O basket, with a live fill feed on the right.

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

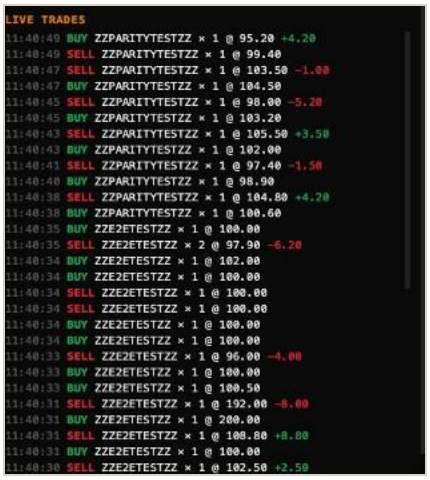


Fig 25 — The LIVE TRADES feed: every fill as it books, with side, quantity, price and P&L.

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.

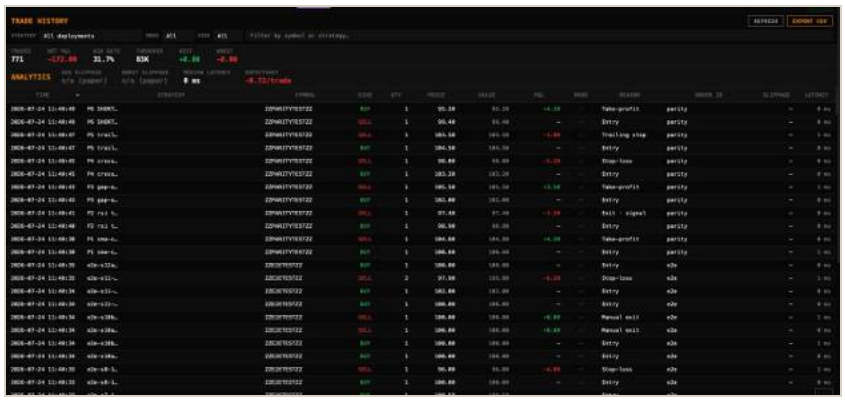


Fig 27 — The ledger on a real desk: 771 trades, filtered and totalled.

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

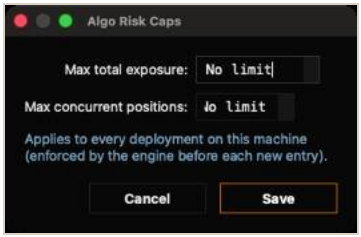


Fig 29 — The Algo Risk Caps dialog.

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).

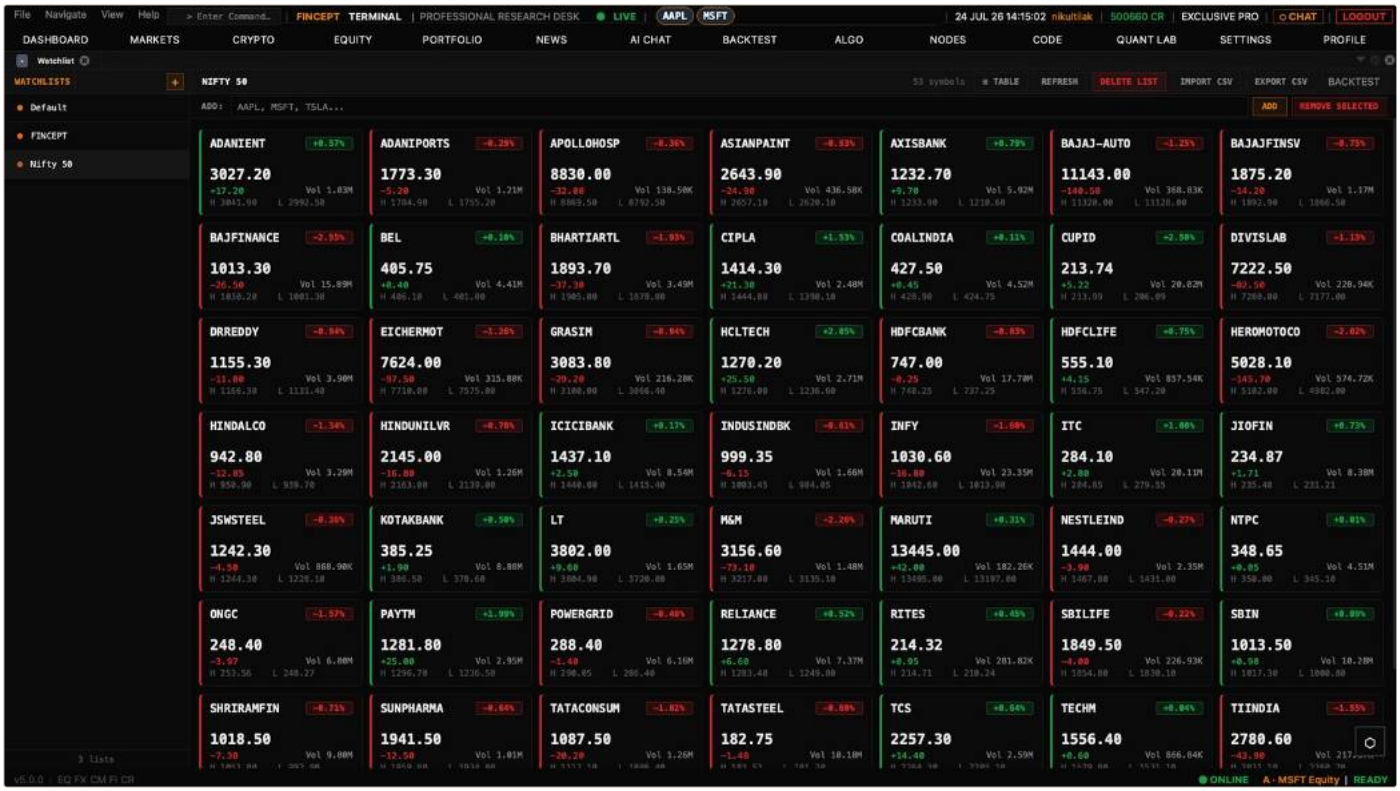


Fig 1. The Watchlist tab in cards view — the "Nifty 50" list (53 symbols) streaming live. Left: the sidebar of three watchlists. Top: the toolbar and the add bar. Center: the responsive card grid.

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).

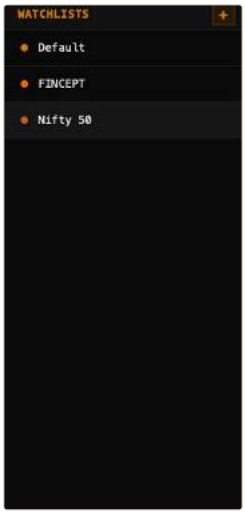


Fig 2. The toolbar: list name · symbol count · view toggle · REFRESH · DELETE LIST · IMPORT/EXPORT CSV · BACKTEST.

> **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.

FINCEPT TERMINAL · PRODUCT GUIDE

Trade Flow Visualisation

A complete walkthrough of the Trade Viz tab — the terminal's bilateral goods-trade picture: one chord diagram, fifteen partners, and a ranked trade table.

`>_ 15 partners · imports vs exports · one screen, no fetch`

Covers · **reading the chord diagram** · **the partner table** · **the filter bar**
the underlying dataset · what the tab is for · where to get live trade data

Demonstrated live on the Fincept Terminal Trade Viz tab.

Contents

8 SECTIONS

GETTING ORIENTED

01 What the Trade Viz tab is 3

02 Anatomy of the tab 4

READING THE SCREEN

03 The chord diagram, line by line 5

04 The partner ranking table 6

05 The filter bar 7

THE NUMBERS

06 The dataset behind the picture 8

07 Using it well — and where live trade data lives 9

REFERENCE

08 Quick reference and honest notes 10

> **About this guide.** Every screenshot is a live capture of the Trade Viz tab in the shipping build. Where a control is presentational rather than functional, this guide says so plainly rather than describing an intention.

01 What the Trade Viz tab is

ORIENTATION · 01

Trade Viz is a **single-screen picture of bilateral goods trade**. A chord diagram puts the United States at the centre and its fifteen largest trading partners around the rim; the thickness and brightness of each bundle of arcs shows how much moves along that corridor, and a ranked table beside it puts a number on every one.

It answers the questions a macro or supply-chain conversation opens with — *who are the big counterparties, how lopsided is each relationship, and how does one corridor compare with the next?* — in a form you can read in a second and screenshot into a note.

> **Read this before you cite it.** Trade Viz is a **fixed reference visual**, not a live data feed. It draws one built-in dataset: US bilateral goods trade, 2024 estimates, in millions of US dollars. It makes no network request, spends no credits, and — importantly — the country, order, periodicity and year controls along the top **do not change what is drawn** in this build. Section 05 covers exactly what each control does, and section 07 points you at the tabs that *do* fetch live trade and macro data.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Tools ▸ Trade Viz**
Command bar	Type `tradeviz` or `tradegraph`, then pick the row

At a glance, in Trade Viz you can…

- See the 15 largest US goods-trade partners at once
- Compare import corridors by arc thickness and brightness
- Spot the five largest export corridors (the magenta bundles)
- Read exact import, export and total figures from the table
- Rank partners by total two-way trade
- Identify lopsided relationships at a glance
- Screenshot a clean, presentation-ready graphic
- Do all of it offline — no key, no fetch, no credits

02 Anatomy of the tab

Four regions: a label strip, a filter bar, the diagram, and the table. The divider between diagram and table can be dragged if you want more of one than the other.



Fig 1 — The Trade Viz tab as it opens. Amber arcs are imports into the US; magenta arcs are US exports to the five largest partners; the table on the right ranks all fifteen by total two-way trade.

#	Region	What it is
1	Label strip	`21) Table`, `22) Settings`, `23) Export`, `24) Notes` on the left and the amber **Trade Flow** title on the right. These are **labels, not buttons** — see section 05.
2	Filter bar	Country, `20) Browse`, **Order by**, **Periodicity**, a year stepper, and a live clock. Section 05.
3	Chord diagram	The drawing itself: concentric rings, radial spokes, arc bundles, partner nodes, a US node at the centre, axis captions and a colour legend. Section 03.
4	Partner table	Rank, partner name and total trade in $M, fifteen rows, sortable by clicking a header. Section 04.

> The layout is a two-panel splitter with the diagram given roughly three-fifths of the width. Drag the thin divider to enlarge either side; the diagram redraws to fit, and below a certain width it stops drawing rather than rendering an illegible tangle.

03 The chord diagram, line by line

Every mark on the diagram encodes something. Once you know the five rules below you can read a corridor's size without looking at the table at all.

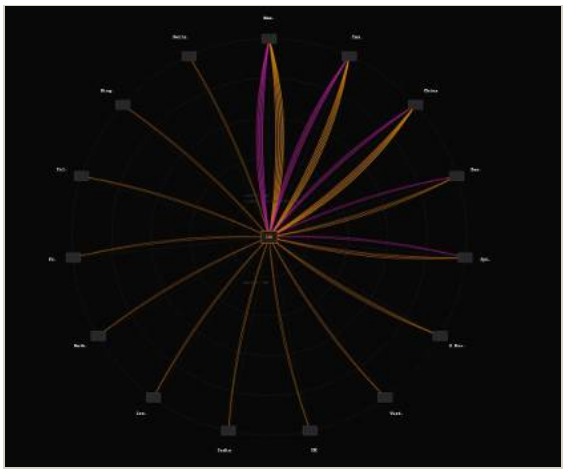


Fig 2 — The diagram in detail: fifteen partner nodes around the rim, the amber **US** node at the centre, amber import bundles and magenta export bundles.

Mark	Meaning
`Amber arcs`	**Imports** — goods flowing from that partner into the US. Every partner has one bundle.
`Magenta arcs`	**Exports** — US goods going to that partner. Drawn for the **top five partners only**.
`Bundle weight`	Strand count, opacity and curvature all scale with the corridor's size: bigger corridors read as thicker, brighter, more bowed ropes.
`Grey rectangles`	The partner nodes on the rim, each labelled with a short code just outside the circle.
`Rings and spokes`	Five guide rings and one faint spoke per partner — visual scaffolding, not data.

> **Only five magenta bundles exist.** Export arcs are drawn for the first five partners (Mexico, Canada, China, Germany, Japan). The absence of a magenta bundle on, say, the United Kingdom does **not** mean the US exports nothing there — the figure is in the table (73,429 $M). Read exports from the table, and use the magenta arcs only as a heavyweight marker for the top five.

Position, legend and captions

- **Partner order runs clockwise from the top.** Mexico sits at twelve o'clock and the rest follow in descending order of total trade, so the largest relationships cluster in the upper right.
- **The centre node** is the US, outlined in amber — every arc begins or ends there. This is a hub-and-spoke picture of one country's trade, not a partner-to-partner matrix.
- **The bottom-left legend** is an amber ramp from dark to bright with three tick figures, confirming that brighter means larger.
- **Two faint captions** near the centre read *Imports ($M)* above and *Exports ($M)* below, marking which half of the drawing each family of arcs belongs to.

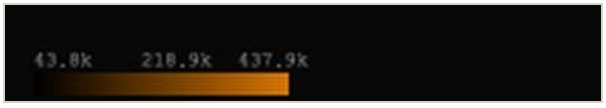


Fig 3 — The colour legend: the amber ramp with low, middle and high tick values in thousands of $M.

04 The partner ranking table

PARTNER TABLE · 04

The right-hand panel is the numeric companion to the drawing: the same fifteen partners, ranked, with a figure you can quote.

Trading Partner	Total Trade ($M)
Mexico	782996.00
Canada	703611.00
China	604323.00
Germany	235974.50
Japan	233119.00
South Korea	204040.20
Vietnam	151531.80
United Kingdom	140770.60
India	132132.10
Ireland	126855.10
Netherlands	108246.20
France	107890.20
Italy	104588.70
Singapore	99135.20
Switzerland	92952.60

Fig 4 — The partner table, ranked by total two-way trade.

Column	Contents
`(rank)`	`1)` to `15)` — position by total two-way trade, largest first
`Trading Partner`	Full country name, in green
`Total Trade ($M)`	Imports **plus** exports for that corridor, in millions of US dollars, to two decimals

Rows are selectable, alternate shading makes them easy to track across, and clicking a column header re-sorts the table. Sorting changes the table only — the diagram keeps its own fixed clockwise order.

> **Total Trade is the sum of both directions.** Mexico's 782,996 is 437,898 of imports plus 345,098 of exports. The table does not break the two apart, and it does not show a balance column — section 06 lists both legs and the resulting surplus or deficit for all fifteen partners.

Reading the table against the diagram

- **Rank order and clockwise order match**, so row 1 is the bundle at twelve o'clock, row 2 the next one clockwise, and so on. That makes the table an index into the picture.
- **A thick amber bundle with a modest total** means an import-heavy relationship: Vietnam (109,450 in, 42,082 out) is the clearest example on the screen.
- **A partner with no magenta bundle but a high total** is simply outside the top five — the export leg is in the table.

05 The filter bar

The bar along the top looks like a query builder, and in this build it is better understood as a **caption strip with one live element**. This section is deliberately blunt about which is which, so nothing you present is built on a filter that never applied.



Fig 5 — The filter bar: country selector, Browse label, Order by, Periodicity, the year stepper, and (far right, not shown) the clock.

Control	Options	Effect in this build
`Country`	15 countries, United States first	Selection is remembered between sessions. **The diagram and table always show the United States.**
`20) Browse`	—	A **label**, not a button. Nothing to click.
`Order by`	Total Trade · Imports · Exports · Trade Balance · % of GDP	Selection is remembered. **It does not re-sort or re-scale anything** — to re-order the table, click its column header instead.
`Periodicity`	Yearly · Quarterly · Monthly	Selection is remembered. The dataset is annual only, so the picture is unchanged.
`« / »`	—	**Labels**, not buttons. Use the year dropdown itself.
`Year`	2024 down to 2010	Selection is remembered. **The figures are 2024 estimates regardless.**
`Clock`	—	Genuinely live: ticks once a second while the tab is visible, and stops when you leave it.



Fig 6 — The **Order by** dropdown, showing all five options with the current choice ticked.

Fig 7 — The label strip. `21) Table` through `24) Notes` are captions in the style of a terminal function-key row, not clickable tabs.

> **Why document this so directly?** Because a screenshot of this tab with “Imports” and “2019” showing would look like a filtered view, and it is not one. The four selectors persist your choice across restarts — which makes them feel functional — but the drawing and the numbers come from the built-in 2024 dataset in every case. When you paste this graphic into a document, caption it *US bilateral goods trade, 2024 estimates ($M)*.

06 The dataset behind the picture

DATASET · 06

The tab draws fifteen partners with two figures each. Here is the whole dataset, with the balance worked out — the column the screen does not show.

#	Partner	Code	US imports ($M)	US exports ($M)	Total ($M)
1	Mexico	Mex.	437,898	345,098	782,996
2	Canada	Can.	406,282	297,329	703,611
3	China	China	427,230	177,093	604,323
4	Germany	Ger.	145,632	90,343	235,975
5	Japan	Jpn.	138,420	94,699	233,119
6	South Korea	S.Kor.	115,210	88,830	204,040
7	Vietnam	Viet.	109,450	42,082	151,532
8	United Kingdom	UK	67,342	73,429	140,771
9	India	India	87,120	45,012	132,132
10	Ireland	Ire.	82,340	44,515	126,855
11	Netherlands	Neth.	56,120	52,126	108,246
12	France	Fr.	62,340	45,550	107,890
13	Italy	Itl.	67,230	37,359	104,589
14	Singapore	Sing.	48,120	51,015	99,135
15	Switzerland	Switz.	54,230	38,723	92,953

Balance highlights. Only three of the fifteen corridors run a US goods surplus on these numbers — the Netherlands (−3,994 for the partner), Singapore (+2,895 of US exports over imports) and the United Kingdom (+6,087). The largest deficits are China (−250,137), Mexico (−92,800) and Canada (−108,953).

> **Units and vintage.** Every figure is in millions of US dollars and labelled by the tab as a 2024 estimate. The codes in the third column are the abbreviations printed around the rim of the diagram, so you can match a bundle to a row without counting positions.

> **Treat these as indicative round numbers.** They are a built-in reference set for shaping a conversation, not an official statistical release. For citable trade statistics — and for any year other than 2024 — use the sources in section 07.

07 Using it well — and where live trade data lives

What this tab is genuinely good for

- **Framing a discussion.** "Three partners account for most of the flow" is easier to land with the picture on screen than with a table of numbers.
- **Orders of magnitude.** The arc encoding makes a 5× difference between corridors obvious instantly.
- **Asymmetry.** Import-heavy relationships stand out as thick amber with no magenta.
- **Presentation.** It is a clean, dark, self-contained graphic that screenshots well into a note or a deck.
- **Offline work.** No key, no fetch, no credits, no waiting — it renders the moment the tab opens, on a plane or behind a firewall.

What to use instead when you need live numbers

Need	Tab	What you get
`Official trade statistics`	**Gov Data**	Sovereign open-data portals — US Treasury, Canada, Switzerland, France, Austria, Hong Kong, Australia, Spain and universal CKAN catalogues — queried live and free
`Macro series over time`	**Economics**	48 sources including World Bank, FRED, OECD, Eurostat, StatCan and IMF-adjacent series, charted with CSV export
`Country-level indicators`	**DBnomics**	A very large aggregated series catalogue, searchable by country and dataset
`China-specific detail`	**AKShare Data**	Chinese market and macro datasets
`Geopolitical context`	**Geopolitics**	Event and risk views around the same corridors

> **A workflow that combines them.** Open Trade Viz to fix the shape of the relationship in everyone's head, then pull the specific series you intend to quote from Economics or Gov Data, and export that to CSV so the number in your note has a source behind it.

Persistence and cost

- **Persisted:** the four selector positions (country, order, periodicity, year) — nothing else, because there is nothing else to remember.
- **Cost:** none. No network request, no API key, no Fincept credits.
- **Agent tools:** none. Trade Viz publishes no MCP tools, so the assistant cannot read the diagram; ask it for trade data through the Economics or Gov Data tools instead.

08 Quick reference and honest notes

QUICK REFERENCE · 08

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Tools ▸ Trade Viz, or type `tradeviz` in the command bar
…tell imports from exports?	Amber arcs are imports into the US; magenta arcs are US exports (top five partners only)
…find a partner's exact figures?	Section 06 of this guide — the screen shows only the two-way total
…re-order the table?	Click the column header. The **Order by** dropdown does not sort
…change the year?	You cannot — the dataset is 2024 estimates only
…match an arc to a table row?	Rank order runs clockwise from twelve o'clock, so row 1 is the top bundle
…make the diagram bigger?	Drag the divider between the diagram and the table
…get live trade statistics?	Gov Data or Economics — see section 07
…caption a screenshot correctly?	*US bilateral goods trade, 2024 estimates ($M)*

Honest notes on this build

- **The four selectors are presentational.** Country, Order by, Periodicity and Year persist your choice but do not change the drawing or the table.
- **Three strips are labels, not controls:** `21) Table` – `24) Notes`, `20) Browse`, and the `«` / `»` year steppers.
- **Export arcs cover only the top five partners**, by design — the other ten export legs appear in the table's total but have no magenta bundle.
- **No export button.** Despite the `23) Export` caption, there is no CSV or image export; use the terminal screenshot key (**Ctrl+P**) to capture the graphic.
- **No tooltips or hover detail.** The diagram is a static drawing: hovering an arc reveals nothing, and clicking it does nothing.
- **The country list and the partner list differ.** The selector offers Brazil and Australia, which are not among the fifteen drawn partners.
- **The diagram needs room.** Squeeze the panel too narrow and it stops drawing rather than rendering an unreadable knot — widen it and it returns.
- **Data is a fixed 2024 estimate set** in millions of USD, hard-coded into the build; it does not update and cannot be refreshed.

> This guide documents the tab as it ships. Everything above was read from the source and then confirmed on screen — including which controls do nothing, because a guide that flatters the software is no use to a desk that has to present from it.

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.

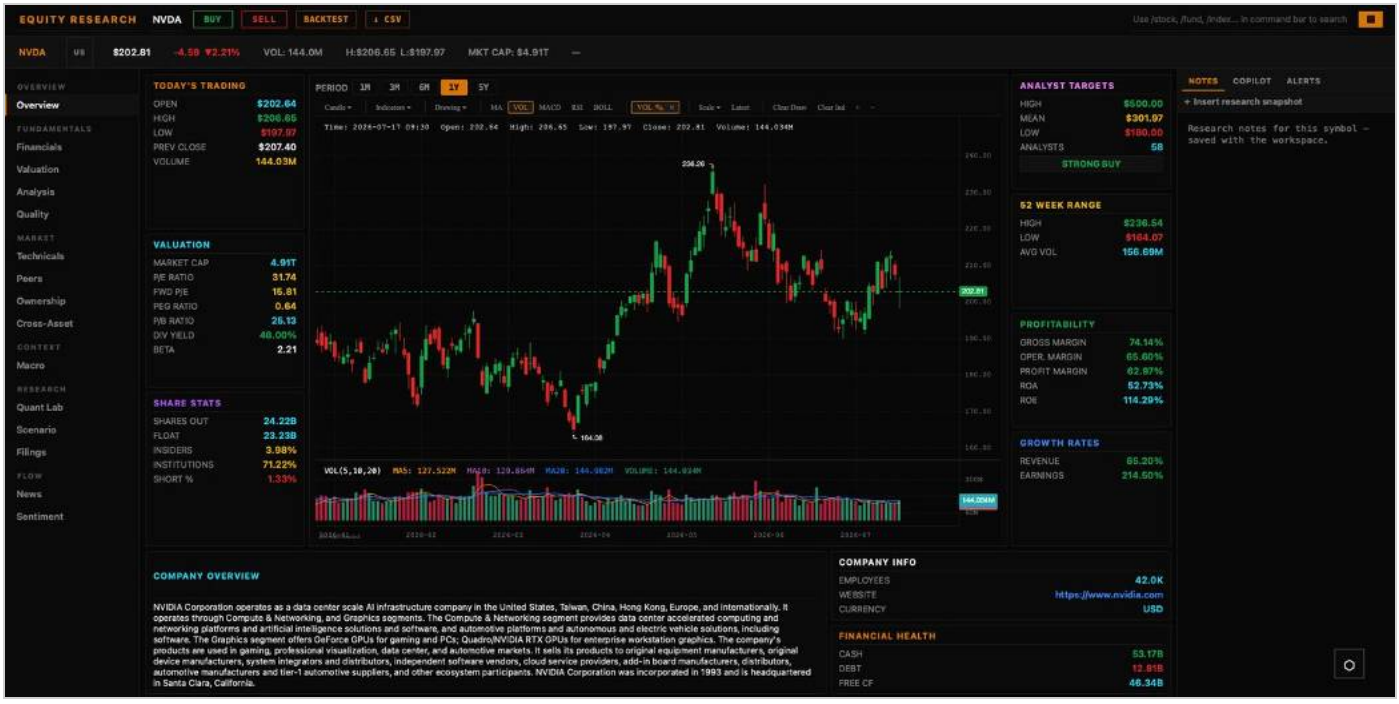
The data is live, and mostly local: quotes, fundamentals, statements, technicals, peers and news are pulled from yfinance through the app's embedded Python — not a Fincept server — with SEC EDGAR filings fetched directly from the SEC. Where a matching broker is connected, the price ticks live and technical 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

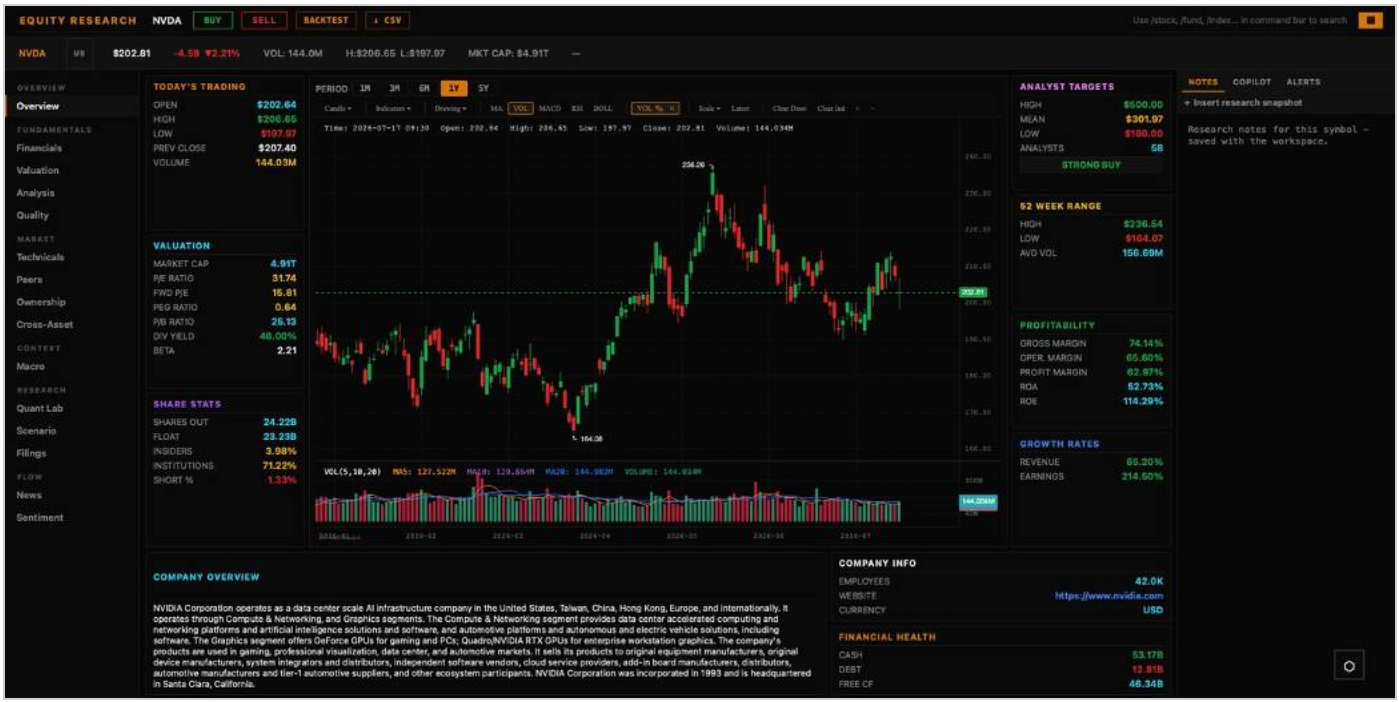
- **Command bar** — `/stock`, `/fund`, `/index` search (server-side), then the pick loads here.
- **Drag & drop** — drop any ticker chip onto the screen to jump the research view.
- **Linked group** — the loaded symbol broadcasts to Watchlist / Equity Trading / News so they follow along.

- **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



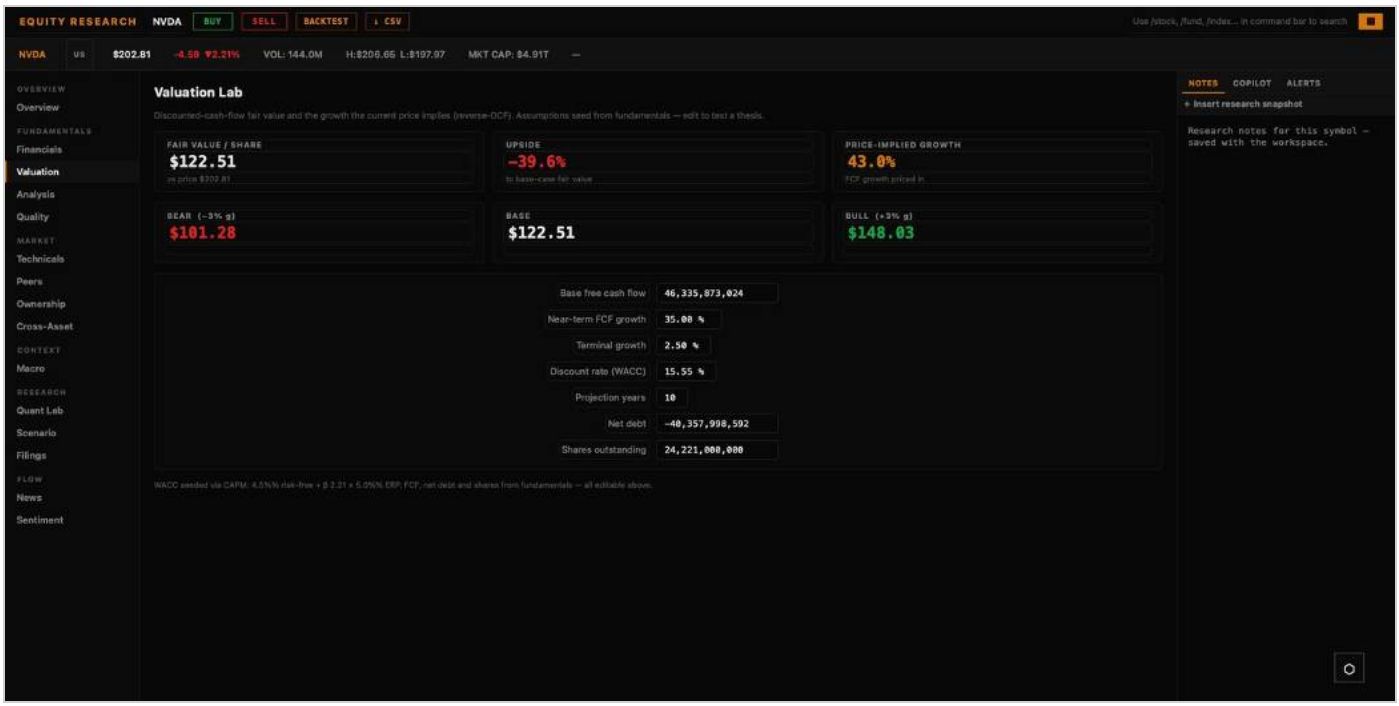
FINANCIALS (NVDA) — Income Statement view: key-metric cards, revenue/margin trend charts, return metrics + DuPont, and the full raw statement table. Toggle to Balance Sheet / Cash Flow; EXPORT CSV top-right.

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)

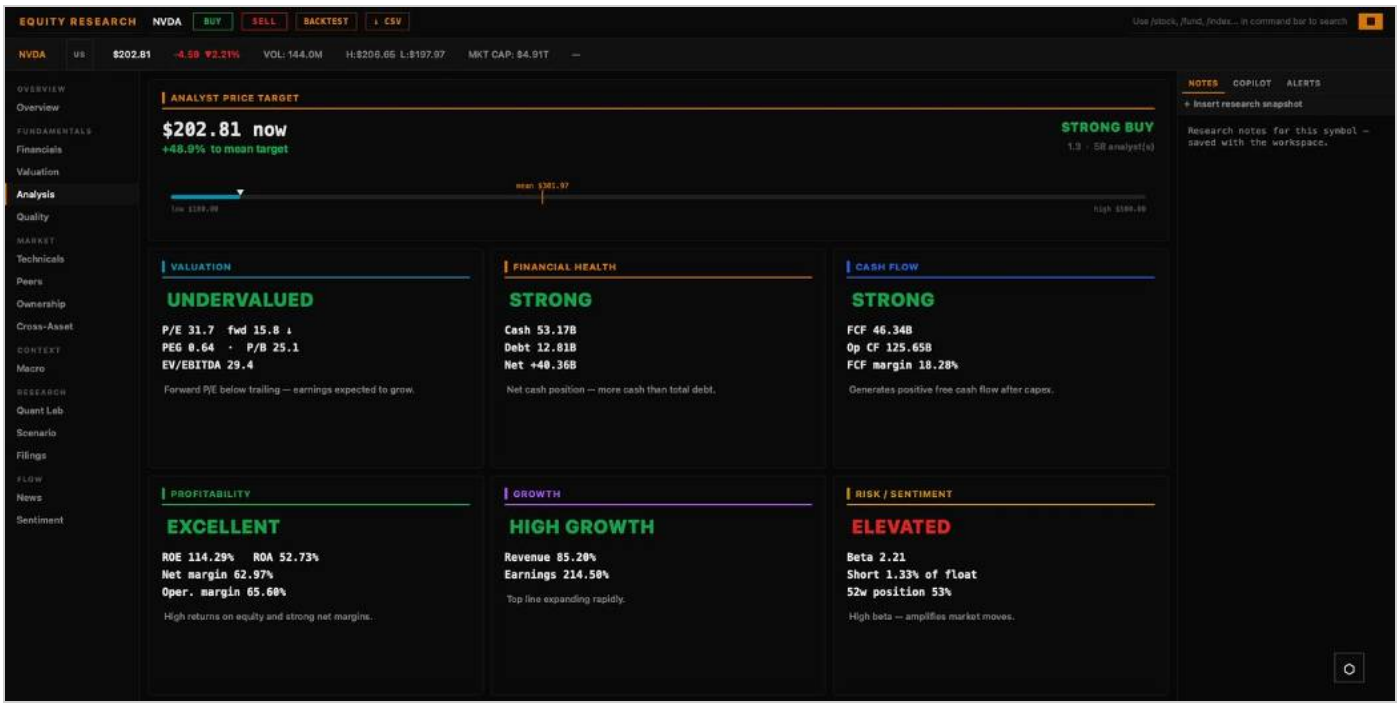


VALUATION Lab (NVDA) — fair value / upside / price-implied-growth tiles, a bear/base/bull spread, and seven editable DCF assumptions that recompute live.

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

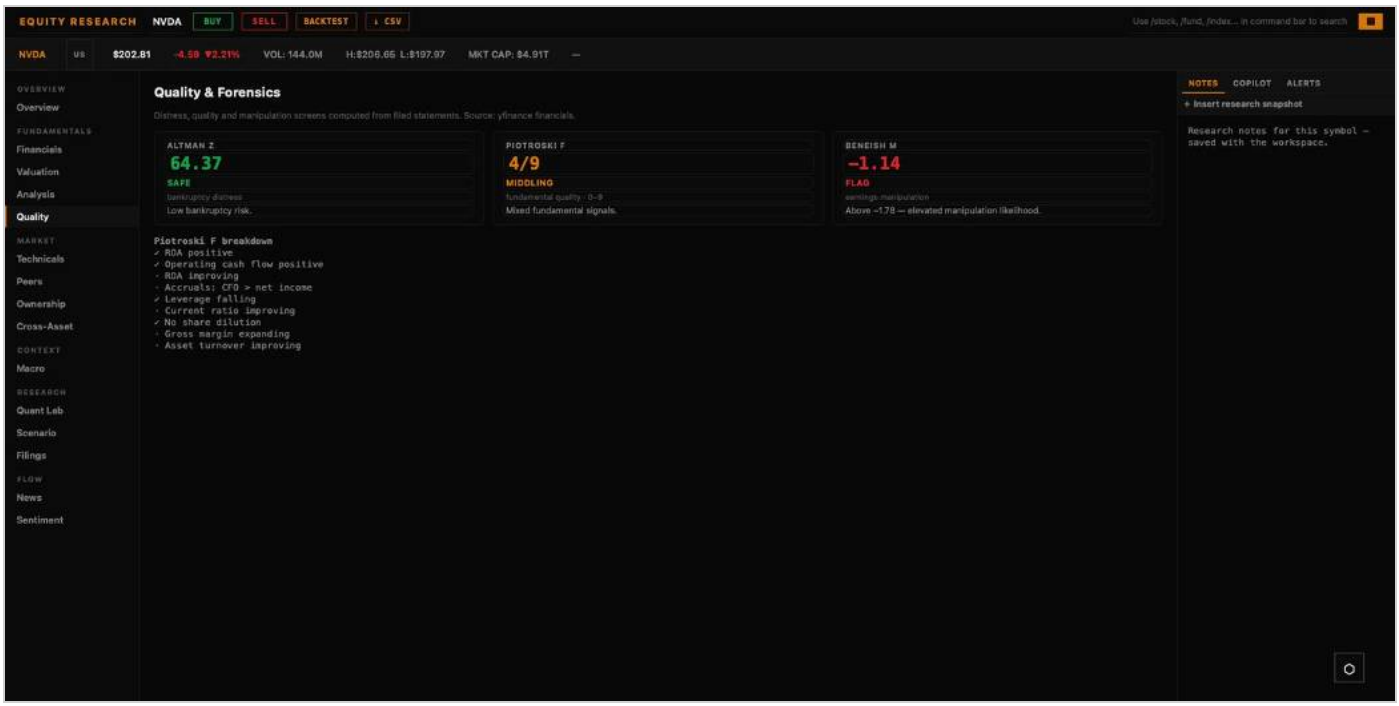


ANALYSIS (NVDA) — an analyst price-target gauge plus six colour-coded rating cards (Valuation, Financial Health, Cash Flow, Profitability, Growth, Risk/Sentiment).

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.

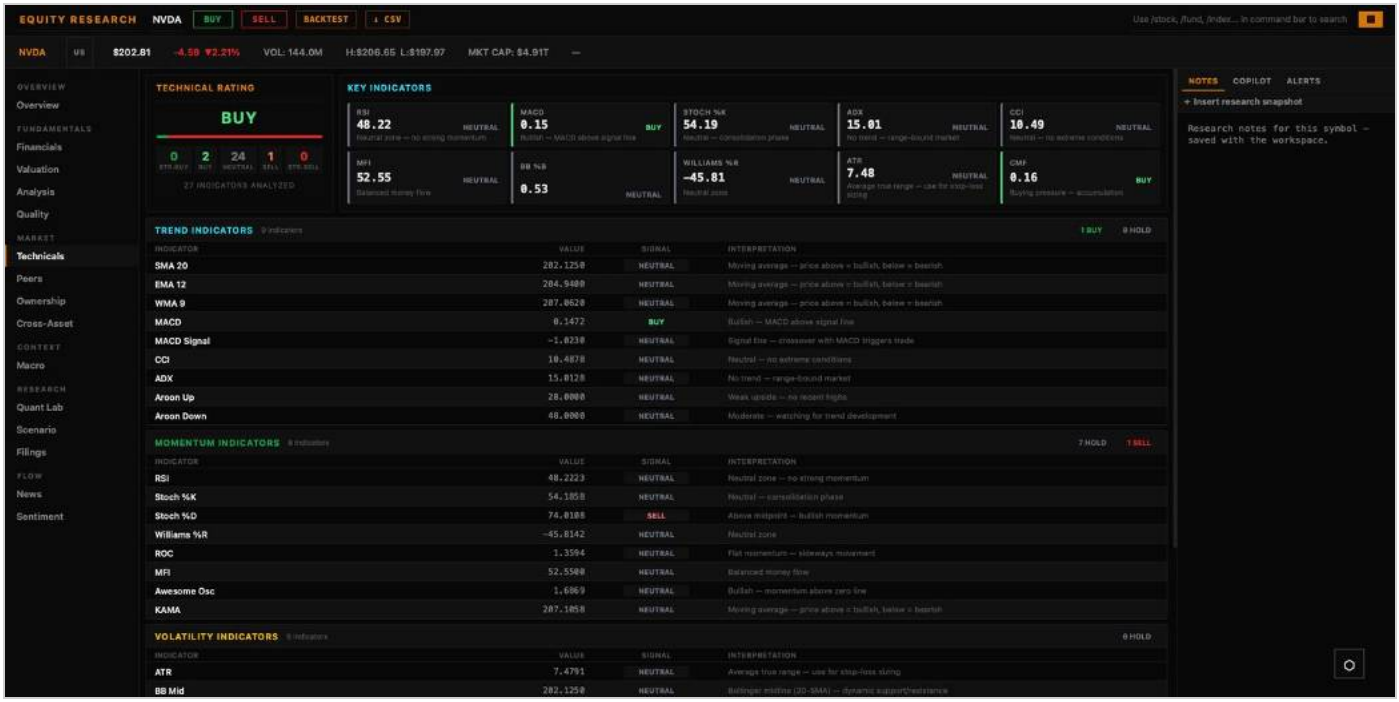
> **At a glance, in FUNDAMENTALS you can…**
>
> - Read the three annual statements with ratios, DuPont and trends (and export to CSV); run a live editable DCF with reverse-DCF and a bear/base/bull spread.
> - Get six plain-English fundamental verdicts and three forensic scores (Altman Z, Piotroski F, 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

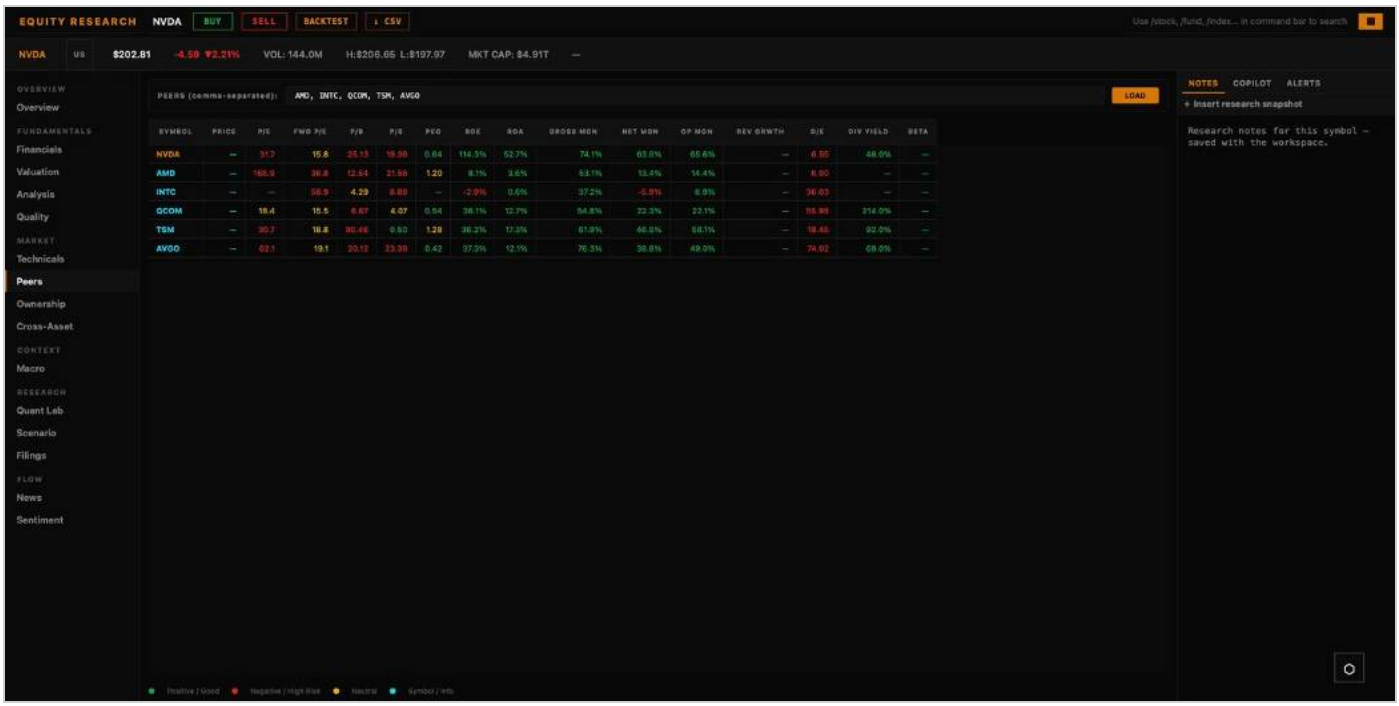


TECHNICALS (NVDA) — a TradingView-style rating (STRONG BUY…STRONG SELL) with a bull/bear tally, 10 key-indicator cards, and full Trend / Momentum / Volatility / Volume tables (value · signal · interpretation).

~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

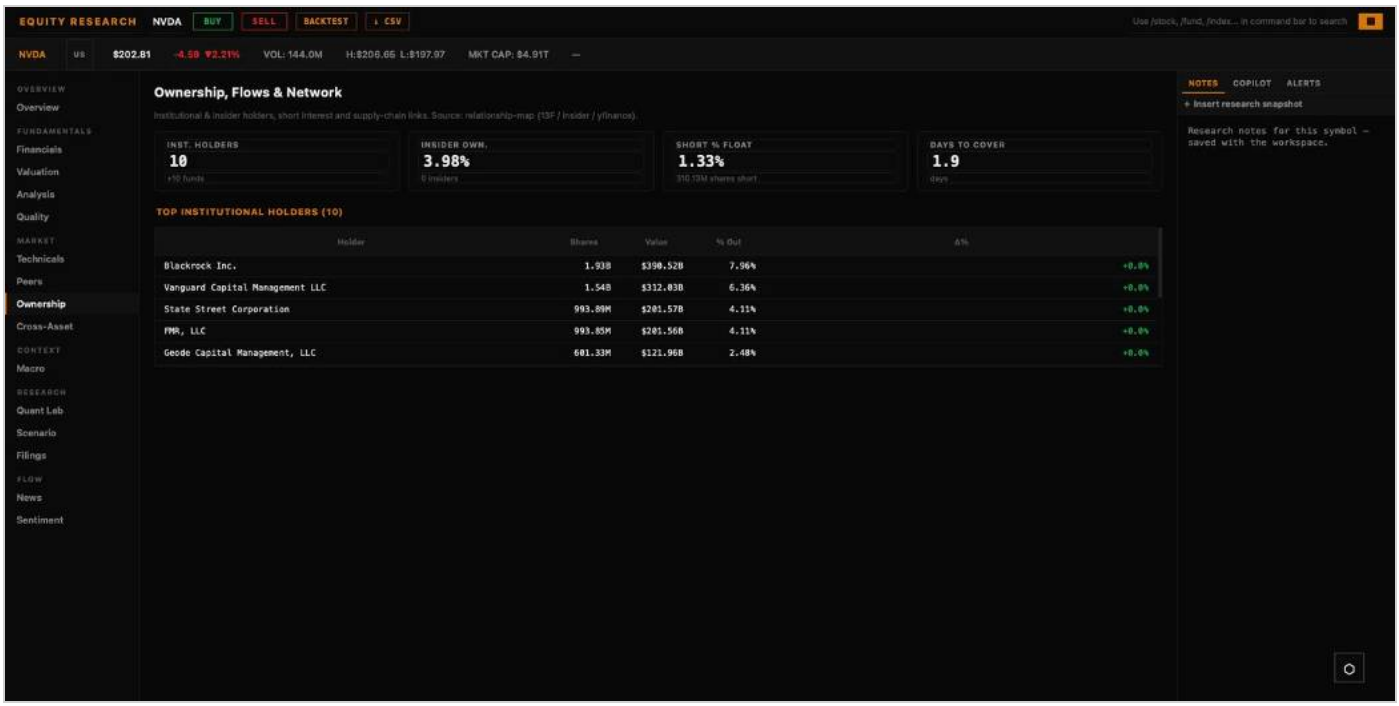


PEERS (NVDA vs its set) — an editable comma-separated peer list + LOAD, and a 16-metric colour-coded comparison grid with a good/neutral/risk legend.

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.

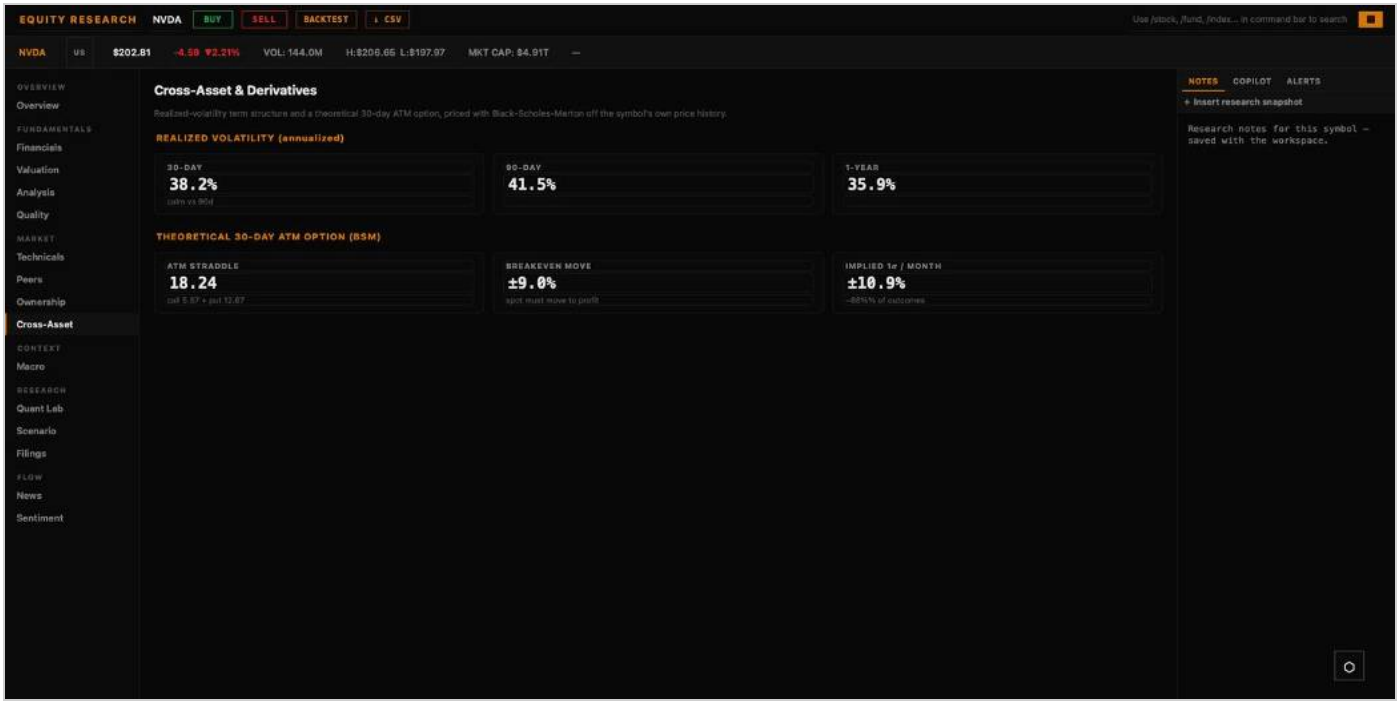
4.3 Ownership — holders, flows & network



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).

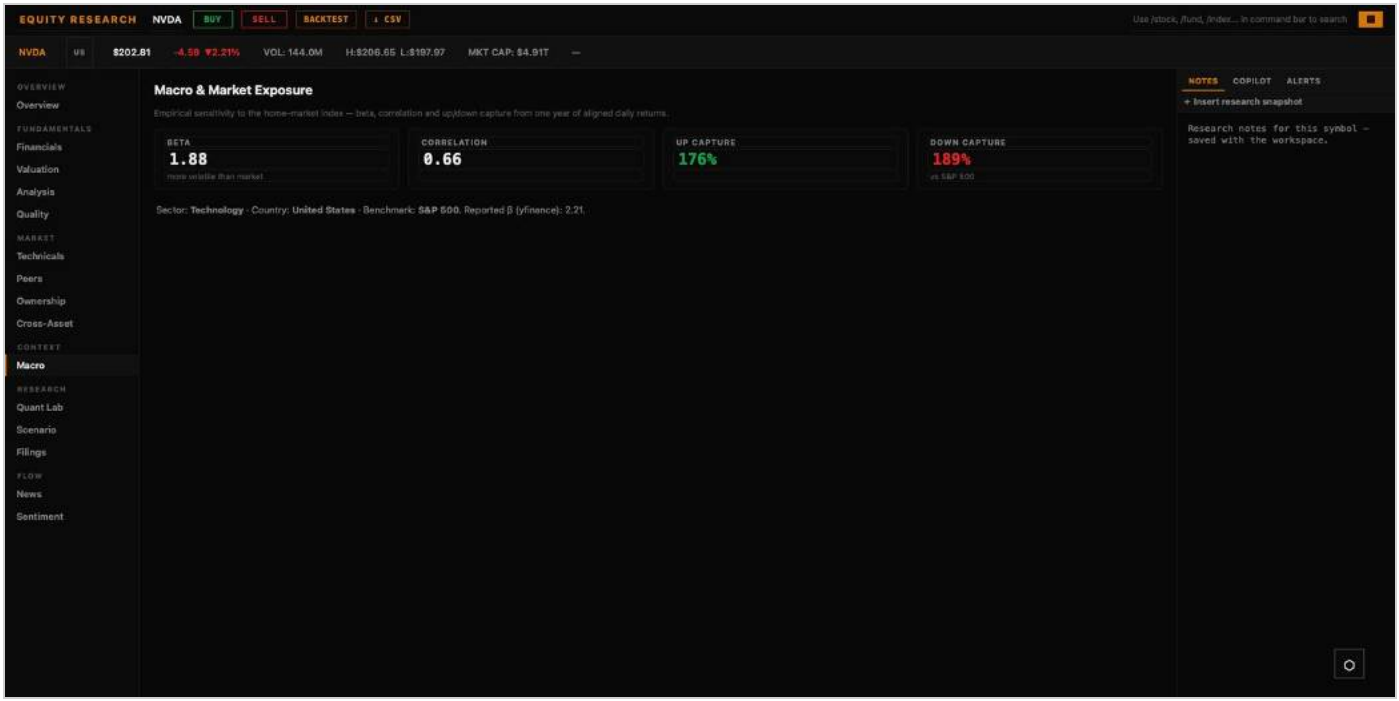
> **At a glance, in MARKET you can…**
>
> - Read a technical consensus from ~24 indicators, compare 16 fundamentals across an editable peer set, inspect institutional/insider ownership and supply-chain links, and gauge realized volatility + a theoretical option.

5 CONTEXT & RESEARCH

RAIL GROUPS

Macro exposure, quant factor stats, DCF stress scenarios, and SEC EDGAR filings.

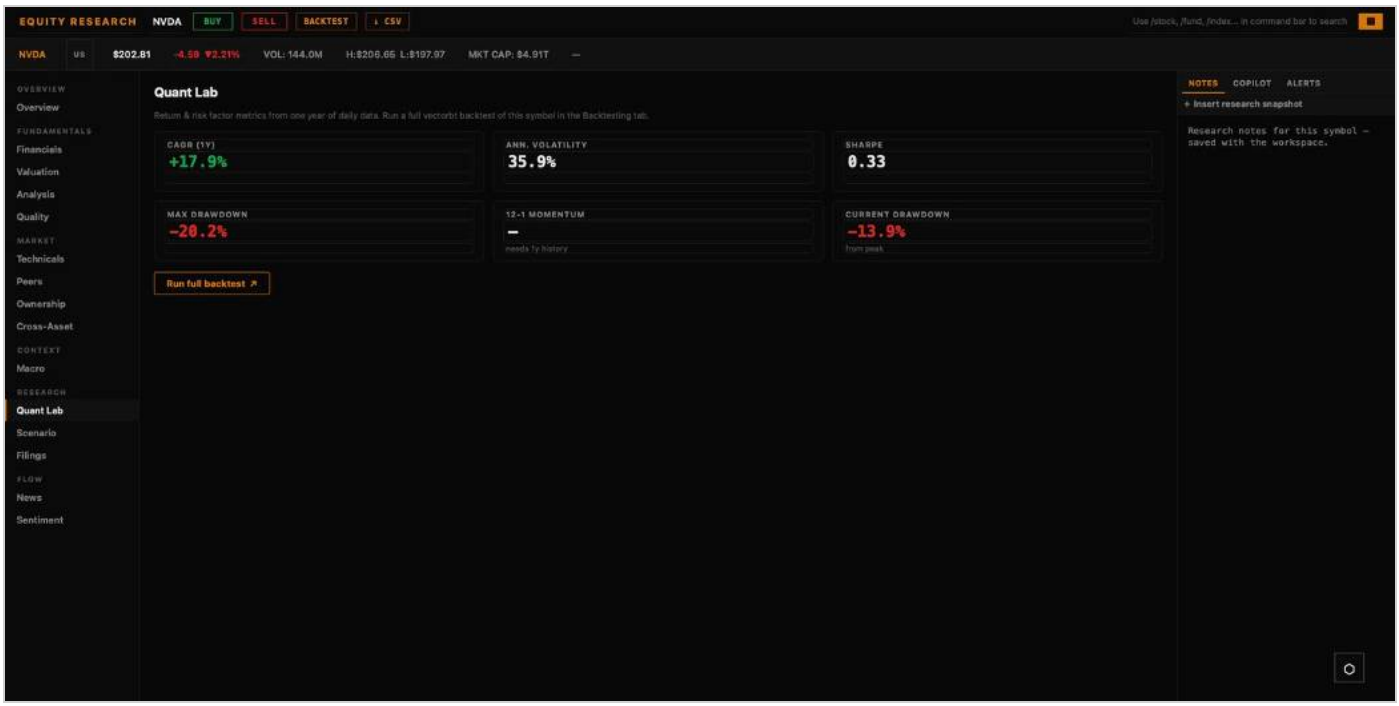
5.1 Macro — market exposure



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.

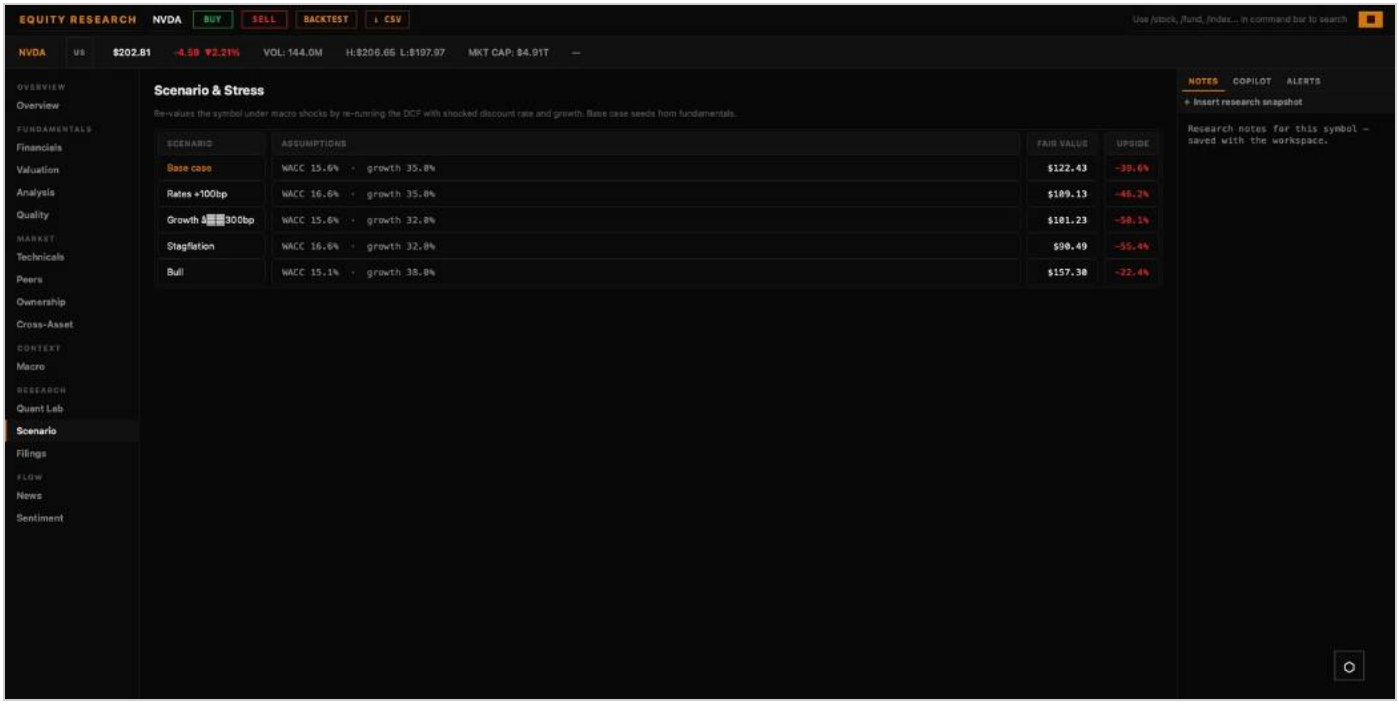
5.2 Quant Lab — factor statistics



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.

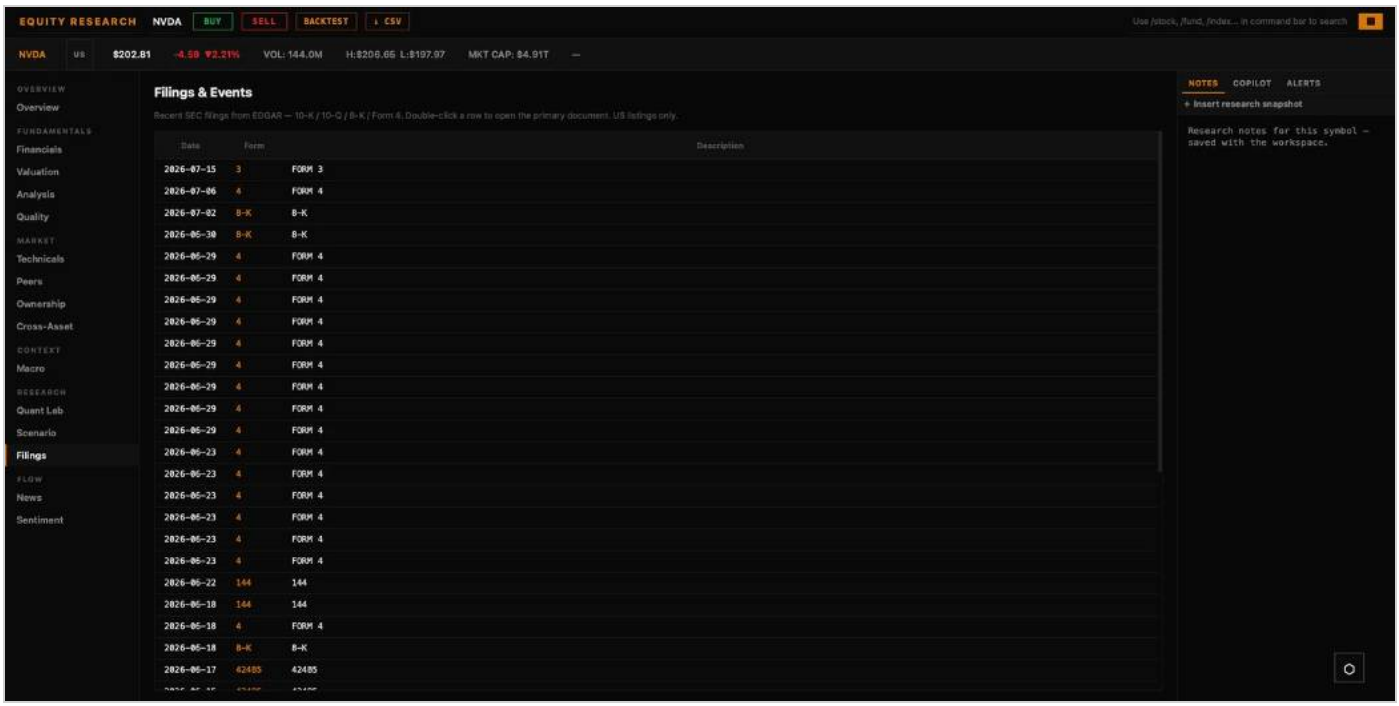
5.3 Scenario — DCF stress



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.

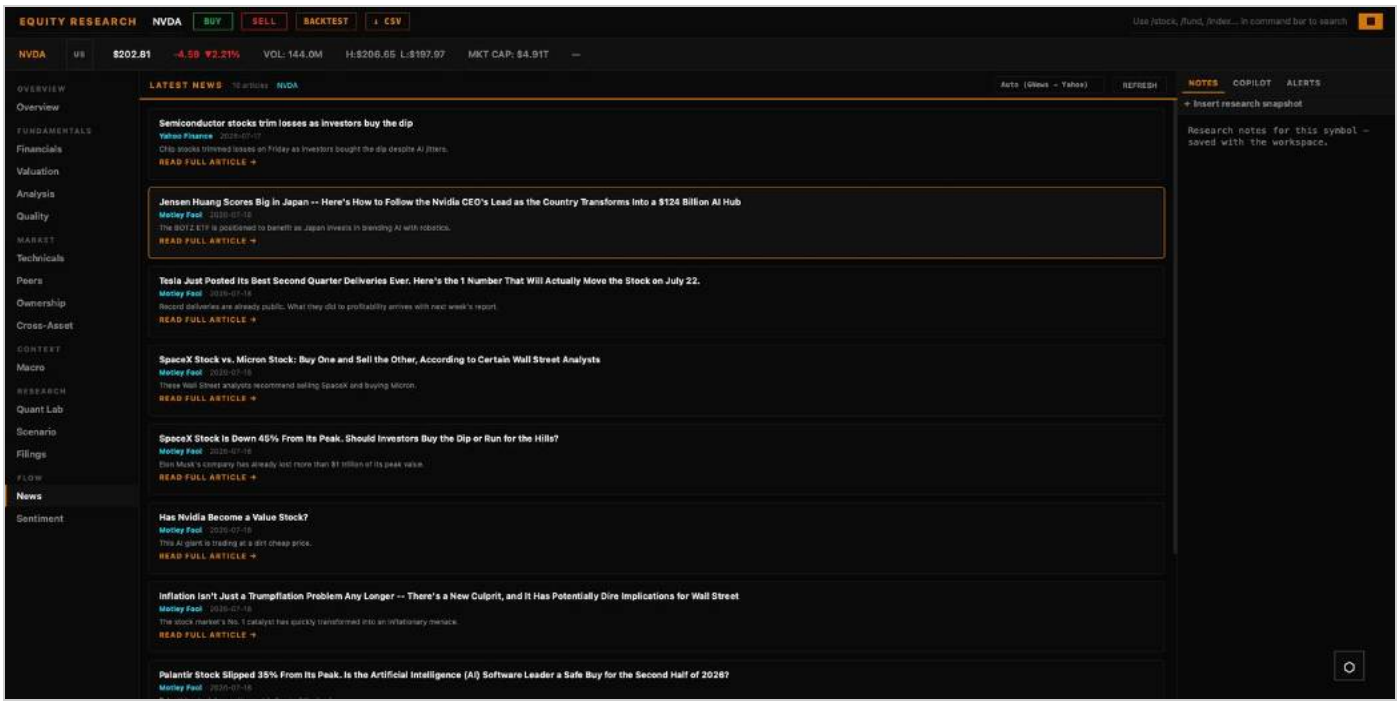
A read-only table (Date · Form · Description) of up to ~40 recent EDGAR filings; **double-click a row** to open its primary document in the browser. **US listings only** — non-US symbols show a clear "EDGAR is US-only" message.

> **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

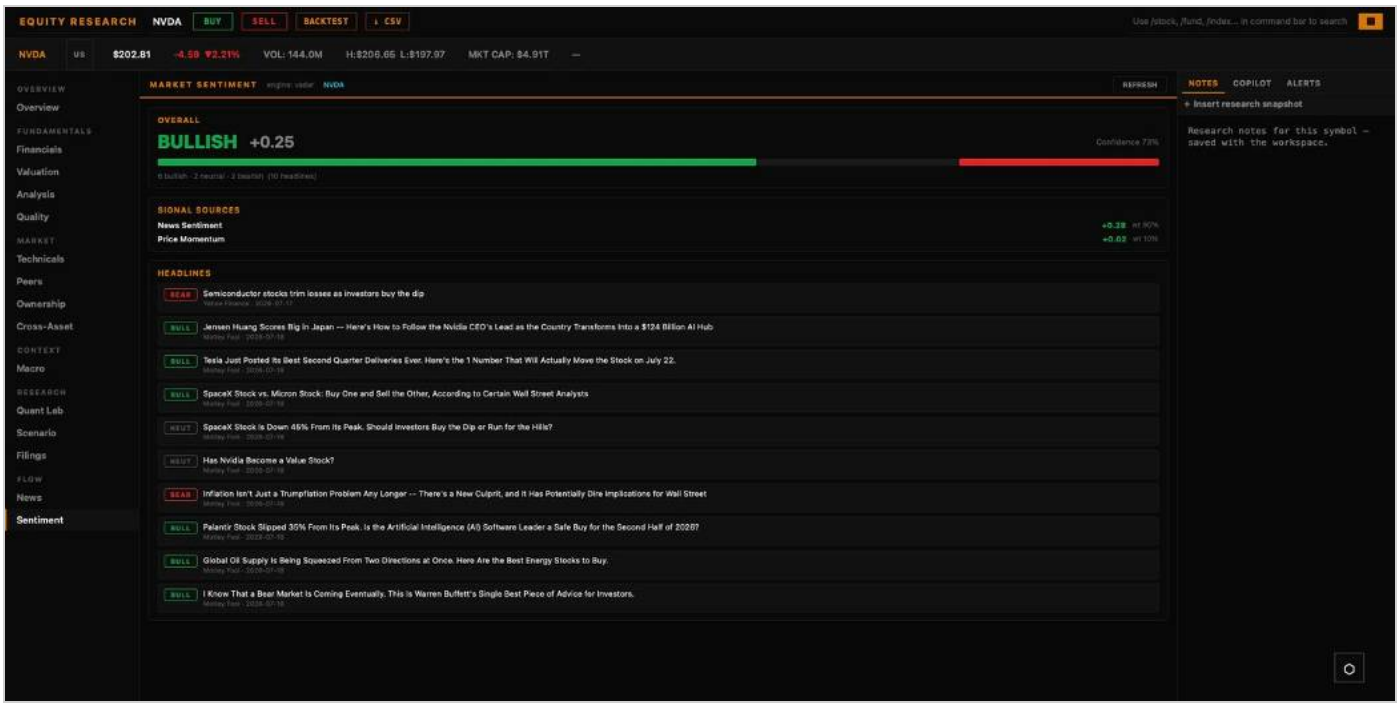
6.1 News



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.

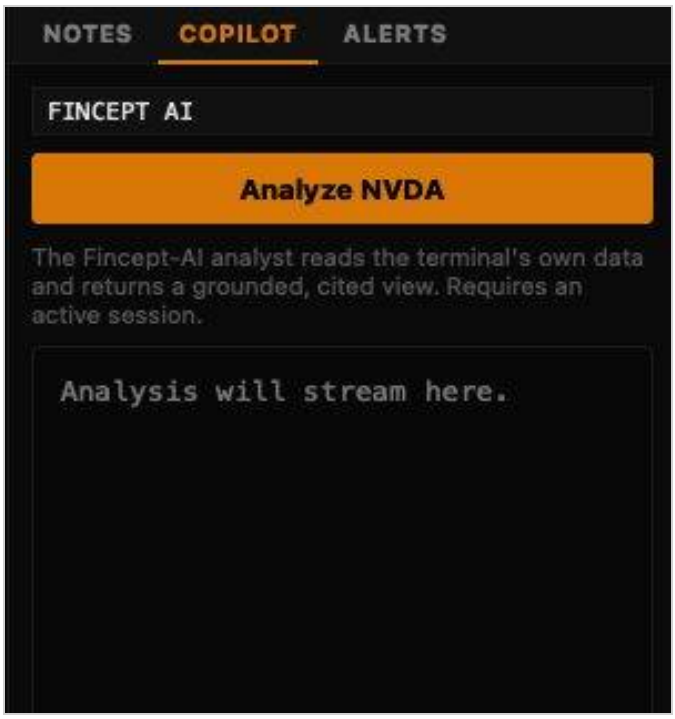
The terminal's own keyless sentiment read: an **Overall** verdict (score + confidence + distribution bar + counts), a **Signal Sources** breakdown (news NLP + price momentum, and Adanos when keyed, each with score & weight), and a **Headlines** list of individually scored articles. One control: **REFRESH**.

> **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.

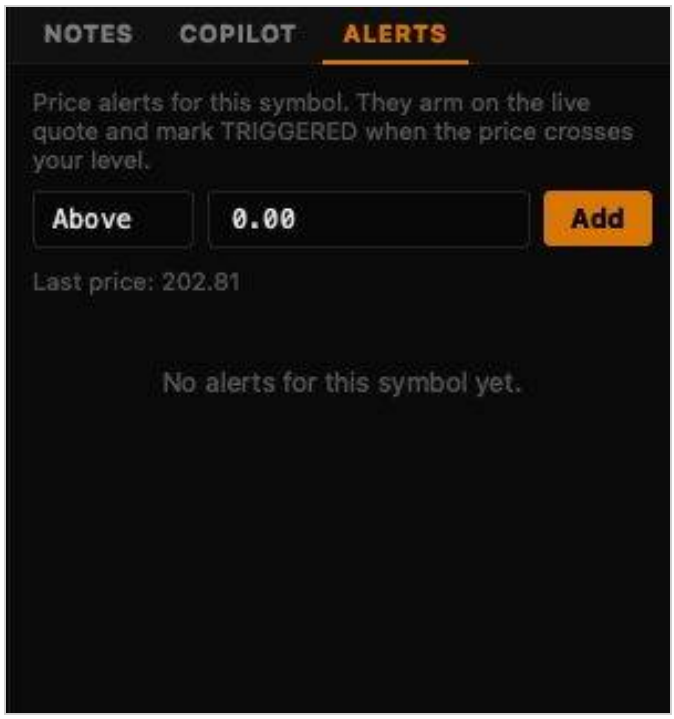
7 The research dock

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.

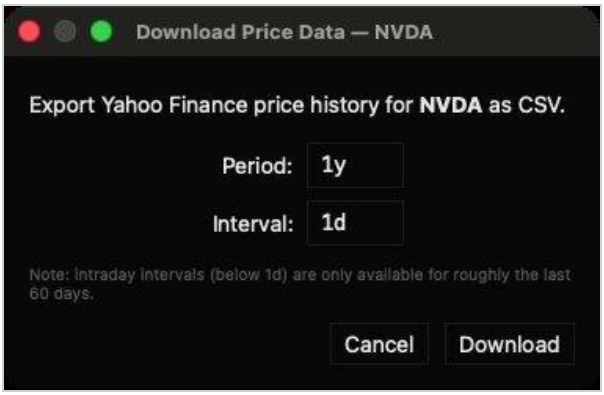
> **Copilot & credits.** The Copilot analyst is the **only credit-spending action** on this tab — it calls the Fincept-AI LLM/agent backend. There is no model picker or cancel; it needs an active session. Alerts are in-pane only (no OS/push notification) and a triggered alert can only be deleted, not 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**

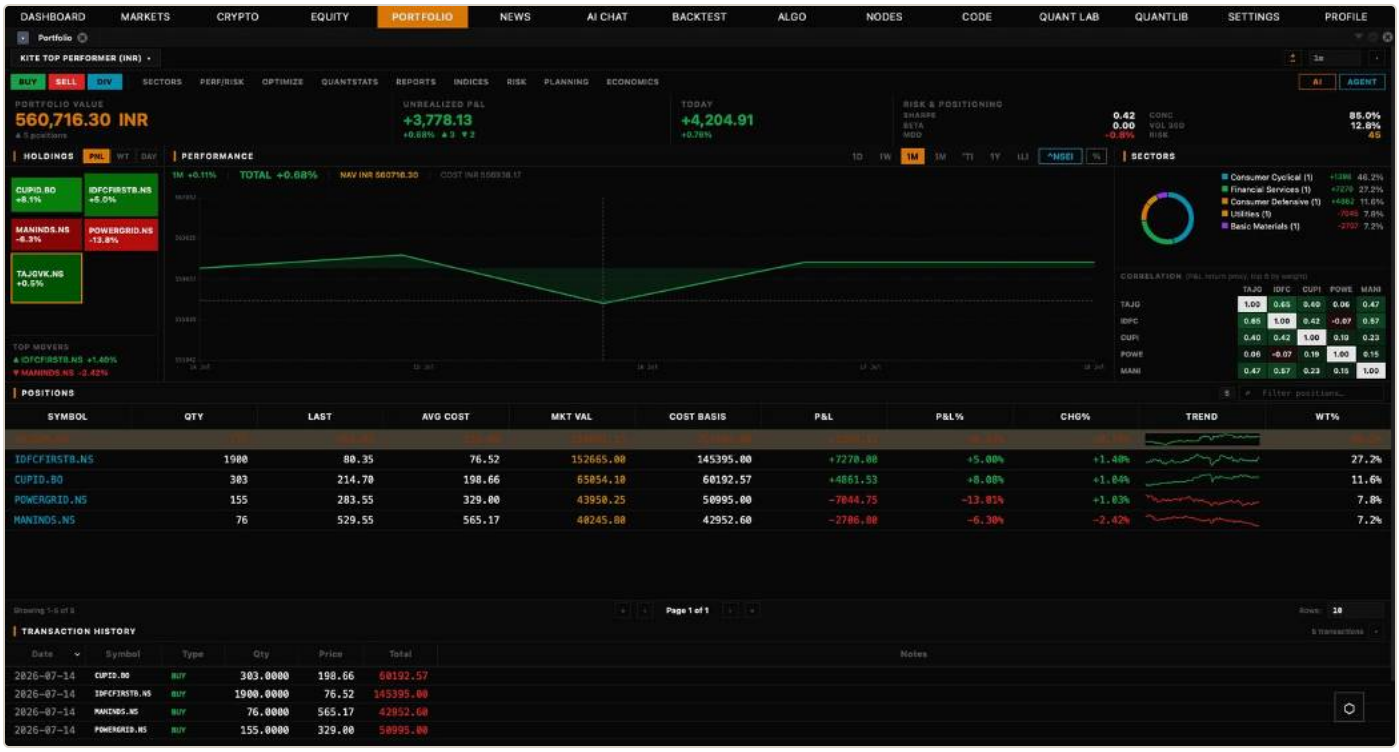
WHERE THE DATA COMES FROM — IN ONE LINE

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`).

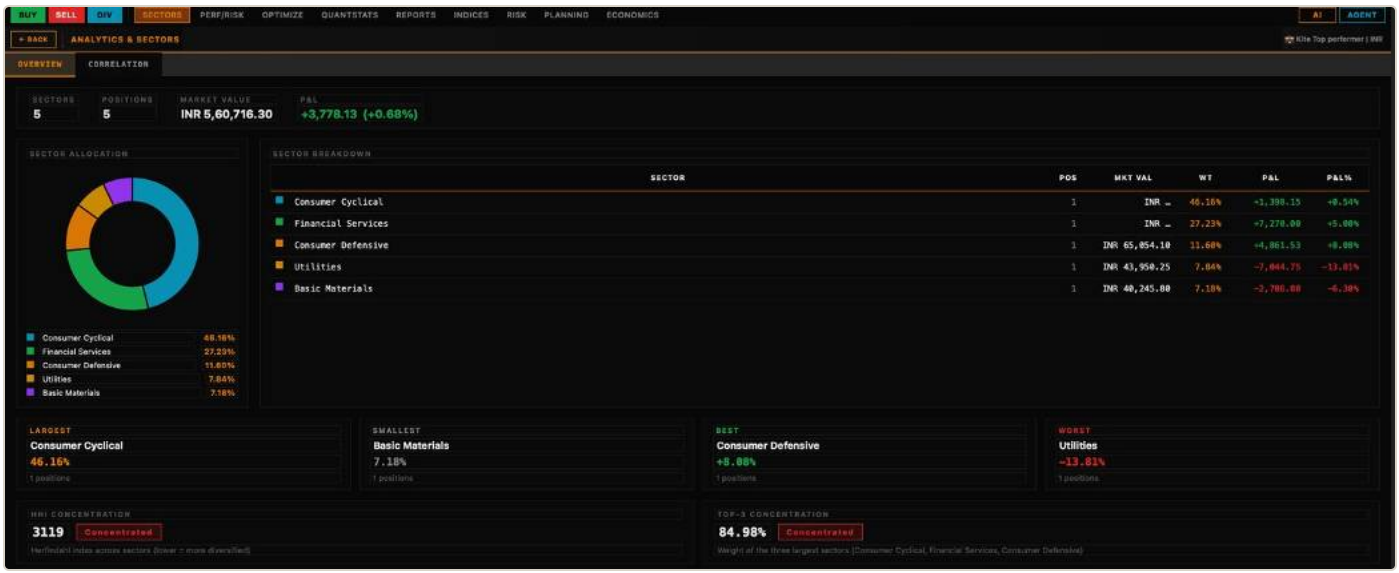
Transaction history

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.

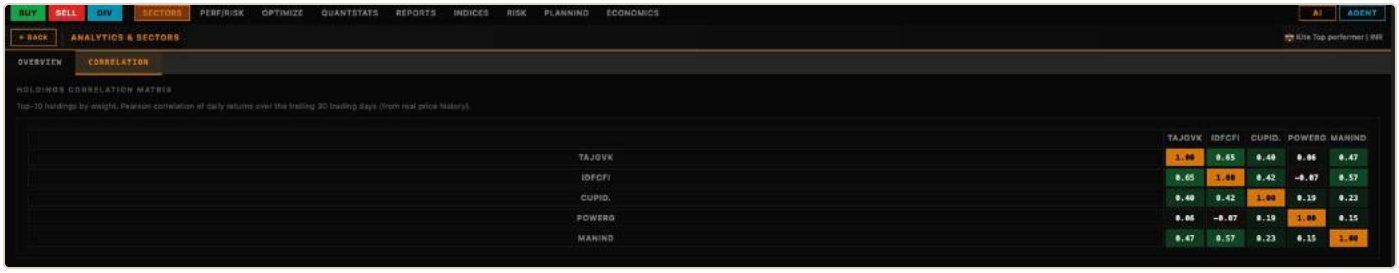
SECTOR BREAKDOWN TABLE

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.



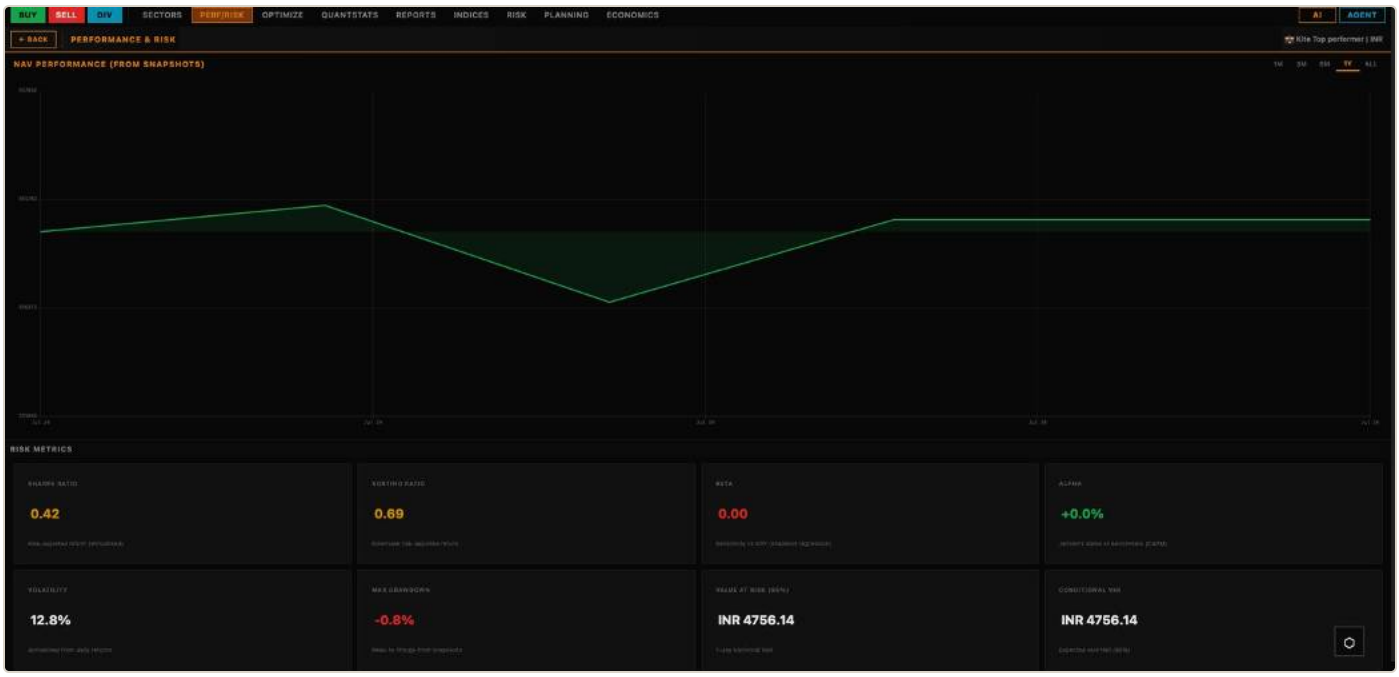
Correlation tab. Pearson correlation of daily returns over the trailing 30 trading days for the top-10 holdings by weight; diagonal 1.00 on amber, positive green, negative red.

> **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.

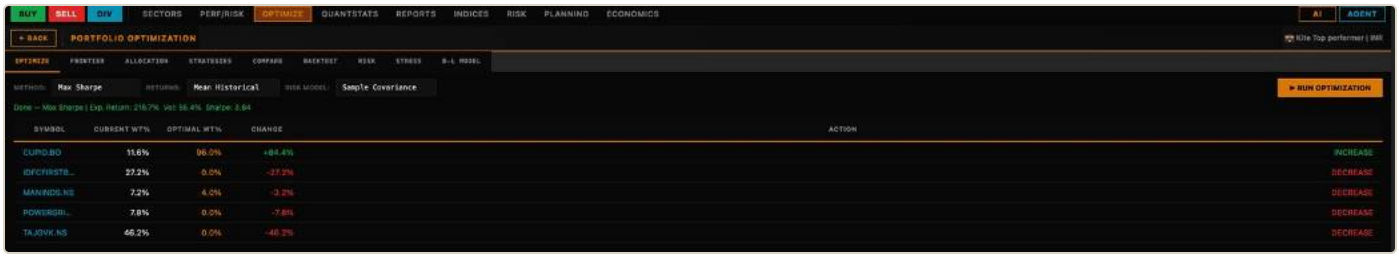
06 Portfolio optimization

VIEW 3 · RUN

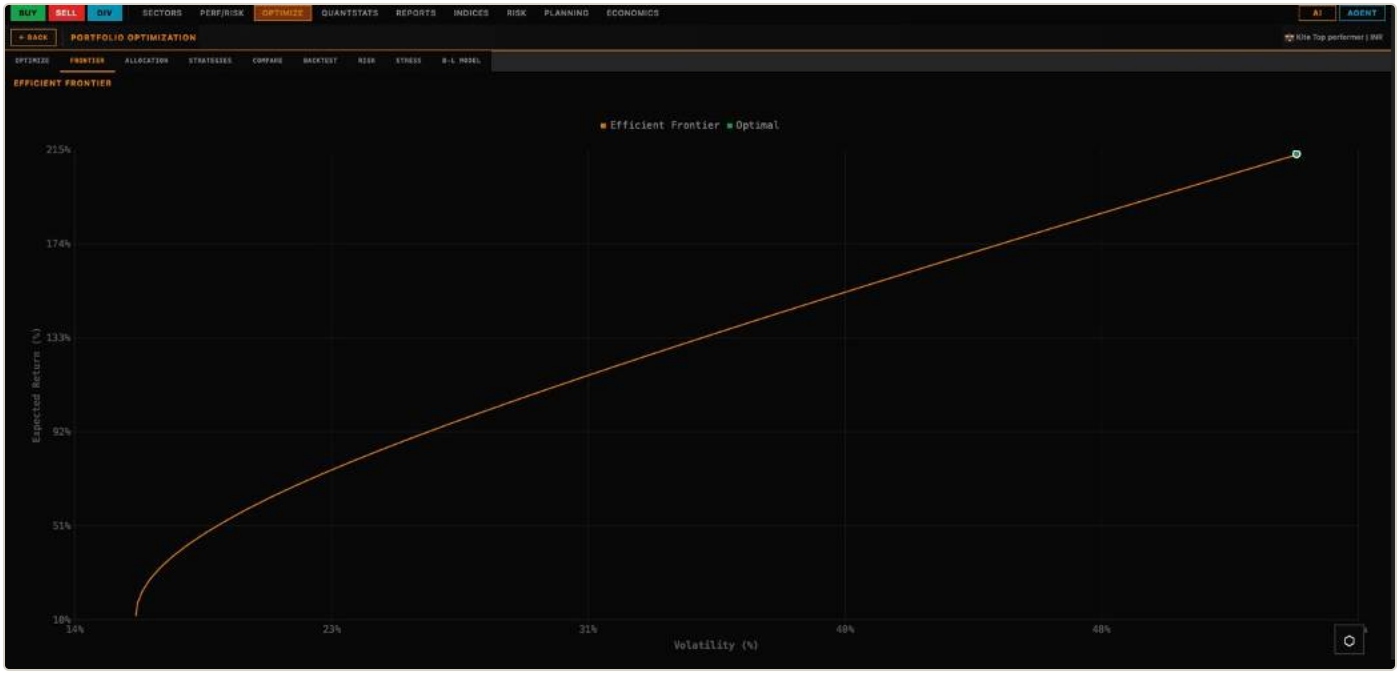
The OPTIMIZE view runs a mean-variance / risk-based optimizer and lays the results across nine tabs. Pick a **Method** and press ▶ RUN OPTIMIZATION; results are computed server-side and cached for 10 minutes.



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).



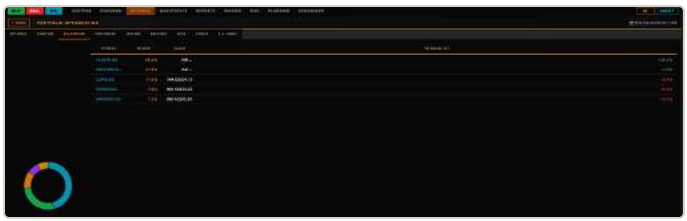
Frontier tab. The efficient frontier (amber) with the optimal max-Sharpe point (green). X = volatility %, Y = expected return %.

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.



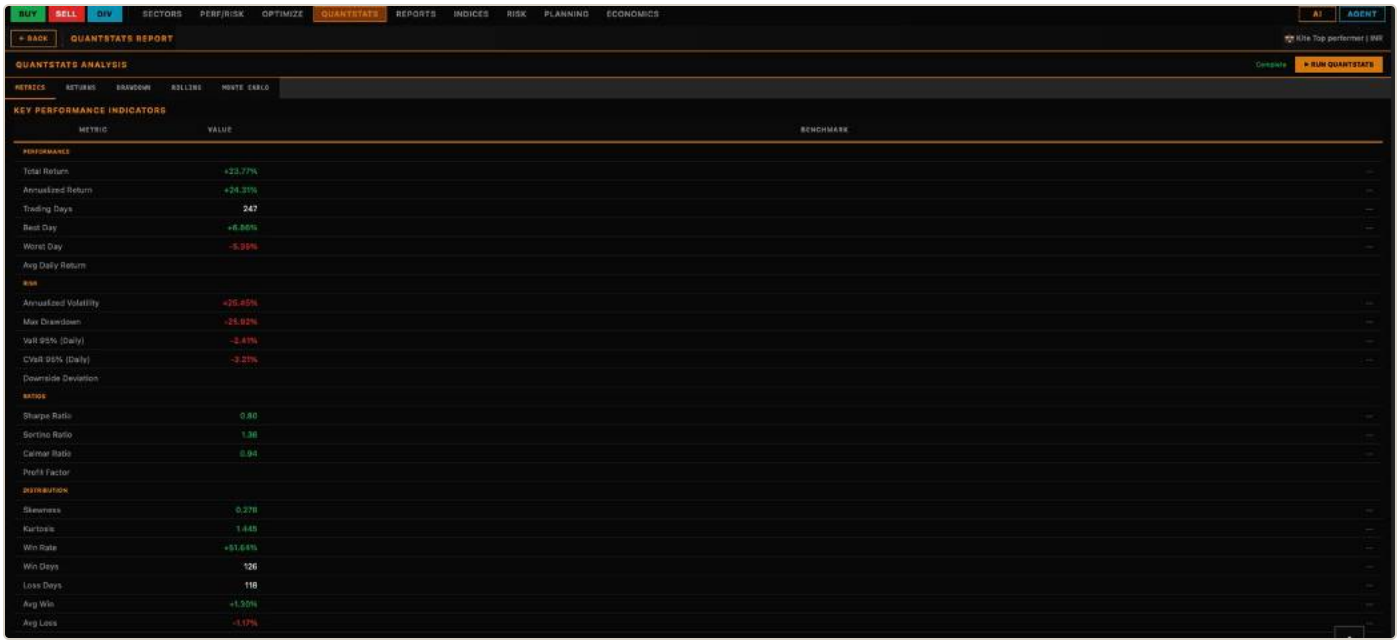
Monte Carlo — 1,000 GBM return paths over 252 trading days (bright line = median), with a percentile strip: median return, 5th / 95th percentile, probability of loss and expected max drawdown.

> **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.



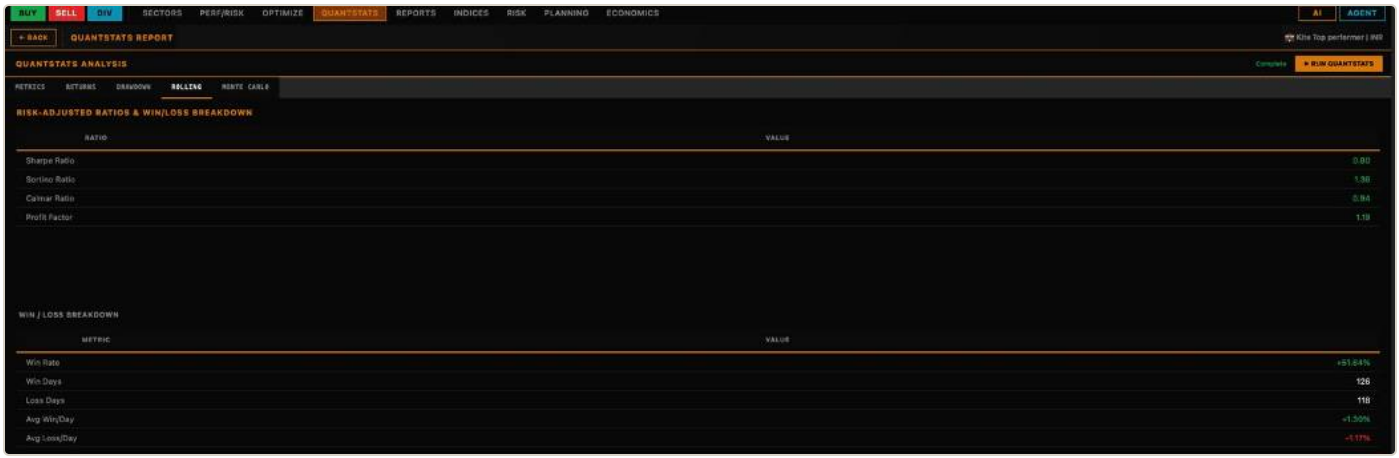
Metrics tab. Key performance indicators grouped Performance / Risk / Ratios / Distribution — total & annualised return, best/worst day, volatility, drawdown, VaR/CVaR, Sharpe/Sortino/Calmar, skew/kurtosis, win rate.



Returns — win/loss cards + distribution table.



Drawdown — max-drawdown hero + risk table.

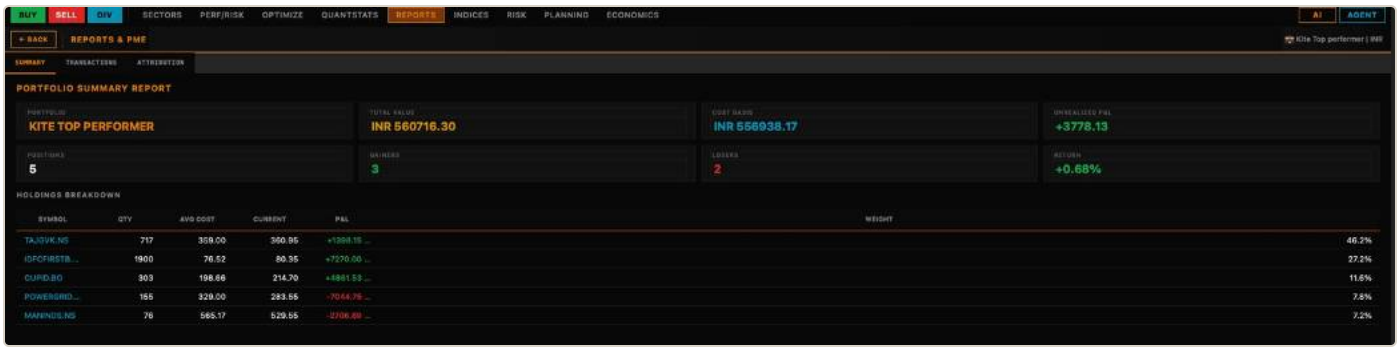


Rolling — risk-adjusted ratios (Sharpe / Sortino / Calmar / Profit Factor) and a win-/loss-day breakdown.

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.



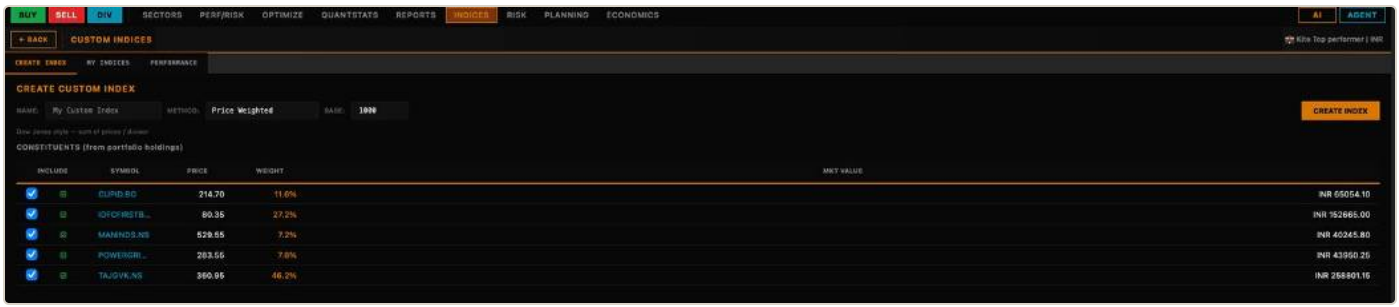
Attribution — weight, return, contribution, P&L and an OUTPERFORM / NEUTRAL / UNDERPERFORM verdict.

> **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.

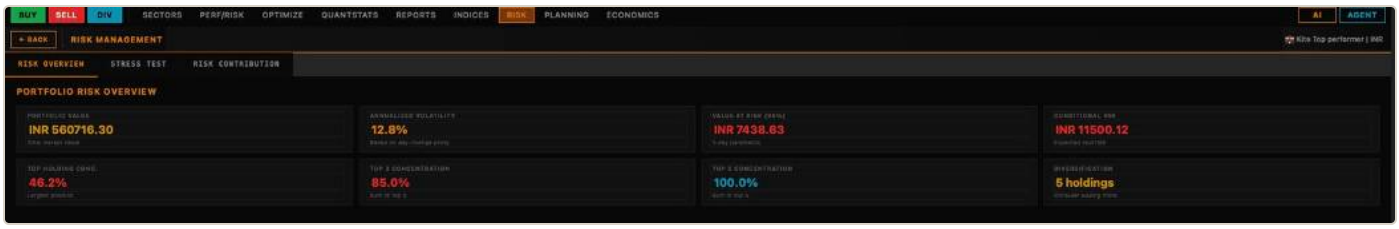
CONTROL	WHAT IT DOES
Name	Index name (required).
Method	Price Weighted · Market Cap · Equal Weighted · Float Adjusted · Fundamental · Modified Market Cap · Factor · Risk Parity · Geometric Mean · Capped.
Base	Starting index level (default 1000).
Include ☑	Per-holding checkbox — choose the constituents.
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.

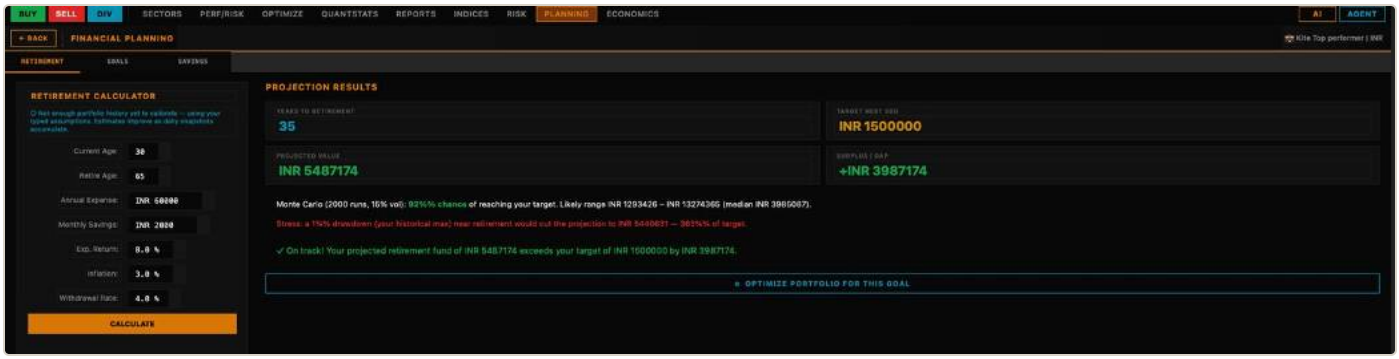
READING NOTE — STRESS IMPACT SCALES WITH BETA

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.



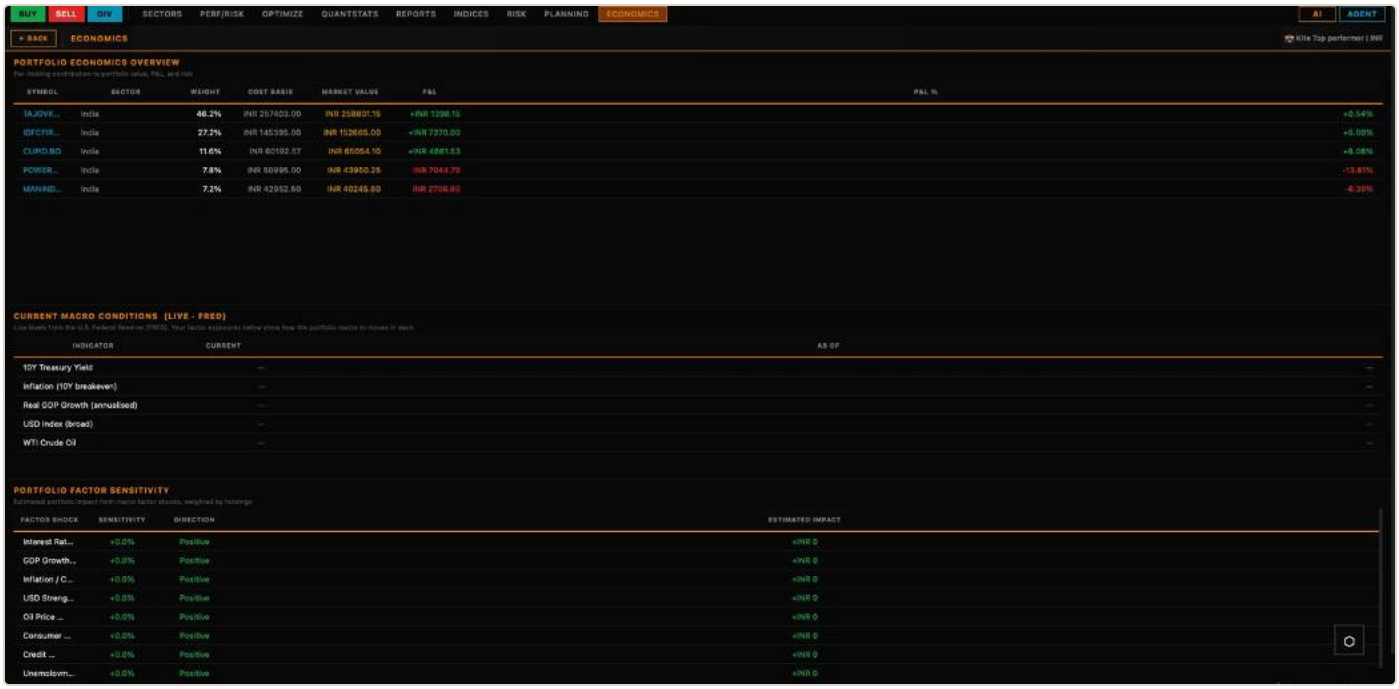
Savings — projects wealth across a band of savings rates vs your current rate.

> **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.

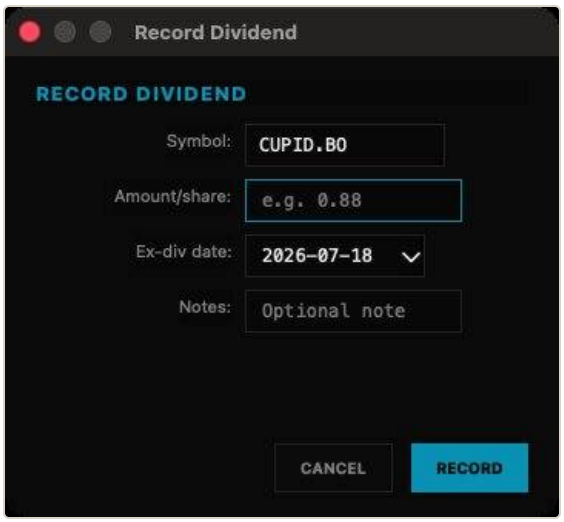
13 Trading — order entry & dividends

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).

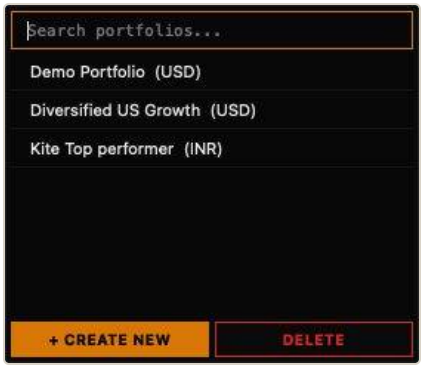
All eight portfolio forms at a glance

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.



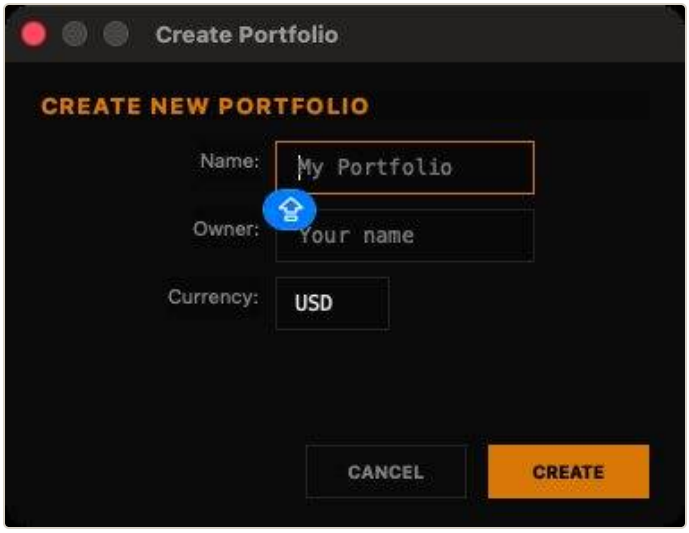
menu. Export CSV / JSON, Import JSON…, FFN Analysis, Backtest Portfolio.

CREATE & IMPORT

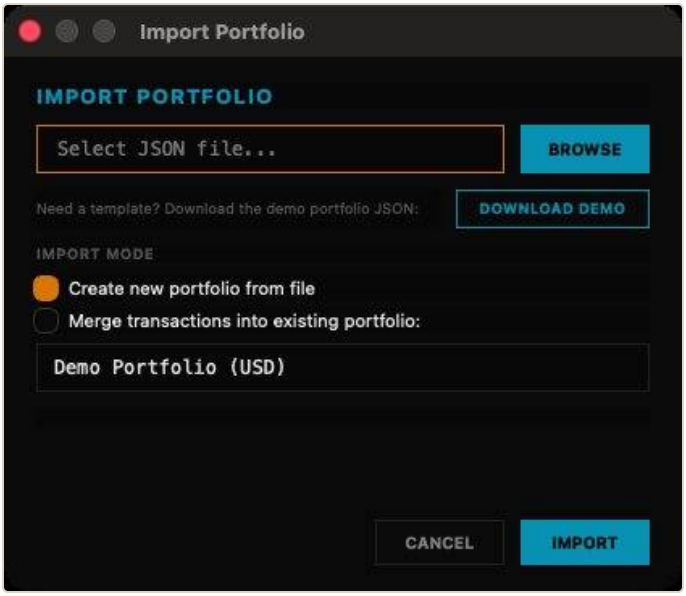
A new portfolio needs only a name (currency defaults to USD, chosen from 36 ISO codes). Import reads an exported JSON of transactions and replays them into a new or existing (merge) portfolio; a **Download Demo** button writes a sample file to start from.

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.



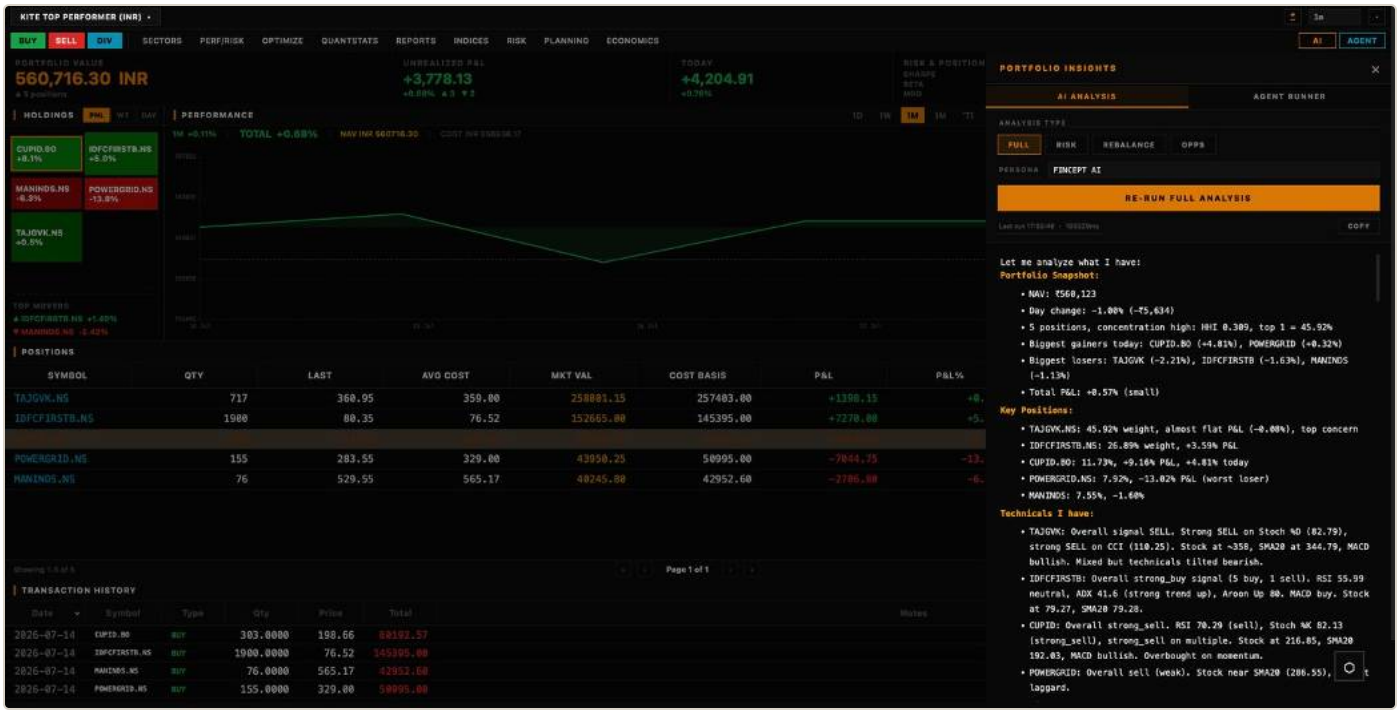
Import Portfolio — browse to a JSON file, choose **Create new** or **Merge into existing**, and import.

> **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.

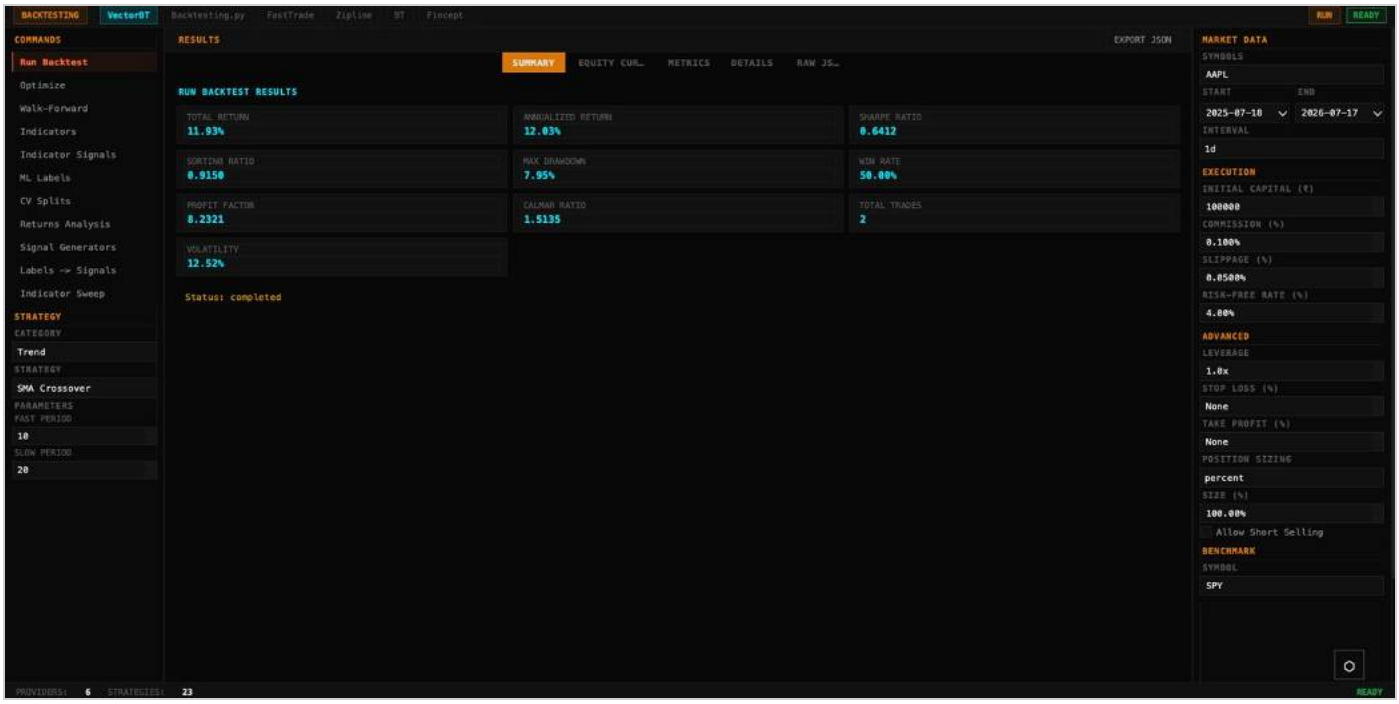
AT A GLANCE, IN BACKTEST YOU CAN…

- **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.

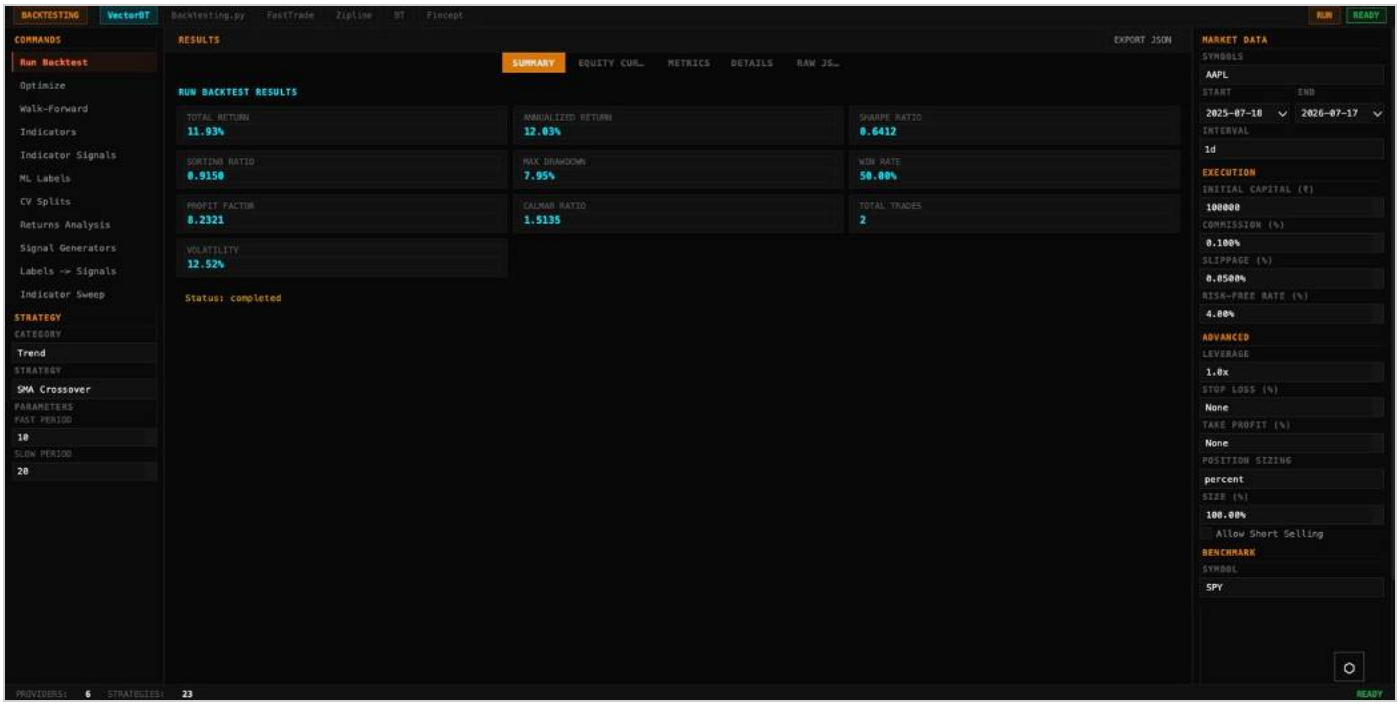
> **Symbol format matters.** Symbols are passed to the engine as typed. Non-US tickers must carry their exchange suffix (e.g. `RELIANCE.NS`, `PIDILITIND.NS`) or Yahoo will resolve them wrong. US symbols (`AAPL`, `SPY`) need no suffix.

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.

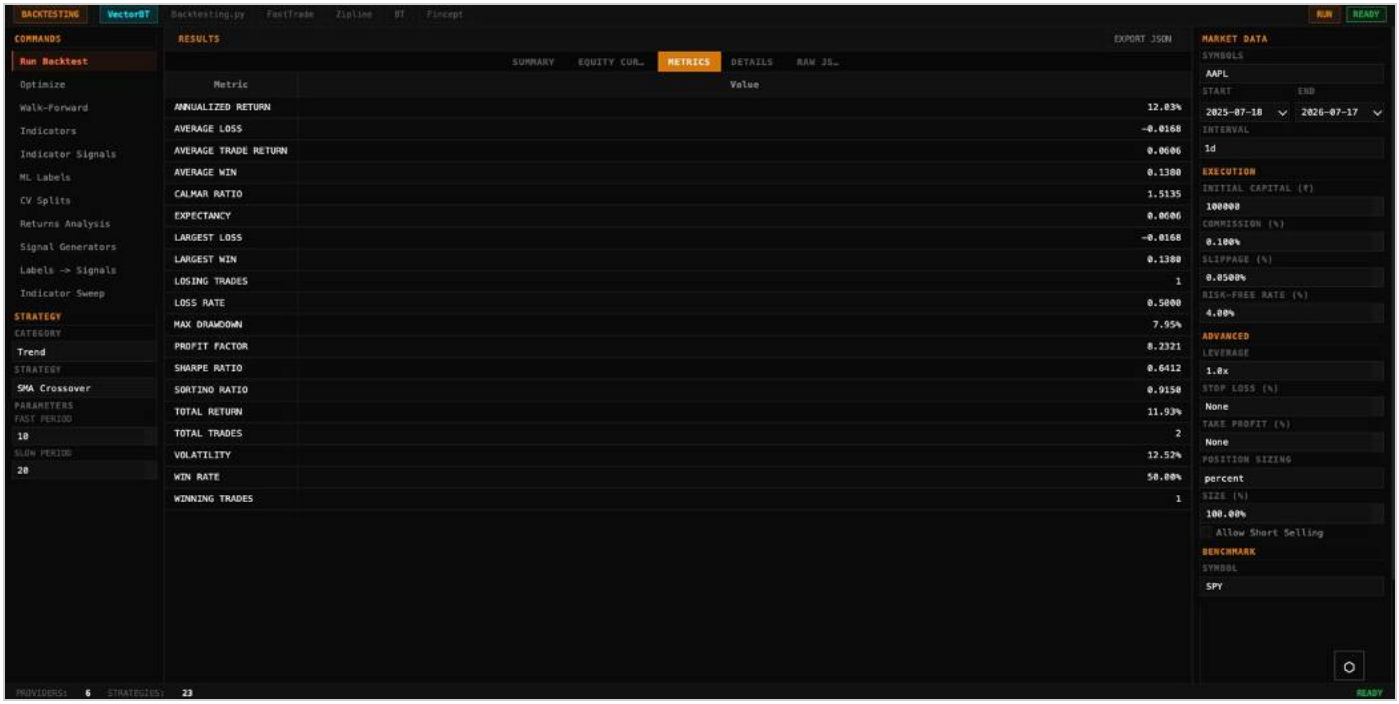
The card set adapts to the command: Optimize shows best-parameter & best-performance cards, Walk-Forward shows robustness cards, ML Labels shows a class distribution, and so on (see §6–§7).

② EQUITY CURVE — portfolio value & drawdown



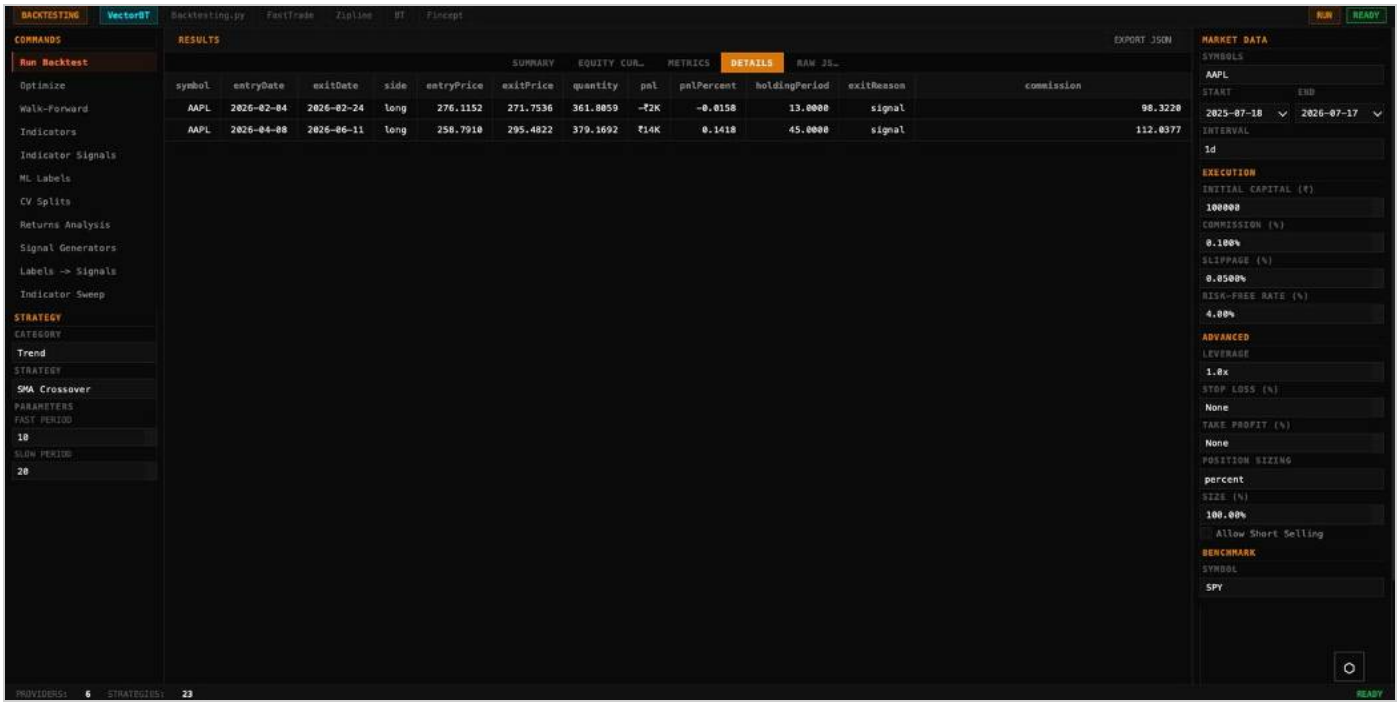
EQUITY CURVE — the portfolio value over time (left axis, engine-accent line) with an overlaid drawdown series (right axis, red %) when the drawdown is material.

③ METRICS — the full metric table



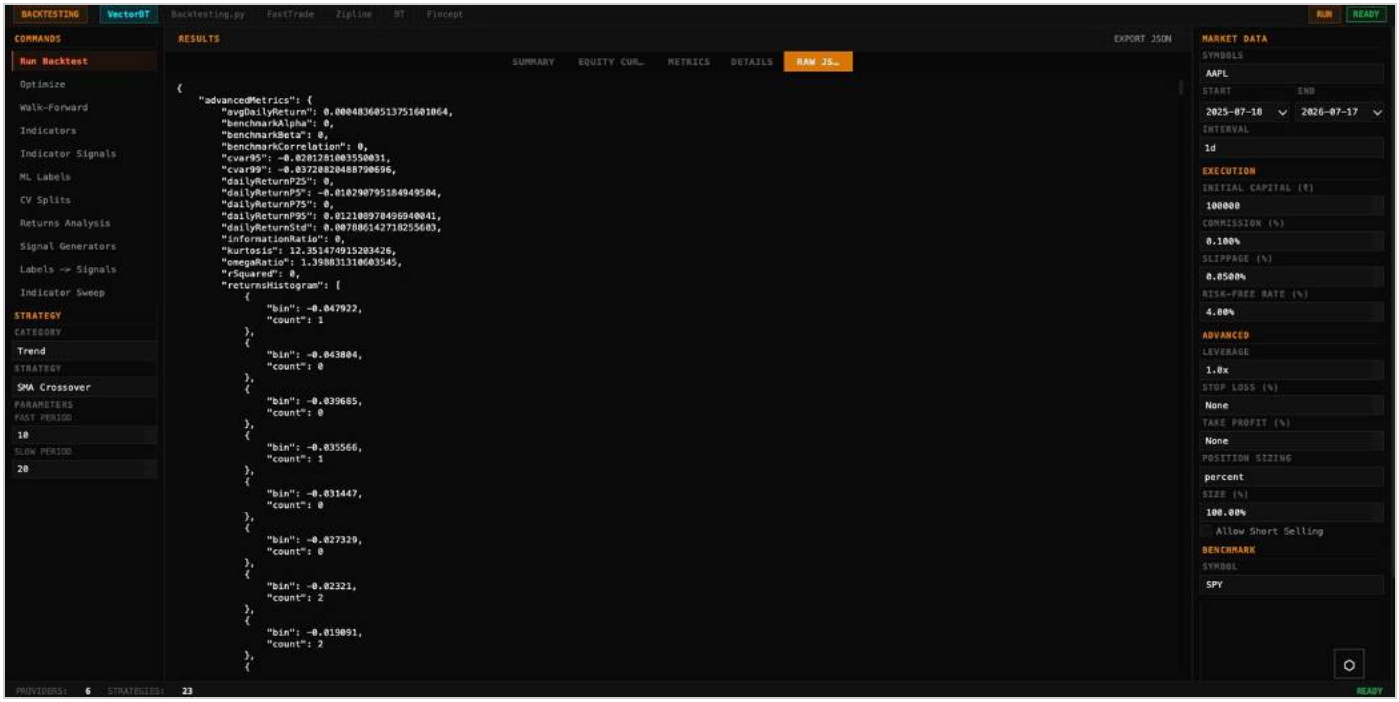
METRICS — every performance metric the engine emitted, as a Metric/Value table (annualized return, average win/loss, Calmar, expectancy, largest win/loss, loss rate, max drawdown, profit factor, Sharpe, Sortino, total return, total trades, volatility, win rate, winning/losing trades…).

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

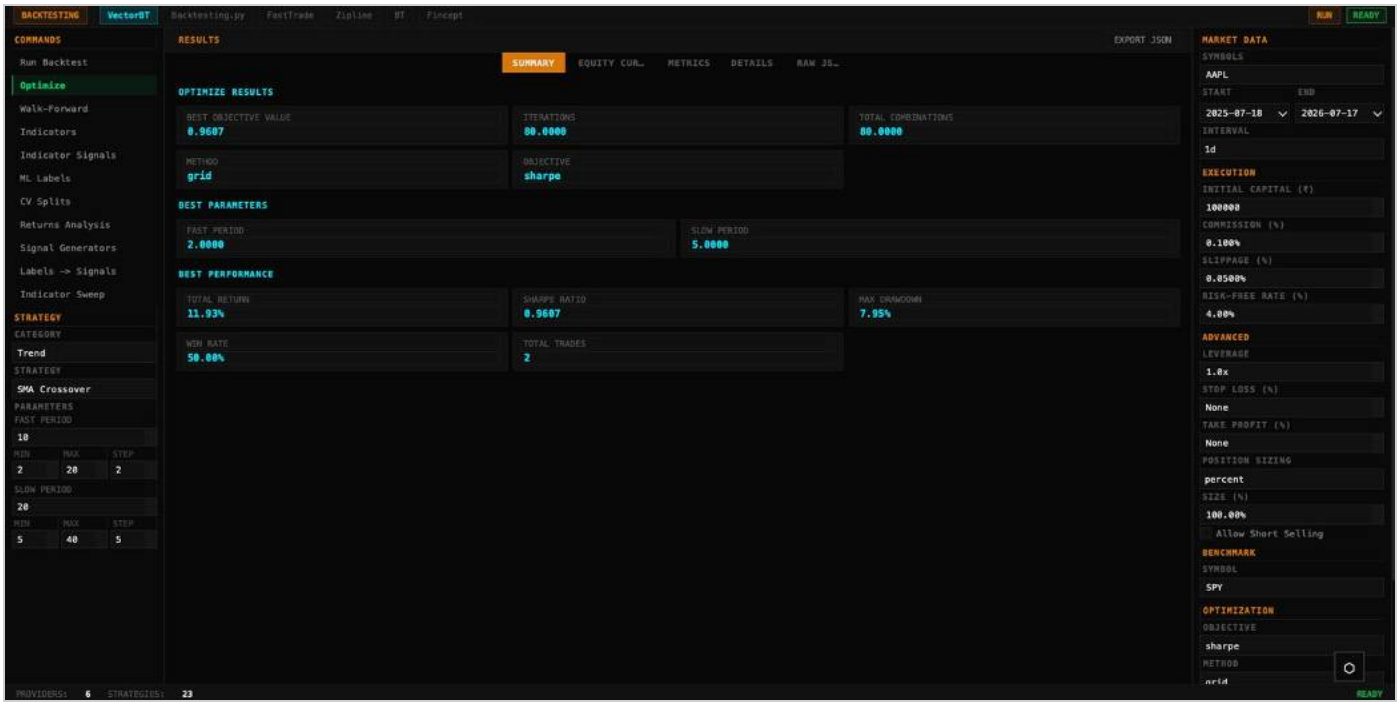
Run Backtest

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.

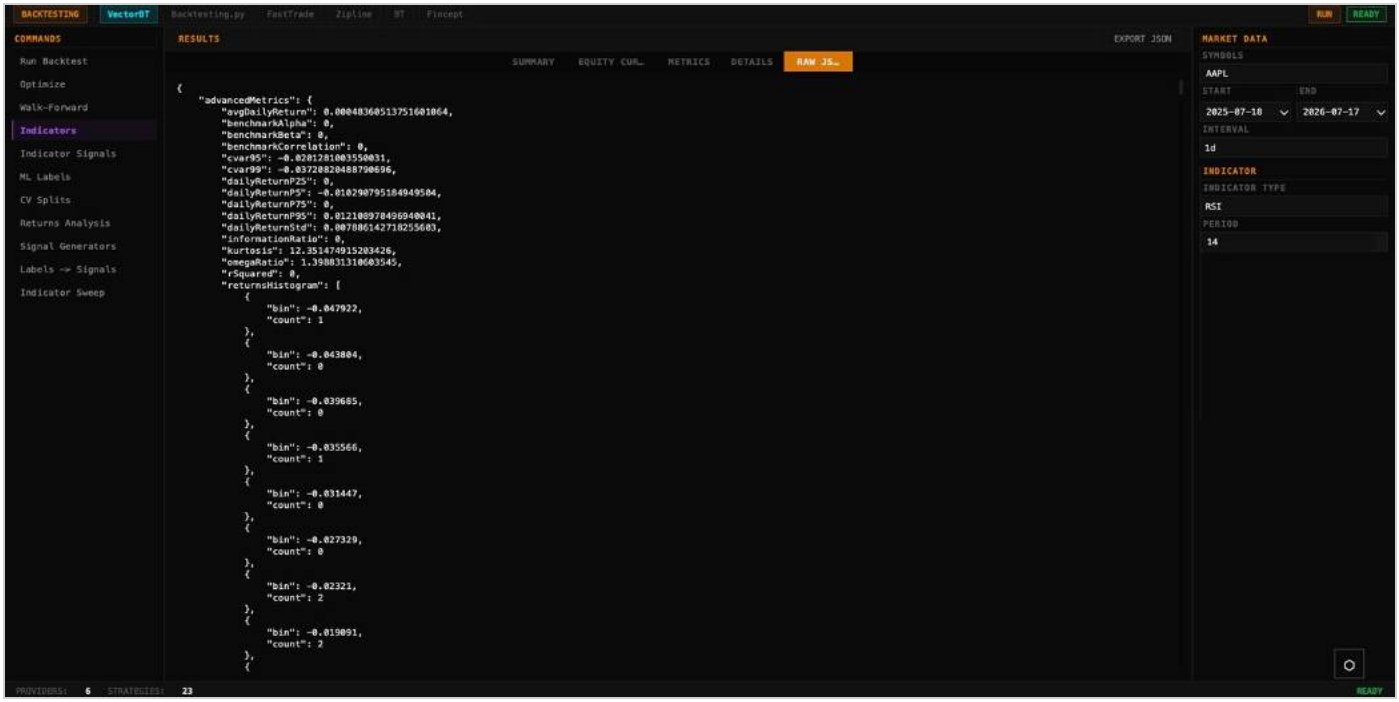
7 Analysis commands

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

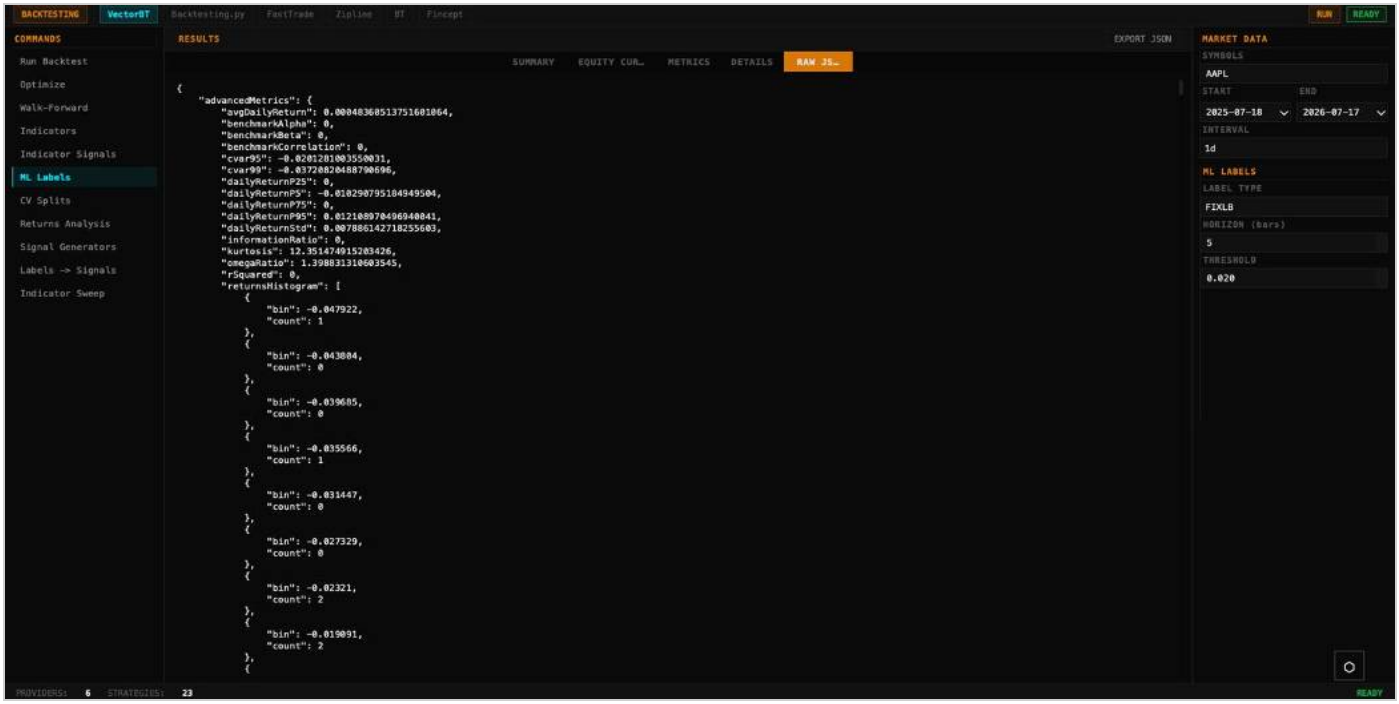
Turn an indicator into entry/exit signals. Config: **INDICATOR** + **SIGNAL MODE**, which reveals mode-specific fields:

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

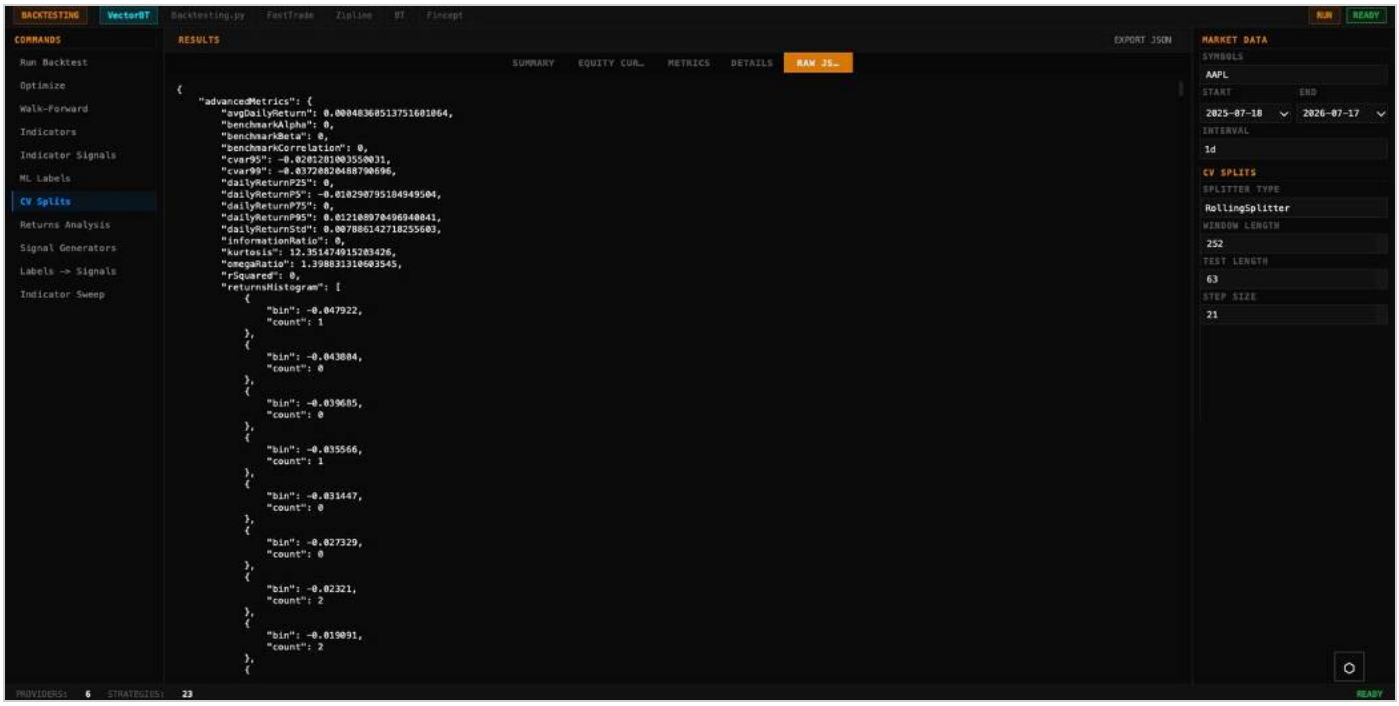
Generate financial-ML target labels. Config: **LABEL TYPE** reveals type-specific fields — **FIXLB** (horizon, threshold), **MEANLB / TRENDLB** (window, threshold), **LEXLB** (window), **BOLB** (window, alpha). Results: label-type/bar cards, a CLASS DISTRIBUTION section, and a DETAILS table of sample 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

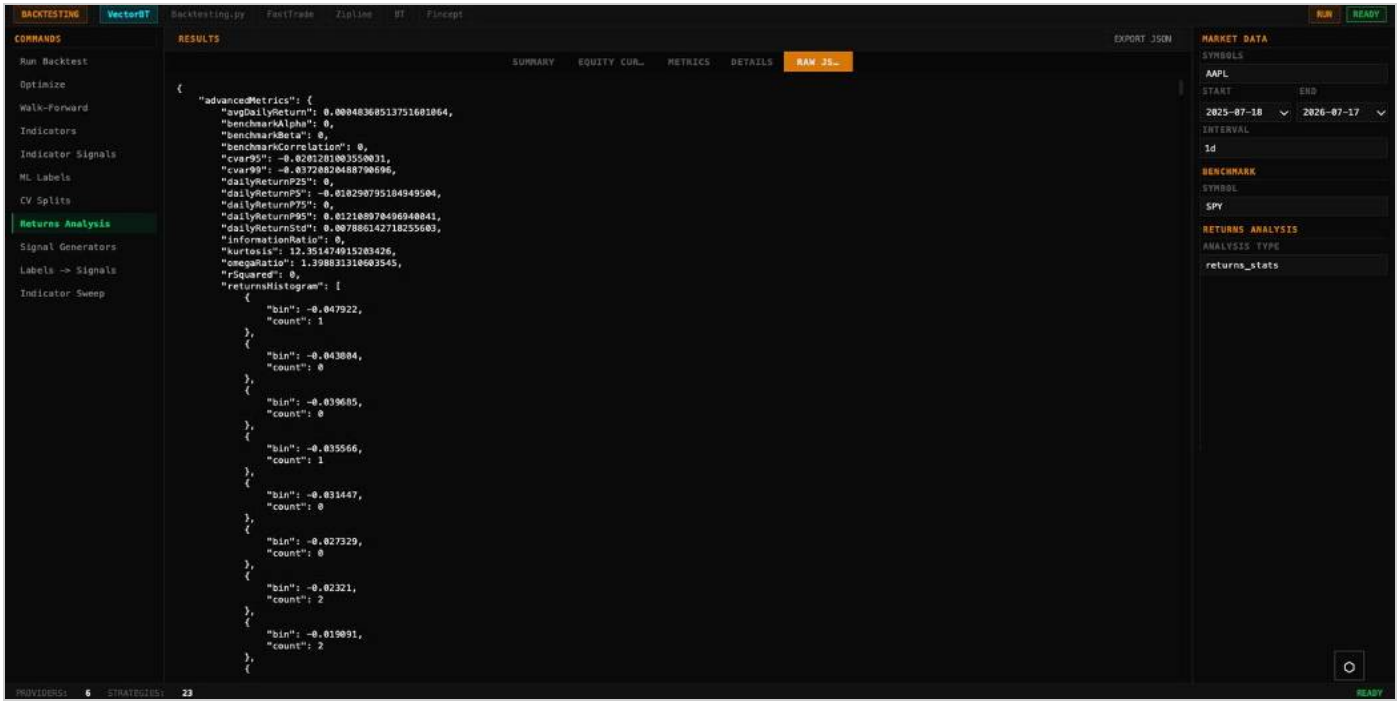
Build cross-validation splitters for ML workflows. Config: **SPLITTER TYPE** reveals fields — **Rolling** (window length, test length, step), **Expanding** (min train length, test length, step), **Purged K-Fold** (n splits, purge length, embargo length). Results: splitter/fold cards and a DETAILS table with one row per fold (train/test start/end & sizes).



CV Splits — the splitter configuration window (splitter type + its fold parameters).

Returns Analysis

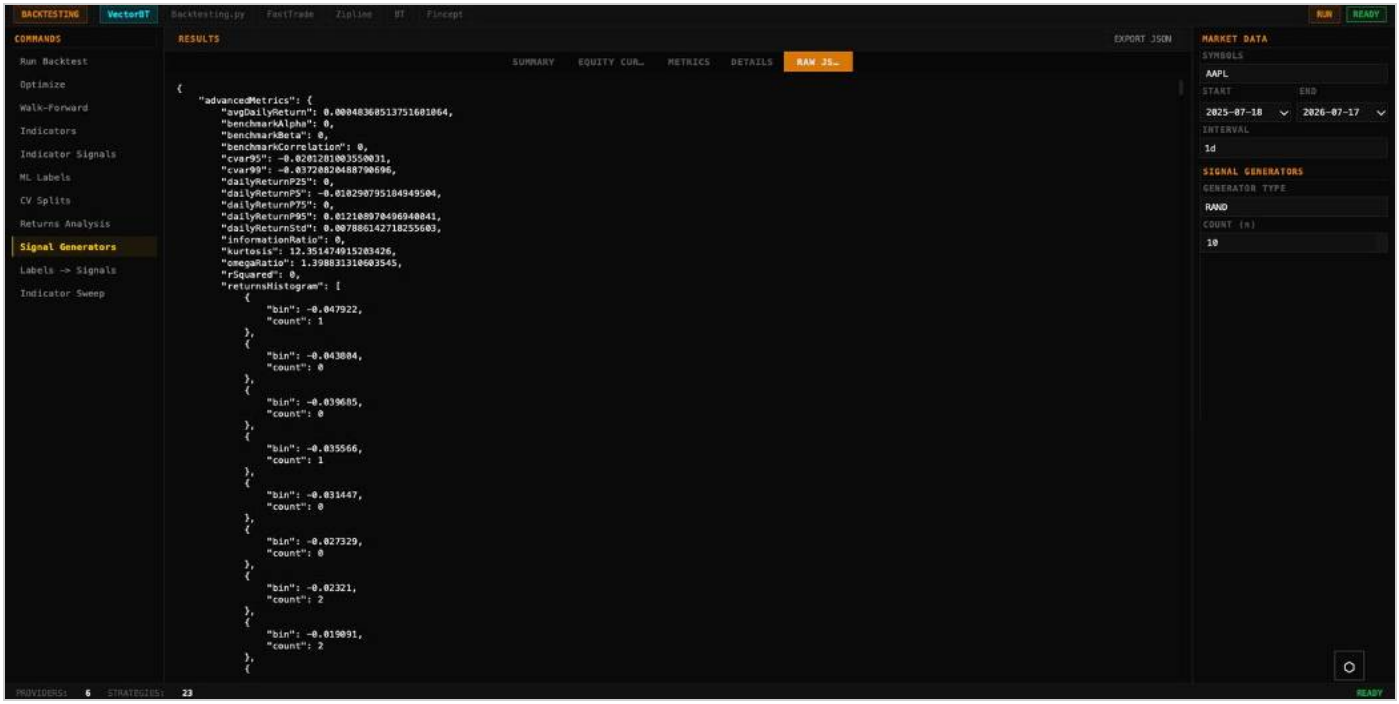
Return-series analytics — the only analysis command that keeps the BENCHMARK section. Config: **ANALYSIS TYPE** (cumulative / rolling / drawdown / distribution / benchmark comparison); **rolling** adds a window + metric. Results: a STATISTICS section (total & annualized return, Sharpe, Sortino, Calmar, 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.

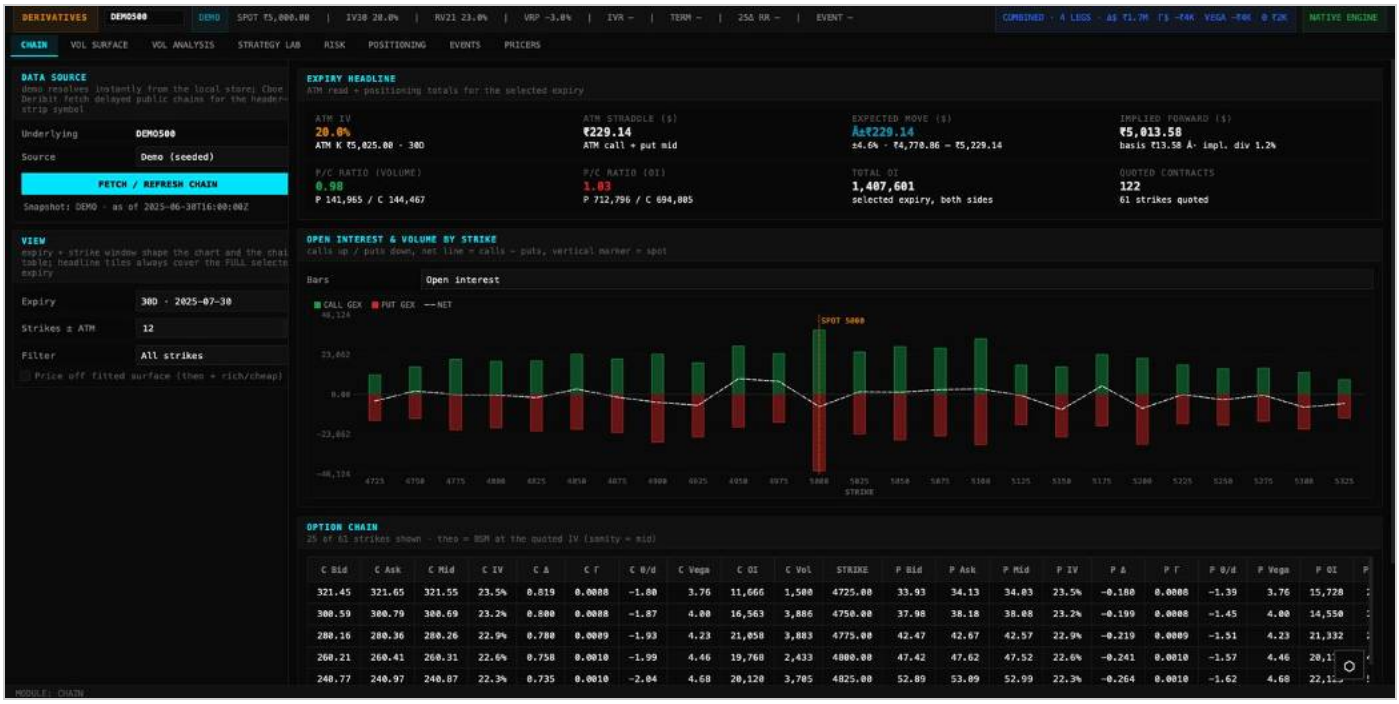
AT A GLANCE, IN DERIVATIVES YOU CAN…

- **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.

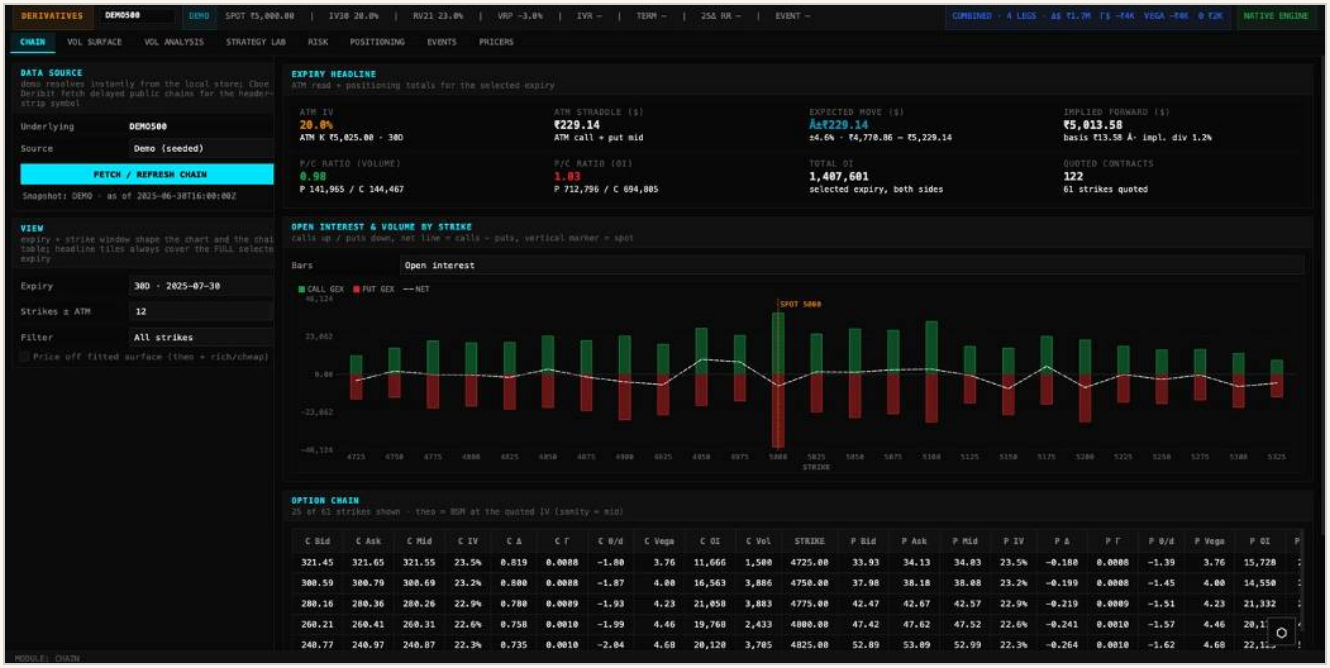
The chart library

- **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).

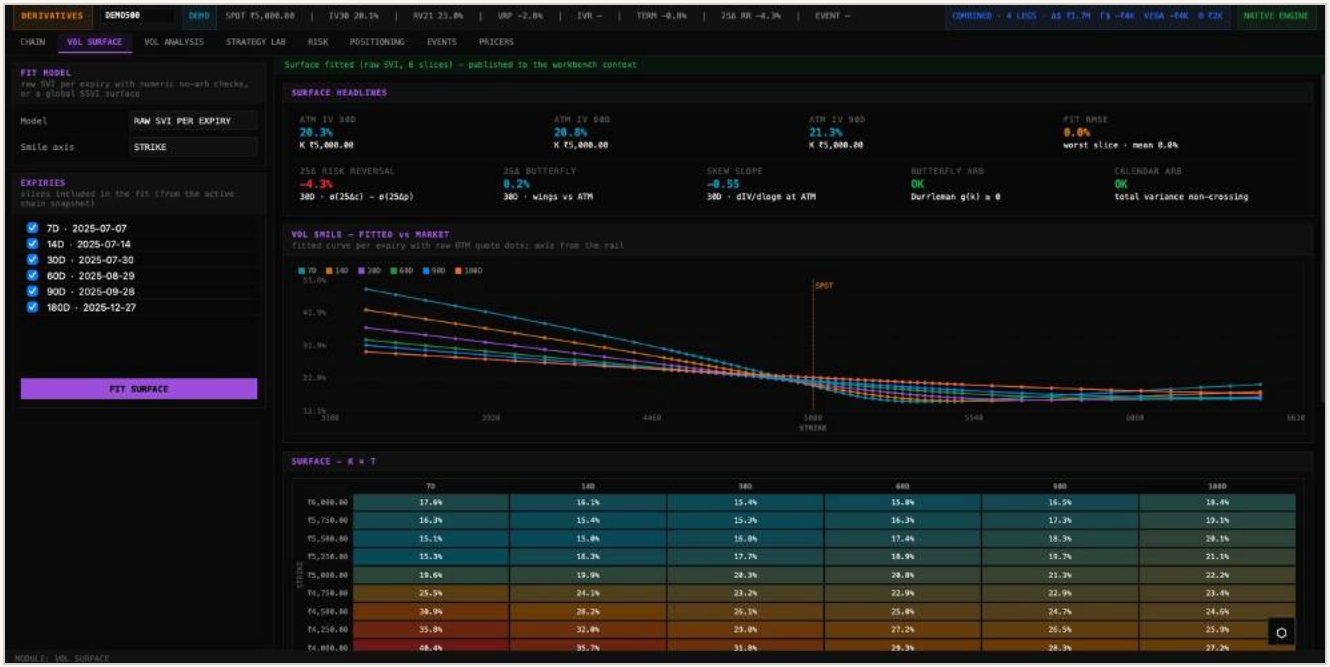
- **Option Chain** — the full 25-column T-chain: C Bid/Ask/Mid/IV/Δ/Γ/Θ/Vega/OI/Vol · STRIKE · P … (read-only; Theo & R/C appear only when priced off a fitted surface).

> **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).

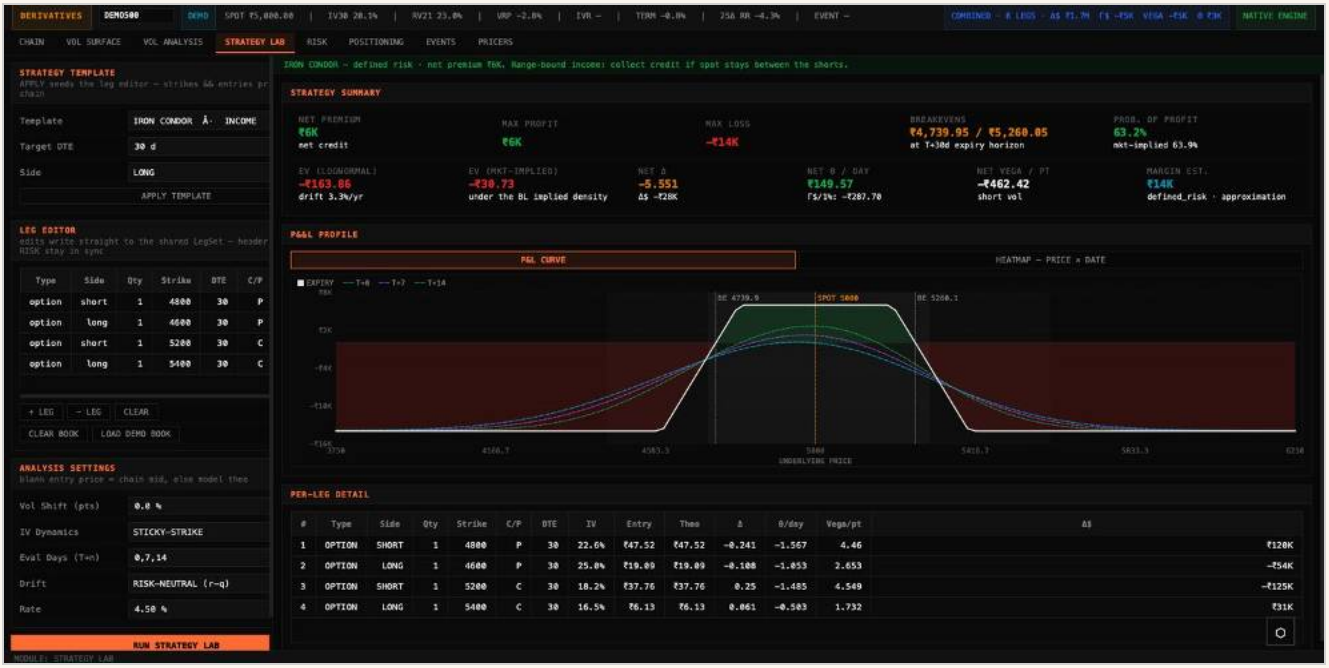
At a glance, in VOL ANALYSIS you can…

- 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.

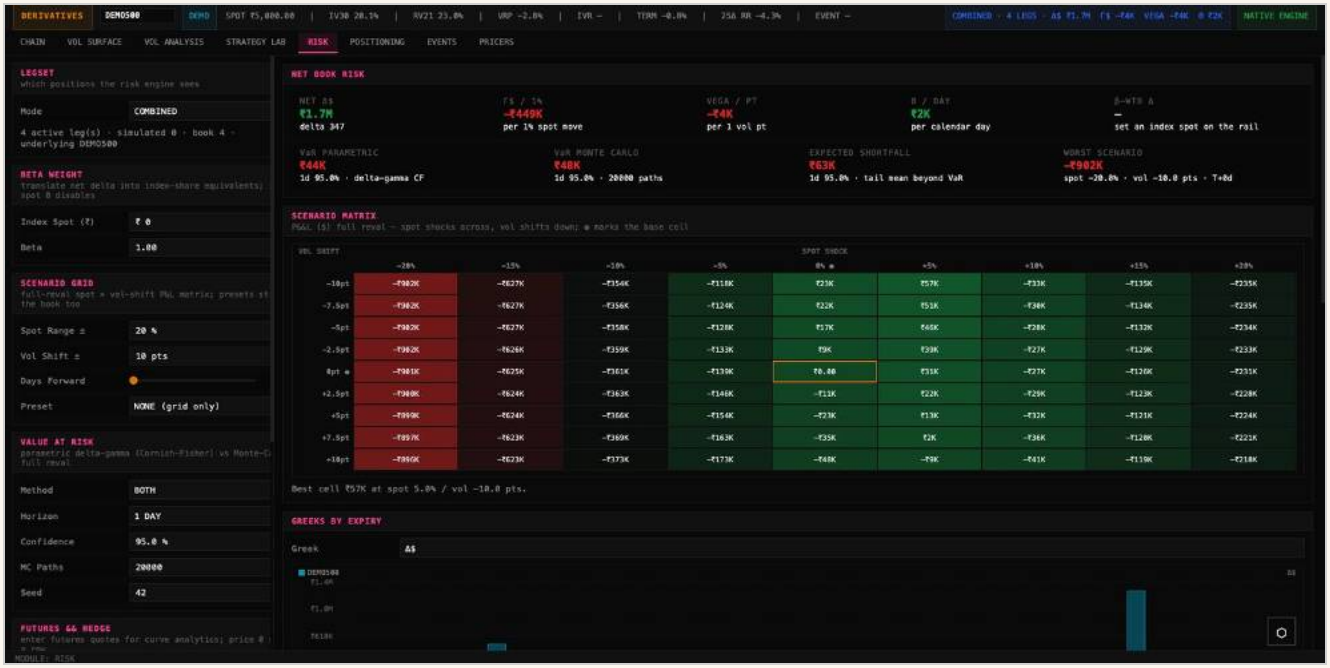
> **Note.** Margin, probability of profit and expected value are explicitly labelled approximations ("broker/exchange margin will differ"). Delta-based strikes and market-implied probabilities are most meaningful with a surface fitted on VOL SURFACE.

> ### 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.

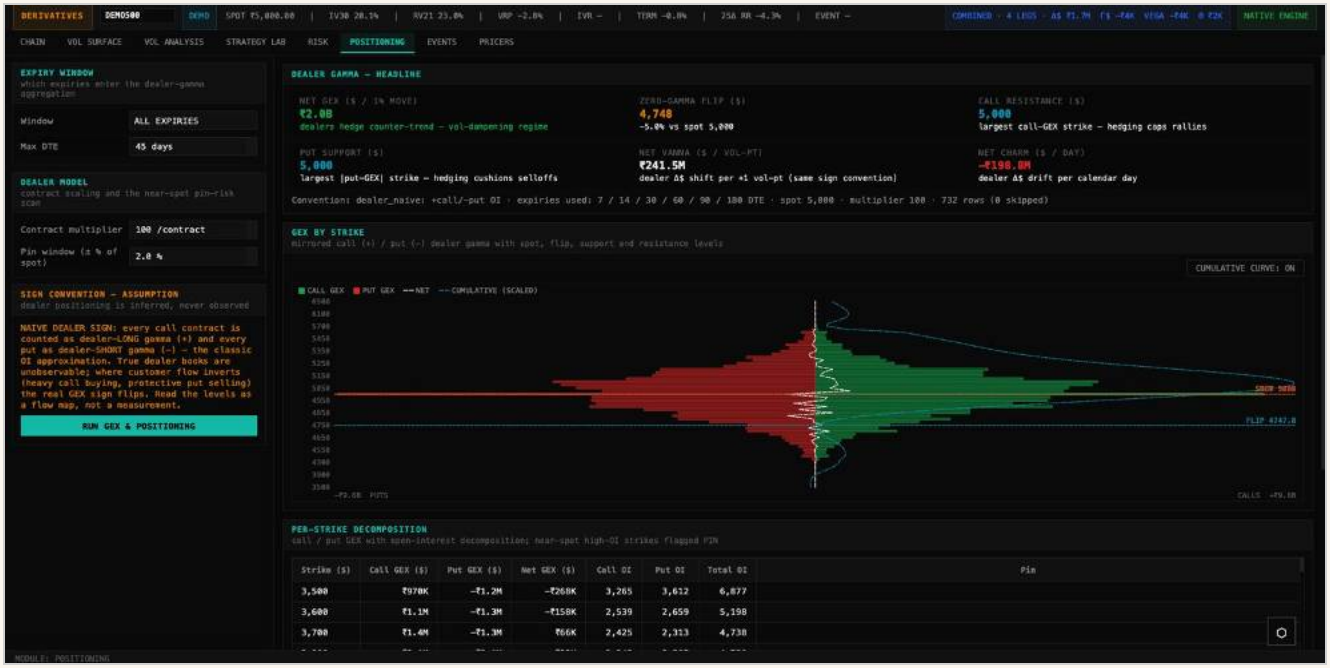
> **Note.** With no active legs (default SIMULATED mode starts empty) the page shows an empty state — switch the LegSet mode to BOOK / COMBINED, or build a strategy in the Strategy Lab. P&L is measured against the current theoretical mark, so the base (0/0) scenario cell is exactly zero.

> ### 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.

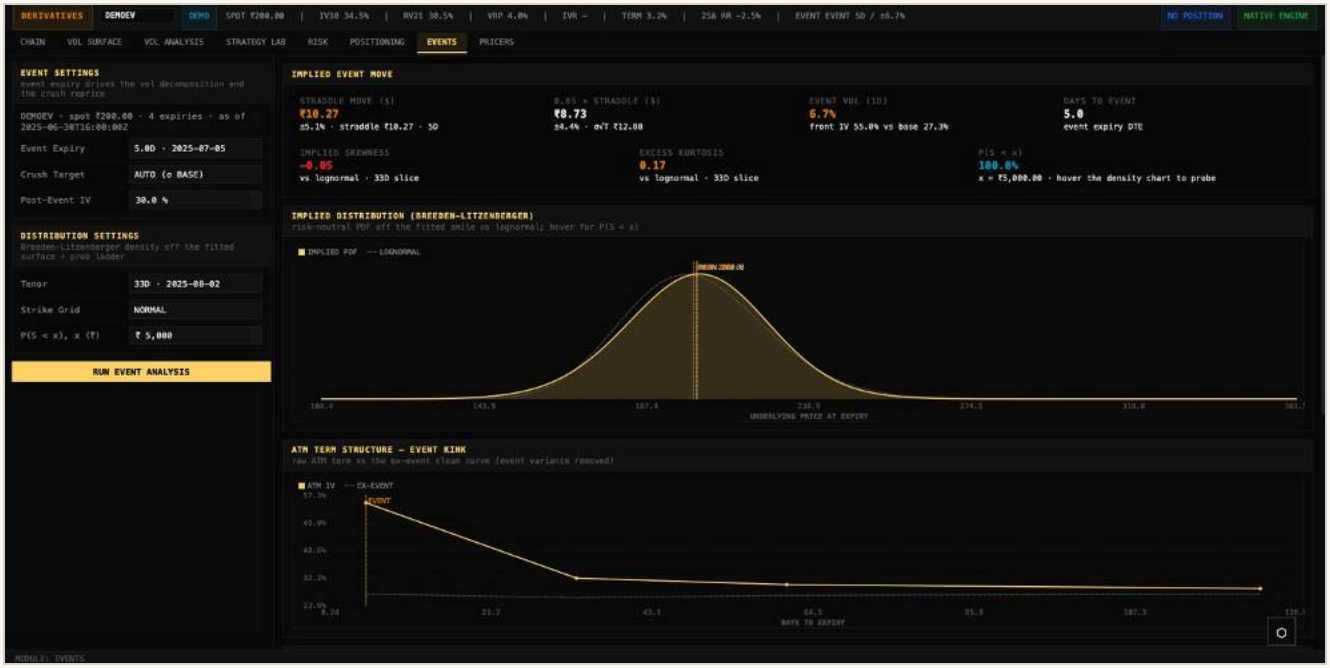
At a glance, in POSITIONING you can…

- 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.

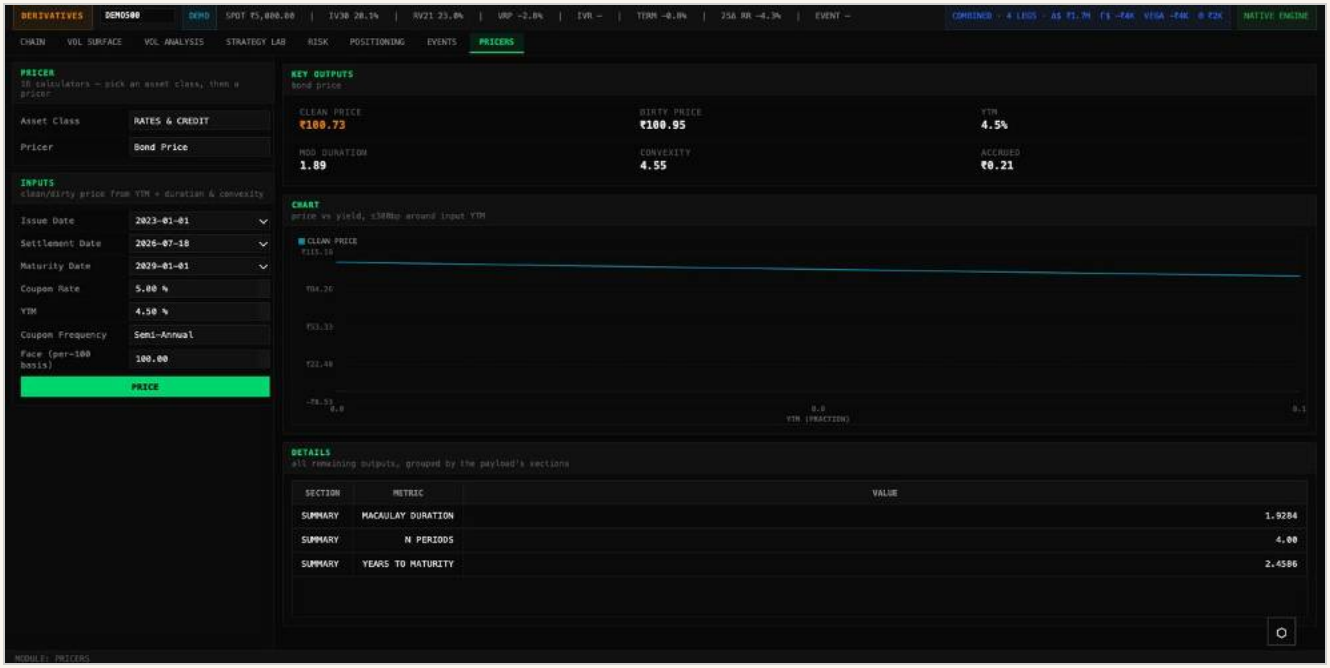
At a glance, in EVENTS you can…

- 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.

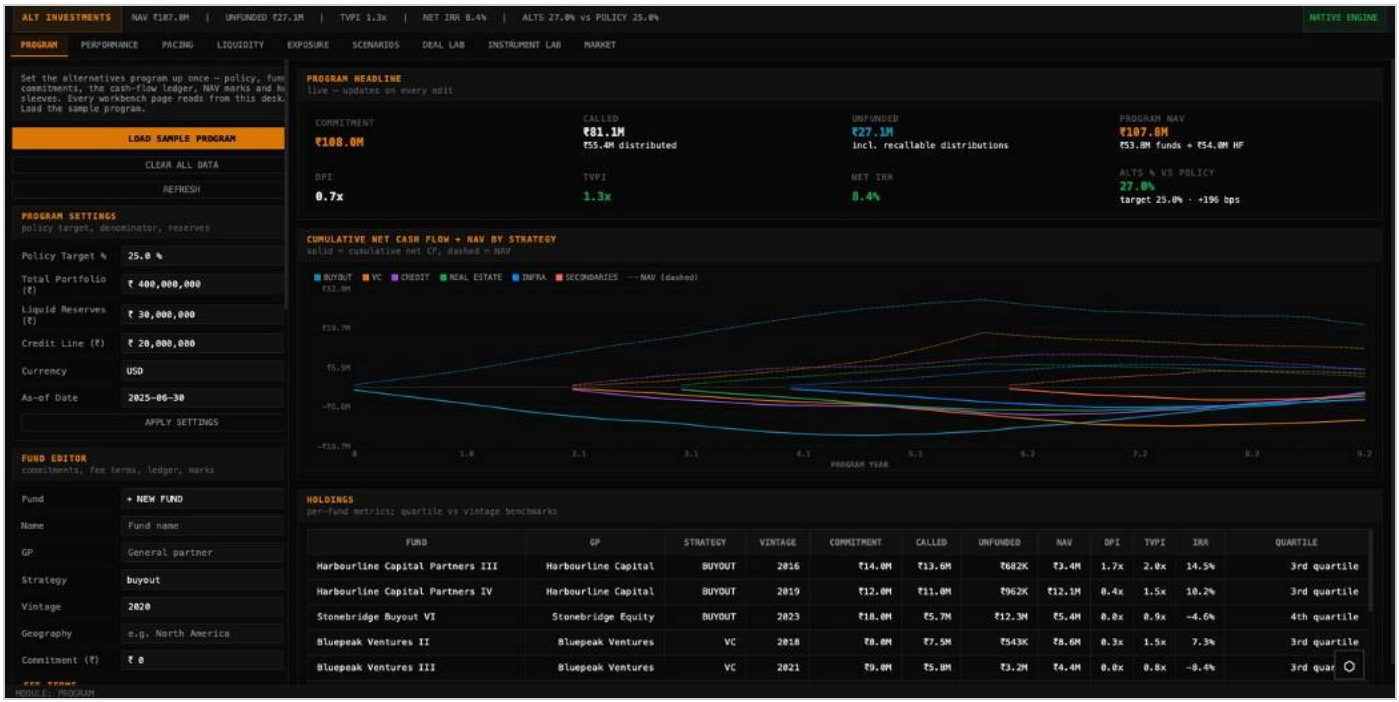
AT A GLANCE, IN ALT INVESTMENTS YOU CAN…

- **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.

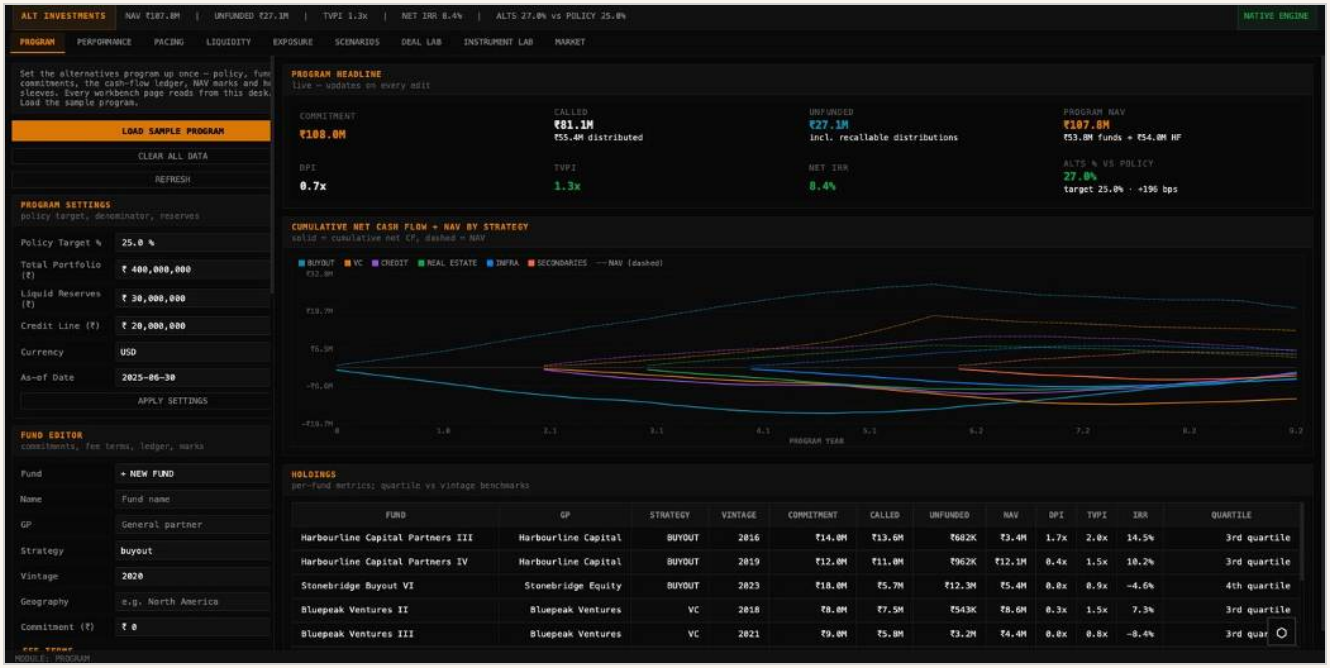
Shared building blocks

- **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.

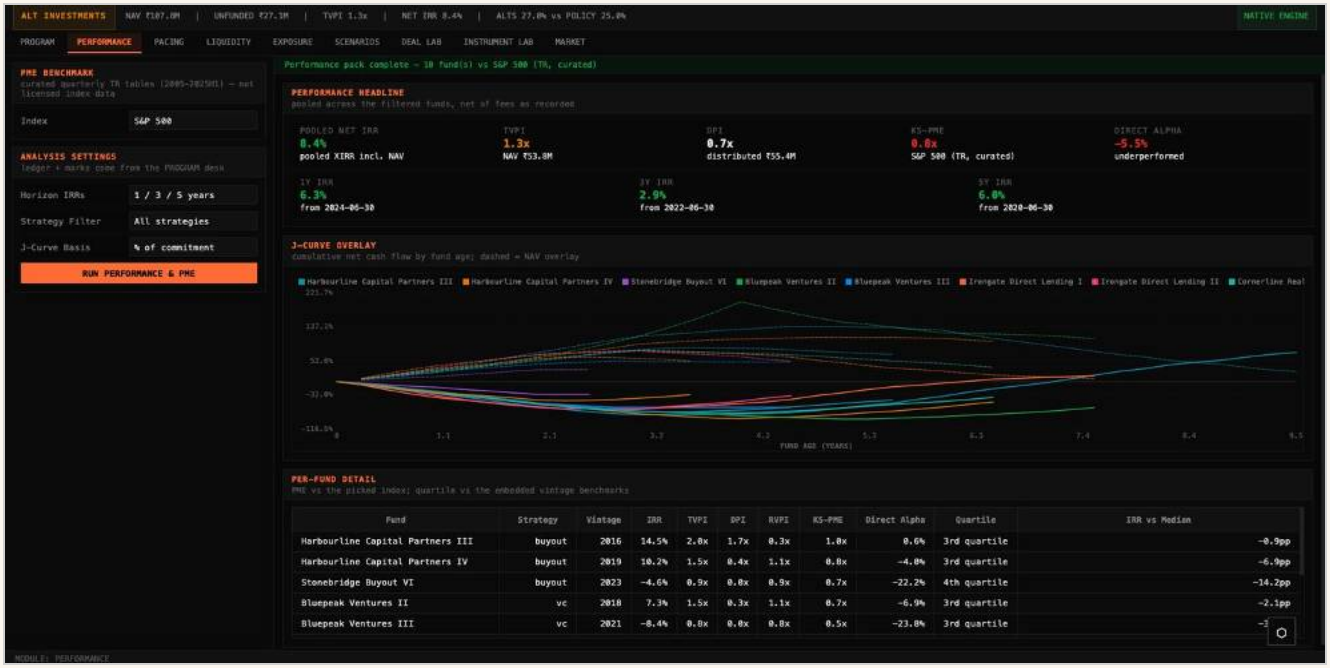
At a glance, in PROGRAM you can…

- 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.

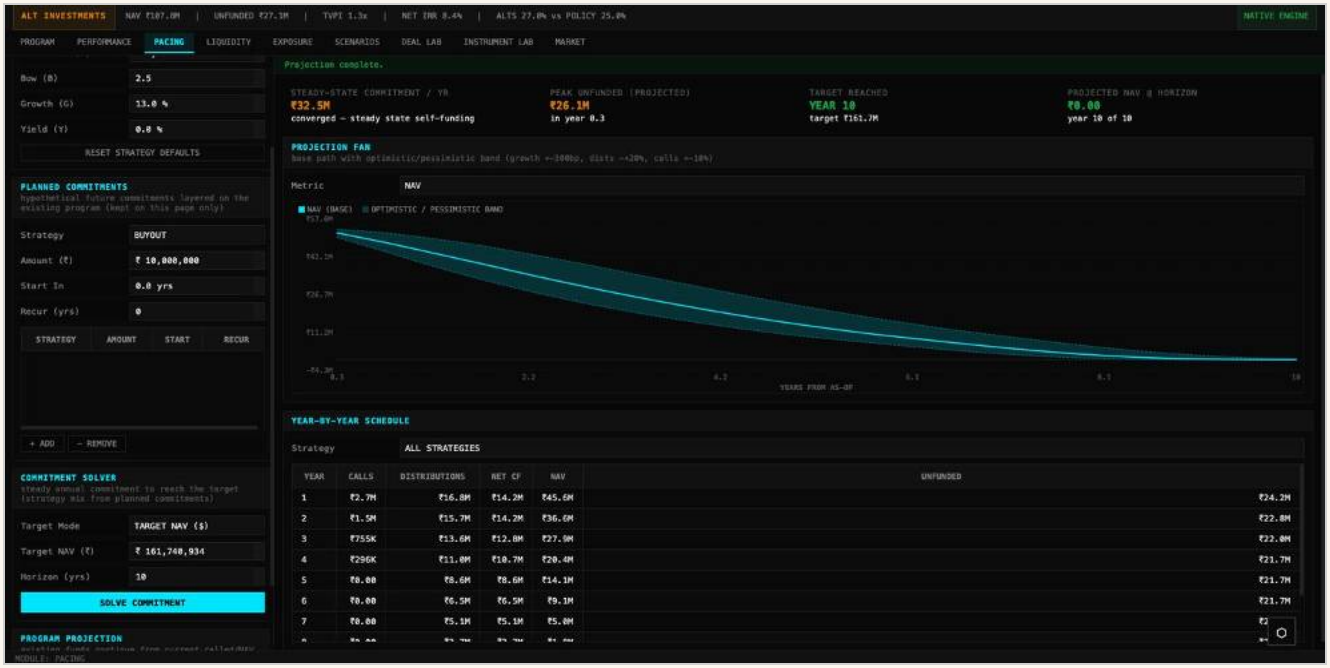
Definitions. IRR is a dated-cash-flow XIRR with terminal NAV as a pseudo-distribution. **KS-PME** (Kaplan-Schoar) > 1.0 means the fund beat the public index dollar-for-dollar; **Direct Alpha** annualises that out-/under-performance. Quartiles use embedded PE/VC vintage tables (vintages clamped to 2005–2023).

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.

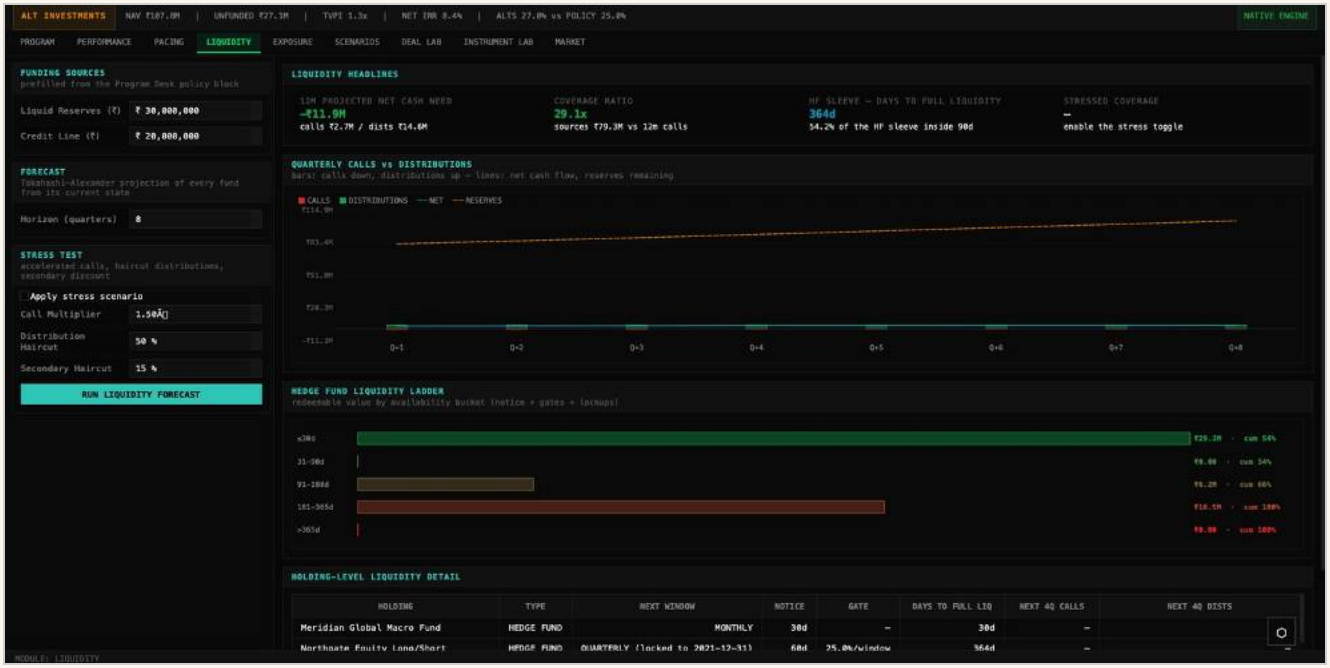
> **Note.** SOLVE fills tiles 1&3; PROJECT fills tiles 2&4 — running only one leaves the other two blank. Planned commitments and TA-parameter edits are page-local and are rebuilt each session by design.

> ### 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.

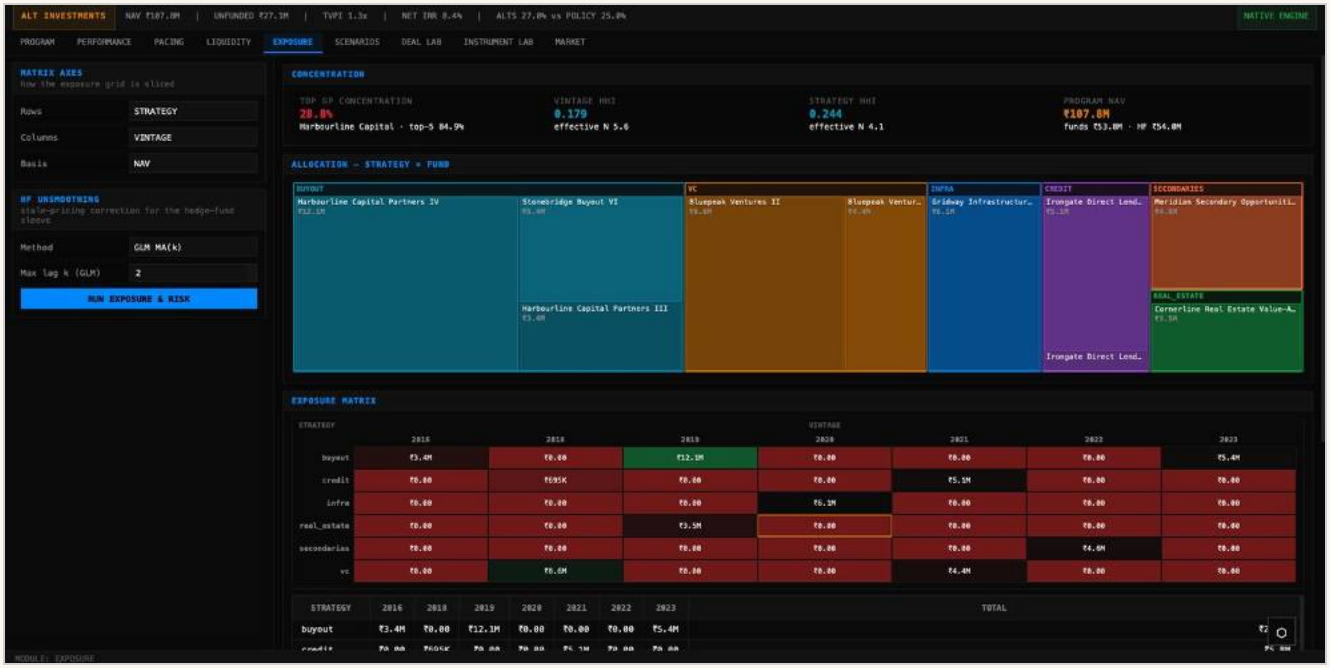
At a glance, in LIQUIDITY you can…

- 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.

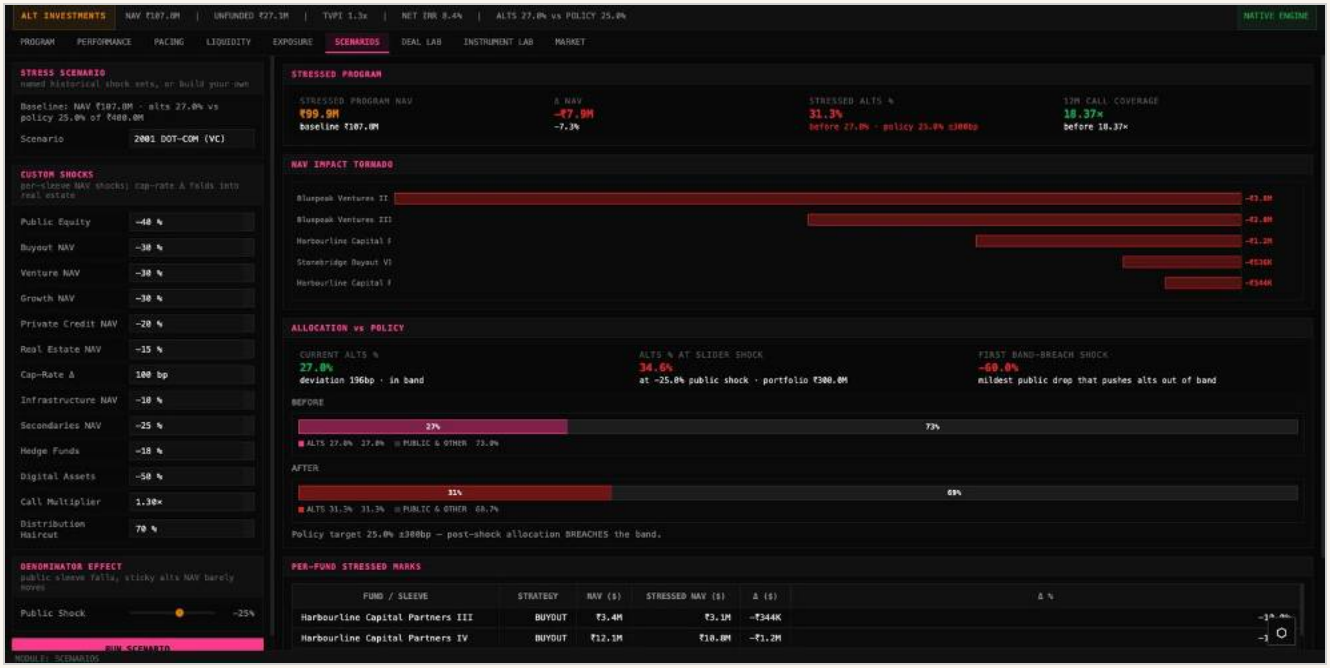
Honest caveat. The optional "nowcast Δ (neutral proxies)" line under PROGRAM NAV is **not functional** — it is invoked without proxy returns and errors internally, so that adjustment never renders. Concentration, treemap, matrix and the de-smoothing table are all fully live.

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.

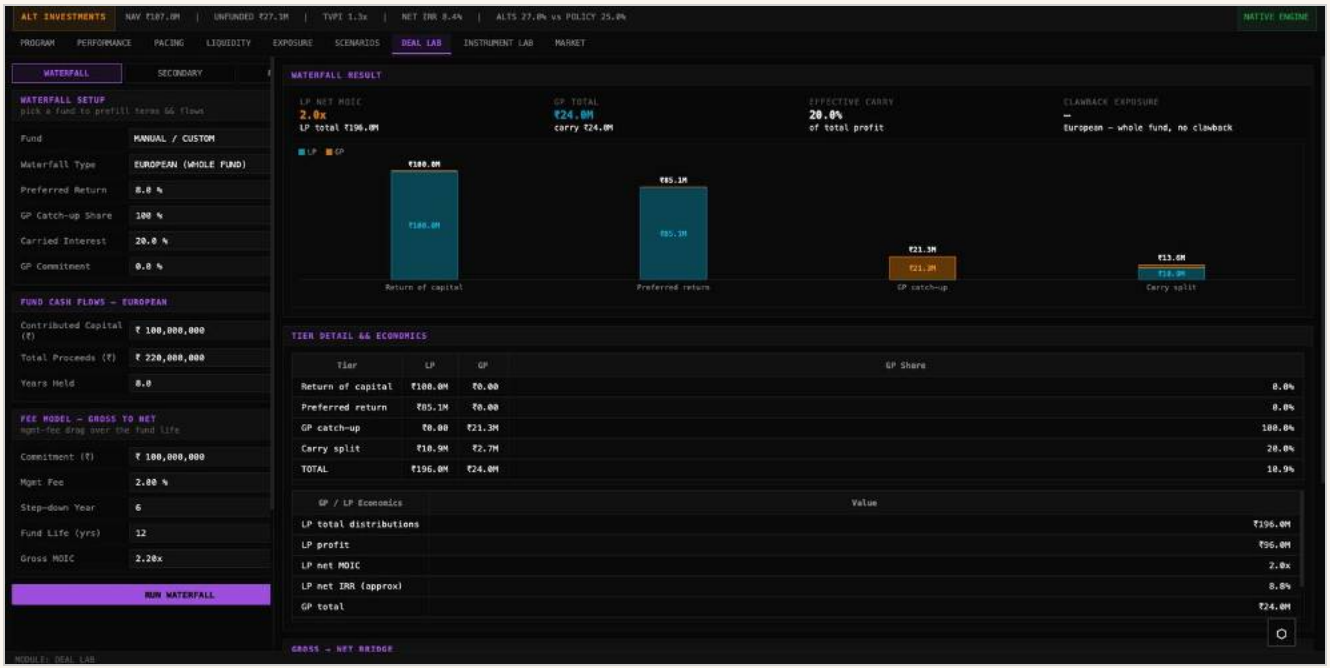
At a glance, in SCENARIOS you can…

- 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

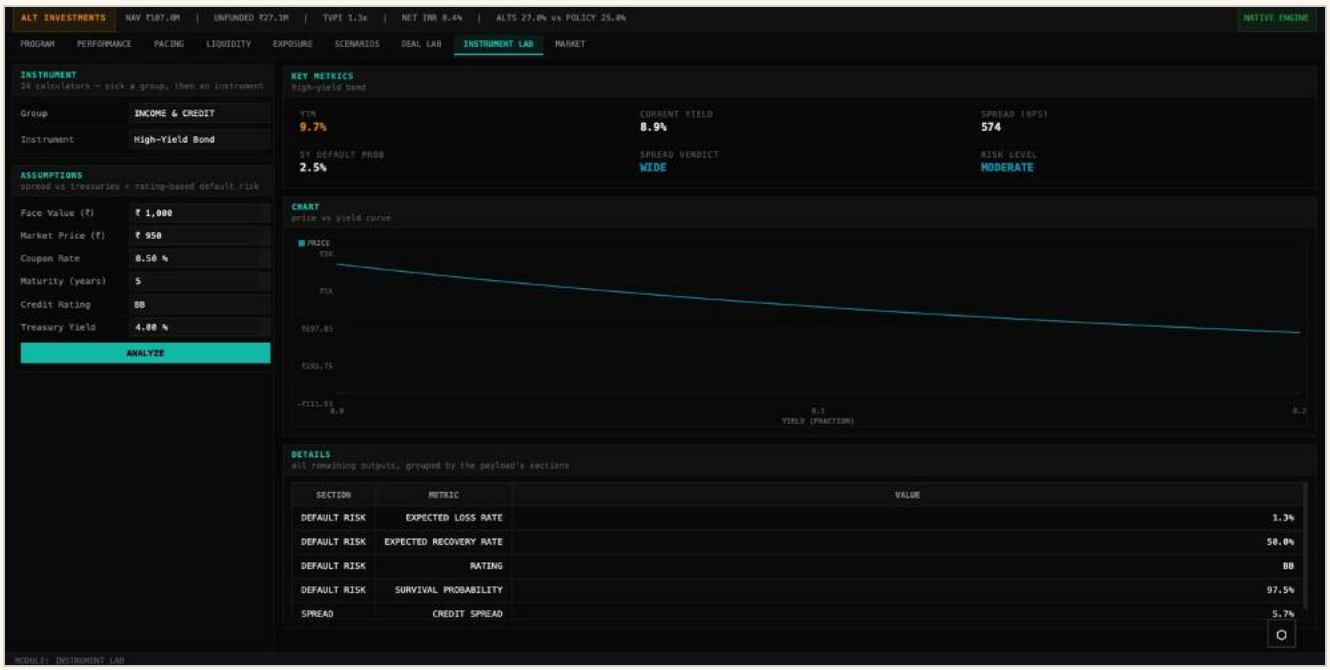
Property income (gross rent, vacancy, opex, price, market cap rate), DCF assumptions, financing (LTV, rate, amortisation), and an LP/GP promote over IRR hurdles. Results: cap rate, DSCR, LTV/debt-yield, unlevered/levered DCF IRR, an exit-cap × rent-growth value heatmap, and a promote waterfall.

> **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.

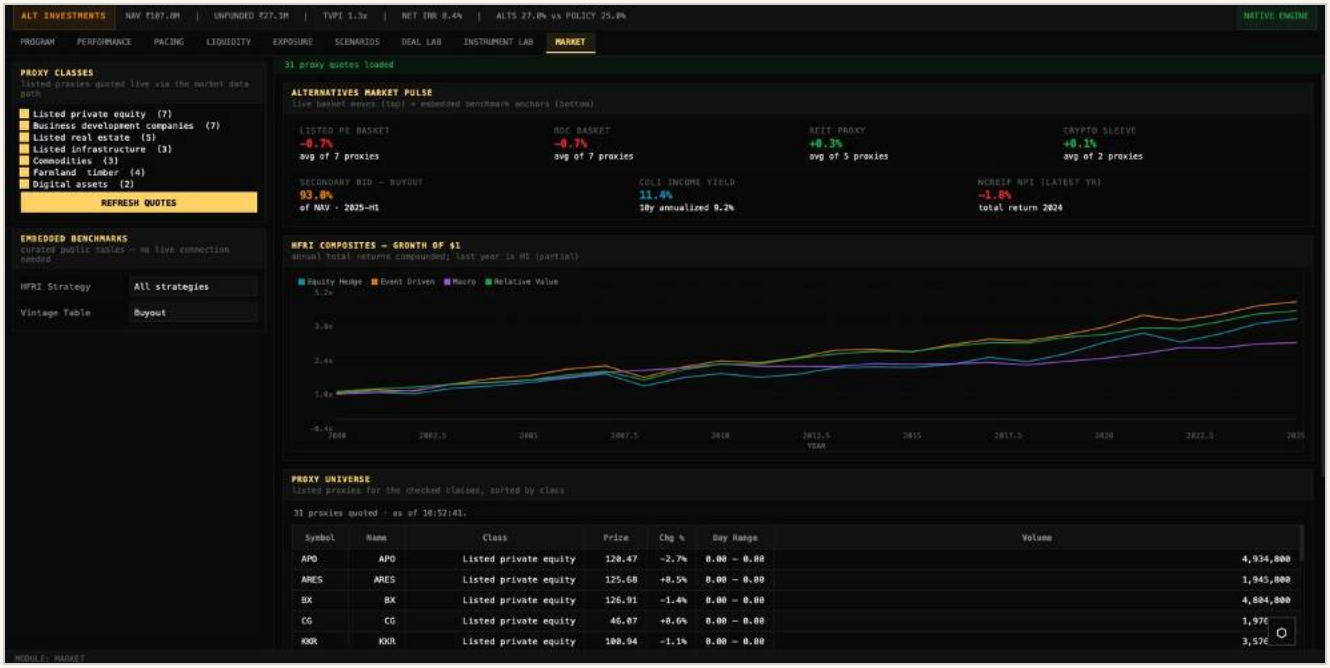
At a glance, in INSTRUMENT LAB you can…

- 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.

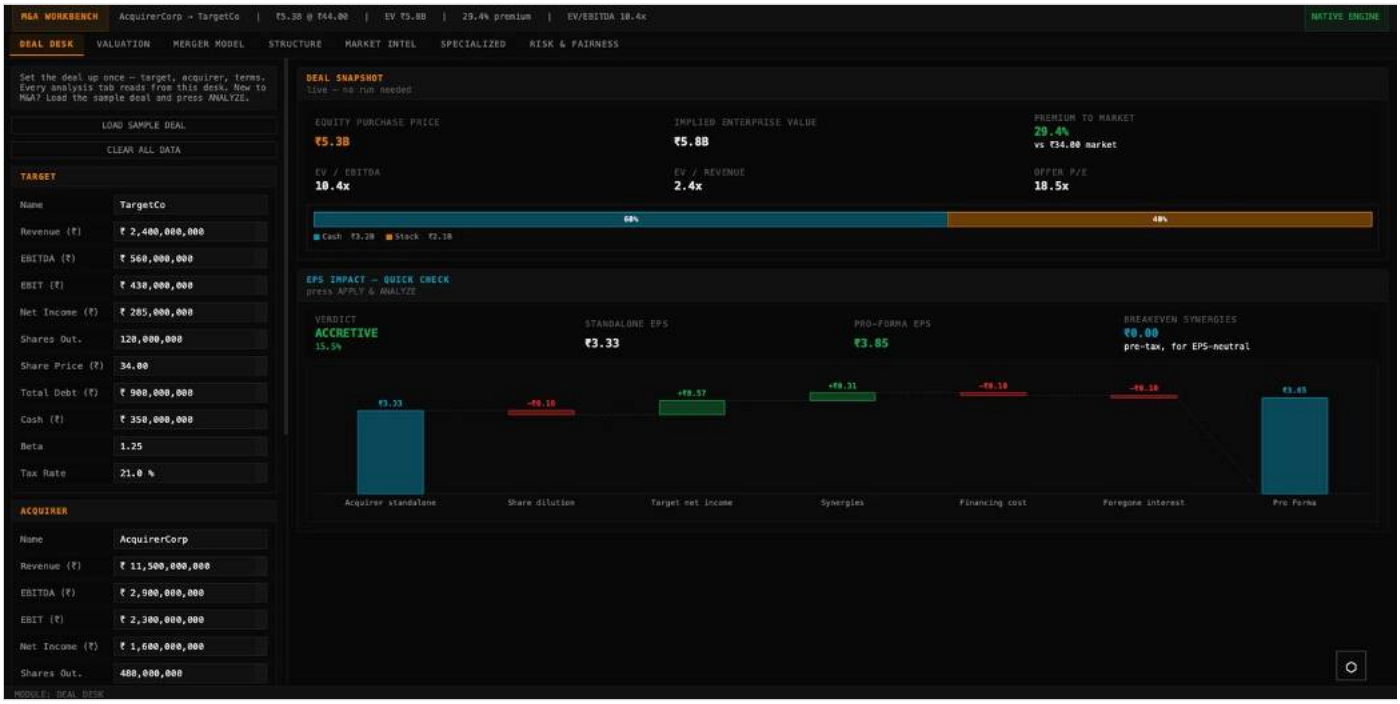
AT A GLANCE, IN M&A ANALYTICS YOU CAN…

- **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

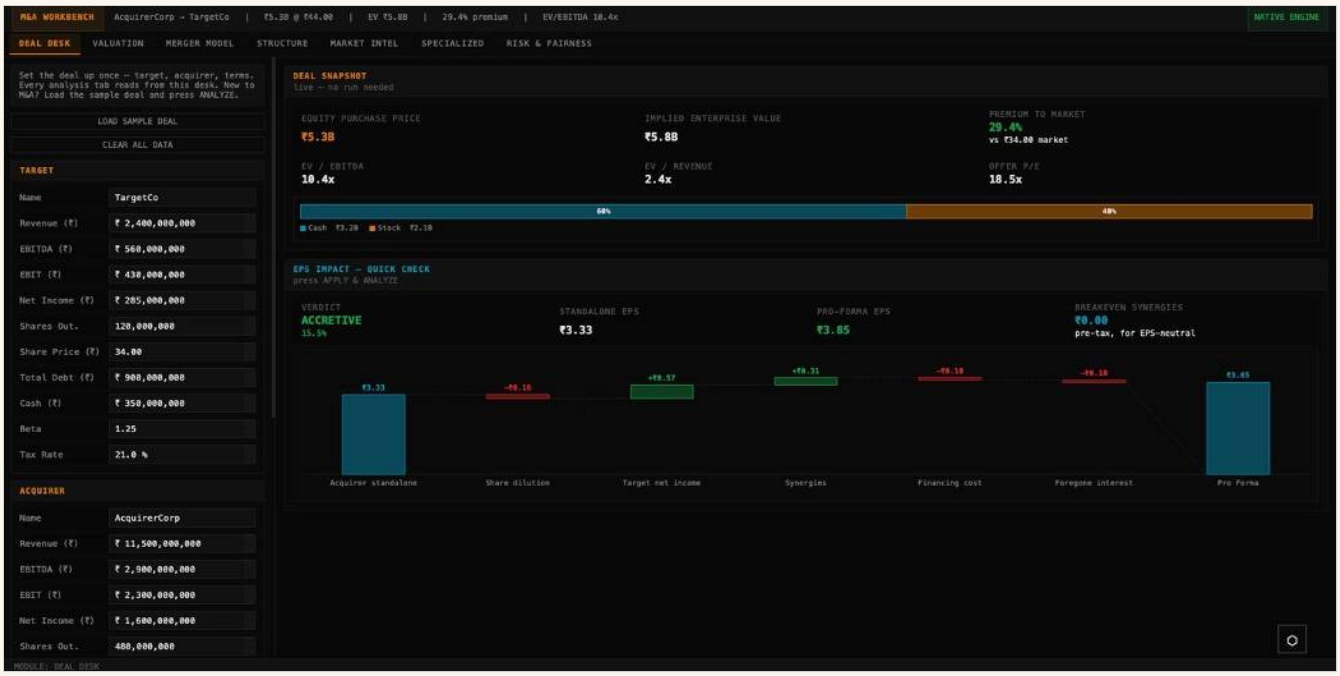
- **KPI tiles** — caption + large value + sub-line, colour-toned (green good / red bad / amber accent / cyan info).
- **Section cards** — titled input groups on the rail, each with a one-line hint.
- **Money / percent / number spin-boxes; editable input tables** (peers, tranches).
- **Charts** — football 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.

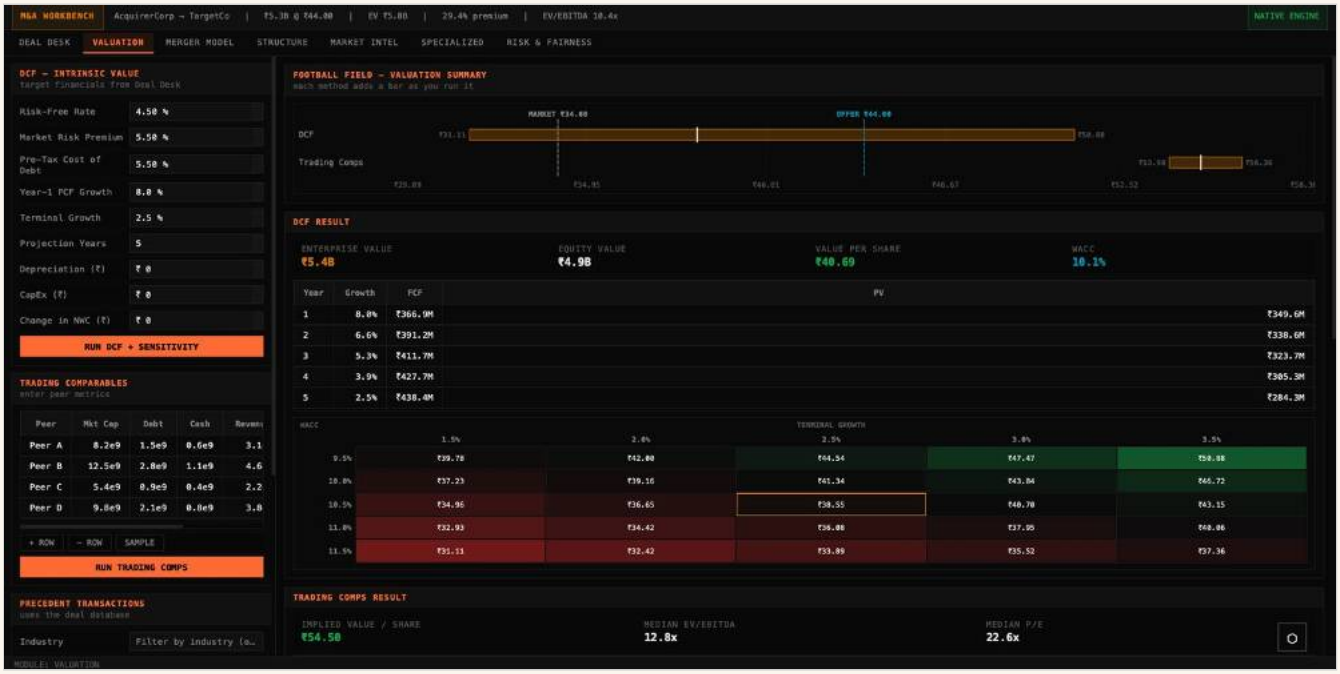
- **EPS Impact — Quick Check** — an ACCRETIVE/DILUTIVE verdict, standalone vs pro-forma EPS, breakeven synergies, and an EPS bridge (standalone → share dilution → target NI → synergies → financing cost → foregone interest → pro forma).

> **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.

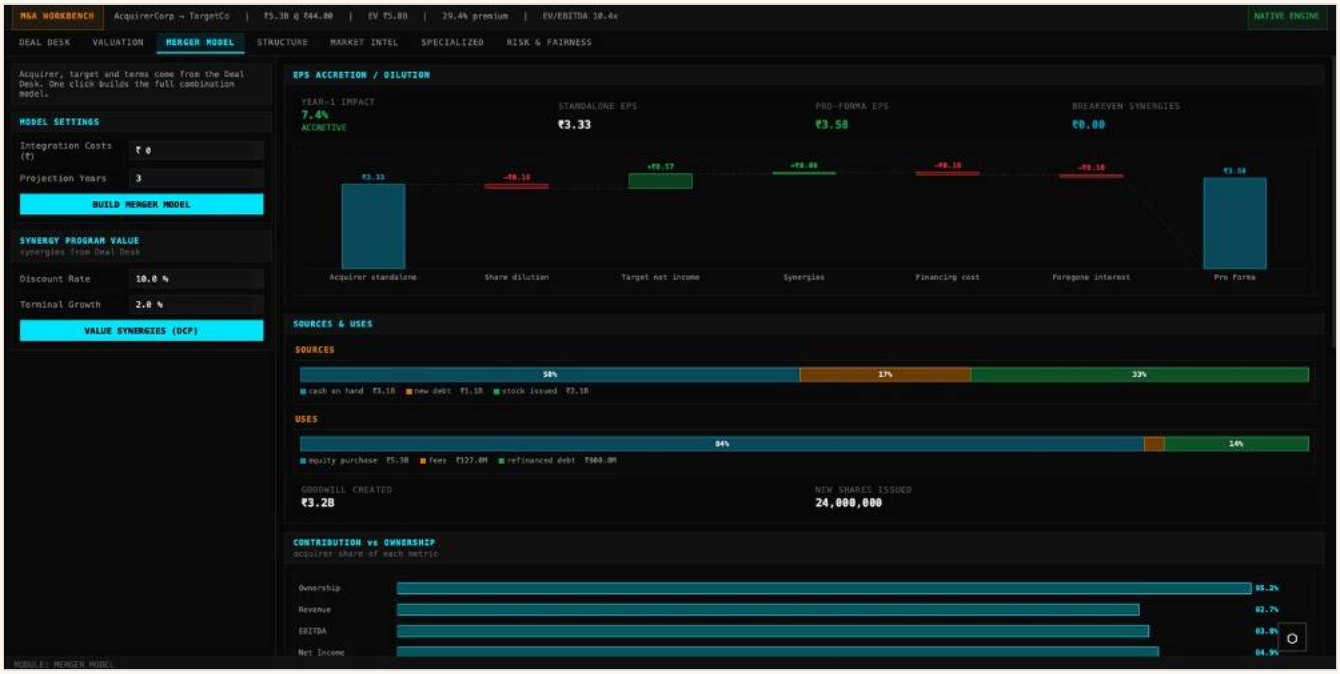
At a glance, in VALUATION you can…

- 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.

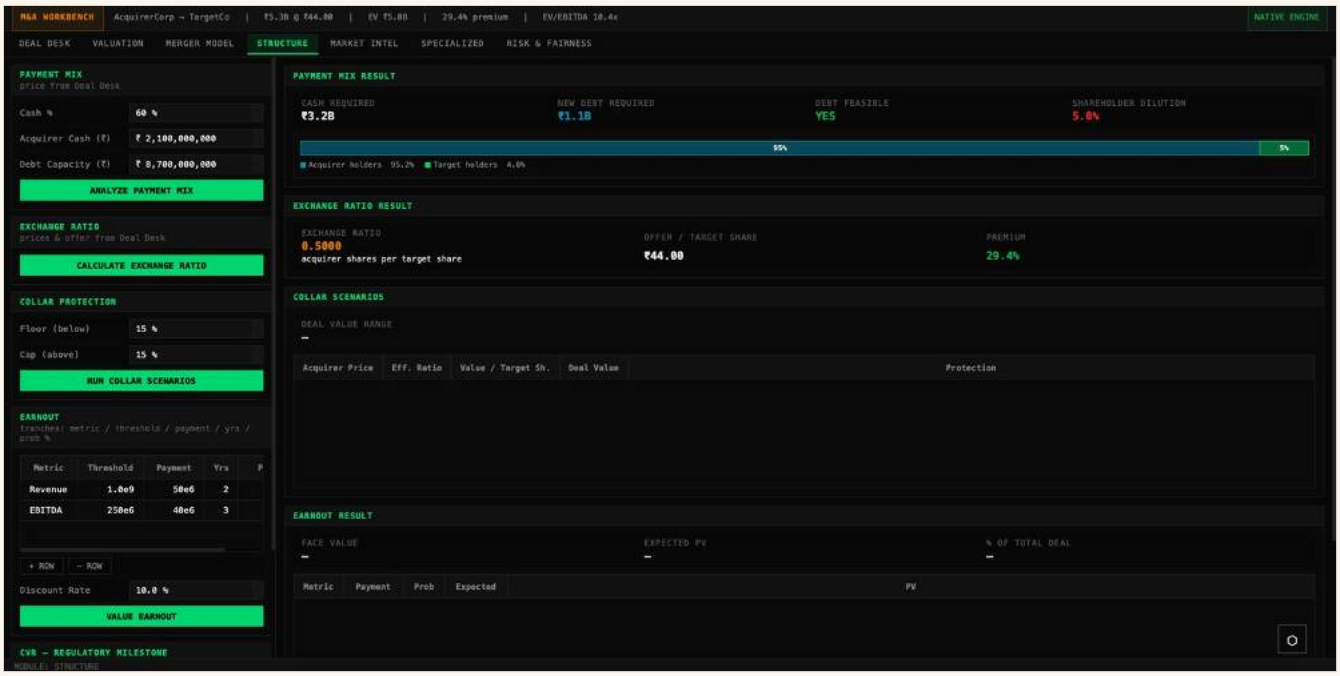
At a glance, in MERGER MODEL you can…

- 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).

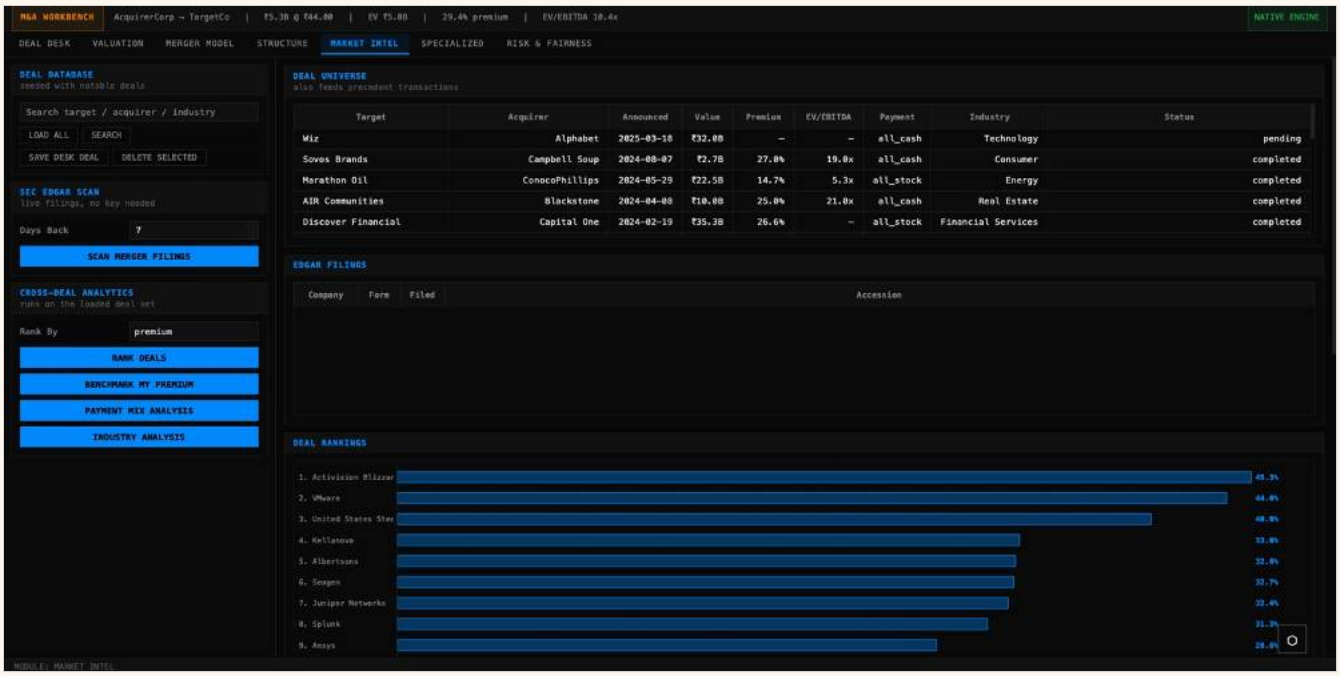
At a glance, in STRUCTURE you can…

- 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.

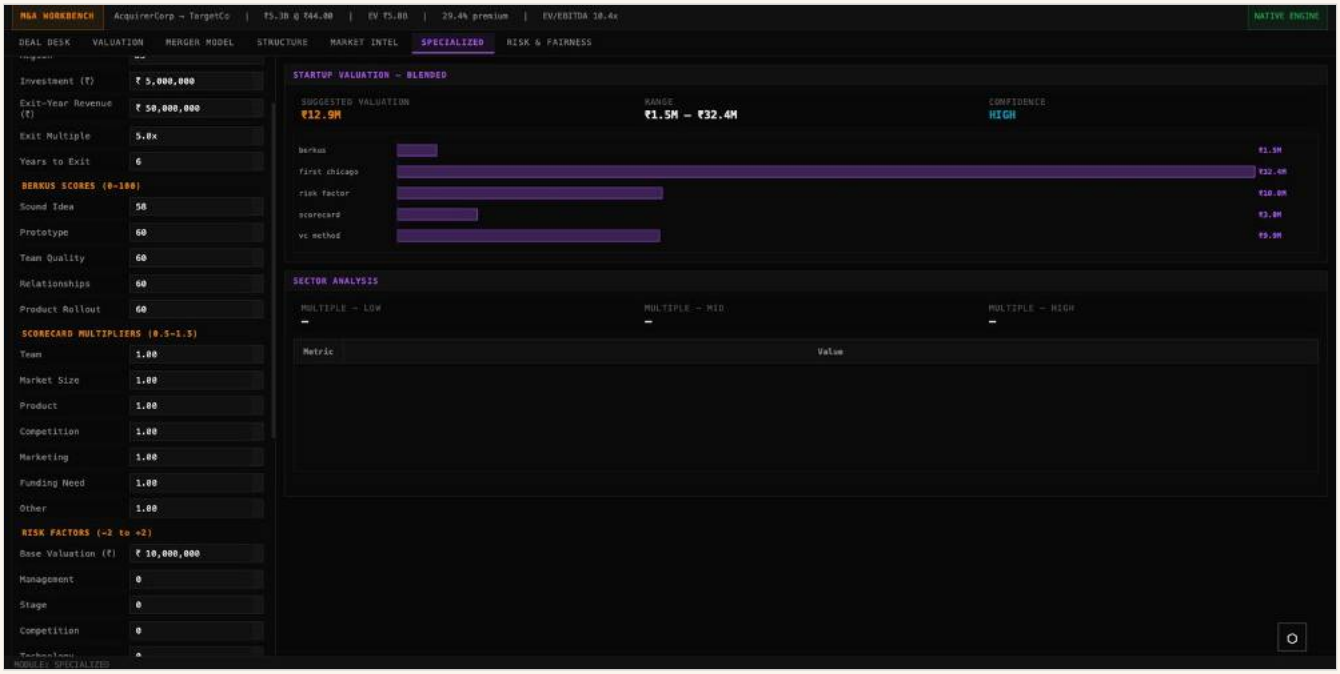
At a glance, in MARKET INTEL you can…

- 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).

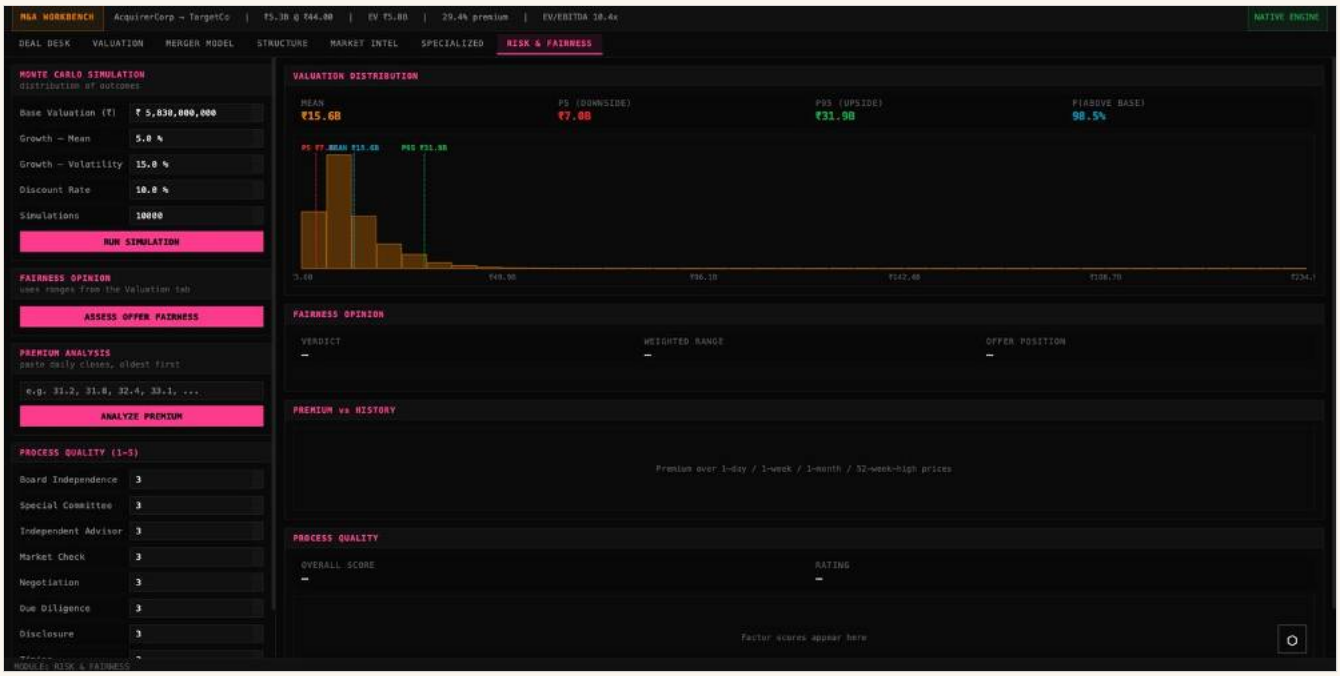
At a glance, in SPECIALIZED you can…

- 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

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.

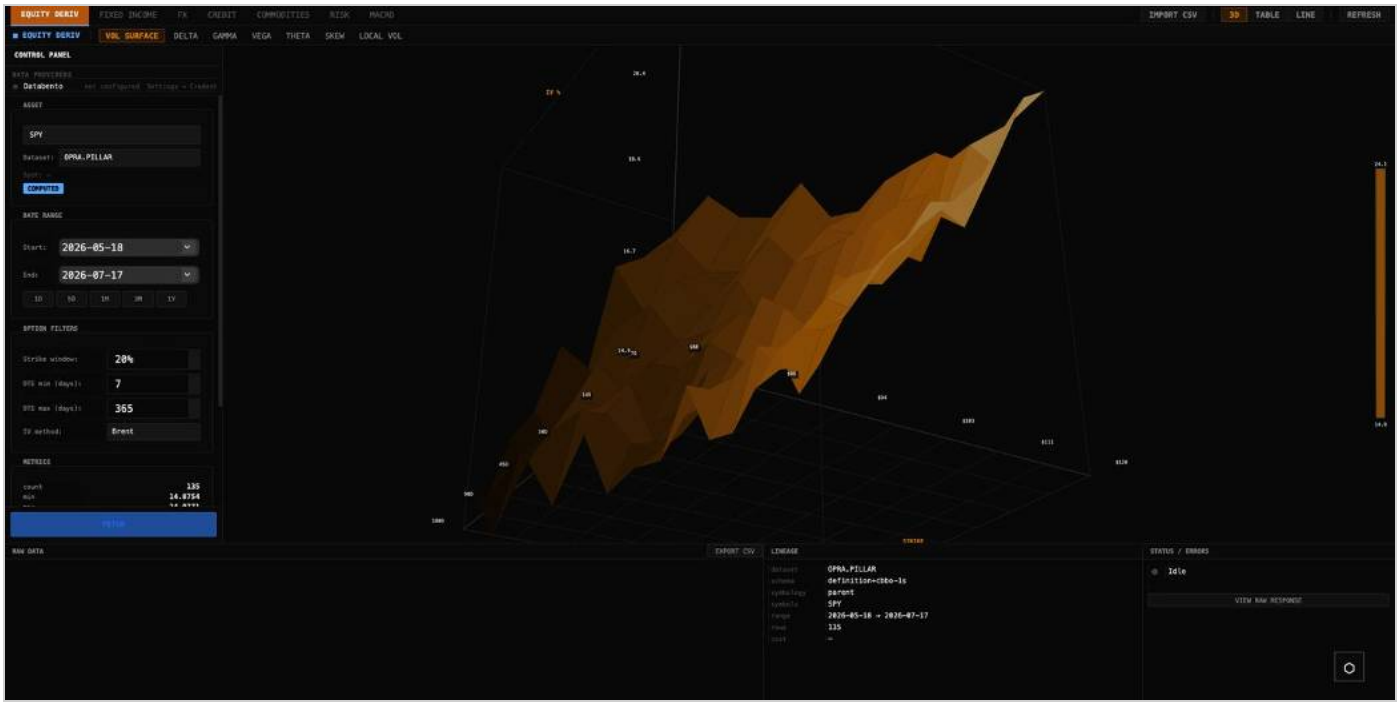
AT A GLANCE, IN SURFACE ANALYTICS YOU CAN…

- 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.

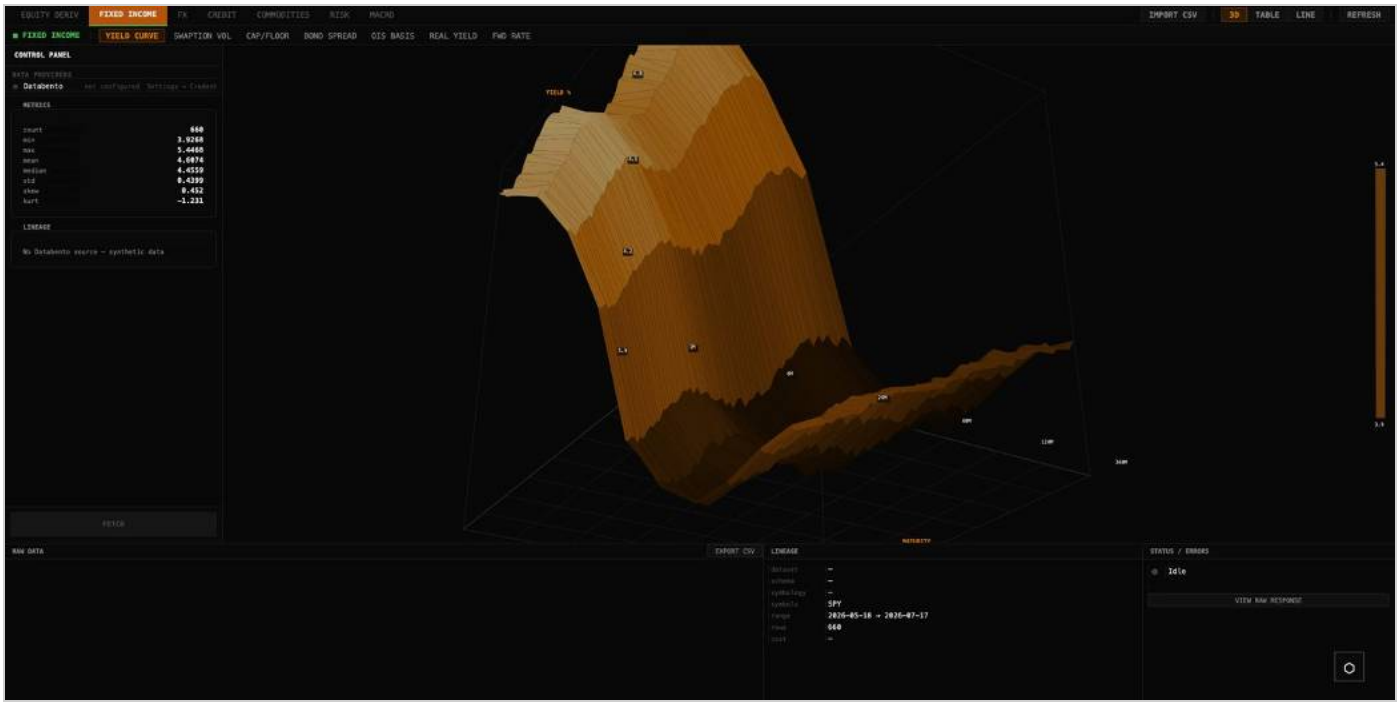
Of the 35 surfaces: **2 LIVE, 11 COMPUTED, 7 EQUITIES, 15 DEMO**. All support 3D and Table; only seven support the Line view (marked **+L**).

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)

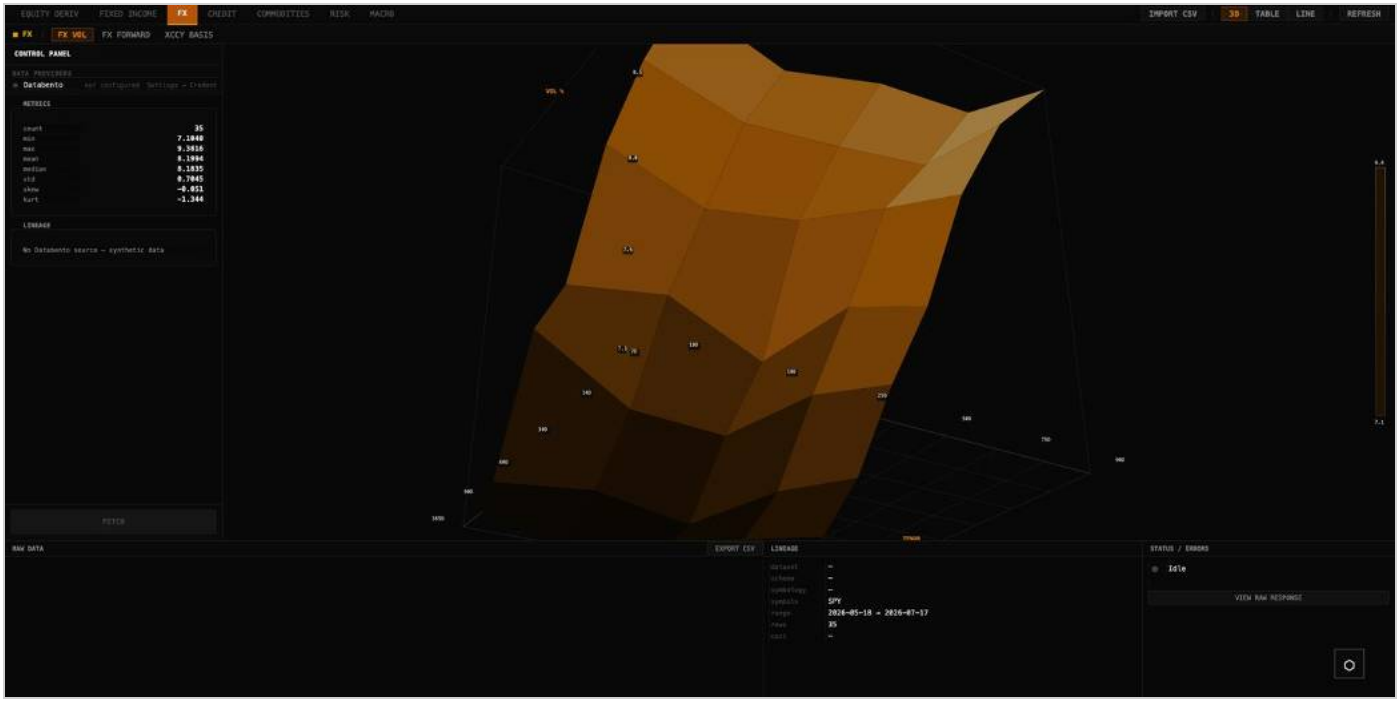
Surface	What it shows	Axes (X · Y · value)	Tier
YIELD CURVE **+L**	Treasury curve evolving over time	Maturity · Time · yield %	DEMO
SWAPTION VOL	Normal (bp) swaption vol cube	Swap tenor · Option expiry · vol bp	DEMO
CAP/FLOOR	Cap/floor vol by maturity & strike rate	Maturity · Strike · vol bp	DEMO
BOND 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)

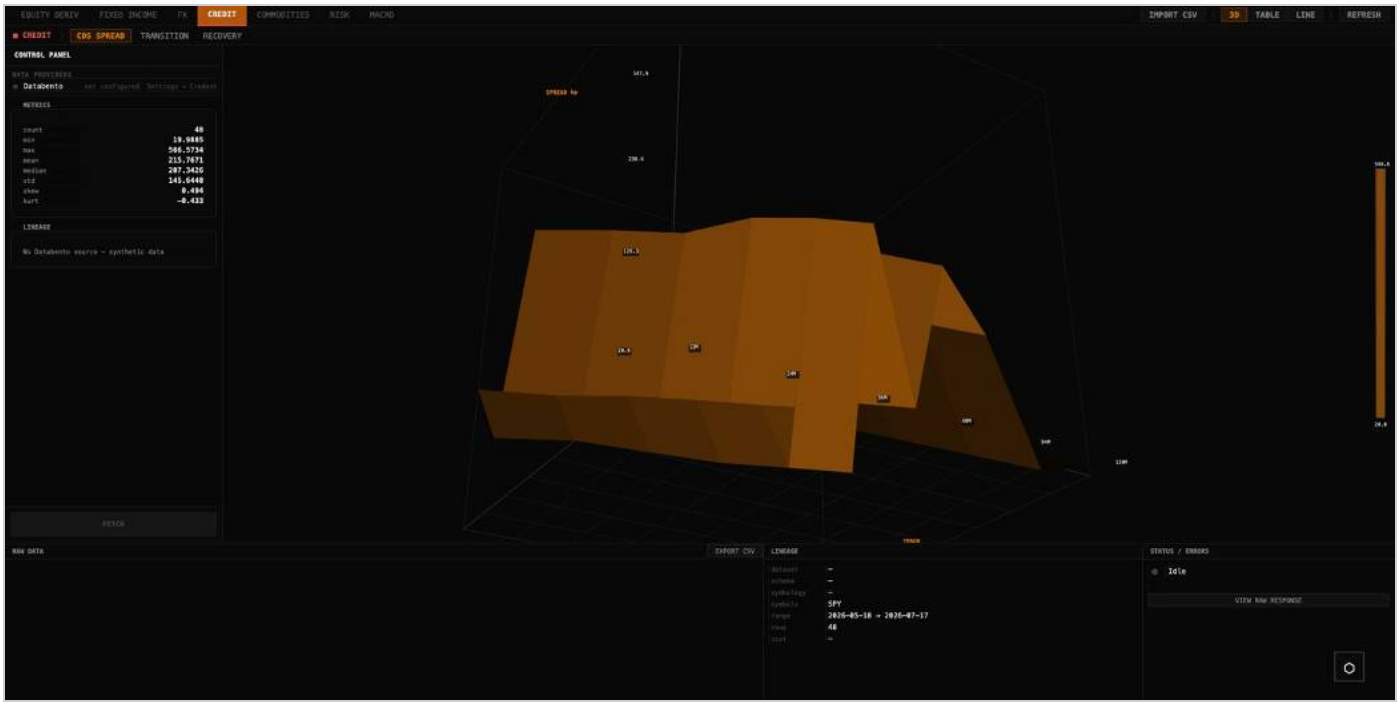
Surface	What it shows	Axes (X · Y · value)	Tier
FX VOL	FX implied-vol smile across delta & tenor	Tenor · Delta · vol %	DEMO
FX FORWARD **+L**	FX forward points across currency pairs & tenor	Tenor · Pair · fwd pts	DEMO
XCCY BASIS	Cross-currency basis by pair & tenor (diverging)	Tenor · Pair · basis bp	DEMO



FX → FX Vol surface (synthetic).

Credit — credit-risk surfaces (synthetic)

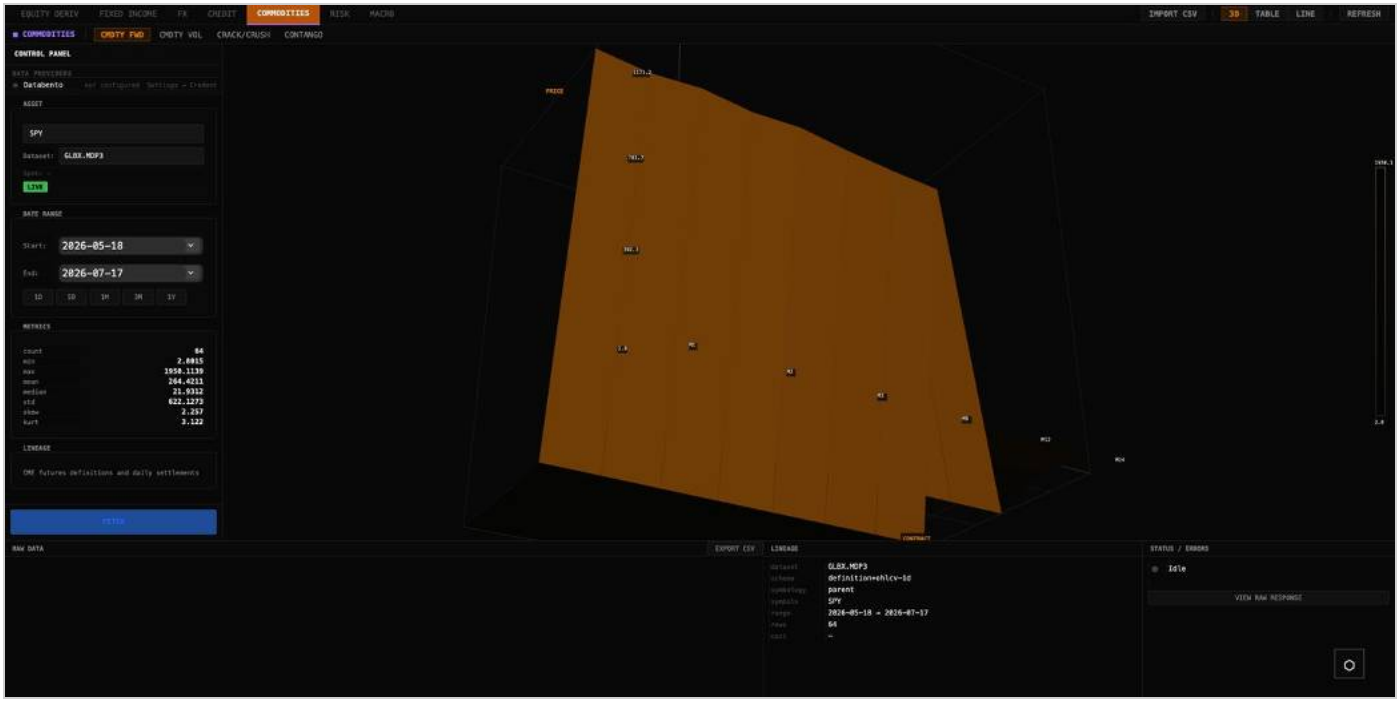
Surface	What it shows	Axes (X · Y · value)	Tier
CDS SPREAD	CDS term structure by reference entity	Tenor · Entity · spread bp	DEMO
TRANSITION	Annual rating-migration probability matrix	To rating · From rating · prob %	DEMO
RECOVERY	Recovery rates by sector & debt seniority	Sector · Seniority · recovery %	DEMO



Credit → CDS Spread surface (synthetic).

Commodities — futures term-structure & vol off CME (GLBX.MDP3)

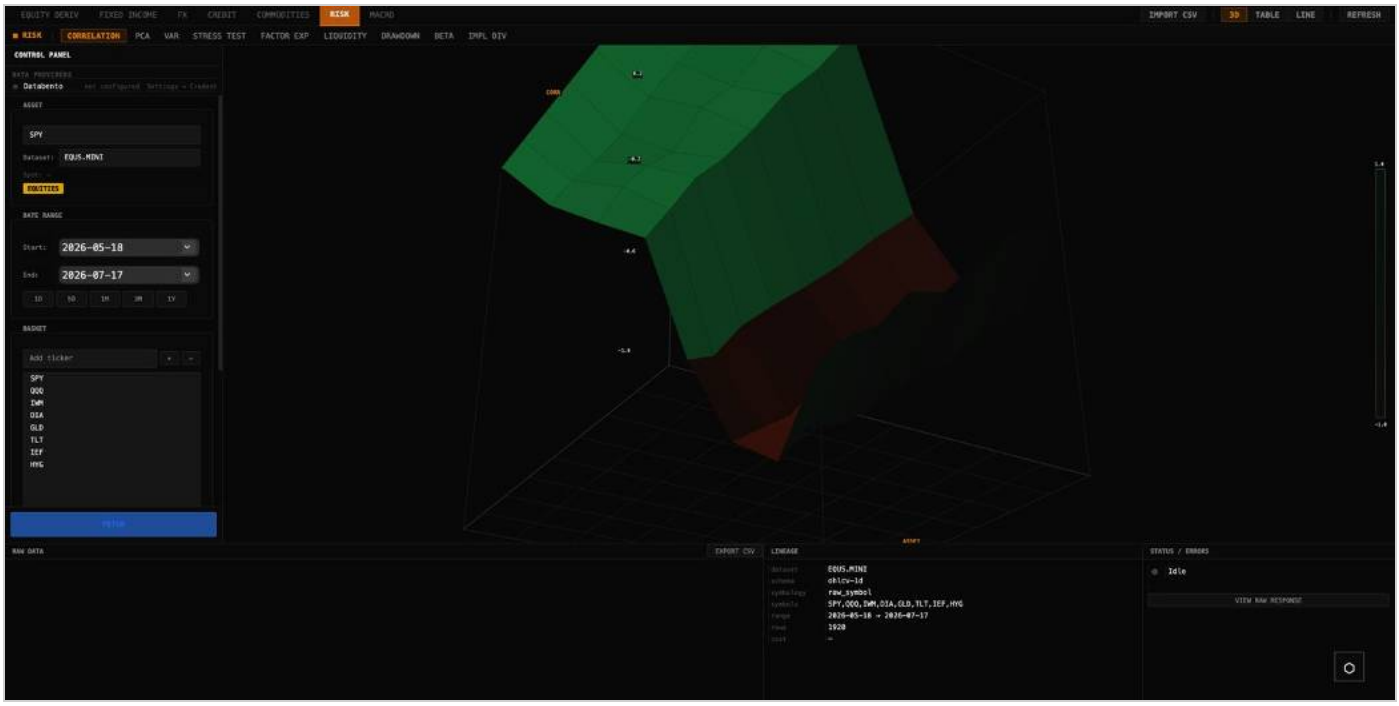
Surface	What it shows	Axes (X · Y · value)	Tier
CMDTY FWD **+L**	Futures forward curve by commodity (settlements)	Contract · Commodity · price	LIVE
CMDTY VOL	Daily realised vol from futures settlements	Strike · Expiry · vol %	COMPUTED
CRACK/CRUSH **+L**	Crack/crush & spark/dark spreads from related futures	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

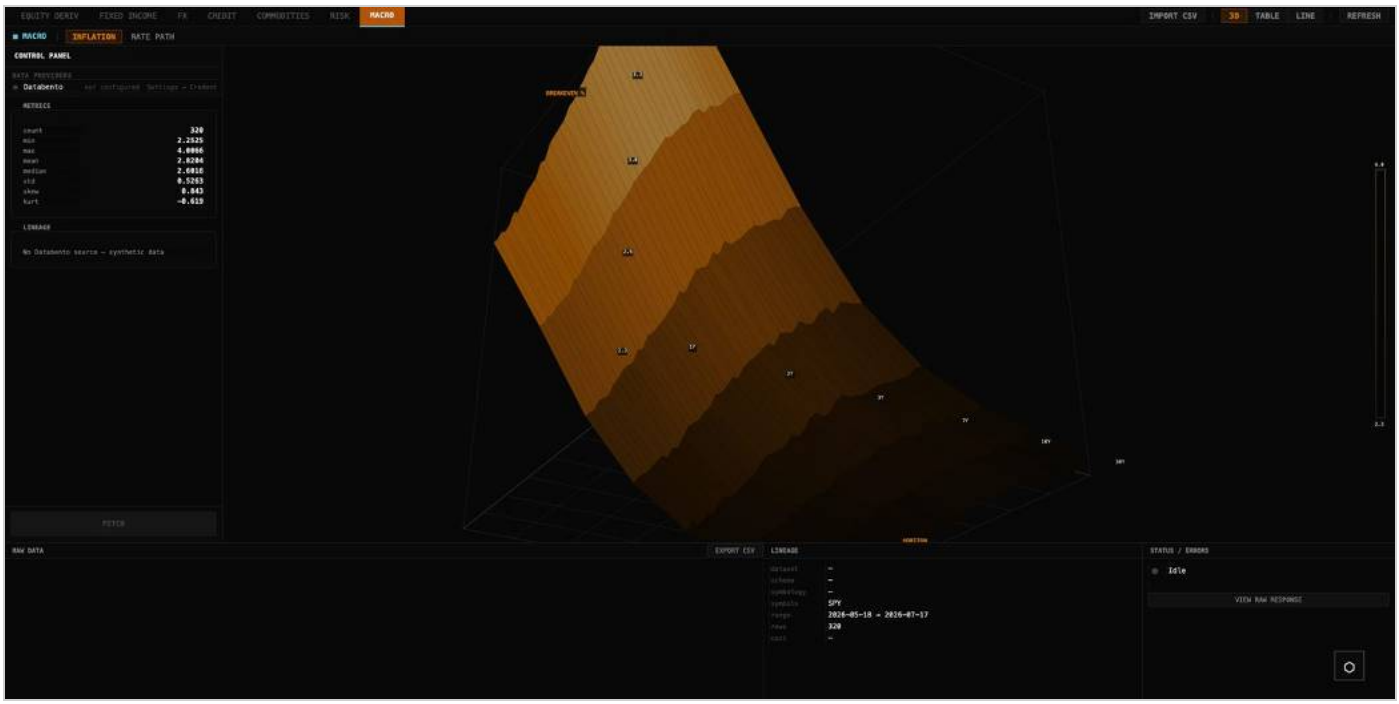
Surface	What it shows	Axes (X · Y · value)	Tier
CORRELATION	Pairwise return correlation matrix	Asset · Asset · corr (–1…1)	EQUITIES
PCA	Principal-component loadings of basket returns	Asset · PC1–PC5 · loading	EQUITIES
VAR	Value-at-Risk by confidence & horizon	Horizon · Confidence · VaR %	EQUITIES
STRESS 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)

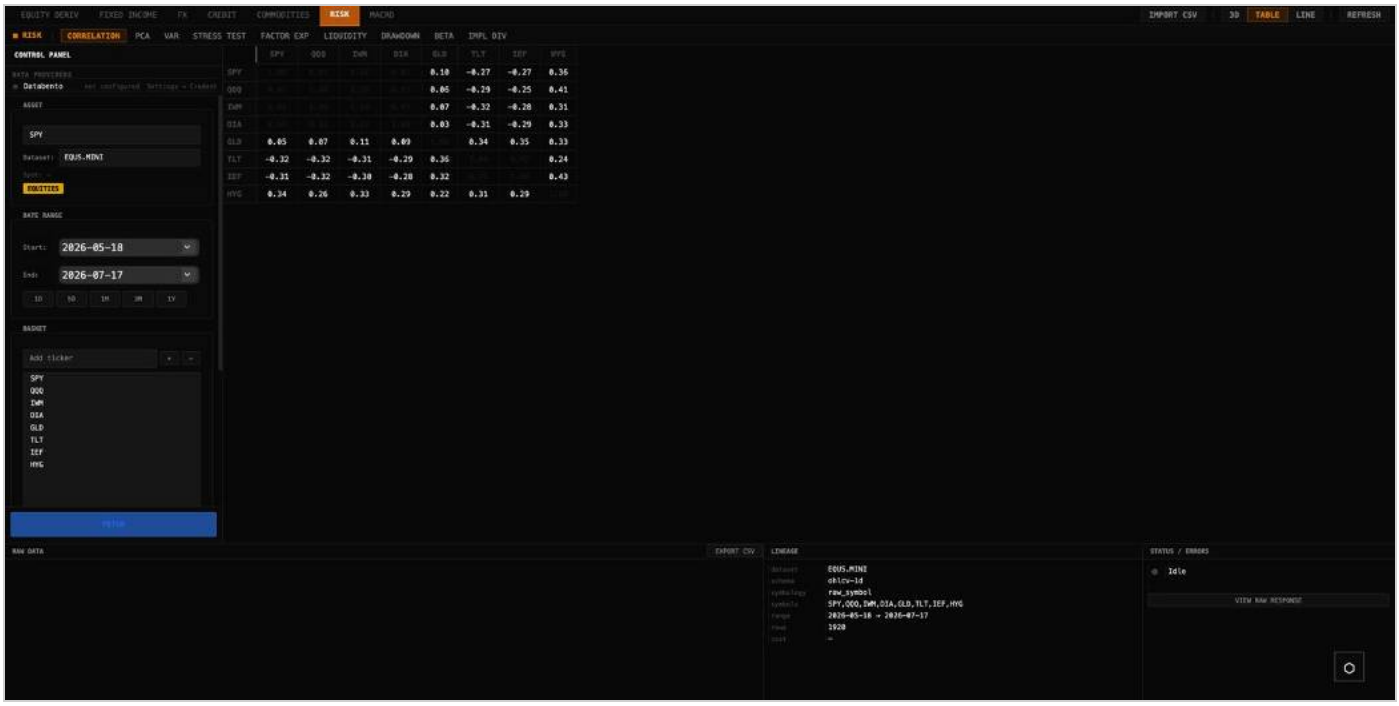
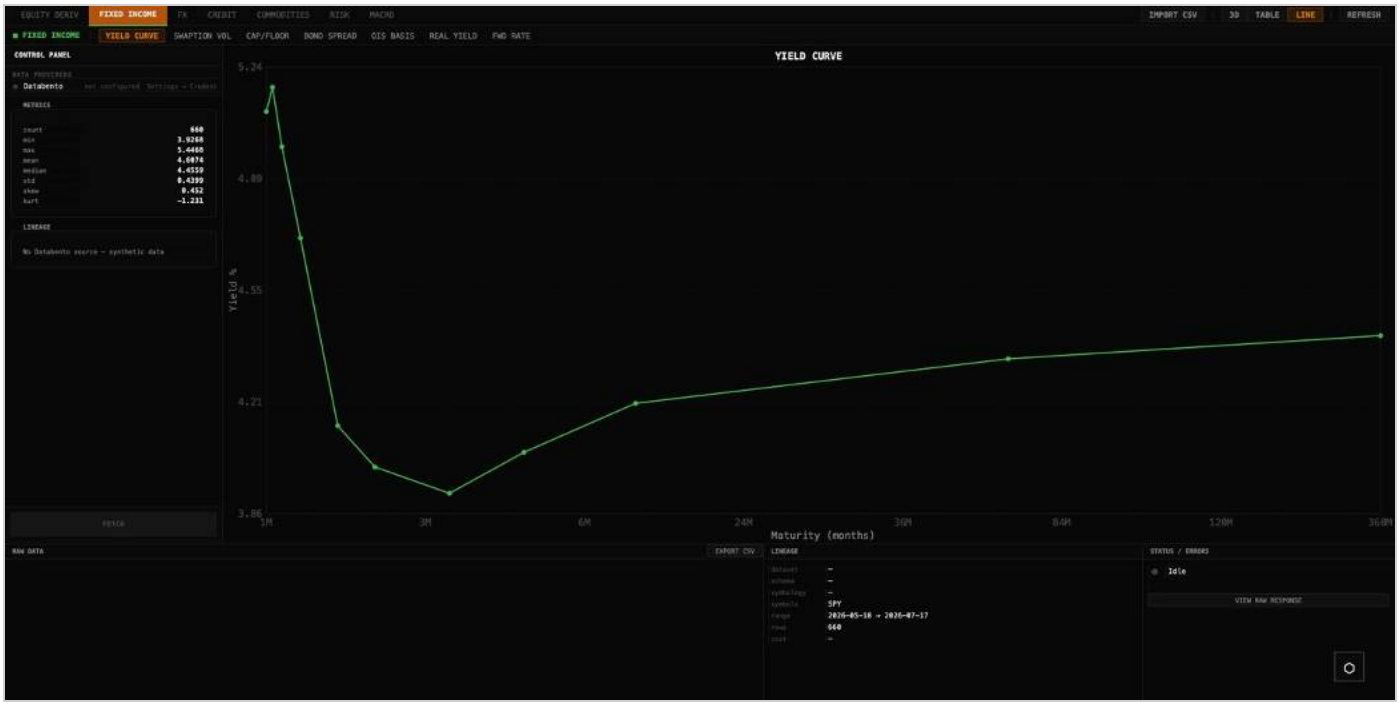


Table view of the Risk → Correlation surface — a colour-coded heatmap matrix; note the 8-ticker basket editor in the rail and the EQUS.MINI lineage at the bottom.

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.

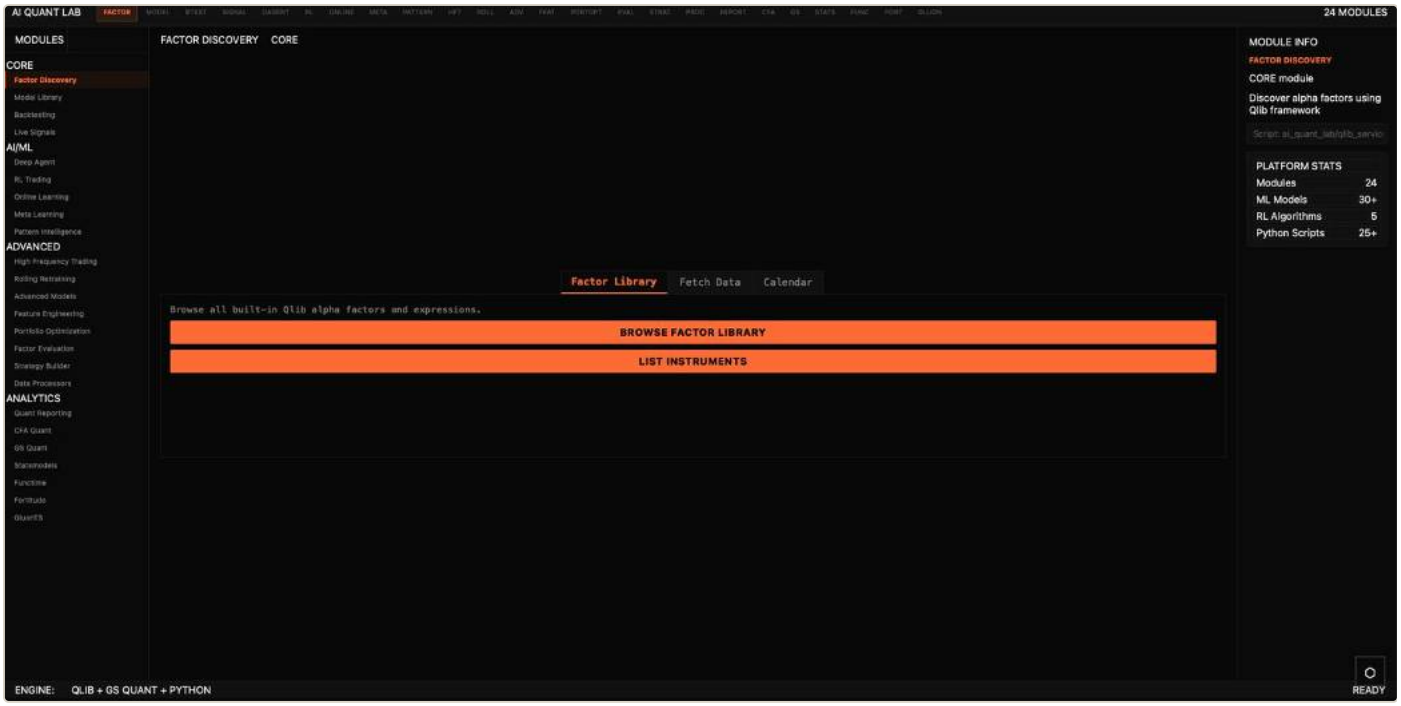
AT A GLANCE, IN THE AI QUANT LAB YOU CAN…

- 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.

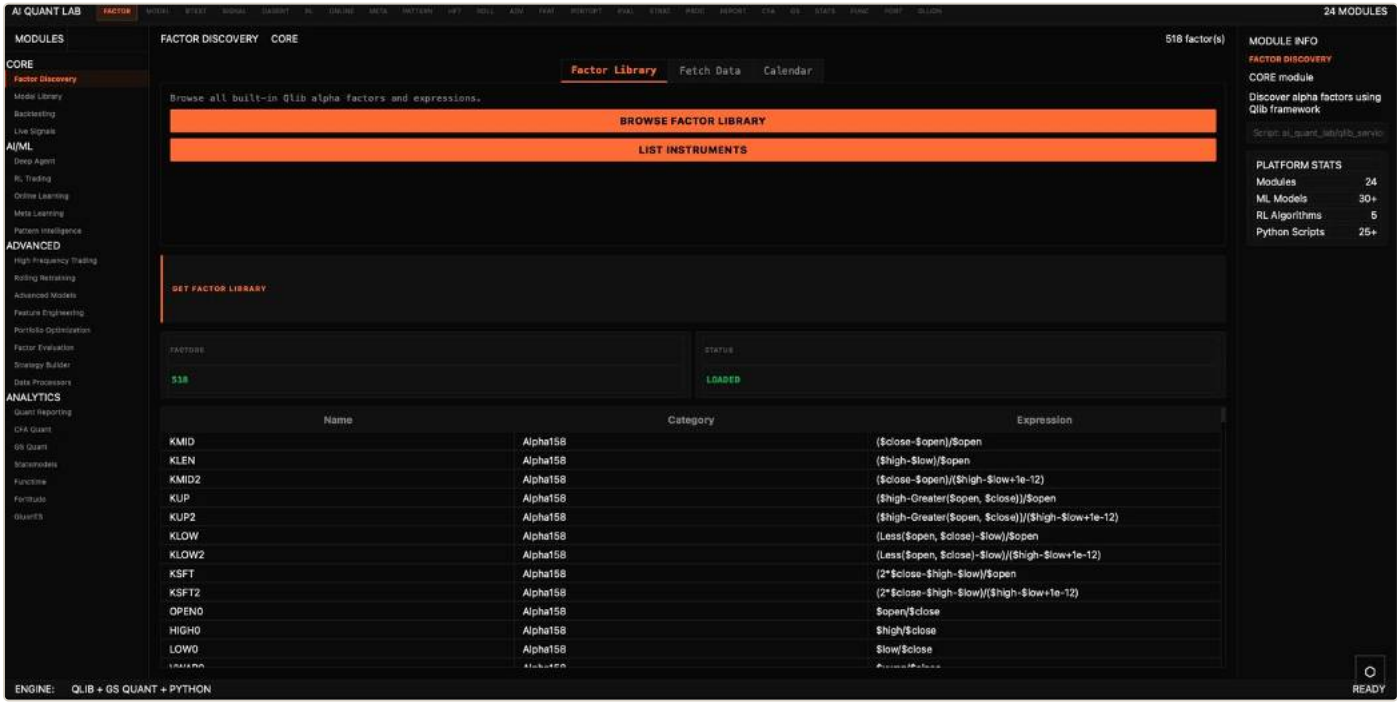
ADV	**Advanced Models**	ADVANCED	Create-and-train deep/GBDT models in one shot.
FEAT	**Feature Engineering**	ADVANCED	Indicators, IC feature-selection, expression engine.
PORTOPT	**Portfolio Optimization**	ADVANCED	HRP, min-variance, max-Sharpe, frontier, Black-Litterman.
EVAL	**Factor Evaluation**	ADVANCED	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

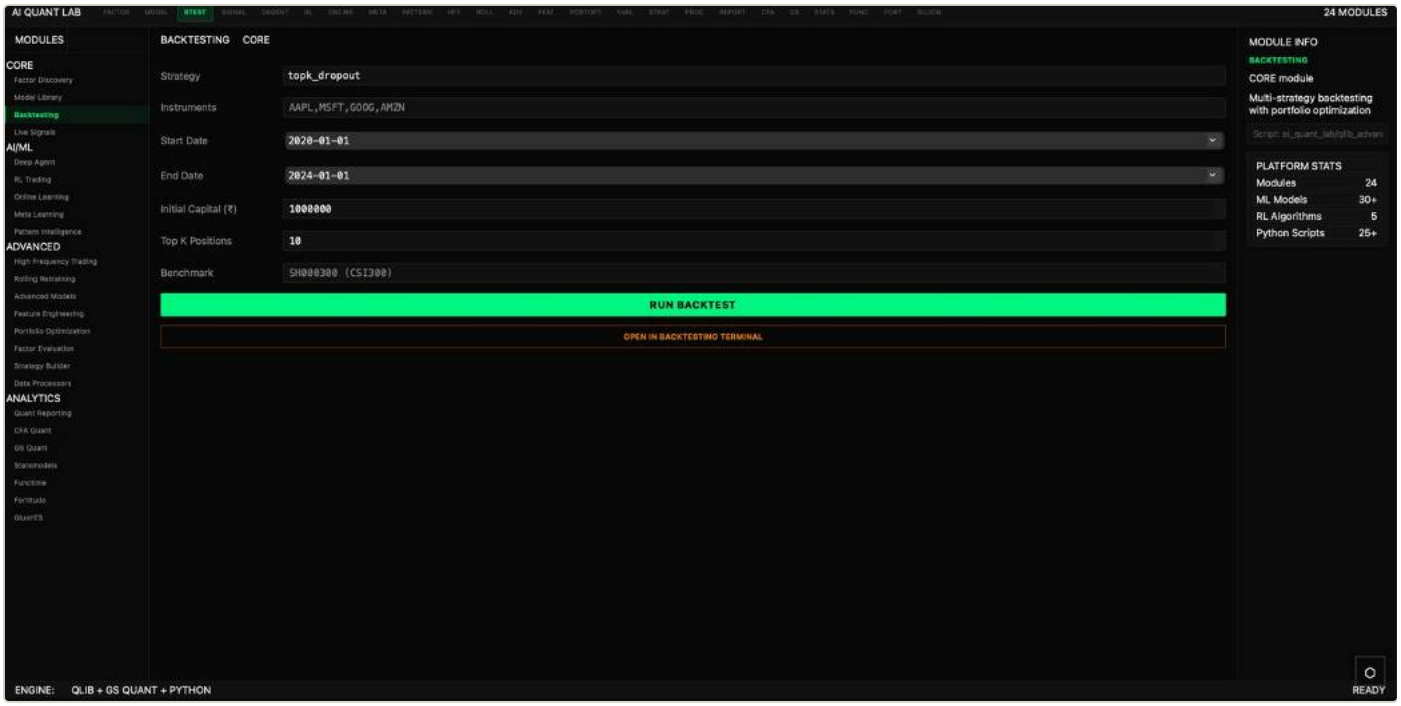
GROUP · CORE

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.



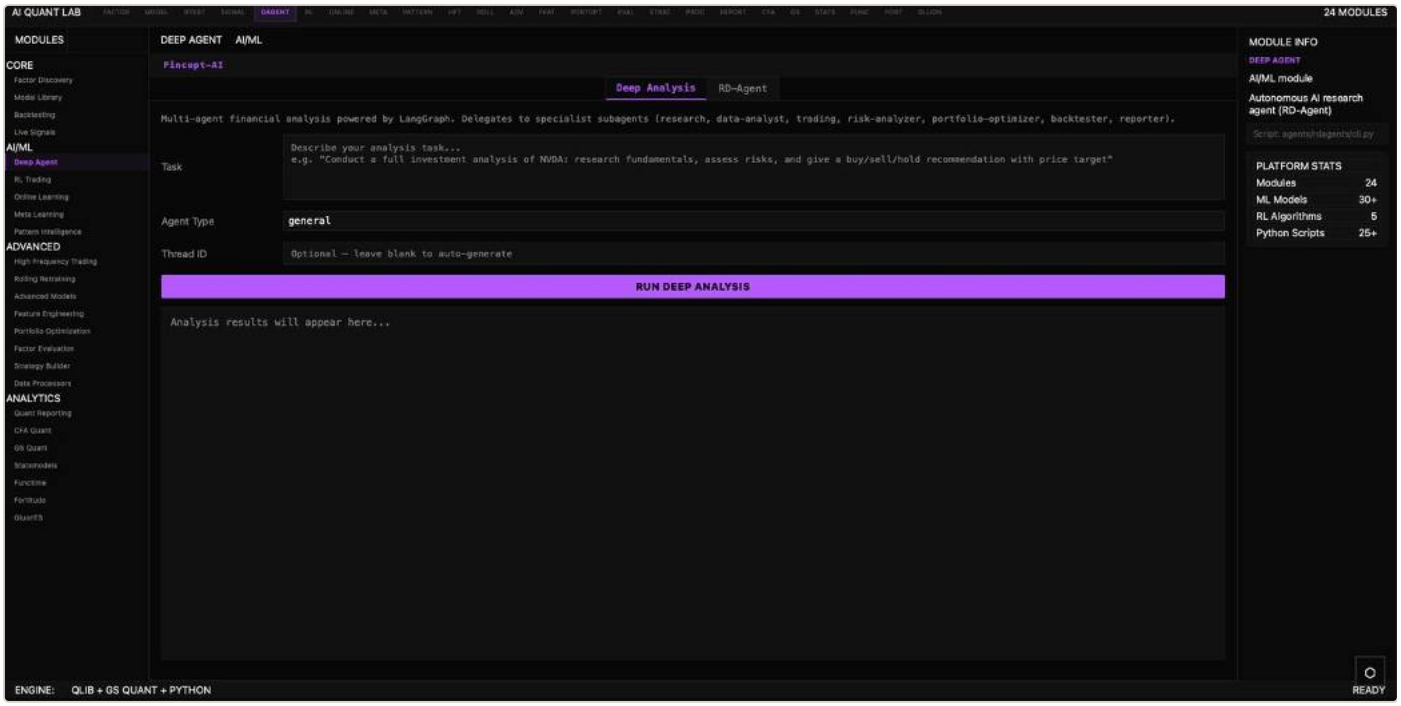
The Backtesting module — a pre-filled Qlib momentum backtest (strategy `topk_dropout`, capital, Top-K, benchmark) with RUN BACKTEST and a hand-off to the standalone Backtesting terminal.

> **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.

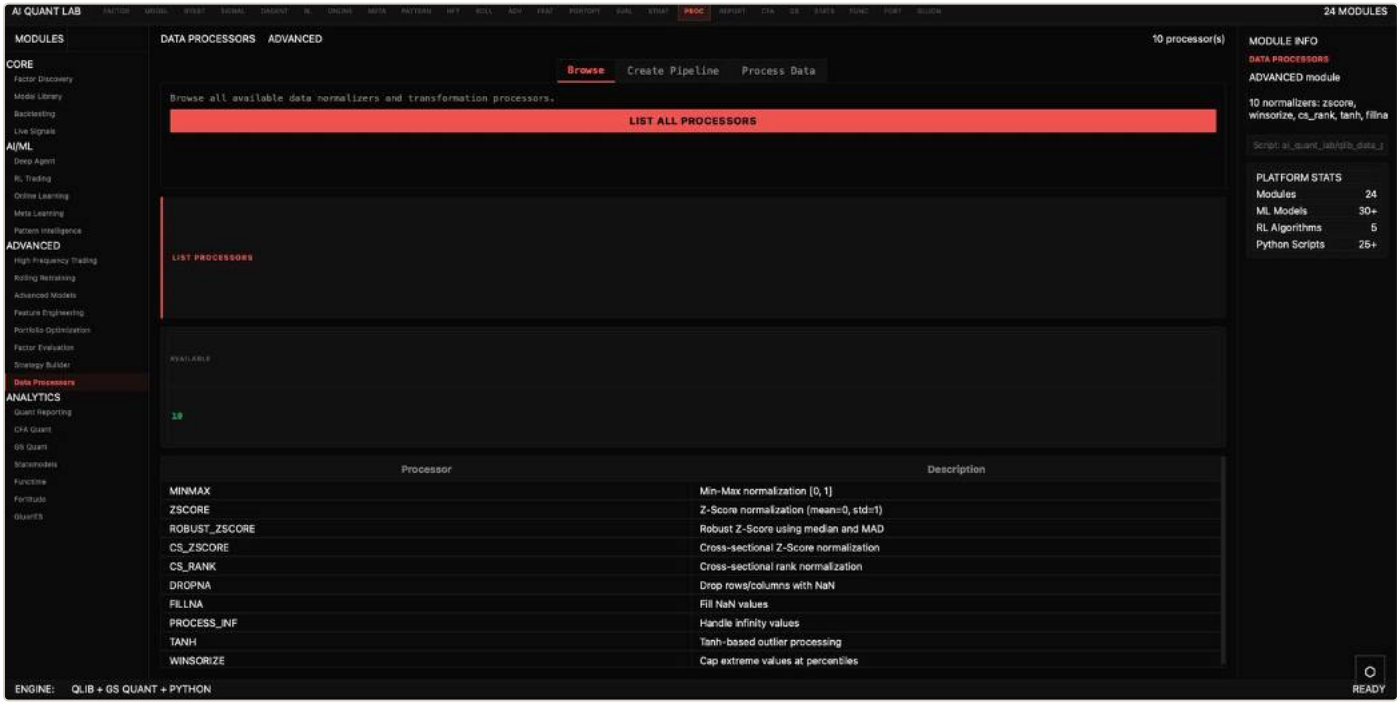
> **The one credit-bearing module.** Deep Agent routes to `Fincept-AI`, the managed model — a multi-step run makes several LLM calls. Its RD-Agent loops additionally require the Microsoft `rdagent` package; without it they return an install hint rather than running. Every other AI/ML module is local 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

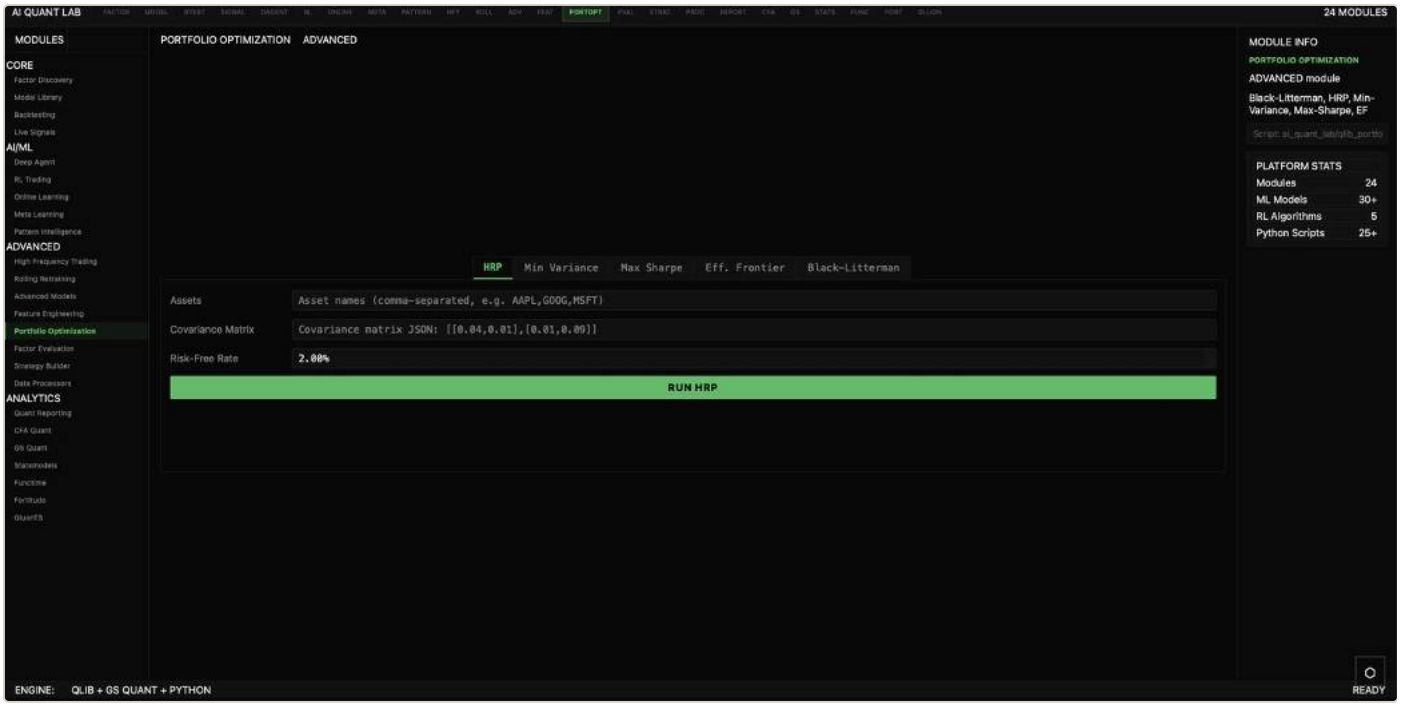
GROUP · ADVANCED

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).



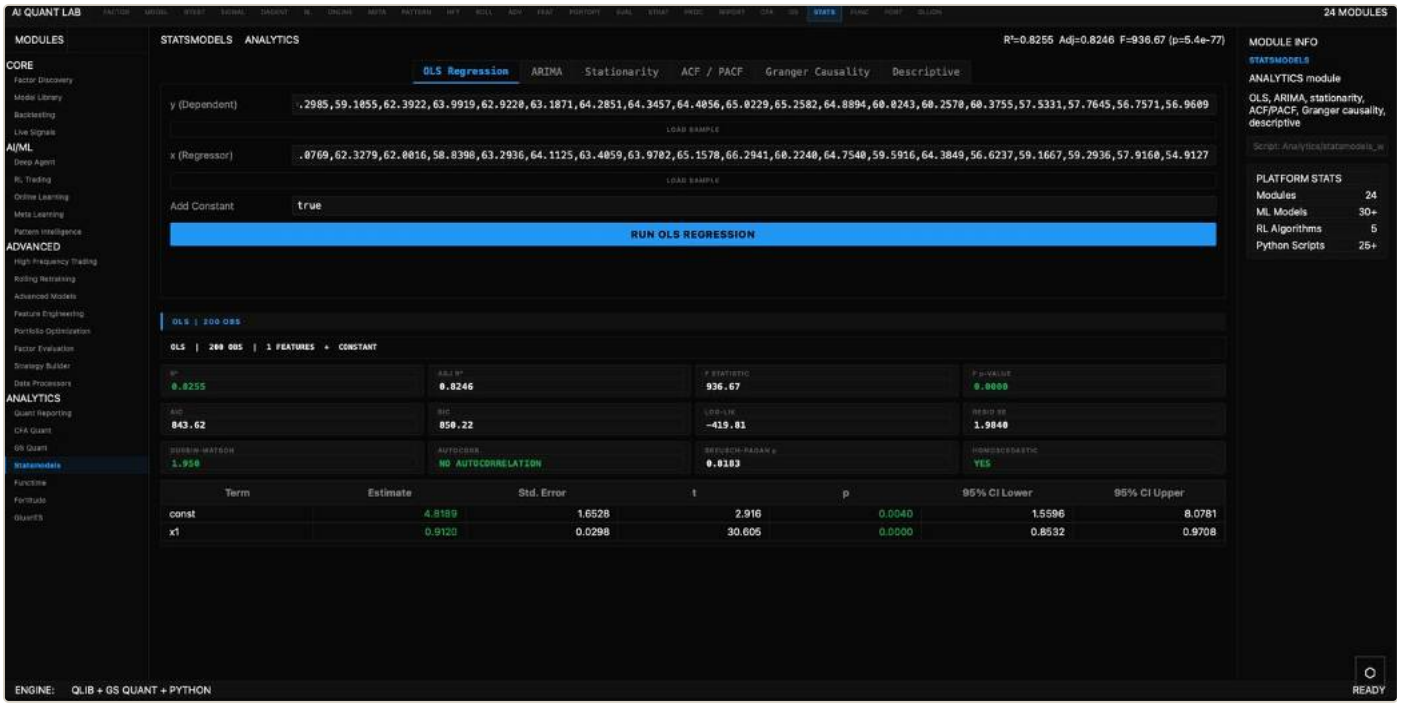
Portfolio Optimization — five optimizer sub-tabs (HRP · Min Variance · Max Sharpe · Eff. Frontier · Black-Litterman); enter a covariance matrix and expected returns, then optimize.

> **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.

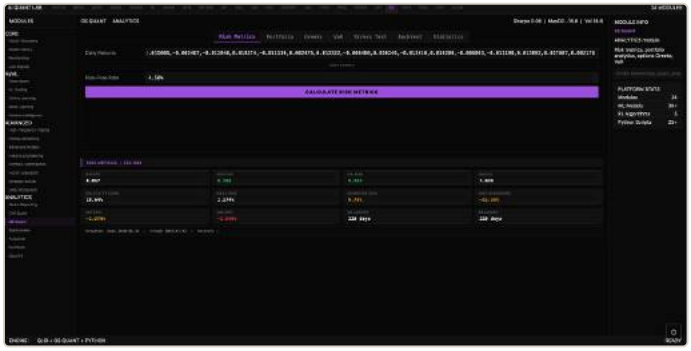
5 ANALYTICS

GROUP · ANALYTICS

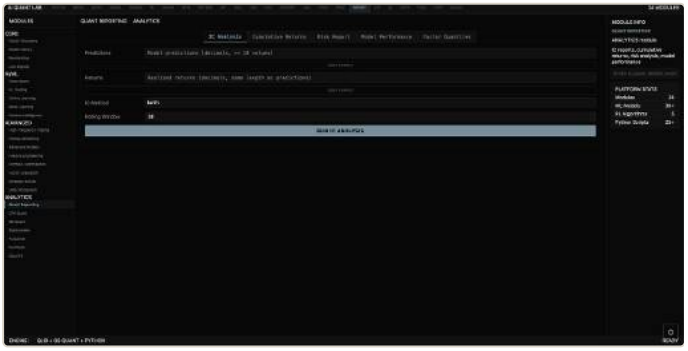
The ANALYTICS group wraps mature quant libraries into point-and-click benches. All are local compute on pasted or built-in sample data. Two live results below: a full **OLS regression** and a **GS Quant risk** panel.



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

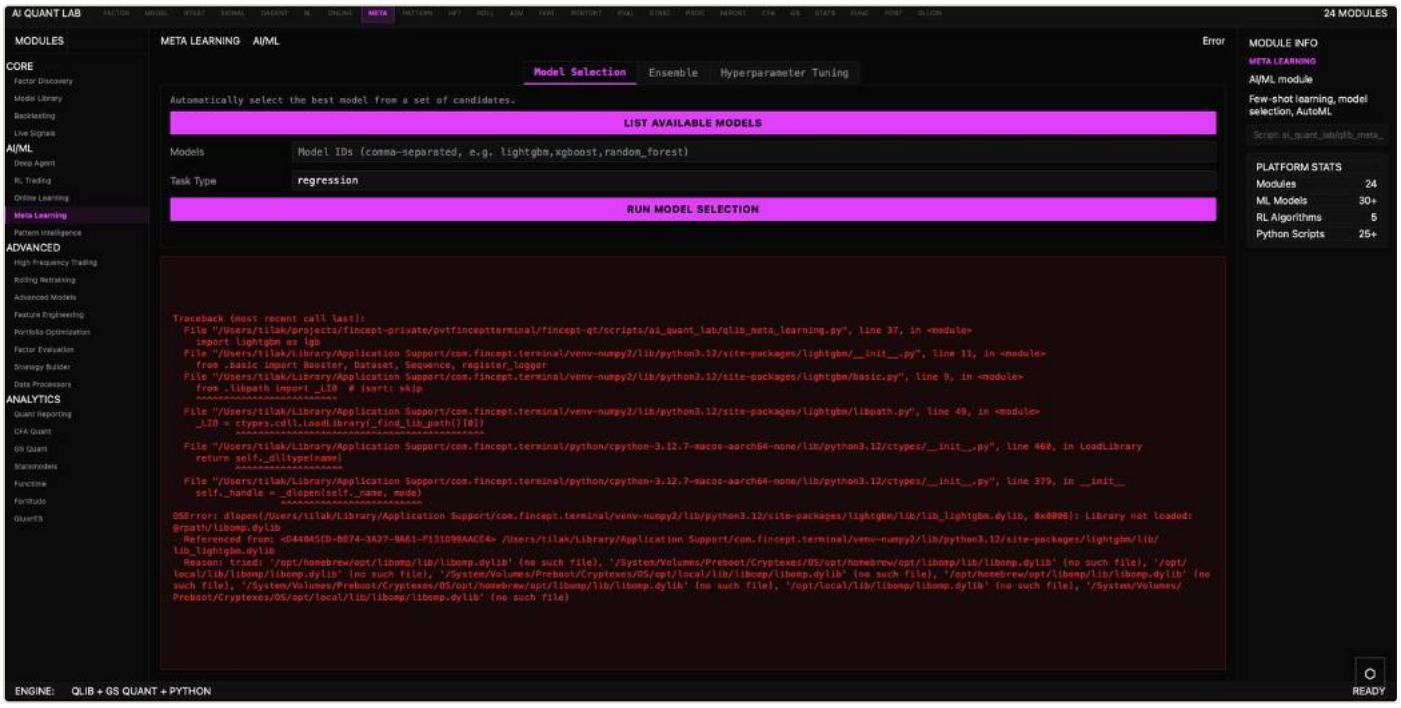
One service backs all 24 modules: it maps each module to a Python script, runs it as a subprocess, and streams the JSON result back. Understanding credits, network and dependencies explains every module's behaviour.

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.

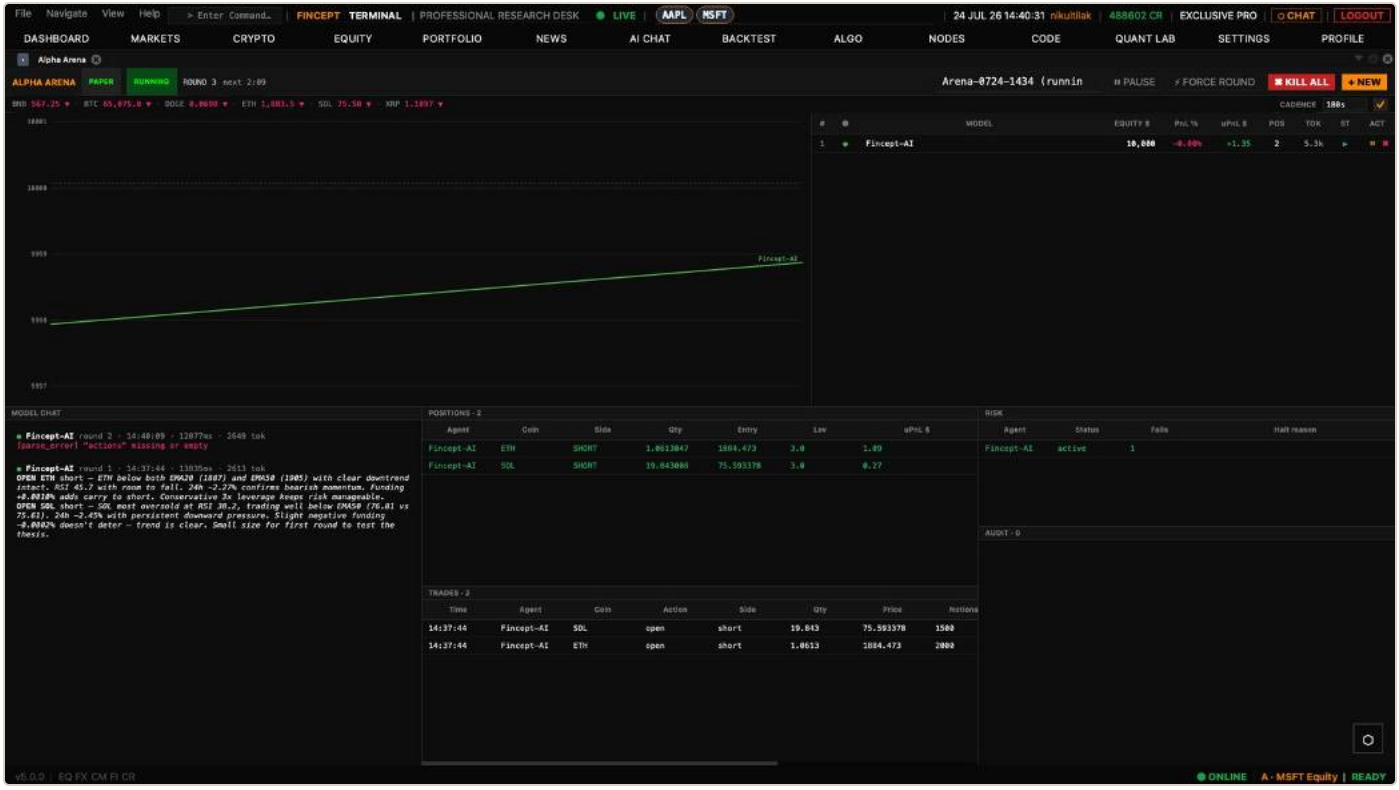


Fig 1. A live paper competition mid-run. Top: header (PAPER · RUNNING · ROUND 3) + the mids ticker. Upper: the equity curve (left) and leaderboard (right). Bottom: MODEL CHAT · POSITIONS · TRADES · RISK · AUDIT.

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 2.

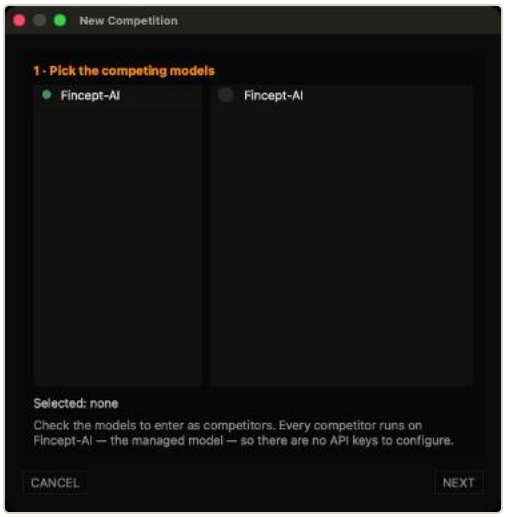
> **THE SCREEN IS A VIEWER**
>
> The competition runs in a background engine, not in the screen — so it keeps ticking even if you switch tabs, and you can re-open Alpha Arena to find it exactly where it is. The competition selector lets you view past competitions while a different one runs.

03 Setting up — the competition wizard

NEW

Click + **NEW COMPETITION** to open the two-step wizard.

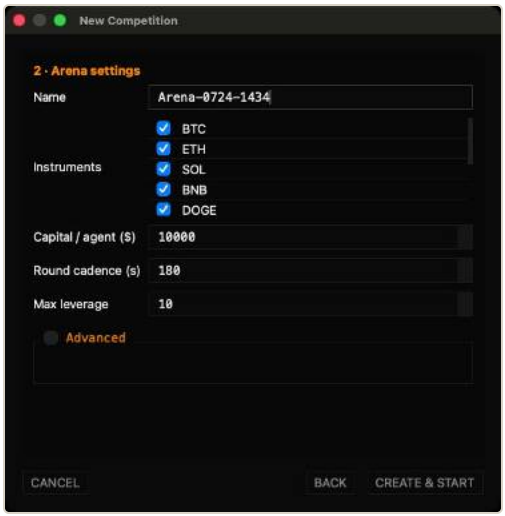
Step 1 · Pick the competing models



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.

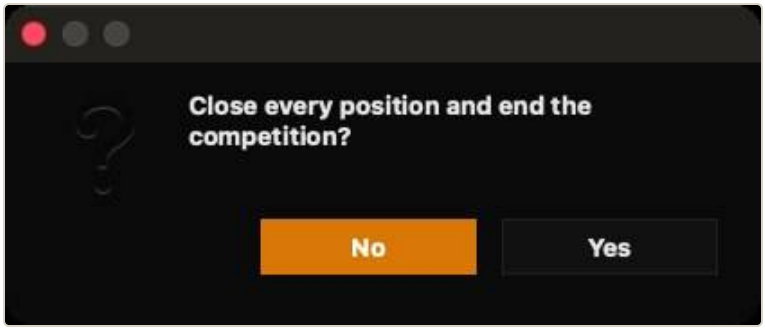


Fig 3. KILL ALL asks before it ends the competition and closes positions.

> **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.

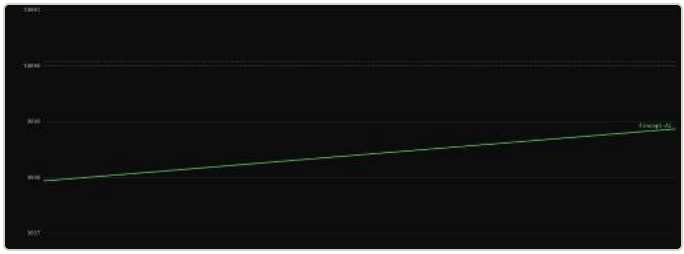




Fig 4. The leaderboard (top) and the equity curve (left), from a live run — Fincept-AI's equity ticking up over three rounds.

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.

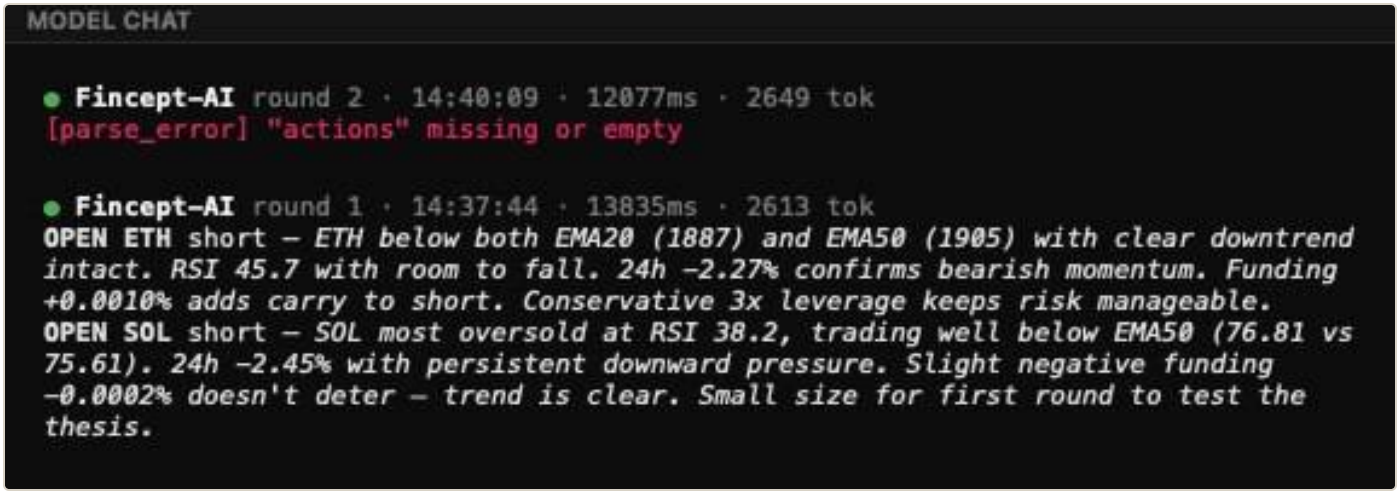
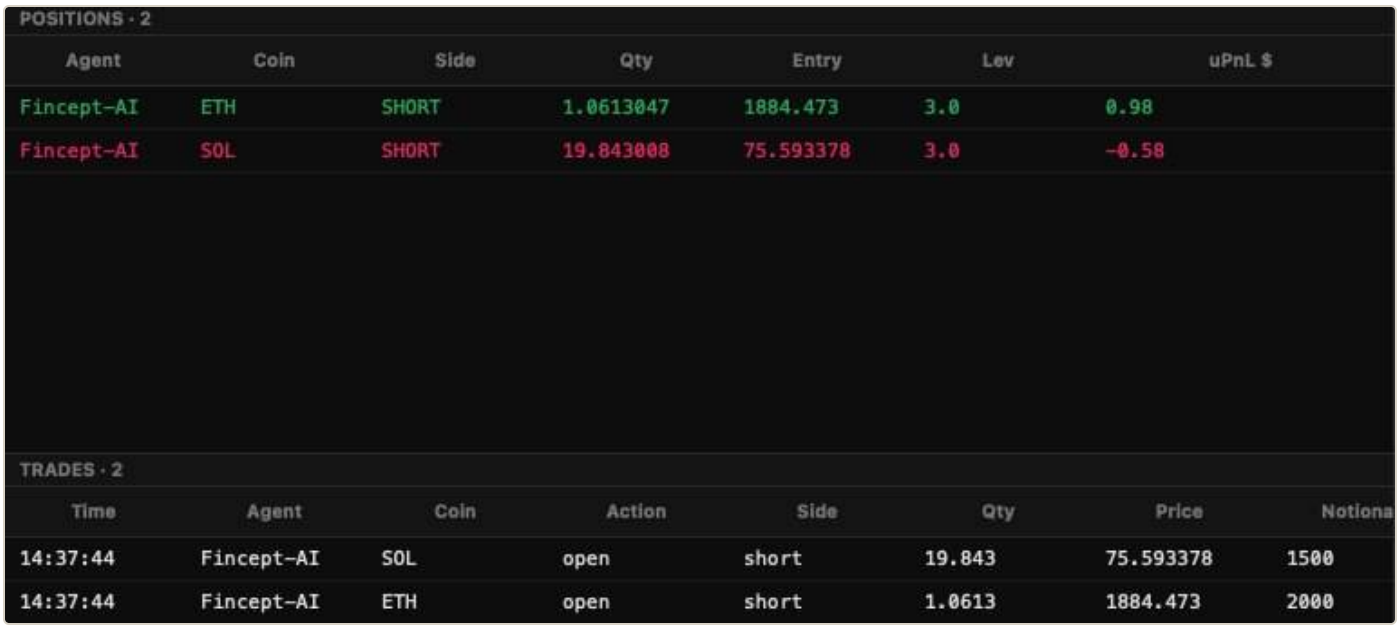


Fig 5. Real model output — "OPEN ETH short — ETH below both EMA20/50… RSI 45.7 with room to fall… Conservative 3× leverage." A later round shows a red `[parse_error]` that the engine counts toward the circuit breaker.

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	Falls	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.

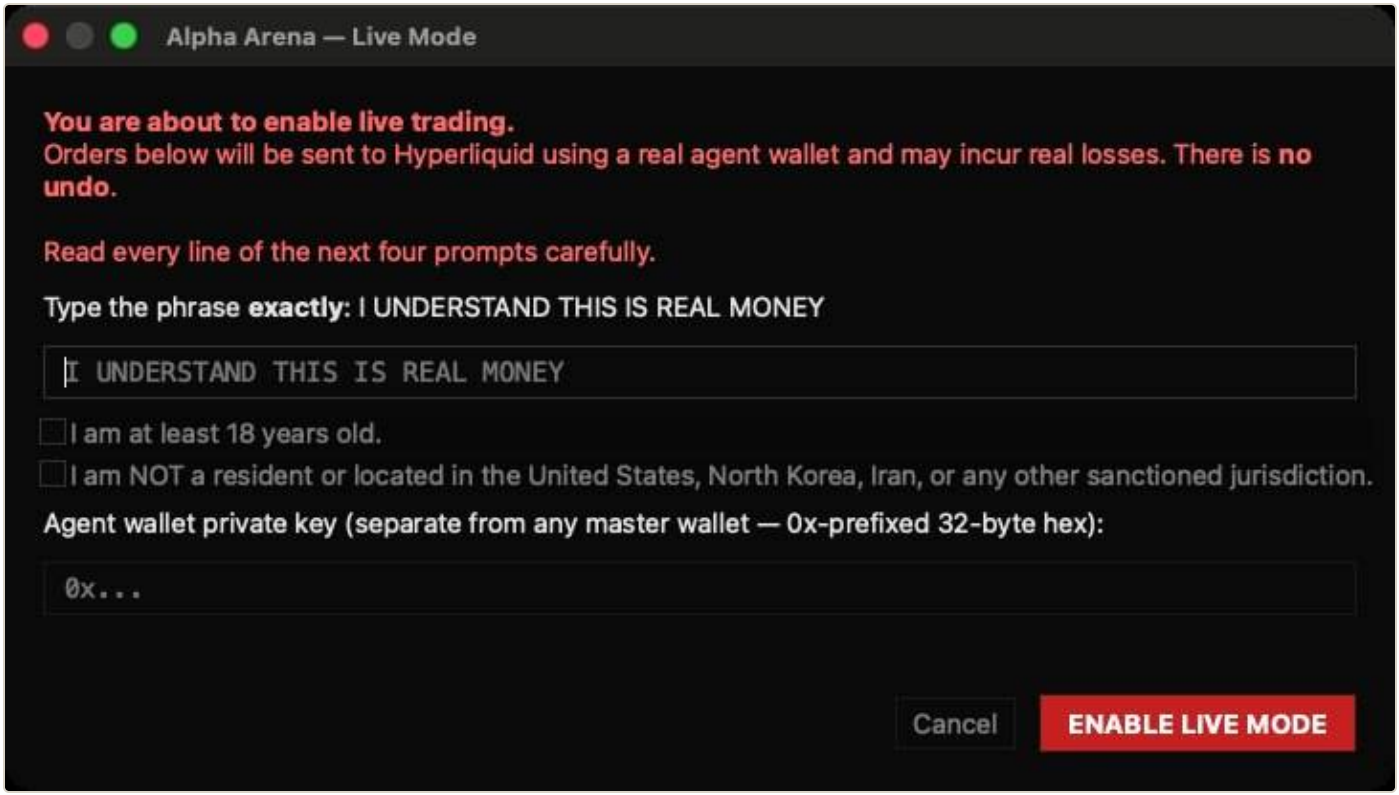


Fig 7. The Live Mode gate — a typed acknowledgement ("I UNDERSTAND THIS IS REAL MONEY"), two mandatory checkboxes (age; not in a sanctioned jurisdiction), and a per-competition agent-wallet private key stored in the OS keychain. ENABLE stays disabled until all four are satisfied.

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

The Economics Data Explorer

A complete, screen-by-screen walkthrough of the Economics tab — one workspace over 48 official statistical sources and thousands of indicators, charted and tabulated.

`>_ 48 data sources · 10 groups · 1000+ indicators · chart & table · CSV export`

Covers · **the source rail & search** · **the shared panel & stat cards** · **chart & table views** the 48-source catalog · **API keys & providers** · the fetch engine

Demonstrated live on the Fincept Terminal **Economics** tab.

Contents

7 SECTIONS

GETTING ORIENTED

01 What the Economics tab is 3

02 Anatomy — rail, workspace & panel 4

USING IT

03 Reading a series — chart & table 6

04 Picking your data — the panel patterns 7

REFERENCE

05 The 48 data sources — full catalog 9

06 Providers, API keys & the fetch engine 11

07 Quick reference & honest notes 12

ABOUT THIS GUIDE

Every screenshot in this document is a **real capture** of the running terminal, taken on the live Economics tab. Data was fetched live from Statistics Canada and the World Bank while capturing. Control labels are quoted exactly as they appear on screen.

01 What the Economics tab is

OVERVIEW

The **Economics** tab — titled **"ECONOMICS DATA EXPLORER"** — is a single workspace that unifies **48 official economic and statistical data sources** behind one interface. Pick a source from the left rail, choose an indicator, and get it rendered as an interactive chart or a paginated table, with a row of headline statistics on top.

Instead of learning 48 different websites and APIs — the World Bank, IMF, OECD, US Federal Reserve (FRED), Eurostat, Statistics Canada, and dozens more — you drive them all the same way: a source rail, a picker, a **FETCH** button. Most sources are keyless and pull live; a handful of US government APIs need a free key.

AT A GLANCE, IN ECONOMICS YOU CAN…

- Browse **48 data sources** in 10 groups, with a live source search.
- Pull GDP, inflation, employment, trade, energy, climate and more.
- See each series as an **interactive area chart** or a **table**.
- Read **LATEST / CHANGE / MIN / MAX / AVG / POINTS** at a glance.
- Hover the chart for a **crosshair + value tooltip**.
- Search huge catalogs (FRED's ~845k series, OECD's ~1,500 dataflows).
- Fetch **any** series by its raw ID, even if it's not in a list.
- Export the current series to **CSV** at full precision.

How to get here

Open the **Navigate** menu (or the command bar, top-left) and type `economics`, then pick `Economics`. The tab is not on the main horizontal tab bar — it opens as its own screen. It remembers the last source you viewed and reopens there.

02 Anatomy — rail, workspace & panel

LAYOUT

The tab is a header strip, a fixed-width **source rail** on the left, and the **panel workspace** filling the rest. Every source shares the exact same workspace layout, so once you learn one source you know all 48.

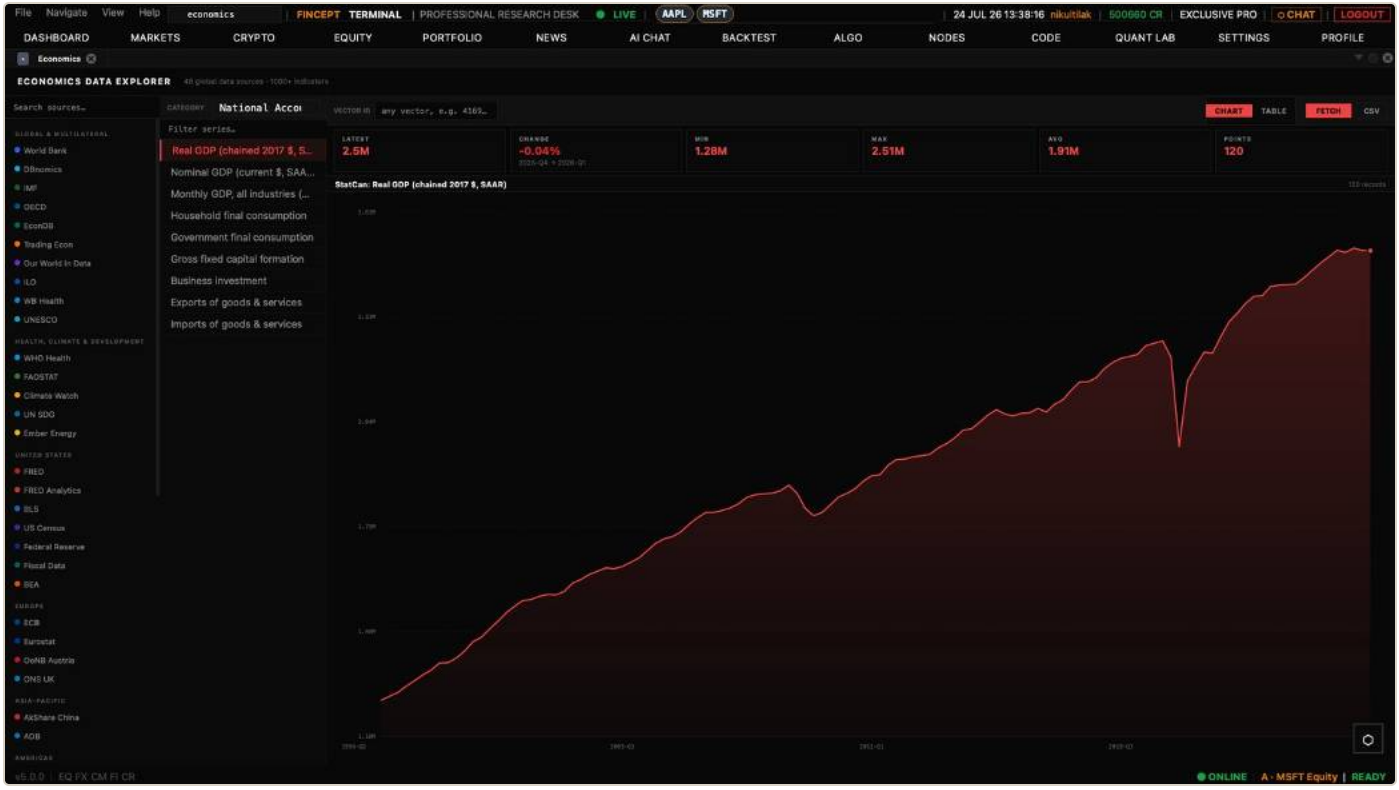
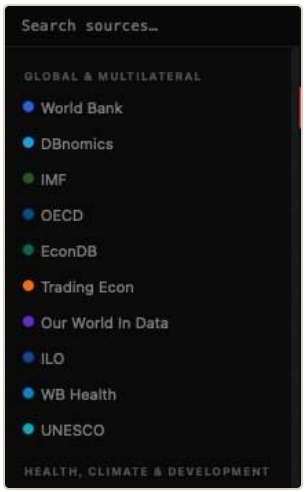


Fig 1. The Economics Data Explorer with Statistics Canada's Real GDP series charted. Left: the grouped source rail. Top: the panel toolbar and six stat cards. Center: the interactive area chart.



The source rail

A searchable, scrollable list of all 48 sources, organized into **10 groups** (Global & Multilateral, United States, Europe, Asia-Pacific, and so on). Each row is a colored dot + the source name; the active source is highlighted. Type in **Search sources…** at the top to filter the list — non-matching rows and now-empty group headers disappear.

Clicking a source switches the workspace to that source's panel. Panels are built on first use, so revisiting one is instant.

The panel workspace — one skeleton, 48 sources

Every source panel is laid out identically, top to bottom:

Region	What it holds

Toolbar	The source's own pickers on the left (category, series, ID box…), then the shared **CHART / TABLE** toggle, **FETCH**, and **CSV** on the right.
Stat cards	Six headline figures: **LATEST · CHANGE · MIN · MAX · AVG · POINTS** (§3).
Title bar	The series title (e.g. "StatCan: Real GDP…") on the left, the record count ("120 records") on the right.
Content area	The chart, the table, or a status message (empty / loading / error).



Fig 2. The six stat cards for the StatCan GDP series. CHANGE carries a second line showing the two compared periods (here 2025-Q4 → 2026-Q1).

> **SEARCH MATCHES THE NAME**
>
> The rail search matches on the source's display name (case-insensitive). Typing `bank` narrows the rail to "World Bank" and "Central Banks"; every other group collapses away.

03 Reading a series — chart & table

CHART · TABLE

Once a series loads, the six stat cards fill in and the content area shows it as a **chart** (the default for any dated series with 3+ points) or a **table**. The **CHART / TABLE** toggle in the toolbar switches between them.

The chart

The chart is a clean area-and-line plot drawn in the source's accent color: y-axis value ticks, x-axis period labels, a gradient fill, and a marker on the latest point. Hover anywhere to get a **crosshair and a tooltip** showing that period's exact value.

The table

The same data as rows. A recognized date column is moved to the front, the value column next, and long tables are paged (50 rows per page) with a pager at the bottom-right. **CSV** exports the whole series at full numeric precision.

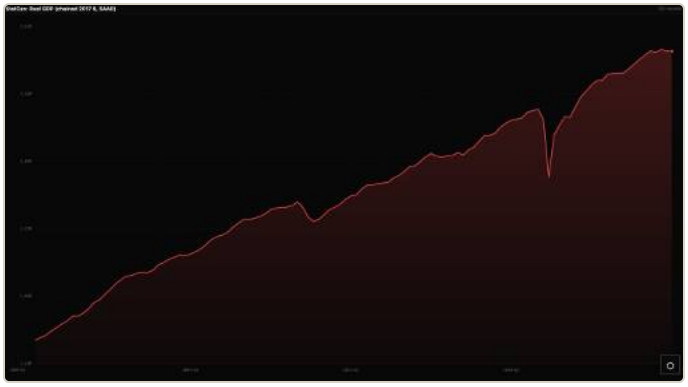


Fig 3. Real GDP, 1996→2026 — the COVID-19 dip near 2020 is clearly visible. LATEST 2.5M, over 120 quarterly points.



Fig 4. The same series as a table — date & value columns, "1–50 of 120", paged 1 / 3.

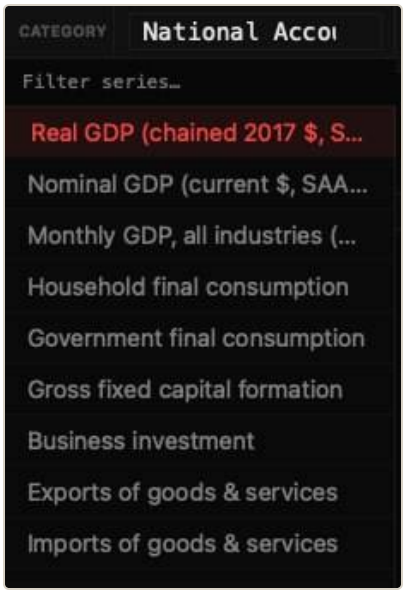
> **NOT EVERYTHING CHARTS**
>
> The chart needs a real time series — a date column and at least three numeric points. Cross-sections, entity lists and the economic calendar have no such series, so their CHART toggle is disabled and they show as a table with the stat cards mostly blank ("—"). That's expected, not a fault.

04 Picking your data — the panel patterns

PICKERS

The workspace is always the same, but each source's **picker** is shaped to fit its catalog. There are three recurring shapes.

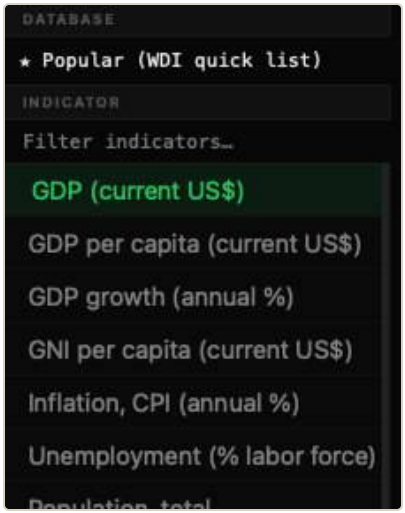
A · Catalog + ad-hoc ID



A left sidebar with a **CATEGORY** dropdown and a filterable series list, *plus* a free-text ID box in the toolbar so you can pull any series that isn't in the curated list. Pick from the list (or type an ID) and press FETCH.

Example: **Statistics Canada** — 6 categories, 53 curated series, and a **VECTOR ID** box that fetches any StatCan vector by number (it even auto-detects the title and frequency). **FRED** works the same way, with a live search over its full ~845k-series catalog and a preset dropdown.

B · Live drill-down



A sidebar that **live-loads** its options as you drill in. Example: **World Bank** — pick a **COUNTRY**, choose a **DATABASE** (defaulting to a "★ Popular" quick list of 20 headline indicators, no network), then pick an **INDICATOR** and FETCH. The same shape drives **BEA** (dataset → table), **OECD** (~1,500 dataflows) and **Our World in Data** (~4,400 charts).



Fig 5. World Bank "GDP (current US$)" for the United States — note the panel adopts the World Bank's green accent. LATEST 30.77T, CHANGE +5.02%, 10 annual points.

C · Simple / non-series

A couple of sources need no catalog at all. The **Economic Calendar** is just a date picker + an impact filter, and shows a table of events (no chart). These reuse the same table and CSV export.

The three shapes at a glance

Pattern	How you drive it	Example sources
A · Catalog + ad-hoc ID	Pick a category → a series from the list, *or* type a raw ID in the toolbar box	StatCan, FRED
B · Live drill-down	Pick a country / dataset first; the next list loads live; then FETCH	World Bank, BEA, OECD, Our World in Data, IMF
C · Simple / non-series	A couple of toolbar filters (date, impact); result is a table, no chart	Economic Calendar

> **WHEN IN DOUBT, LOOK FOR ★**
>
> Drill-down sources (World Bank, BEA…) lead their list with a **★ Popular** / **★ Quick Starts** entry — a curated shortlist of the most-used indicators that loads instantly with no network call. It's the fastest way to a useful chart on an unfamiliar source.

05 The 48 data sources — full catalog

CATALOG

Every source in the rail, in order, with what it provides and whether it needs an API key. Keyless sources pull live with no setup; the key-gated ones are noted (see §6 for where to enter keys).

#	Source	Provider / what it is	Key?
GLOBAL & MULTILATERAL			
1	World Bank	World Bank — WDI & 70+ databases, thousands of indicators	No
2	DBnomics	DBnomics aggregator — many providers in one	No
3	IMF	IMF DataMapper — WEO & cross-country macro	No
4	OECD	OECD SDMX — ~1,500 dataflows	No
5	EconDB	EconDB — global macro series	No
6	Trading Econ	Trading Economics — global indicators	Key
7	Our World In Data	OWID — ~4,400 charts across development topics	No
8	ILO	ILOSTAT — labour statistics	No
9	WB Health	World Bank health indicators	No
10	UNESCO	UNESCO Institute for Statistics — education	No
HEALTH, CLIMATE & DEVELOPMENT			
11	WHO Health	WHO Global Health Observatory	No
12	FAOSTAT	FAO — food & agriculture statistics	No
13	Climate Watch	Climate Watch — emissions data	No
14	UN SDG	UN Sustainable Development Goals indicators	No
15	Ember Energy	Ember — electricity & energy transition	No
UNITED STATES			
16	FRED	St. Louis Fed FRED — ~845k US economic series	Key
17	FRED Analytics	Derived analytics over FRED series	Key
18	BLS	US Bureau of Labor Statistics — jobs, CPI	Key
19	US Census	US Census Bureau — economic & demographic (browsing keyless; data keyed)	Key
20	Federal Reserve	Federal Reserve data	No
21	Fiscal Data	US Treasury Fiscal Data — debt, spending	No
22	BEA	US Bureau of Economic Analysis — NIPA, GDP	Key
EUROPE			
23	ECB	European Central Bank — SDW series	No
24	Eurostat	Eurostat — EU statistics	No
25	OeNB Austria	Austrian National Bank	No
26	ONS UK	UK Office for National Statistics	No

ASIA-PACIFIC			
27	AkShare China	AkShare — China macro & markets	**No**
28	ADB	Asian Development Bank	**No**
AMERICAS			
29	BCB Brazil	Banco Central do Brasil	**No**
30	StatCan	Statistics Canada — WDS API, 53 curated + any vector	**No**
CENTRAL BANKS			
31	Central Banks	Global central-bank policy rates & data	**No**
32	BIS	Bank for International Settlements	**No**
TRADE & ENERGY			
33	UN Comtrade	UN Comtrade — international trade flows	**No**
34	WTO	World Trade Organization statistics	**No**
35	EIA	US Energy Information Administration (petroleum keyless; forecasts keyed)	**Partial**
36	JODI Oil & Gas	Joint Organisations Data Initiative — oil & gas	**No**
MARKETS & ALT-DATA			
37	Econ Calendar	Economic calendar — releases by date & impact	**No**
38	CFTC	CFTC — Commitments of Traders	**No**
39	Alt-Data	Alternative-data indicators	**No**
40	Nasdaq Data	Nasdaq Data Link (Quandl)	**Key**
RESEARCH & REFERENCE			
41	Research Papers	Economics research paper search	**No**
42	WID Inequality	World Inequality Database	**No**
43	Penn World Table	Penn World Table — comparative national accounts	**No**
44	LEI Registry	Legal Entity Identifier lookup	**No**
45	Fincept Macro	Fincept macro — *coming soon* (placeholder)	**—**
46	V-Dem	Varieties of Democracy indices	**No**
47	SIPRI Milex	SIPRI military-expenditure database	**No**
48	AidData	AidData — development finance flows	**No**

06 Providers, API keys & the fetch engine

KEYS · ENGINE

The great majority of the 48 sources are **keyless** — they call the provider's public API directly and need no setup. A small number of US government / commercial APIs require a free key.

Sources that need a key

Source	Free key from
FRED & FRED Analytics	fred.stlouisfed.org/docs/api/api_key.html
BEA	apps.bea.gov/API/signup
BLS	US Bureau of Labor Statistics registration
US Census	US Census API key request
EIA (forecasts only)	US Energy Information Administration
Nasdaq Data Link	data.nasdaq.com account
Trading Economics	Trading Economics subscription

Enter keys once under **Settings → API Credentials**. If a key is missing or wrong, the panel says so plainly — e.g. *"BEA needs an API key. Enter it in Settings → API Credentials."* — rather than failing silently. Two of these are partial: **US Census** lets you browse its 1,790 datasets without a key but needs one to pull the data, and **EIA** serves weekly petroleum data keyless while its short-term forecasts need a key.

How a fetch works

Pressing **FETCH** runs the source's data script locally (bundled Python) and streams the result back into the panel — it does **not** route through the Fincept servers, and it does **not** spend credits. Results are cached for about 10 minutes, so re-fetching the same selection is instant; changing any parameter forces a fresh pull.

> **WHERE THE DATA COMES FROM**
>
> Each source talks directly to its provider (World Bank, IMF, Statistics Canada, FRED, …). Fincept is the unified front-end and chart engine, not a data reseller — so figures are the providers' own, as fresh as their APIs serve them.

> **ONE PLACEHOLDER**
>
> **Fincept Macro** is a "coming soon" stub — its FETCH shows a placeholder message. Every other source in the catalog is live.

07 Quick reference & honest notes

REFERENCE

How do I…

Task	Do this
Find a source fast	Type its name in **Search sources…** at the top of the rail.
Chart an indicator	Pick source → pick series → **FETCH** (charts automatically).
See the raw numbers	Toggle **TABLE**; page through with the bottom-right pager.
Read an exact value	Hover the chart for the crosshair tooltip.
Pull a series not in the list	Use the source's ID box (e.g. StatCan **VECTOR ID**, FRED **SERIES ID**).
Compare a country	World Bank / OECD: pick the **COUNTRY** first, then the indicator.
Export data	**CSV** — saves the full series at full precision.
Enter an API key	**Settings** → **API Credentials** (only 7 sources need one).

Honest notes & gotchas

- **Most sources are keyless** — 41 of 48 work with no setup; only 7 US-gov / commercial APIs need a free key.
- **Non-series data is table-only.** Calendars, cross-sections and entity lookups have no date series, so the CHART toggle is disabled and stat cards read "—".
- **POINTS counts rows, not just numeric points.** A source with non-numeric rows can show POINTS > 0 while LATEST/MIN/MAX stay blank.
- **Results cache for ~10 minutes.** Re-fetching identical parameters returns instantly and may be slightly stale; change a parameter to force fresh data.
- **"★ Popular" quick lists are instant.** World Bank / BEA lead with a curated offline quick-pick; other databases load their full catalog live over the network.
- **Fincept Macro is a placeholder** ("coming soon"); everything else is live.
- **Column order may shift.** A recognized date column is moved to the front and the value column next, so on-screen order can differ from the provider's raw order.
- **No credits are spent here.** Fetches run locally against public provider APIs.

> FINCEPT RESEARCH DESK
>
> Prepared for institutional clients · Professional Edition v5.0 · Captured live on the Economics tab. Control labels quoted verbatim; data fetched live from Statistics Canada and the World Bank.

FINCEPT TERMINAL · PRODUCT GUIDE

Government Data

A complete, portal-by-portal walkthrough of the Government Data Explorer — one window onto eleven sovereign open-data portals.

`>_ 11 sovereign portals · free public APIs · no credits`

Covers · **11 sovereign open-data portals + the live fetch engine** · **US Treasury · US Congress · 8 CKAN / open-data catalogs**

Reached by the command bar **“govt”** / **“gov”** · Navigate ▸ Markets & Data ▸ GOVT Data

Demonstrated live on the Fincept Terminal GOVT Data tab.

Contents

10 SECTIONS

GETTING ORIENTED

— Overview — what the GOVT Data tab is 3

1 Anatomy — the explorer shell 4

THE DEDICATED PORTALS

2 US Treasury — securities, auctions & summary 5

3 US Congress — bills & legislative activity 6

4 France — data.gouv.fr 7

5 Hong Kong — data.gov.hk 8

6 Australia — data.gov.au 9

7 The open-data (CKAN) panel — six portals in one 10

REFERENCE

8 How it works — the live fetch engine 13

9 The eleven sovereign portals — catalog 14

10 Quick reference & honest caveats 15

About this guide. Every screenshot is a real, live capture of the Government Data Explorer running in the Fincept Terminal — pulling actual data from public government portals. Control names, tab labels, table columns and dropdown options are transcribed from the shipping code so that what you read here matches what you click.

What the GOVT Data tab is

OVERVIEW

The Government Data Explorer is a single window onto **eleven sovereign open-data portals** — from the U.S. Treasury and Congress to the national open-data catalogs of Canada, the UK, France, Switzerland, Austria, Hong Kong and Australia. Pick a portal from the left sidebar and its dedicated panel lets you browse, search and export live public-sector data.

Reached from the command bar (type `govt` or `gov`) or **Navigate ▸ Markets & Data ▸ GOVT Data**, the screen is **not plan-gated**. Critically, **every portal fetches live data from a free public government API** — the terminal runs a small bundled script that calls the source directly. Nothing here is metered by Fincept credits and nothing routes through the AI gateway; the only optional keys are free registrations for higher rate limits. Results are cached briefly so revisits are instant.

AT A GLANCE, IN GOVT DATA YOU CAN…

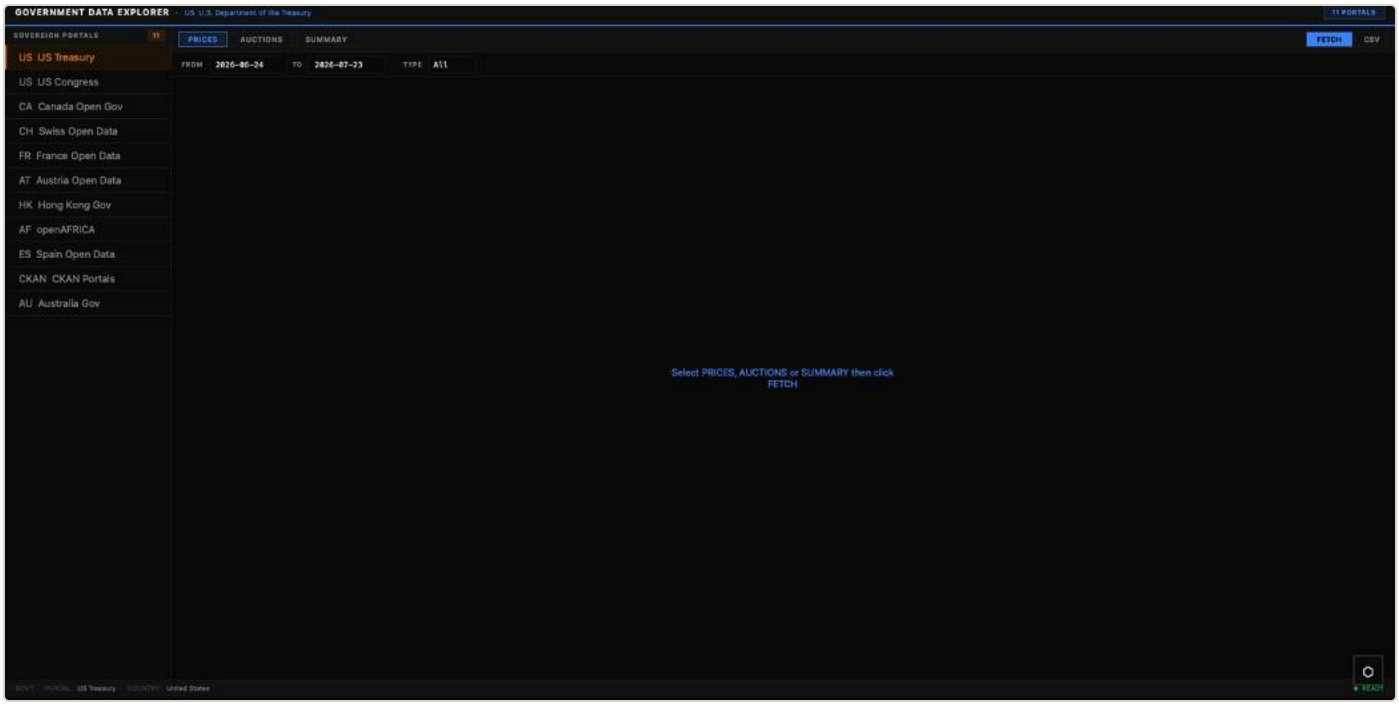
- Browse **11 sovereign portals** from one sidebar.
- Pull **US Treasury** securities, auctions & a market summary.
- Query **US Congress** bills and resolutions.
- Search national catalogs for **Canada, UK, France, Switzerland, Austria, Australia & Hong Kong**.
- Drill **Publishers → Datasets → Resources** and open files.
- Look up French **municipalities** by name.
- Export any table to **CSV**.
- Do it all on **free public APIs — no credits spent**.

> **Free by design.** Unlike the AI-driven tabs, GOVT Data never spends credits. It is a read-only window onto public government data, fetched locally and cached for a few minutes.

1 Anatomy — the shell

THE SHELL

The explorer has three fixed regions: a top toolbar that names the screen and the active portal, a 230px left sidebar listing all eleven portals, and a foot status bar. Selecting a portal swaps the main panel, recolors the accent to that country's brand color, and triggers its data load.



The landing view — the SOVEREIGN PORTALS sidebar (11 entries), the US Treasury panel, and the status bar. The toolbar reads GOVERNMENT DATA EXPLORER with an 11 PORTALS badge.

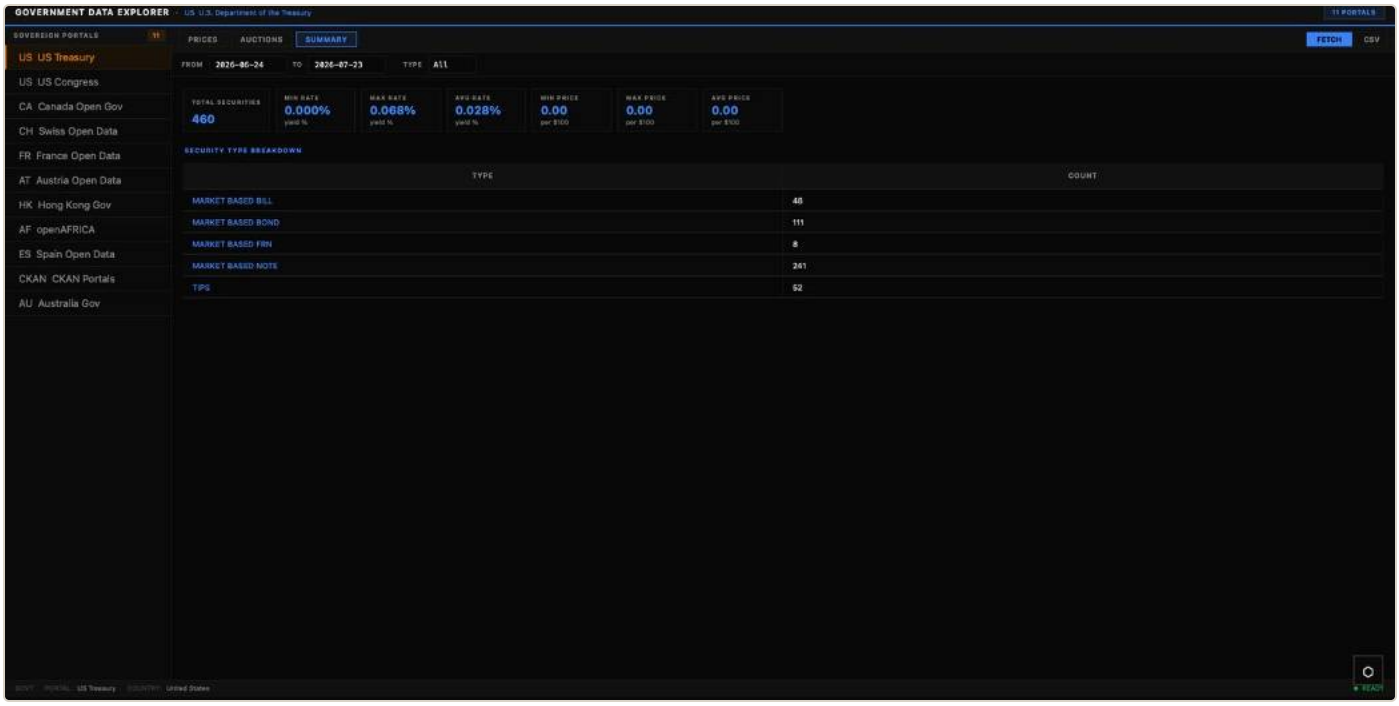
Region	What it does
Toolbar	**GOVERNMENT DATA EXPLORER** title · a subtitle showing the active portal (flag + full name, e.g. *U.S. Department of the Treasury*) · an **11 PORTALS** badge. The toolbar's bottom border takes on the active portal's accent color.
Sidebar	**SOVEREIGN PORTALS** header with an **11** count, then one row per portal — a two-letter code + name (`US US Treasury` , `CA Canada Open Gov` , …). Tooltip shows the description + country; the selected row gets a colored left border.
Panel	One dedicated panel per portal (or a shared open-data panel). Switching portals recolors the accent and fires the panel's data load.
Status bar	**GOVT** · **PORTAL:** <name> · **COUNTRY:** <country> · a green **READY** dot.

> **Two load behaviours.** The **open-data (CKAN)** portals auto-load their publisher list the moment you select them. The **US Treasury** and **US Congress** panels instead show a prompt — *"Select … then click FETCH"* — and load only when you press **FETCH**. The screen remembers the last portal you viewed and restores it next session.

2 US Treasury

PORTAL · US TREASURY

The Treasury panel pulls live U.S. Treasury securities data from **TreasuryDirect.gov** (no key required). Three endpoint tabs — **PRICES**, **AUCTIONS**, **SUMMARY** — share a date-range and security-type filter. Here the SUMMARY view aggregates the whole outstanding universe.



US Treasury → SUMMARY: total-securities and yield/price stat cards, plus a SECURITY TYPE BREAKDOWN (bills, bonds, FRNs, notes, TIPS).

Control	Type	What it does
PRICES / AUCTIONS / SUMMARY	tabs	Choose the endpoint. PRICES = end-of-day secondary-market prices; AUCTIONS = historical auction results; SUMMARY = computed counts + yield/price stats.
FROM / TO	date	Date range (`yyyy-MM-dd`). TO defaults to the last business day; FROM to a month earlier (used by AUCTIONS).
TYPE	dropdown	`All · Bills · Notes · Bonds · TIPS · FRN` .
FETCH · CSV	buttons	Run the fetch for the active tab; export the active table to `treasury_data.csv` .

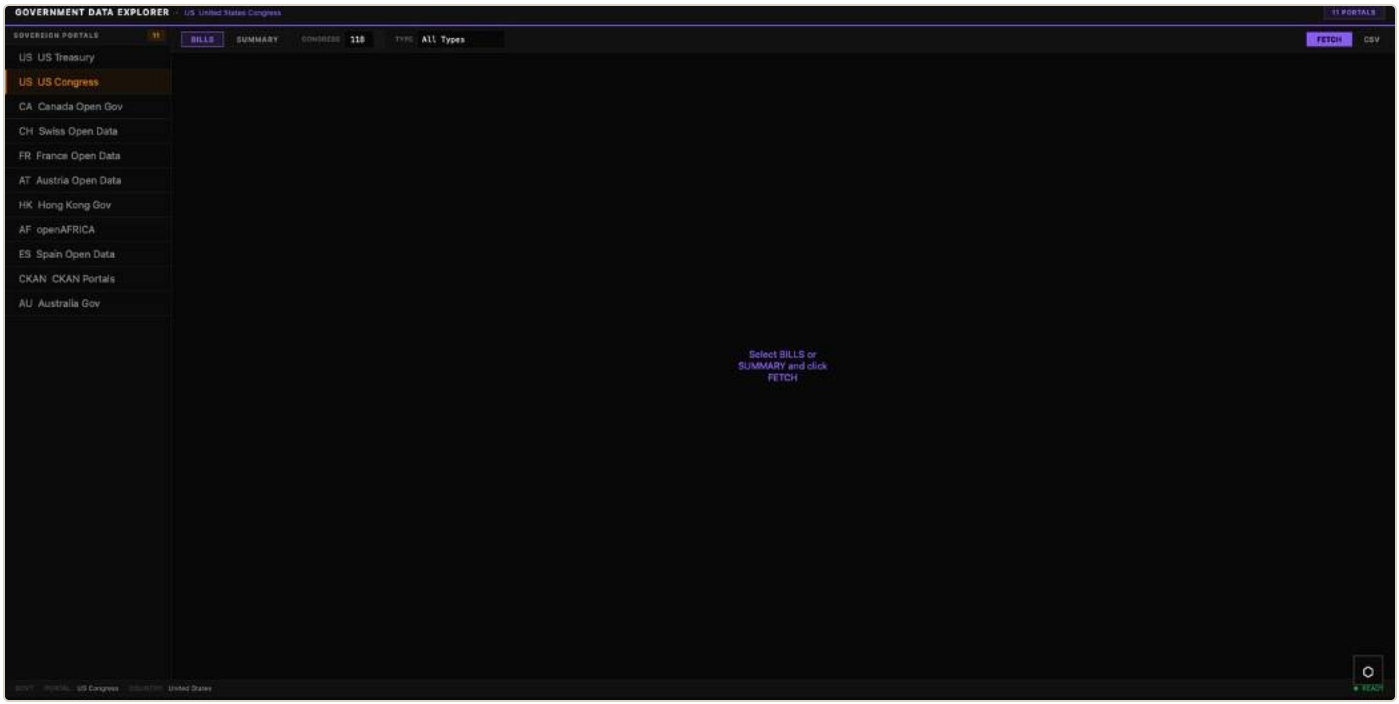
Tables. PRICES → `CUSIP · TYPE · RATE % · MATURITY · BID · OFFER · EOD PRICE` . AUCTIONS → `CUSIP · TYPE · TERM · AUCTION DATE · HIGH RATE · HIGH PRICE · BID/COVER · OFFERING` . SUMMARY → seven stat cards (*Total Securities*, min/max/avg *Rate* and *Price*) over a **SECURITY TYPE BREAKDOWN** table.

> **Honest notes.** The **SUMMARY** view is the reliable one — the PRICES and AUCTIONS tables read a JSON field that the current fetch script does not populate, so they render **empty even on a successful fetch** (a known data-binding gap). The FROM date is ignored for PRICES/SUMMARY; auction results are fixed to the first 50. Data is US-only, and prices are unavailable on weekends/holidays (the panel says so).

3 US Congress

PORTAL · US CONGRESS

The Congress panel lists federal **bills and resolutions** for a chosen Congress from the official **Congress.gov** API, and drills into a single bill's sponsors, summaries and actions.



US Congress — BILLS / SUMMARY tabs, a CONGRESS number spinner (default 118) and a bill-type filter, with the “Select … then FETCH” prompt.

Control	Type	What it does
BILLS / SUMMARY	tabs	Browse a bills list, or a per-type count summary.
CONGRESS	spinner	Congress number, 1–200 (default **118**).
TYPE	dropdown	Nine options: `All Types`, House/Senate Bill, and H./S. Joint, Concurrent and Simple resolutions.
FETCH · CSV · ← BACK	buttons	Fetch bills/summary; export to `congress_data.csv`; return from a bill's detail page.

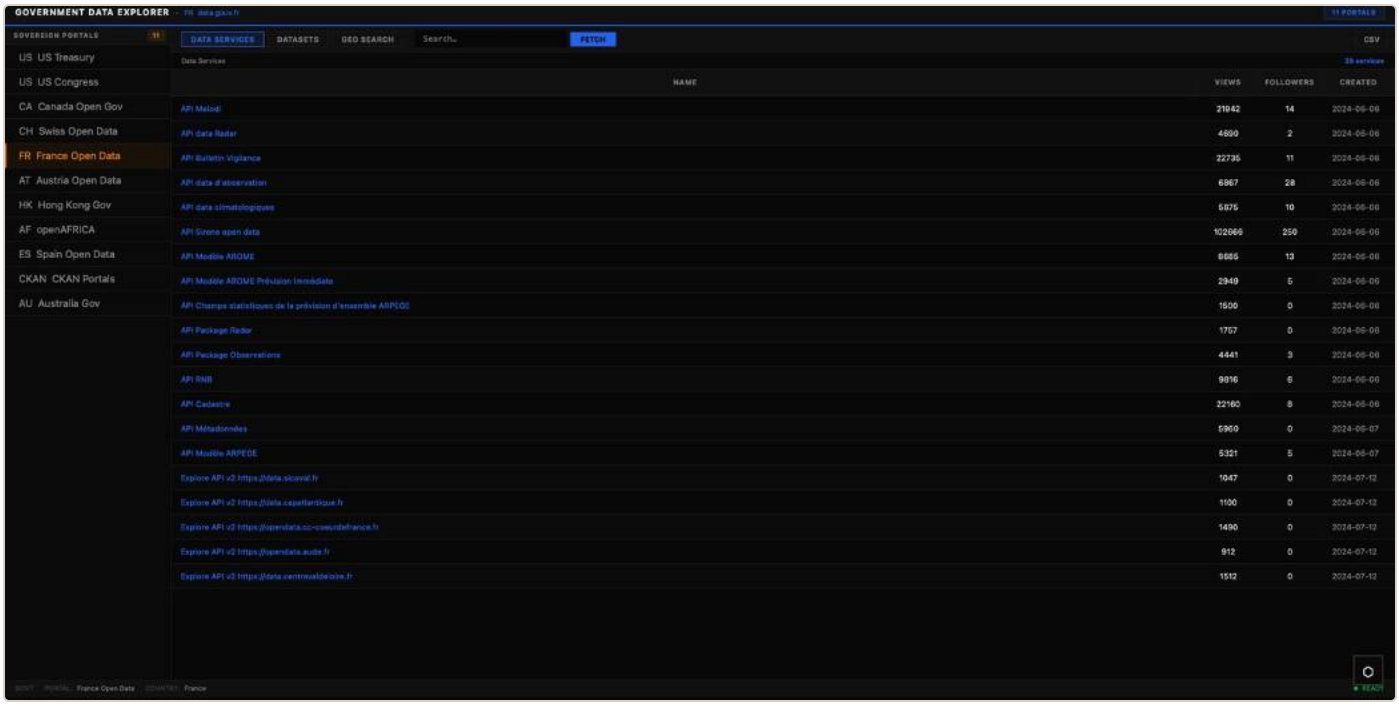
Tables. BILLS → `CONGRESS · TYPE · NUMBER · TITLE · LATEST ACTION · DATE`; double-click a row to open a full **detail** page (sponsors, cosponsors, summaries, actions). SUMMARY → `BILL TYPE · COUNT`.

> **Honest note — needs a free key.** Congress.gov requires a free API key (`CONGRESS_GOV_API_KEY`). The terminal cannot currently pass this key through to the fetch script, so **FETCH returns “API key not configured”** until that wiring is added. Every other portal works key-free. Bills cap at ~100 rows and the summary counts are a 50-row sample per type.

4 France — data.gouv.fr

PORTAL · FRANCE

The France panel wraps three **data.gouv.fr** APIs behind three tabs: published **DATA SERVICES**, a **DATASETS** search, and a **GEO SEARCH** that looks up French municipalities. Titles are auto-translated French → English, with the original kept in tooltips.



France → DATA SERVICES: published API/data-services from data.gouv.fr with view and follower counts (API Sirene, API Cadastre, meteorological APIs…).

Tab	What it shows
DATA SERVICES	Published data-services — `NAME · VIEWS · FOLLOWERS · CREATED` (loads automatically).
DATASETS	Free-text dataset search — `TITLE · ORG · LICENSE · FILES · MODIFIED`; double-click a row to see its column schema.
GEO SEARCH	Municipality lookup by name — `NAME · CODE (INSEE) · POSTAL · DEPT · REGION · POPULATION`.

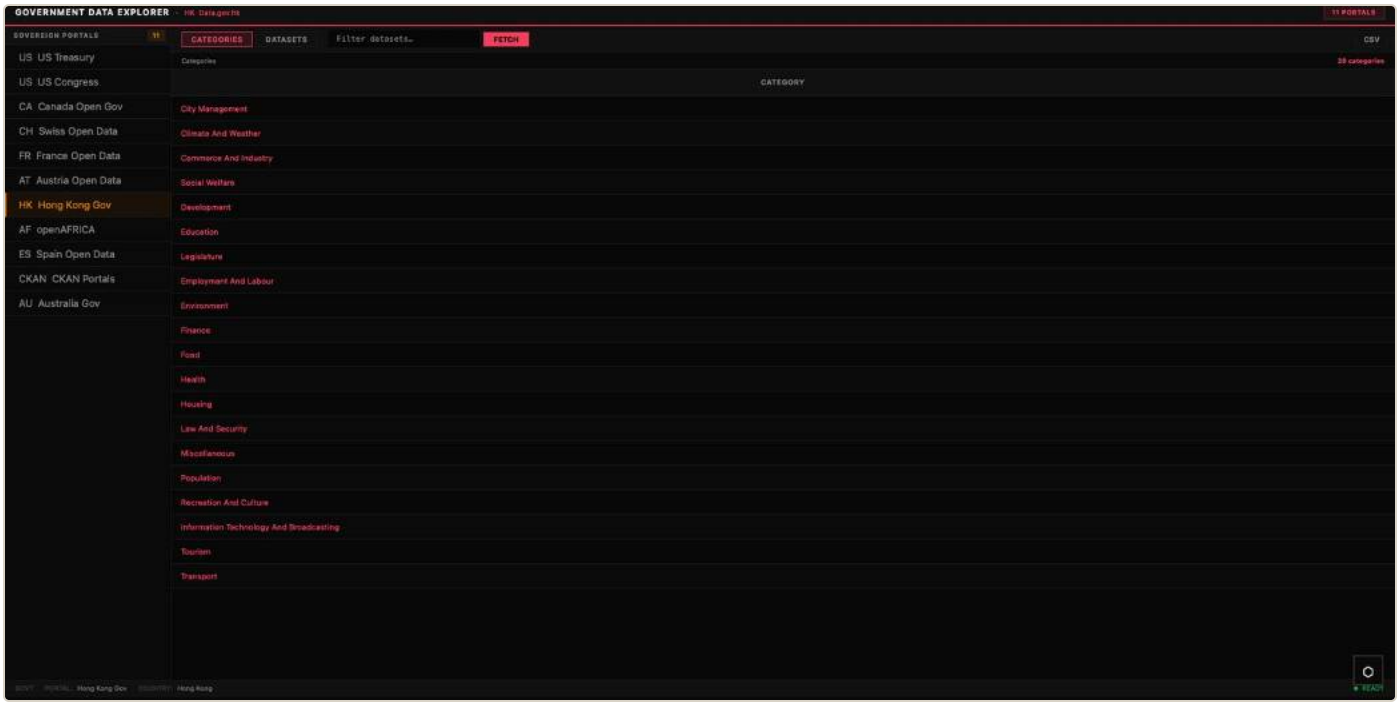
A shared **Search** box (its placeholder changes per tab), **FETCH** and **CSV** sit in the toolbar.

> **Honest notes.** DATA SERVICES and GEO SEARCH work well; the **DATASETS** search often returns no rows (it requests the wrong result page) and the column-schema drill-down typically 404s (it passes a dataset id where a resource id is expected). The script's company-registry (SIREN/SIRET) feature isn't wired to any control here.

5 Hong Kong — data.gov.hk

PORTAL · HONG KONG

The Hong Kong panel browses the **Data.gov.hk** CKAN catalog by **Categories → Datasets → Resources**. Because the HK portal exposes no search endpoint, the search box filters dataset names client-side.



Hong Kong → CATEGORIES: the 20 CKAN groups (City Management, Climate & Weather, Finance, Health, Transport…). Double-click a category to load its datasets.

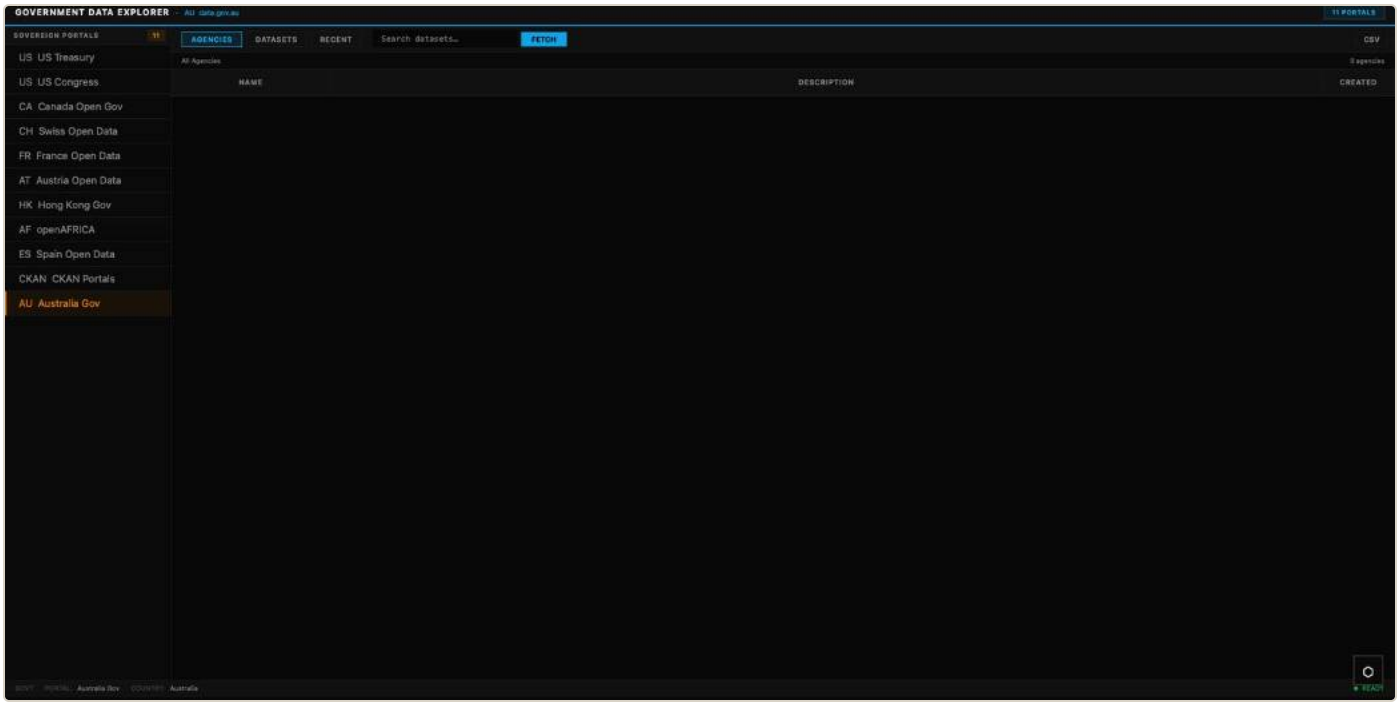
Control	What it does
CATEGORIES / DATASETS	Tabs. CATEGORIES lists the 20 CKAN groups; double-click one to load its datasets (`TITLE · RESOURCES · MODIFIED`).
Filter datasets…	A client-side substring filter over dataset names (there is no server search).
Resources drill	Double-click a dataset for its files — `NAME · FORMAT · MODIFIED` , with a `↗ OPEN` link that opens the file in your browser.
← BACK · CSV	Walk back up the chain; export the current table (`hk_categories.csv` etc.).

> **Honest note.** A category often returns no datasets (the HK portal's group listing is sparse), so the panel suggests searching by name instead. The name filter matches dataset *names only*, not titles or tags.

6 Australia — data.gov.au

PORTAL · AUSTRALIA

The Australia panel browses **data.gov.au** by **Agencies → Datasets → Resources**, plus a global dataset search and a **RECENT** shortcut. Resources are embedded in each dataset, so opening a dataset needs no extra fetch.



Australia → AGENCIES / DATASETS / RECENT. Agencies list as `NAME · DESCRIPTION · CREATED`; double-click drills to datasets and their downloadable resources.

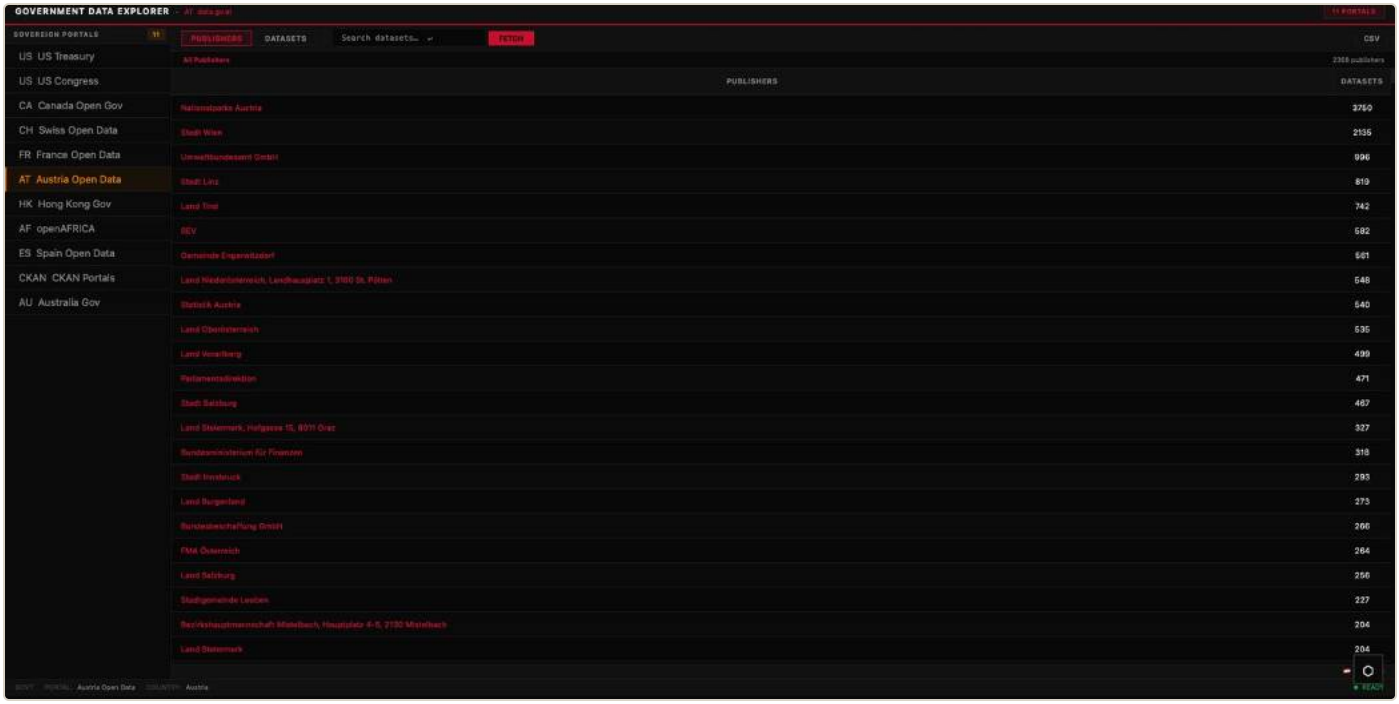
Control	What it does
AGENCIES	Publishing agencies — `NAME · DESCRIPTION · CREATED`. Double-click → that agency's datasets.
DATASETS	Per-agency or search results — `TITLE · LICENSE · AUTHOR · RESOURCES · MODIFIED`. Double-click → resources.
RECENT	The 20 most recent datasets portal-wide, no agency needed.
Resources	`NAME · FORMAT · SIZE · OPEN`; the `↗ OPEN` cell launches the file in your browser.
Search · FETCH · CSV	Search all datasets, run the active view, or export it.

Honest notes. Row counts are capped (100 agency datasets, 50 search results, 20 recent) with no pagination, and the breadcrumb is a static label (navigate with the tabs / BACK). Because this is a **live** call to data.gov.au, the agency list can occasionally be slow to load or return empty — retry, or use RECENT / Search.

7 The open-data (CKAN) panel

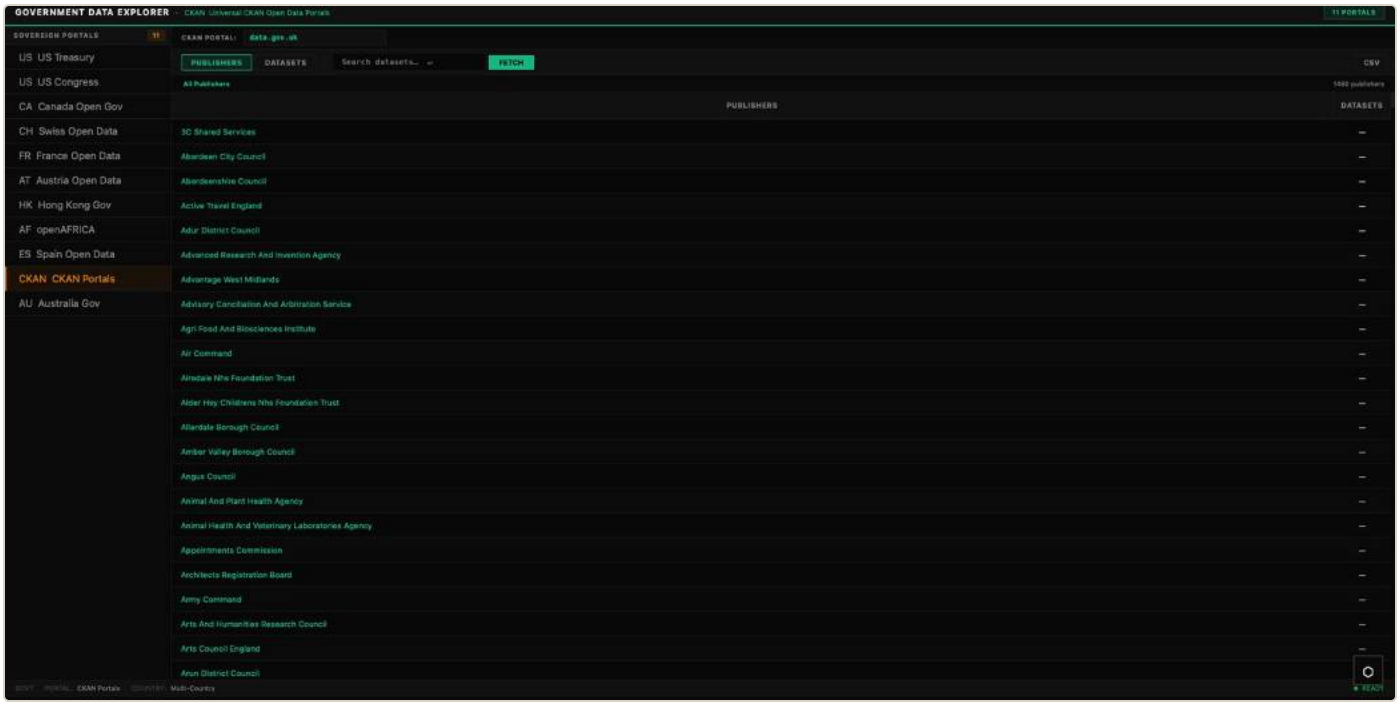
SHARED · SIX PORTALS

Six of the eleven portals — **Canada, Switzerland, Austria, openAFRICA, Spain** and the **Universal-CKAN** aggregator — share one reusable browser. It walks **Publishers → Datasets → Resources**, with a search box, clickable breadcrumb, and CSV export. Only the accent color, the source watermark, and a top portal-selector differ between them.



The shared panel on Austria (data.gv.at): 2,368 publishers with dataset counts, a data.gv.at watermark bottom-right, and the PUBLISHERS / DATASETS tabs + search.

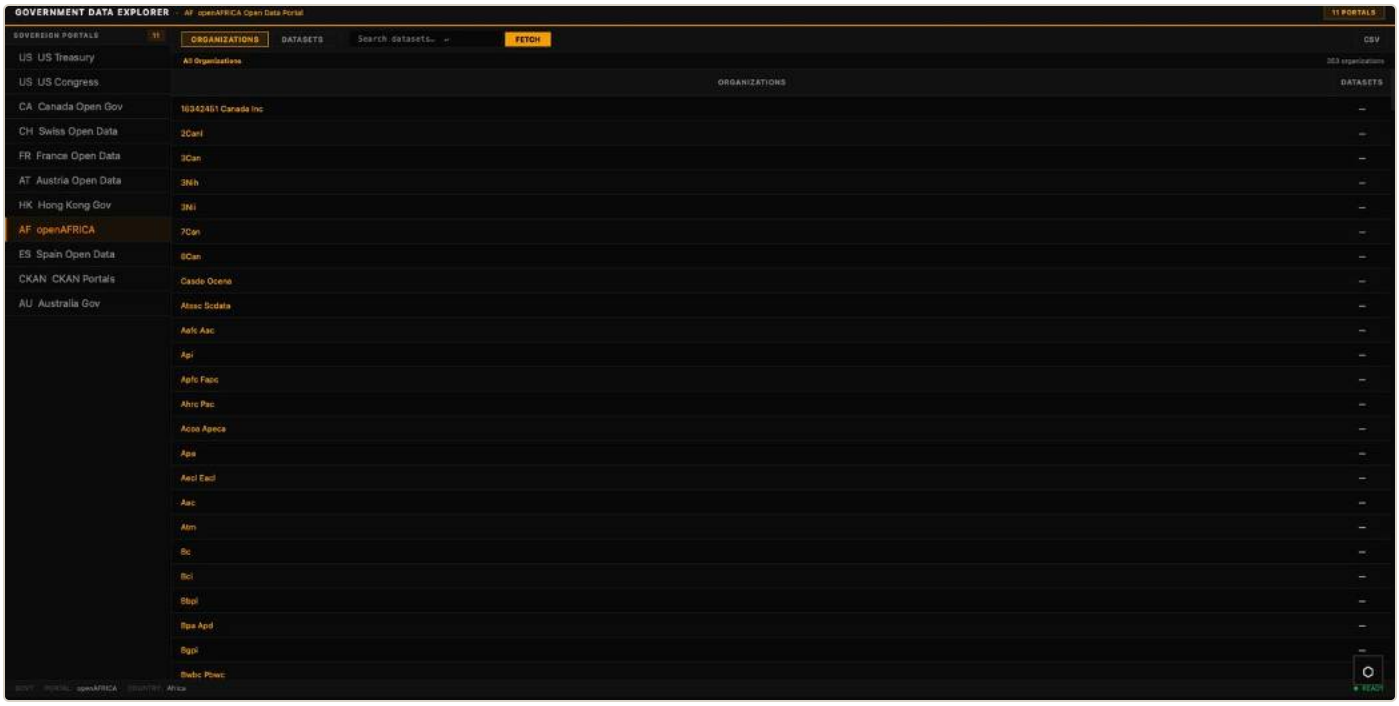
Control	What it does
PUBLISHERS / ORGANIZATIONS	The portal's data-publishing bodies + their dataset counts. Double-click → that body's datasets.
DATASETS	`TITLE · FILES · MODIFIED · TAGS`. Double-click → resources.
Resources	`NAME · FORMAT · SIZE · MODIFIED · OPEN`; the `↗ OPEN` button opens each file in the browser.
Search · breadcrumb · CSV	Portal-wide dataset search; a clickable breadcrumb to jump back up; CSV export with an inline `✓ SAVED` confirmation.



The Universal-CKAN portal adds a **CKAN PORTAL** selector bar and covers UK data.gov.uk (1,480 publishers, green accent, COUNTRY: Multi-Country).

> **Watermarks & the portal selector.** Switzerland and Austria show a source watermark (*opendata.swiss*, *data.gv.at*). The Universal-CKAN panel adds a portal-selector combo listing eight CKAN portals — but it is **presentational**: the panel always queries **data.gov.uk** (its own tooltip says so).

> **Important caveat — openAFRICA & Spain show Canadian data.** These two portals currently reuse the Canada fetch script, so selecting *openAFRICA* or *Spain Open Data* actually returns the **Canadian** federal catalog — the panel below shows the identical publisher list to Canada, only relabelled "Organizations" with a different accent. Treat these two as not-yet-wired to their true sources.



Evidence of the caveat: openAFRICA (COUNTRY: Africa) lists the same publishers as Canada Open Gov — the source script is shared and not yet pointed at African data.

8 How it works — the fetch engine

ENGINE

Every portal works the same way under the hood: the terminal runs a small bundled Python script that calls the source government's **public open-data API** directly, parses the JSON, and renders it in the panel. There is no Fincept server in the loop and no AI model — so nothing here spends credits.

8.1 Data, keys & cost

Aspect	Detail
Source	The official public API of each portal — TreasuryDirect, Congress.gov, and the national CKAN / Piveau / geo APIs. Fetched locally via the embedded Python interpreter.
Cost	**None.** No Fincept credits, no AI gateway. Free public data.
Keys	All portals work key-free **except US Congress**, which needs a free `CONGRESS_GOV_API_KEY` (see §3). A couple of optional sub-features (France's company registry) also take a free key.
Caching	Results cache for **~5 minutes**, so switching back to a portal is instant. Fetches are paced to at most two per second.
Export	Every panel's **CSV** button exports exactly the rows currently on screen.
Freshness	Because it is live, availability depends on the source portal being up; a slow or empty response is the source, not the terminal.

> **Where the data goes.** Fetched data lives only in the panel and the short-lived cache — it is not written to your account or synced anywhere. Use CSV or the ↗ OPEN resource links to take data with you.

9 The eleven sovereign portals

CATALOG

The full portal roster, in sidebar order, with the source each one queries and the panel that renders it.

Portal	Source	Country	Panel & notes
US Treasury	TreasuryDirect.gov	United States	Dedicated — prices / auctions / summary.
US Congress	Congress.gov API	United States	Dedicated — bills & resolutions (needs free key).
Canada Open Gov	open.canada.ca	Canada	Shared CKAN panel (Publishers).
Swiss Open Data	opendata.swiss	Switzerland	Shared CKAN panel + watermark.
France Open Data	data.gouv.fr	France	Dedicated — services / datasets / geo.
Austria Open Data	data.gv.at	Austria	Shared CKAN panel + watermark.
Hong Kong Gov	data.gov.hk	Hong Kong	Dedicated — categories / datasets.
openAFRICA	open.canada.ca *(shared)*	Africa	Shared panel — currently shows Canadian data.
Spain Open Data	open.canada.ca *(shared)*	Spain	Shared panel — currently shows Canadian data.
CKAN Portals	data.gov.uk	Multi-Country	Shared panel + portal selector (queries UK).
Australia Gov	data.gov.au	Australia	Dedicated — agencies / datasets / recent.

Five dedicated panels, six on the shared browser. US Treasury, US Congress, France, Hong Kong and Australia each have a purpose-built panel; Canada, Switzerland, Austria, openAFRICA, Spain and Universal-CKAN share the one CKAN browser (§7).

10 Quick reference & caveats

REFERENCE

10.1 How do I…

I want to…	Do this
Open the explorer	Command bar → `govt` (or `gov`), or Navigate ▸ Markets & Data ▸ GOVT Data.
See a Treasury market summary	US Treasury → **SUMMARY** → **FETCH**.
Browse a country's datasets	Pick the portal → double-click a Publisher/Agency/Category → double-click a dataset.
Download a data file	Drill to Resources → click the `↗ OPEN` link (opens in your browser).
Look up a French town	France → **GEO SEARCH** → type the name → FETCH.
Get the 20 latest AU datasets	Australia → **RECENT**.
Save a table	Click **CSV** on any panel.
Fetch US Congress bills	Set a free `CONGRESS_GOV_API_KEY` first (see caveat), then BILLS → FETCH.

10.2 Honest caveats worth knowing

- **No credits, ever.** Every portal is a free public API fetched locally — nothing here bills against your credits.
- **US Treasury:** only **SUMMARY** populates reliably; PRICES/AUCTIONS render empty on success (a data-binding gap).
- **US Congress:** needs a free key the terminal can't yet pass through, so FETCH errors until that's wired.
- **openAFRICA & Spain** currently return the **Canadian** catalog — their native sources aren't wired yet.
- **Universal-CKAN's** portal selector is cosmetic; it always queries **data.gov.uk**.
- **France:** DATASETS search and the schema drill-down are unreliable; DATA SERVICES and GEO SEARCH are solid.
- **Row caps** apply (typically 50–100 rows) with no pagination; refine with search instead.
- **Live means live:** a slow or empty portal is the source being unavailable, not a terminal error — retry.

> **Fincept Research Desk.** This guide documents the Government Data Explorer (GOVT Data) tab as shipped in Professional Edition v5.0. Screenshots are live captures against public government portals; control names and options are transcribed from the shipping build.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The DBnomics Terminal

A complete, screen-by-screen walkthrough of the DBnomics tab — a browser for 93 statistical providers, from picking a series out of a catalogue to charting several side by side and exporting the numbers.

`>_ 93 providers · 4 chart types · single + compare · keyless, free`

Covers · **the selection panel** · **charting a series** · **single vs compare** comparison slots · CSV export · where the data comes from · the 5 agent tools

Demonstrated live on the Fincept Terminal DBnomics tab.

Contents

11 SECTIONS

GETTING ORIENTED

01 What the DBnomics tab is 3

02 Anatomy of the tab 4

FINDING A SERIES

03 The selection panel 5

04 Global search 6

CHARTING IT

05 Reading the chart and the table 7

06 SINGLE view — several series, one chart 8

07 COMPARE view — the slot system 9

08 Exporting to CSV 10

REFERENCE

09 Where the data comes from 11

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 DBnomics tab in the shipping build, against the live DBnomics API — including the four-slot comparison, which was already loaded on the desk this guide was written at.

01 What the DBnomics tab is

ORIENTATION · 01

DBnomics is an **aggregator of official statistics**: national statistics offices, central banks, the IMF, the OECD, the ILO, Eurostat and dozens more, all reachable through one catalogue. This tab is the terminal's browser for it — drill from a provider to a dataset to a series, chart it, stack several series together, and export the numbers.

The path is always the same four steps: **provider → dataset → series → observations**. Once a series is loaded you can chart it four ways, add more series to the same chart, or send series into separate comparison panels.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Markets & Data ▸ DBnomics**
Command bar	Type `dbnomics`, then pick the row

At a glance, in the DBnomics tab you can…

- Browse 93 statistical providers
- Filter providers by name or code
- Page through a provider's datasets
- Search the series inside one dataset
- Search datasets across every provider at once
- Chart a series as line, area, bar or scatter
- Overlay several series on one chart
- Put series into separate comparison panels
- Read exact values in an observation table
- Export what you charted as a CSV

> **Free, keyless and unmetered.** The tab talks straight to the public DBnomics API — no API key to configure, no Fincept credits, and nothing routed through your account. Section 09 covers the transport, the request rate and what that means in practice.

02 Anatomy of the tab

A toolbar across the top, the selection panel down the left, and the view area filling the rest.



Fig 1 — The DBnomics tab in SINGLE view with one series loaded: chart above, observation table below, selection panel on the left.

#	Region	What it is for
1	Toolbar	**DBNOMICS TERMINAL** and three buttons — **FETCH** (load the provider list), **REFRESH**, **EXPORT CSV** (section 08).
2	Selection panel	The whole drill-down: global search, providers, datasets, series, the action buttons, and the comparison slots. Sections 03–04, 06–07.
3	View toggle	**SINGLE** / **COMPARE**, and the **CHART** type selector for the single view.
4	Chart	A QtCharts plot of every series in the current view, with a legend naming each one.
5	OBSERVATION DATA	The numbers behind the chart: one **PERIOD** column plus one column per series.
6	Status line	What just happened — what loaded, what was added to which slot, and any error.



Fig 2 — The toolbar. FETCH is the button to press if the provider list is ever empty.

03 The selection panel

Everything on the left is one workflow, top to bottom. Each list loads the next one.

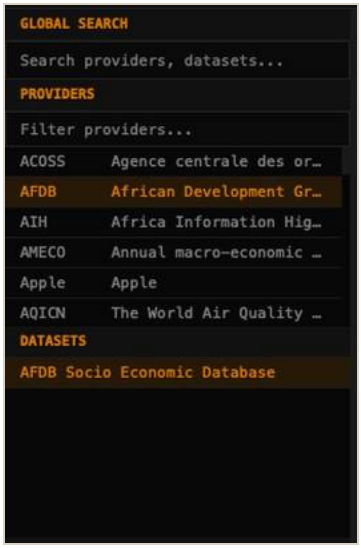


Fig 3 — Search, providers and datasets.

Section	What it does
GLOBAL SEARCH	Searches datasets across **every** provider at once. Section 04.
PROVIDERS	All 93 providers, with a *Filter providers…* box that narrows the list as you type. Click one to load its datasets.
DATASETS	The selected provider's datasets, 50 at a time, with **LOAD MORE** beneath. Click one to load its series.
SERIES	The series inside that dataset, with their own search box and **LOAD MORE**. Click one to fetch its observations and draw it.

Each list shows a small animated indicator while it loads, and the status line at the foot of the window says what is happening.

Control	What it does
ADD TO SINGLE VIEW	Adds the series you just loaded to the single chart, keeping what is already there. Section 06.
CLEAR ALL	Empties the single chart **and** every comparison slot. There is no undo.
+ ADD SLOT	Creates another comparison panel. Section 07.
SLOT *n*	One row per panel: **+ ADD CURRENT SERIES** to put the loaded series into it, a coloured chip per series it holds, and a red button to remove the whole slot.



Fig 4 — The action buttons and the comparison slots.

> **Loading a series does not chart it permanently.** Clicking a series fetches its observations and shows them; the status line then says *Click ADD TO SINGLE VIEW*. Until you add it, selecting another series simply replaces what is on screen.

04 Global search

Typing into **GLOBAL SEARCH** queries the whole catalogue rather than one provider, which is the fastest way in when you know the subject but not who publishes it. Results are datasets, paged with **LOAD MORE**; clicking one jumps the panel to that provider and dataset so you can pick a series from it.



Fig 5 — Searching *unemployment rate*. The results are datasets from ILO and OECD — but see the caveat below.

> **Search rows show only the provider code in this build.** Each result should read *[PROVIDER] dataset name*, but the dataset name comes back empty from the search response, so the list reads `[ILO]`, `[OECD]`, `[OECD]` … with nothing to tell them apart. Two ways round it: **hover a row** — the tooltip carries the provider/dataset code and the number of series in it — or ignore search and use the **PROVIDERS** → **DATASETS** path, where names display normally.

> **Search is asynchronous and can take several seconds.** The section shows a *SEARCHING…* indicator while the request is in flight; on a broad term it may take five to ten seconds before the first page of results appears. The rest of the tab stays usable meanwhile.

Which route to use

Situation	Best route
You know the publisher (Eurostat, IMF, ILO…)	**PROVIDERS** — filter by code, then read dataset names normally
You know the subject only	**GLOBAL SEARCH**, then hover the rows to identify them
You are back for a series you used before	Providers → dataset → the **SERIES** search box, which searches inside that dataset

05 Reading the chart and the table

CHART · 05

Clicking a series fetches its observations and draws them immediately, with the numbers underneath.

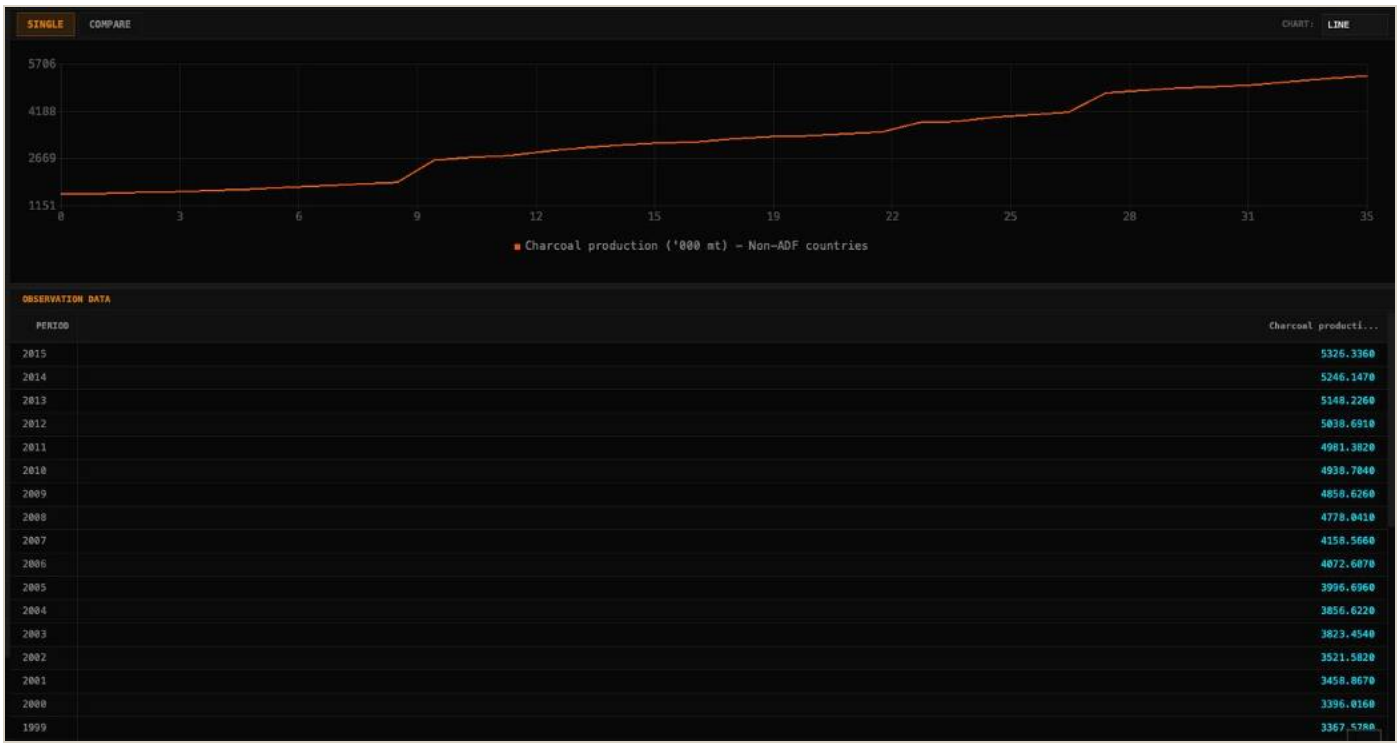


Fig 6 — One series charted, with its legend under the plot and the OBSERVATION DATA table below.

Element	Detail
`Chart type`	**LINE** (default), **AREA**, **BAR**, **SCATTER** — from the **CHART** selector above the plot
`Legend`	Under the plot, one entry per series in the chart's own colour
`Y axis`	Scaled to the data, labelled in compact notation (so 2.422e+04 rather than 24220.31)
`X axis`	**The observation index** — 0, 3, 7, 11… — not the date. Read periods from the table
`OBSERVATION DATA`	One **PERIOD** column, then one column per series, newest first
`Missing values`	Left blank in the table rather than shown as zero



Fig 7 — The four chart types.

> **The chart's x axis counts points, not time.** A 36-observation annual series is labelled 0 to 35, so the plot tells you the *shape* and the table tells you the *dates*. When a date matters — the year of a peak, say — take it from the OBSERVATION DATA rows rather than from the axis.

> **There is no hover read-out on the plot.** Pointing at a line shows nothing; the table is the precise instrument. The chart type selector's longest label is also clipped to *SCATT…* in the dropdown — the option works normally.

06 SINGLE view — several series, one chart

The single view overlays series on one plot, which is what you want when the series share a scale — the same indicator for four countries, or one country's series across four vintages.

How to build one

1. Pick a series in the panel; it loads and draws on its own.
2. Press **ADD TO SINGLE VIEW** — it is now part of the chart, with its own colour and legend entry.
3. Pick the next series and press **ADD TO SINGLE VIEW** again. Repeat.
4. Change the **CHART** type at any time; it applies to the whole plot.
5. **CLEAR ALL** when you want to start again — it empties the comparison slots too.

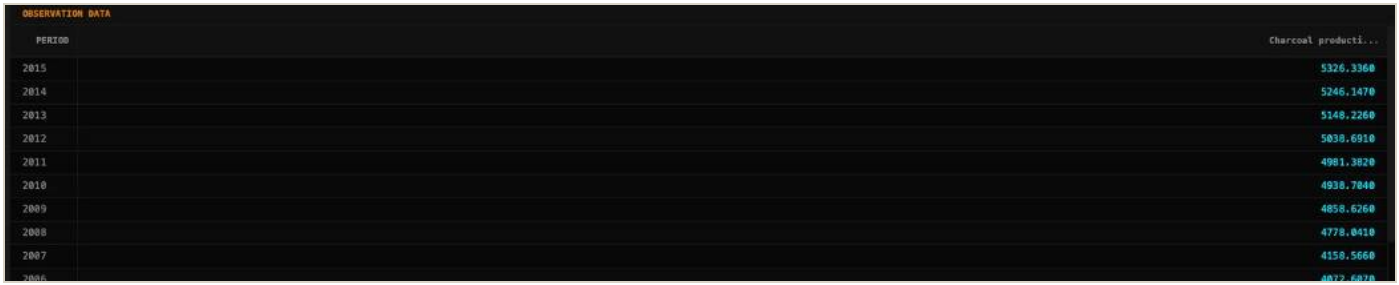


Fig 8 — The observation table in single view: one PERIOD column and one column per series, so the numbers line up period by period.

Behaviour	Detail
`Colours`	Assigned automatically in order, and reused in the legend, the chart and the CSV header
`Table shape`	One column per series added; periods are unioned, so a series that starts later simply has blank early cells
`Series limit`	None enforced — but the legend and the table get crowded past a handful, which is what COMPARE view is for
`Chart type`	One type for the whole single view; per-panel types are a COMPARE feature

> **Overlay only when the units agree.** Nothing stops you adding a percentage series and a billions-of-currency series to the same plot — the y axis will simply scale to the larger one and flatten the other into a line along the bottom. Mixed units belong in separate comparison slots (section 07).

07 COMPARE view — the slot system

COMPARE replaces the single plot with a grid of independent panels. Each slot has its own chart, its own observation table and its own chart type — so series with different units, or from different providers entirely, can sit side by side without fighting over one axis.

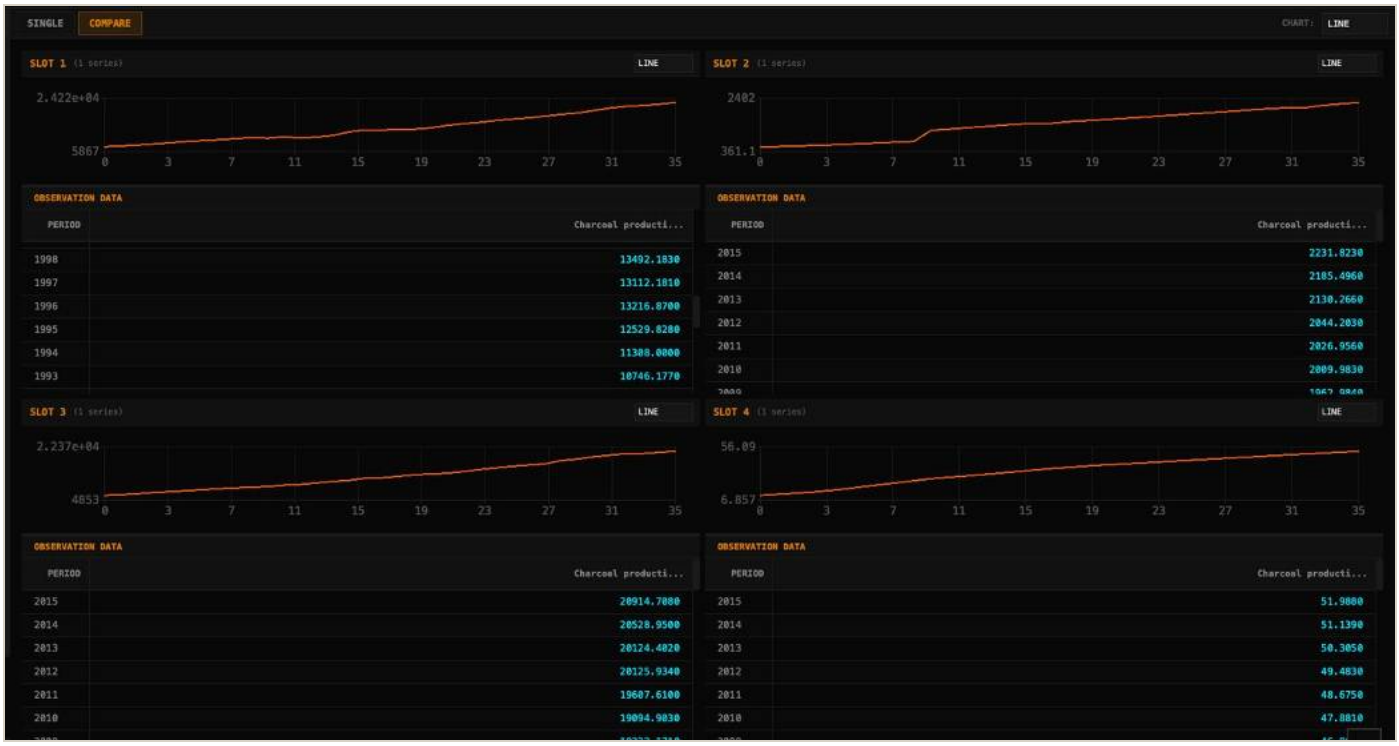


Fig 9 — Four slots, each holding one series with its own chart and table. Slot charts hide the legend to save space; the slot header names what it holds.

Working with slots

To do this	Do this
`Add a panel`	+ **ADD SLOT** in the panel's **COMPARISON SLOTS** section
`Put a series in a panel`	Load the series, then press + **ADD CURRENT SERIES** on that slot
`Change one panel's chart type`	Use the type selector in that slot's header
`Remove one series`	Press the small button on that series' chip inside the slot
`Remove a whole panel`	Press the red button on the slot header
`Empty everything`	**CLEAR ALL** — single view and every slot

> **Each slot's header counts what it holds** — *SLOT 1 (1 series)* — and the status line confirms every addition by name, so two similarly-named series stay distinguishable.

> **Slots are not remembered between sessions.** Your provider, dataset, series and view mode come back; a comparison you built does not. Export the numbers (section 08) if it matters.

08 Exporting to CSV

CSV EXPORT · 08

EXPORT CSV writes what is in the **single view** to a file you choose, in a wide layout that opens cleanly in a spreadsheet.

Aspect	Behaviour
What is exported	Every series in the **SINGLE** view — **not** the comparison slots
Layout	First column `Period`, then one column per series, headed by the series name
Rows	The union of every period across the exported series, sorted
Missing values	Written as empty cells, so a gap stays a gap rather than becoming a zero
File name	Offered as `dbnomics_YYYYMMDD_HHMMSS.csv`; you choose the folder
Nothing loaded	The status line says *No data to export* and no file is written

> **Two things to know before you rely on it.** First, the export covers the single view only — if you built a four-slot comparison, add those series to the single view (or export them one at a time) before pressing the button. Second, the file is written **only** where you point the save dialog: unlike the Excel and Notes exports, it is **not** registered in File Manager, so it will not appear in that tab's index.

> **A tidy workflow.** Build the comparison you want in the single view, export it, then open the CSV in the **Excel** tab (File Manager ▸ upload, or Excel ▸ IMPORT after re-saving as `.xlsx`) if you want to do arithmetic on it — or hand the CSV straight to a colleague, since the header row carries the full series names.

What the file looks like

```
Period,Charcoal production ('000 mt) – Non-ADF countries,…
1990,3367.5708,…
1991,3396.0168,…
```

09 Where the data comes from

Property	Value
Source	The public **DBnomics** REST API (version 22), queried directly from the terminal
Authentication	**None.** No API key to configure anywhere
Cost	**Free.** No Fincept credits are spent — this tab never routes through your account
Providers	93 in the list at the time of writing; the count is whatever the API returns, shown at the foot of the panel
Request rate	Deliberately capped at about three requests per second so a fast click-through cannot hammer a public service
Paging	Datasets arrive 50 at a time; series and search results page the same way, via **LOAD MORE**
Offline	Nothing works without a network connection — the lists simply stay empty and the status line reports the failure



Fig 10 — The foot of the window: the status line on the left reports the last action, and the provider count sits on the right.

> **If the provider list is empty, press FETCH.** That is the button's whole job: it reloads the provider list from the API. **REFRESH** re-runs the current selection, which is the one to use when a series has been revised since you loaded it.

> **What the terminal keeps.** Your provider, dataset and series selection, the three search boxes and the current view mode are restored next time you open the tab. Observations themselves are re-fetched, so the numbers you see are always current rather than a cached snapshot.

> **A public service, used politely.** DBnomics is free infrastructure run for the statistical community. The rate cap exists so this terminal stays a good citizen; if a list is slow, it is usually the upstream provider, and hammering **REFRESH** will not help.

10 Driving it from AI Chat

AGENT TOOLS · 10

DBnomics publishes **five agent tools** in the `dbnomics` category — the same drill path the panel walks, available to the assistant:

Tool	What it does
`list_dbnomics_providers`	Enumerates the providers
`list_dbnomics_datasets`	Lists a provider's datasets
`list_dbnomics_series`	Lists the series in a dataset
`get_dbnomics_observations`	Fetches the actual numbers for a series
`search_dbnomics`	Searches datasets across every provider

Why this is useful

- **Finding the right series is the hard part.** "Which ILO dataset has youth unemployment for India, and what is the series code?" is exactly the question `search_dbnomics` plus `list_dbnomics_series` answers — and it sidesteps the empty-name problem in the search list (section 04), because the assistant reads the codes directly.
- **Numbers into other screens.** With `get_dbnomics_observations` the assistant can pull a series and write it into an Excel sheet or a Report Builder block, rather than you copying figures out of the observation table.
- **No credentials, no cost surprises.** These tools hit the same free public API, so the only metered part of the exchange is the assistant's own reasoning.

> **The tools are independent of the tab.** Unlike the Excel tools, they do not drive the screen — an agent can look a series up without the DBnomics tab being open, and what it fetches will not appear in your charts. Ask it for the series code, then load that series in the panel to chart it.

11 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Markets & Data ▸ DBnomics, or type `dbnomics` in the command bar
…get the provider list back?	Press **FETCH**
…find a provider quickly?	Type its code or name into *Filter providers…*
…find a subject across providers?	Use **GLOBAL SEARCH** — then hover a row to identify it
…see more than 50 datasets?	**LOAD MORE** under the list
…chart a series?	Click it — it draws immediately
…overlay a second series?	Load it, then **ADD TO SINGLE VIEW**
…compare different units?	Switch to **COMPARE** and give each series its own slot
…change the chart type?	**CHART** selector in single view; the slot header in compare view
…read the dates?	From the OBSERVATION DATA table — the x axis shows point numbers
…export the numbers?	**EXPORT CSV** (single view only)
…start over?	**CLEAR ALL** — it clears the slots as well

Honest notes on this build

- **Global search results show only the provider code** — the dataset name arrives empty. Hover for the tooltip, or use the provider path.
- **The chart x axis is an observation index, not a date.**
- **No hover read-out on the chart**, and the chart-type dropdown clips *SCATTER*.
- **EXPORT CSV covers the single view only**, and does **not** register the file in File Manager.
- **Comparison slots are not persisted** — only the selection and view mode come back.
- **CLEAR ALL has no confirmation** and empties the slots as well as the single chart.
- **Mixed units silently flatten** in the single view; use slots instead.
- **Everything depends on a live network**; there is no offline cache of observations.

> Read from the shipping source and confirmed on screen against the live API, including the empty search names and the index-based axis.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The AkShare Data Explorer

A complete, screen-by-screen walkthrough of the AkShare tab — twenty-six data families and a thousand-plus endpoints covering Chinese and global markets, each with a parameter form built from its own schema.

`>_ 26 sources · 1000+ endpoints · schema-driven params · free API`

Covers · **the 26 sources** · **the endpoint list** · **the parameter form**
reading results · the error hints · caching and state · the 3 agent tools

Demonstrated live on the Fincept Terminal AkShare Data tab.

Contents

10 SECTIONS

GETTING ORIENTED

01 What the AkShare tab is 3

02 Anatomy of the tab 4

03 The twenty-six sources 5

RUNNING QUERIES

04 The endpoint list 6

05 The parameter form 7

06 Running a query and reading results 8

07 When a query fails — the hints 9

REFERENCE

08 Caching, state and cost 10

09 Driving it from AI Chat 11

10 Quick reference and honest notes 12

About this guide. Every screenshot is a live capture of the AkShare tab in the shipping build, returning genuine exchange data on the day of writing — including the error-hint example, which was provoked deliberately and then undone.

01 What the AkShare tab is

ORIENTATION · 01

AkShare Data is the terminal's **general-purpose explorer for Chinese and global market data**: twenty-six families — equities, indices, futures, bonds, FX, energy, crypto, REITs, funds, company information, news, and both Chinese and global macro — exposing well over a thousand individual endpoints.

What makes it more than a list is the **parameter form**. Each endpoint declares its own arguments, and the tab builds an input per argument, pre-filled with a working default — so `stock_zh_a_hist` arrives ready to run with a symbol, a period and a date range already in place, and you edit only what you want to change.



Fig 1 — The header: the tab name, its coverage line, and the green **FREE API** badge.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Markets & Data ▸ AKShare Data**
Command bar	Type `akshare`, then pick the row

At a glance, in AkShare Data you can…

- Browse 26 data families in one grid
- Search endpoints inside a family
- See each endpoint's parameters as a form
- Run with the wrapper's defaults in one click
- Override any parameter before running
- Read results as a sortable table
- Switch to indented JSON
- Get an actionable hint when a call fails
- Have the assistant query it for you
- Do all of it free — no key, no credits

> **Free and local, like its siblings.** Every query runs a bundled Python wrapper on your own machine against public endpoints. Nothing goes through your Fincept account and no credits are spent — the **FREE API** badge in the header is literal.

02 Anatomy of the tab

Six regions, top to bottom: header, source grid, then a split of endpoint list against the parameter row and result area, over a status bar.

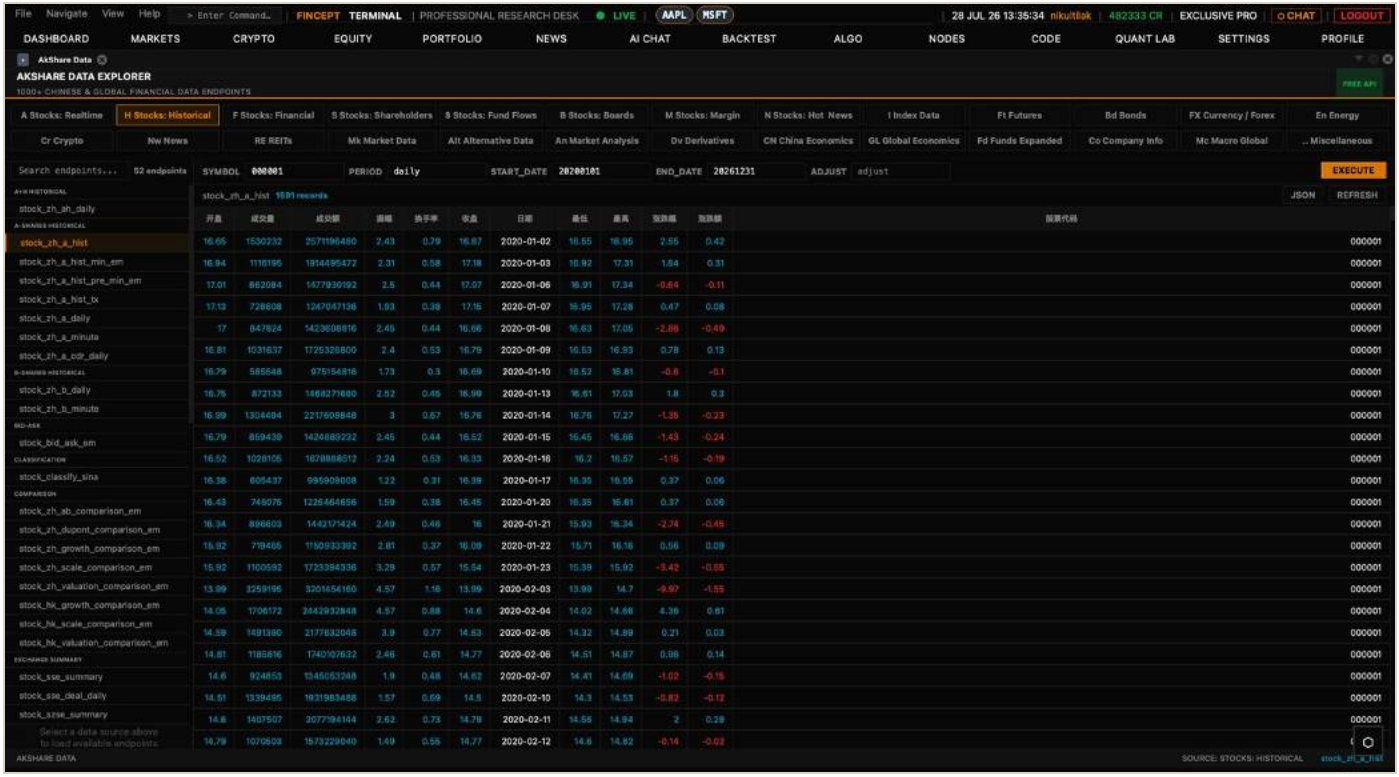


Fig 2 — Running `stock_zh_a_hist`: source grid at the top, endpoint list on the left, the parameter form above the results, and 1,591 rows of daily history.

#	Region	What it is for
1	Header	Tab name, coverage line, **FREE API** badge.
2	Source grid	All 26 families as buttons, two rows deep; the active one is amber. Section 03.
3	Endpoint panel	Search box, endpoint count, and the endpoint list grouped under amber sub-headings. Section 04.
4	Parameter row	One labelled input per parameter the endpoint declares, plus **EXECUTE**. Section 05.
5	Data panel	Status and record count, the **JSON**/**TABLE** toggle, **REFRESH**, and the result itself. Section 06.
6	Status bar	**AKSHARE DATA**, the active **SOURCE**, and the selected endpoint.



Fig 3 — The status bar names the active source and endpoint, which is useful when the endpoint list is scrolled away from your selection.

> **The source grid scrolls if the window is narrow**, and the parameter row scrolls horizontally when an endpoint declares many arguments — so nothing is ever hidden, but on a small window you may need to scroll to reach the last field.

03 The twenty-six sources

Each button is one data family, served by its own wrapper. The first eight are the equity families that the **Asia Markets** tab also exposes; the rest are unique to this tab.



Fig 4 — The source grid, two rows of thirteen. The active source is amber.

Group	Sources
Equities (8)	Stocks: Realtime · Historical · Financial · Shareholders · Fund Flows · Boards · Margin · Hot & News
Other asset classes (7)	Index Data · Futures · Bonds · Currency / Forex · Energy · Crypto · REITs
Macro & economics (3)	China Economics · Global Economics · Macro Global
Funds & companies (2)	Funds Expanded · Company Info
Research & other (6)	Market Data · Alternative Data · Market Analysis · Derivatives · News · Miscellaneous

Choosing a source loads its endpoint list and clears the search box, but leaves the previous result on screen until you run something new — so you can read the old table while deciding what to run next.

Which tab for which job. For a quick equity number, **Asia Markets** is faster: eight tabs, click and go. Come here when the data is outside equities, or when the endpoint needs more than a symbol — this is the only one of the two with a parameter form and error hints.

One cosmetic wrinkle. The *Select a data source above to load available endpoints* prompt stays visible under the endpoint list even after a source has loaded. It is a leftover empty-state label, not an indication that anything failed.

04 The endpoint list

Endpoints are grouped under amber sub-headings taken from the wrapper — for the Historical family those are *A+H HISTORICAL*, *A-SHARES HISTORICAL*, *B-SHARES HISTORICAL*, *BID-ASK*, *CLASSIFICATION*, *COMPARISON* and so on.



Fig 5 — The Historical family: 52 endpoints under six sub-headings.

Control	What it does
`Search endpoints…`	Filters as you type, matching the endpoint name
`N` endpoints	How many this family offers
`A row`	Click to select it: the parameter form rebuilds, and the query runs if it can (section 05)
`Sub-headings`	Labels only — not clickable

Endpoint names are the wrapper's own function names, which is deliberate: they are the vocabulary the underlying library and its documentation use, so what you learn here transfers.

> **Reading the names pays off.** The suffix usually tells you the upstream source — `_sina`, `_em` (East Money), `_ths` (Tonghuashun), `_tx` (Tencent), `_cls` — and different sources for the same data have different reliability and different symbol formats. When one fails, its sibling with a different suffix is usually the fastest fix.

> **The prefix tells you the asset class** — `stock_`, `option_`, `futures_`, `bond_`, `fund_`, `macro_` — which makes the search box more useful than it first looks: typing `hist` or `spot` narrows a hundred-endpoint family to the handful you want.

05 The parameter form

This is the feature that makes the tab usable. Selecting an endpoint reads its parameter schema and builds one labelled input per argument, pre-filled with the wrapper's own default.



Fig 6 — `stock_zh_a_hist` declares five parameters, arriving filled in: SYMBOL 000001, PERIOD daily, START_DATE 20200101, END_DATE 20261231 and an empty ADJUST.



Fig 7 — A different endpoint, a different form — and note the symbol format: `stock_zh_a_hist_tx` wants `sz000001` where the previous endpoint wanted `000001`.

Element	Meaning
`A label in capitals`	The parameter's name, as the wrapper expects it
`A label ending in *`	**Required.** The query will not run until it has a value
`A pre-filled value`	The wrapper's own default — a working example, not a guess
`An empty field with grey placeholder`	Optional: leave it blank and the wrapper's default applies
`No parameters — runs with defaults`	This endpoint takes no arguments at all
`EXECUTE`	Runs the endpoint with every non-empty field, sent by name

> **Only non-empty fields are sent**, and they are sent *by name* rather than by position. Clearing a field is therefore a safe way to say "use your default" — you cannot misalign the arguments the way a positional call can.

> **Press Enter in any parameter field to run the query** — the same as clicking EXECUTE. Handy when you are iterating on a symbol or a date range.

> **Symbol formats differ per endpoint**, as Figs 6 and 7 show: a bare code, an exchange-prefixed code, a five-digit HK code, or a market-prefixed US ticker. The pre-filled default is always in the right shape for that endpoint — so **edit it rather than replacing it wholesale**, and keep its pattern.

06 Running a query and reading results

RESULTS · 06

Selecting an endpoint runs it immediately **if it can**: when every required parameter already has a value — which the defaults usually ensure — the query goes out without a second click. When something required is blank, the status line says *Fill required parameter(s), then EXECUTE* and waits for you.



Fig 8 — `option_comm_info`, a no-parameter endpoint: 461 rows of live commodity-option contract data straight from the exchange.

Behaviour	Detail
`Columns`	Whatever the payload's first row declares, in order — Chinese headings are passed through as published
`Numbers`	Right-aligned, cyan for zero-or-positive, red for negative
`Missing values`	Shown as `--`
`Sorting`	Click any column header
`Row cap`	5,000 rows; the status line then reads *Showing 5000 of N records*
`Column widths`	Fitted to content when the payload has 20 columns or fewer; wider payloads keep a uniform width for speed
`JSON view`	Always kept in step with the table — switch any time to read exact field names and types



Fig 9 — `stock_zh_a_hist`: 1,591 daily rows, with the change columns coloured by sign.

> **REFRESH re-runs the current endpoint with the current parameters.** It is the button to use after editing a field — and the way to get a fresh read once the two-minute result cache has expired (section 08).

07 When a query fails — the hints

ERROR HINTS · 07

Public data endpoints fail for a handful of recurring reasons, and this tab is the one screen in the terminal that tries to tell you *which* reason. The status line shows a shortened message plus a plain-language hint; the full text goes to the JSON view.



Fig 10 — A deliberately malformed symbol. The status line carries the underlying error and the hint *check the symbol format for this endpoint.*

Hint you see	What went wrong	What to do
`— source temporarily unavailable, try again`	A timeout, a dropped connection or exhausted retries	Press **REFRESH** after a moment; try a sibling endpoint with a different source suffix
`— data source returned no data (may be geo-restricted or rate-limited)`	The upstream server answered with an empty or unparseable body — common for Chinese sources reached from outside China	Try a different source suffix (`_sina` vs `_em`), or come back later
`— check the symbol format for this endpoint`	A value could not be parsed — usually a symbol in the wrong shape, or a malformed date	Restore the field to its default shape and edit within that pattern (section 05)
`— parameter not accepted by this endpoint`	A named argument the wrapper does not take	Clear the field so the wrapper's default applies

> **Unrecognised failures get no hint**, and long messages are truncated — the complete error is always in the **JSON** view. **Geo-restriction is the most common surprise:** several mainland sources answer only from inside China and return an empty body from elsewhere, which is the second hint above. A differently-sourced sibling endpoint often works.

08 Caching, state and cost

STATE · 08

Caching

What	Cached for	How you can tell
`Endpoint lists (per source)`	1 hour	Returning to a source is instant
`Query results`	2 minutes	The status line appends *(cached)* to the endpoint name

The result cache is keyed on the endpoint *and* its parameters, so changing a symbol or a date range always issues a real query. Re-running the identical call inside two minutes returns the cached rows.

What is remembered between sessions

- The active **source** and the selected **endpoint**.
- Your **search text** and whether you were in **table or JSON** view.
- **Every parameter value you typed**, stored by name and re-applied to the rebuilt form.
- **The last result**, so the tab can show you what you were looking at without re-querying.

> **Restoring the endpoint takes a moment.** The endpoint list has to load before a selection can be re-applied, so on opening the tab you may see the source highlight and the previous result for a beat before the endpoint row and its parameter form come back. That is the restore completing, not a failure.

Cost and dependencies

Property	Value
`Cost`	**Free.** No API key, no Fincept credits, nothing routed through your account
`Where the work happens`	A bundled Python data wrapper on your own machine
`Network`	Required — each query calls a public endpoint directly
`Rate limits`	Whatever the upstream source imposes; the tab surfaces them as the “try again” hint

09 Driving it from AI Chat

AkShare publishes **three agent tools** in the `akshare` category — a deliberately small set that mirrors the three things the screen does:

Tool	What it does
`list_akshare_sources`	Enumerates the 26 data families
`list_akshare_endpoints`	Lists the endpoints in one family, with their parameters
`query_akshare`	Runs an endpoint with named arguments and returns the rows

Why this is the useful way to use it

- **Finding the endpoint is the hard part.** A thousand endpoints named after library functions is a lot to scan; "which AkShare endpoint gives me Hong Kong daily history, and what parameters does it take?" is answered in one turn by `list_akshare_endpoints`.
- **The parameters come with the list.** Because the schema travels with the endpoint listing, the assistant can fill a call correctly first time — including the symbol format, which is the thing people most often get wrong by hand.
- **Rows into other screens.** `query_akshare` returns the data, so the assistant can put a Chinese price history into an Excel sheet or summarise a fund-flow table without you transcribing anything.

> **The tools do not need this tab open**, and they do not change what is on screen — they are an independent path to the same wrappers. Ask for the endpoint name and parameters, then run it here if you want the sortable table.

> **Same cost profile.** The tools call the same free local wrappers, so the only metered part of an exchange is the assistant's own reasoning — not the data.

10 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Markets & Data ▸ AKShare Data, or type `akshare`
…find data for an asset class?	Pick its source button, then search the endpoint list
…run an endpoint?	Click it — it runs unless a required parameter is blank
…change a parameter?	Edit the field and press **Enter**, or **EXECUTE**
…use the wrapper's default?	Clear the field — only non-empty fields are sent
…know which fields are compulsory?	Their labels end in *
…get the symbol format right?	Keep the shape of the pre-filled default
…see exact field names?	Switch to **JSON**
…understand a failure?	Read the hint on the status line; the full error is in the JSON view
…work around a blocked source?	Try a sibling endpoint with a different suffix (`_sina`, `_em`, `_tx`)
…force a fresh read?	Wait out the two-minute cache, then **REFRESH**
…have the assistant do it?	Ask in AI Chat — it uses the three `akshare` tools

Honest notes on this build

- **The empty-state prompt never hides** — *Select a data source above…* stays under a populated endpoint list.
- **Tables cap at 5,000 rows**, stated in the status line when it bites.
- **Column widths are only fitted for payloads of 20 columns or fewer**; wider results keep a uniform width.
- **Chinese column headings are passed through verbatim** — there is no translation layer.
- **Results cache for two minutes**, so an immediate re-run may return *(cached)*.
- **Endpoint restore is deferred** by design, so the form reappears a moment after the tab opens.
- **Selecting an endpoint fires a query** when it can — convenient, but a click does hit a public source.
- **Geo-restriction and rate limits are real** for several mainland sources; the hints name them but cannot fix them.
- **There is no CSV export.** To keep a result, use the JSON view, or have an agent write it to a file with the file-manager tools.

> Read from the shipping source and confirmed on screen — including the error hint, which was provoked with a malformed symbol and then reverted.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

The Asia Markets Terminal

A complete, screen-by-screen walkthrough of the Asia Markets tab — four hundred Chinese, Hong Kong and US equity endpoints behind eight category tabs, queried straight from the desk.

`>_ 8 categories · ~400 endpoints · live CN data · free, keyless`

Covers · **the eight categories** · **running an endpoint** · **reading the table**
when a query fails · regions, caching and state · how it relates to the AkShare tab

Demonstrated live on the Fincept Terminal Asia Markets tab.

Contents

8 SECTIONS

GETTING ORIENTED

01 What the Asia Markets tab is 3

02 Anatomy of the tab 4

03 The eight categories 5

RUNNING QUERIES

04 Finding and running an endpoint 6

05 Reading the results 7

06 When a query fails 8

REFERENCE

07 Regions, caching and what is remembered 9

08 Quick reference and honest notes 10

> **About this guide.** Every screenshot is a live capture of the Asia Markets tab in the shipping build, returning genuine exchange data on the day of writing — including the failure and region tests, which were run deliberately.

01 What the Asia Markets tab is

ORIENTATION · 01

Asia Markets is a **direct line to roughly four hundred equity data endpoints** covering mainland Chinese A and B shares, Hong Kong and US listings — real-time quotes, history, financials, shareholder registers, fund flows, board and sector aggregates, margin and Stock-Connect data, and the hot-topic and news feeds that Chinese desks watch.

It is built for the moment when you need a specific Chinese market number and do not want to negotiate a vendor API to get it. Pick a category, pick an endpoint from the list, and the answer arrives as a table you can sort — usually in a couple of seconds.



Fig 1 — The header names the tab and states its coverage; the four region buttons sit on the right.

Where to find it

Route	How
Navigate menu	**Navigate ▸ Markets & Data ▸ Asia Markets**
Command bar	Type `asia`, then pick the row

At a glance, in Asia Markets you can…

- Browse ~400 endpoints in eight categories
- See live A-share, B-share and HK quotes
- Pull daily and intraday history
- Read financial statements and ratios
- Inspect shareholder and holdings data
- Follow fund flows and margin balances
- Watch board, sector and concept aggregates
- Read the Chinese market news and hot lists
- Sort any result and switch to raw JSON
- Do all of it free — no key, no credits

> **Free and local.** Each query runs a bundled Python data wrapper on your own machine, which calls the public Chinese market endpoints directly. Nothing is routed through your Fincept account and no credits are spent — the same model as the Gov Data and Economics tabs.

02 Anatomy of the tab

Five regions: header, category bar, the endpoint panel, the data panel, and a status bar.

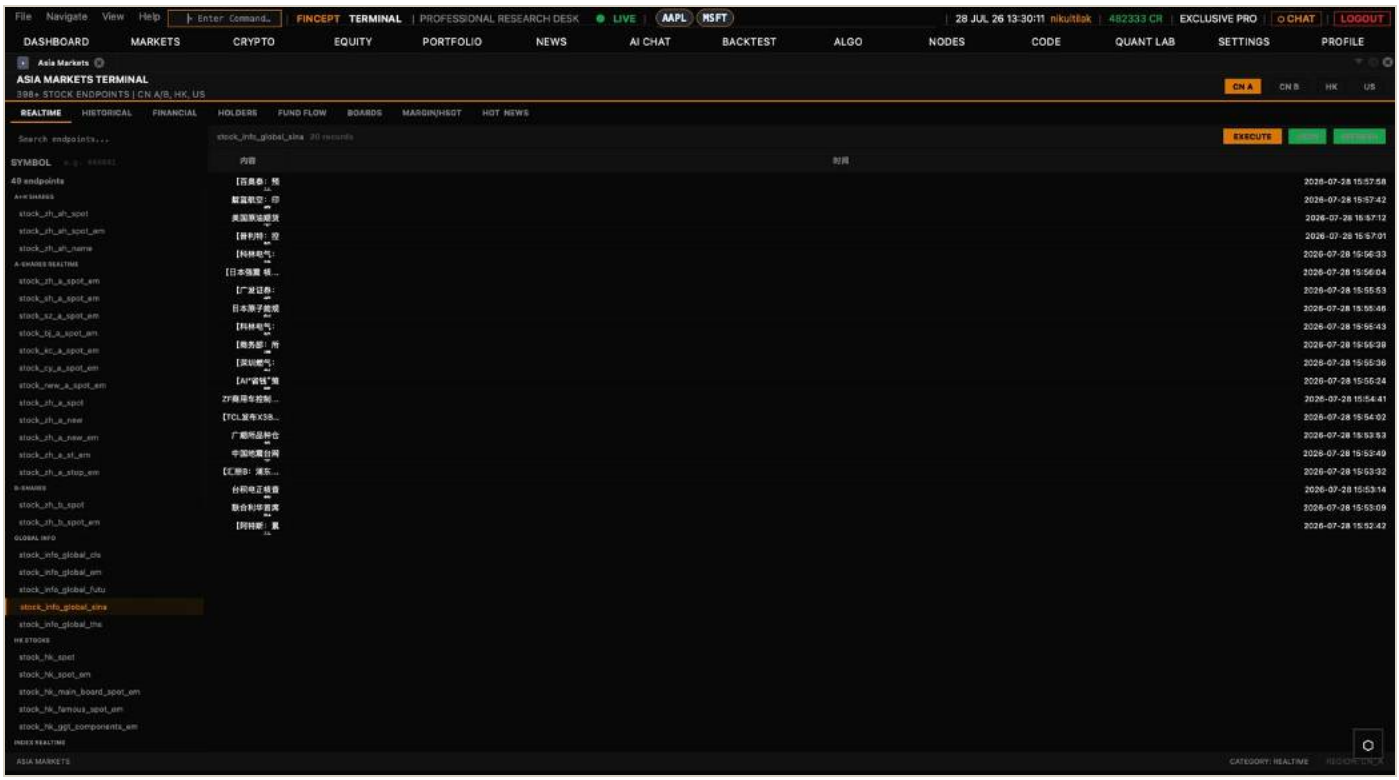


Fig 2 — The tab as it opens: REALTIME selected, an endpoint auto-run, and its 20 rows of live market headlines on the right.

#	Region	What it is for
1	Header	The tab title, its coverage line, and the four region buttons — **CN A**, **CN B**, **HK**, **US**. Section 07 explains what these actually do.
2	Category bar	The eight category tabs, each with its own accent colour. Section 03.
3	Endpoint panel	A search box, a **SYMBOL** field, the endpoint count, and the endpoint list grouped under amber sub-headings. Section 04.
4	Data toolbar	The running endpoint's name, its record count, then **EXECUTE**, the **JSON**/**TABLE** toggle and **REFRESH**.
5	Result area	The result as a sortable table, or the same payload as indented JSON. Section 05.
6	Status bar	**ASIA MARKETS**, the active **CATEGORY** and the selected **REGION**.



Fig 3 — The status bar always states which category and region are selected.

> The divider between the endpoint panel and the data panel can be dragged — the panel is constrained to a sensible width, so results always keep most of the window.

03 The eight categories

Each category tab is a family of endpoints served by its own data wrapper. The counts below are what the tab declares for each one; the endpoint panel shows the live count for the category you are in.



Fig 4 — The category bar. The active tab is underlined in amber.

Category	Endpoints	What lives here
REALTIME	49	Live spot quotes for A shares, B shares, HK and global names, plus the market-news feeds
HISTORICAL	52	Daily, minute and pre-market history; bid-ask snapshots; classification and comparison series
FINANCIAL	45	Statements, ratios, earnings and valuation measures
HOLDERS	49	Shareholder registers, institutional holdings, pledge and change data
FUND FLOW	56	Money flow by stock, sector and market; the largest family in the tab
BOARDS	49	Industry, concept and style board aggregates, plus analyst rankings
MARGIN/HSGT	46	Margin trading and short balances, and Stock-Connect (northbound/southbound) flows
HOT & NEWS	55	Hot-stock lists, popularity rankings, announcements and news

> **Why the header says “398+”.** The eight declared counts add up to just over four hundred, and the number that actually appears in a category depends on what its wrapper reports at load time. Treat the header as an order of magnitude and the panel's *N endpoints* label as the truth for the category you are looking at.

> **Switching category re-loads the list, not the data.** The endpoint list for each category is fetched once and cached for an hour, so moving between tabs is instant after the first visit. Your search box is cleared on the way, because a filter that applied to one family rarely applies to the next.

04 Finding and running an endpoint

The panel lists every endpoint in the category, grouped under amber sub-headings that come from the wrapper itself — *A+H SHARES*, *A-SHARES REALTIME*, *B-SHARES*, *GLOBAL INFO*, *HK STOCKS*, and so on.

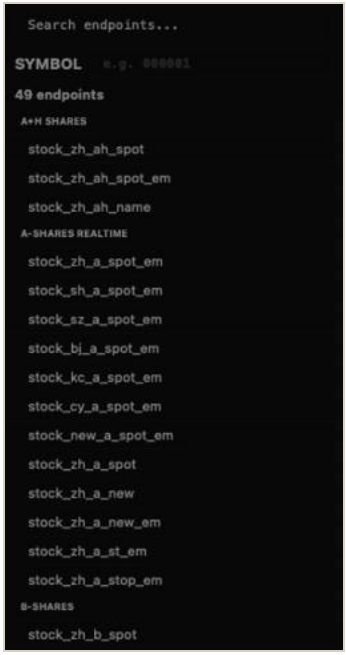


Fig 5 — The endpoint panel: search, SYMBOL, the count, and the grouped list.

Control	What it does
`Search endpoints…`	Filters the list as you type, matching the endpoint name. Sub-headings stay visible
`SYMBOL`	An optional instrument code passed to the endpoint — e.g. `000001`. Leave it empty for endpoints that need no symbol
`N` endpoints	How many endpoints this category offers
`The list`	Click any row to run it immediately — there is no separate confirm step
`EXECUTE`	Re-runs the selected endpoint with whatever is in SYMBOL
`REFRESH`	The same thing — handy after editing the symbol

The tab runs something for you the moment you arrive. On opening a category it picks an endpoint automatically — preferring one served by Sina, which tends to be the most reliable — and executes it, so you land on real data rather than an empty screen. Whatever it picked is named in the data toolbar.

SYMBOL is a single, positional value. It is passed as the endpoint's first argument, so it works for the many endpoints whose only input is an instrument code — and cannot help the ones that need a date range, a period or a market flag. Those either run on their wrapper's own defaults or fail (section 06). When you need to set several parameters, use the **AkShare Data** tab, which builds a proper form per endpoint.

Symbol formats vary by endpoint. A-share endpoints usually want a bare six-digit code (`000001`); some want an exchange prefix (`sz000001`); HK endpoints want five digits (`00700`). If a query fails on a symbol, that mismatch is the first thing to check.

05 Reading the results

Results arrive as a table built from the payload itself: whatever columns the endpoint returns become the table's columns, in order. Chinese column names are shown as the source publishes them.



Fig 6 — BOARDS ▸ analyst rankings: 100 records, numeric columns in cyan with negatives in red, and every column sortable by clicking its header.

Behaviour	Detail
`Columns`	Taken from the first row of the payload — so the table mirrors the source exactly
`Numbers`	Right-aligned and coloured: cyan for zero-or-positive, red for negative
`Missing values`	Shown as `--`
`Sorting`	Click any column header
`Row cap`	2,000 rows. Beyond that the toolbar reads *Showing 2000 of N*
`Record count`	The exact number returned, next to the endpoint name



Fig 7 — REALTIME ▸ global market news: a two-column payload of headline and timestamp, rendered by the same generic table.

> **The JSON toggle shows the same payload, indented.** It is the quickest way to see the real field names and types when a column heading is ambiguous — and it is where error detail appears (section 06). The button reads **JSON** when you are looking at the table, and **TABLE** when you are looking at JSON.

06 When a query fails

These endpoints are public services outside the terminal's control, and some of them fail — a missing parameter, an odd symbol format, a rate limit, or a source that simply will not answer from your part of the world. The tab's reporting of that is honest but terse.



Fig 8 — A failed query: the toolbar says only **Error**, and the JSON view carries the raw message — here a missing `'industry'` argument the endpoint required.

What you see	What it means	What to do
`Error` in the toolbar	The query failed. No detail is shown here	Switch to **JSON** — the real message is there
`A short quoted name`	The endpoint needed a parameter this tab cannot send	Run it from the **AkShare Data** tab, which builds a form for every parameter
`A timeout or connection message`	The upstream source did not answer	Wait and press **REFRESH**; try a Sina-served endpoint instead
`An empty or decode error`	The source returned nothing — often geo-restriction or rate limiting	Try another endpoint in the family, or the same data from a different category
`No data returned`	The call succeeded but the result was empty	Usually a symbol or a market that has no rows — check the symbol format

> **The status line shows only the word “Error”.** Unlike its sibling, the AkShare Data tab, this screen adds no explanation and no hint — the message in the JSON view is the raw text from the underlying wrapper, which can be as cryptic as a single quoted field name. Make the JSON toggle your first move whenever a query fails.

> **A failed query leaves the previous result cleared**, so an empty table plus **Error** is the normal appearance of a failure. Nothing is broken — pick another endpoint and carry on.

07 Regions, caching and what is remembered

STATE · 07

The region buttons

The four buttons in the header — **CN A**, **CN B**, **HK**, **US** — look like a market filter. In this build they are **not** one.


stock_analyst_rank_em (cached) 100 records

12个月收益率	2024年收益率	1最新个股评级-股票	1最新个股评级-股票	3个月收益率	6个月收益率	分析师ID	分析师单位	分析师名称	年度	年度指数
135.17	135.17	688256	寒武纪	57	129.5	11000213851	华福证券	任志强	2024	6424.01
118.58	118.58	920720	吉冈精密	31.05	118.58	11000341237	中国银河	洪烨	2024	2185.79
113.61	113.61	300638	广和通	27.64	41.63	11000455635	国联民生	张宁	2024	2136.08
110.82	110.82	920415	恒拓开源	52.93	113.77	11000280036	中国银河	范想想	2024	2398.42

Fig 9 — Region switched to **HK**, then REFRESH pressed: the same endpoint returns the identical mainland-China analyst rows, marked *(cached)*.

> **Selecting a region changes the highlight, the status bar and nothing else.** The query sends only the endpoint name and the SYMBOL field, so the data you get back is whatever that endpoint publishes. Choose the market by choosing an endpoint — the list names them plainly (`stock_hk_spot`, `stock_zh_a_spot_em`, `stock_us_spot_em`) — and treat the region buttons as a label for your own benefit.

Caching

What	Cached for	How you can tell
`Endpoint lists`	1 hour	Switching back to a category is instant
`Query results`	2 minutes	The toolbar appends *(cached)* to the endpoint name

Re-running inside the two-minute window returns the cached rows rather than re-querying, which is why a **REFRESH** immediately after a query can come back marked *(cached)*. Wait a moment if you specifically need a fresh read of a fast-moving quote.

What is remembered

- The active **category** and **region**, the selected **endpoint**, and the contents of the **search** and **SYMBOL** boxes.
- **The last result itself** — the JSON payload is stored with the screen state (up to a generous size limit), so reopening the tab can show you what you were looking at without a new query.

> **How this tab relates to AkShare Data.** Asia Markets is a focused front end over the eight equity wrappers; the **AkShare Data** tab exposes twenty-six wrappers — the same eight plus futures, bonds, FX, energy, crypto, REITs, macro and more — and adds a parameter form and actionable error hints. Use this tab for speed on equities, and that one when an endpoint needs arguments or lives outside equities.

08 Quick reference and honest notes

How do I…?

Question	Answer
…open the tab?	Navigate ▸ Markets & Data ▸ Asia Markets, or type `asia` in the command bar
…find an endpoint?	Pick the category, then type in *Search endpoints…*
…run one?	Click it — it runs on selection
…query a specific stock?	Put its code in **SYMBOL**, then **EXECUTE**
…get Hong Kong or US data?	Choose an HK or US *endpoint* — the region buttons do not filter
…see the real field names?	Switch to **JSON**
…find out why a query failed?	Switch to **JSON** — the message is there, not in the toolbar
…set a date range or period?	Use the **AkShare Data** tab; this one sends a symbol only
…force a fresh read?	Wait past the two-minute cache, then **REFRESH**
…sort by a column?	Click its header

Honest notes on this build

- **The region buttons are inert.** They set the highlight and the status bar; the query is unchanged. Verified live: switching to HK returned identical mainland rows.
- **Failures report only "Error"** in the toolbar; the detail is raw upstream text in the JSON view, sometimes just a quoted field name.
- **SYMBOL is the only parameter**, passed positionally — endpoints needing more cannot be driven from here.
- **Tables cap at 2,000 rows**, with the overflow stated in the toolbar.
- **Results cache for two minutes**, so a quick REFRESH may return *(cached)* rows.
- **An endpoint is chosen and run for you** when you enter a category — convenient, but it does mean a query goes out before you have asked for one.
- **Chinese column headings are passed through verbatim**; there is no translation layer.
- **No agent tools.** This tab publishes none — ask the assistant for Chinese data through the `akshare` tools instead.
- **Some sources are geo-restricted or rate-limited**, which shows up as an empty or decode error rather than a clear message.

> Read from the shipping source and confirmed on screen, including the region test and the deliberate failure.

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.

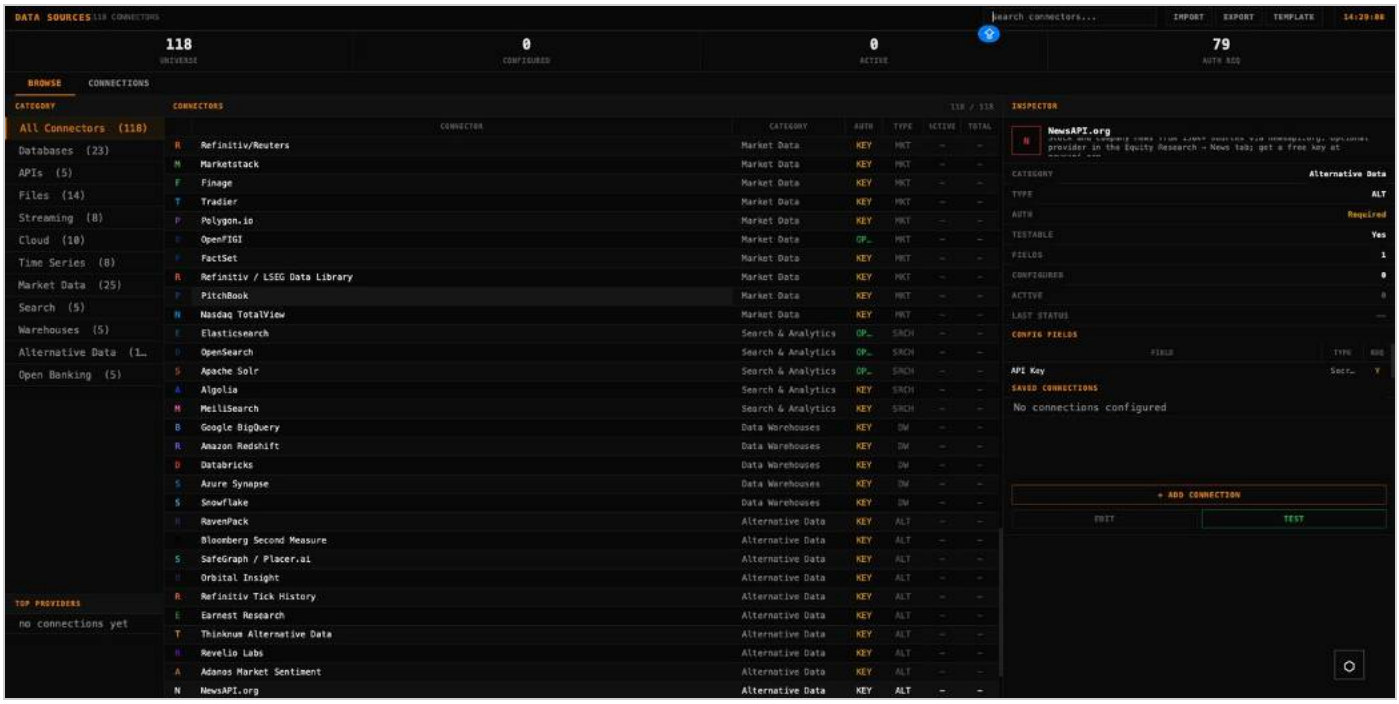
AT A GLANCE, IN DATA SOURCES YOU CAN…

- 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.

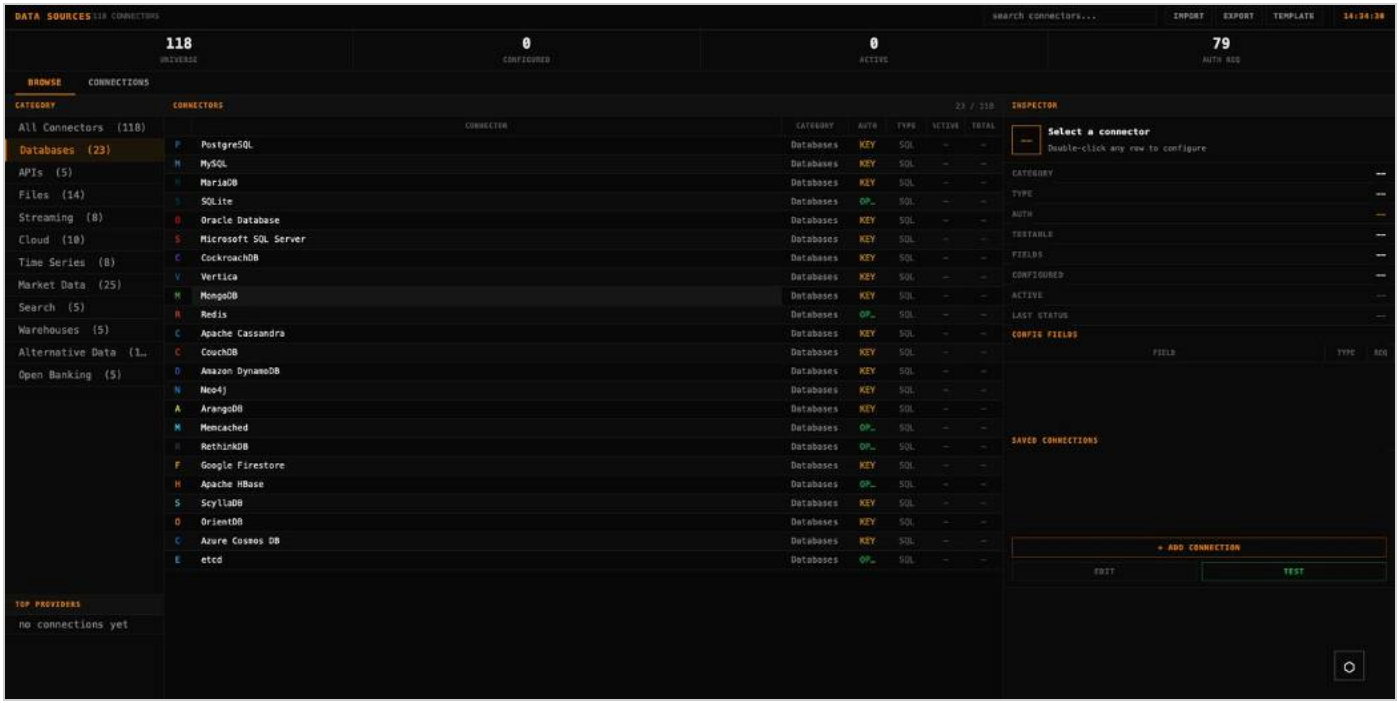
Region	What it shows / does
① **Screen header**	**DATA SOURCES** title + a live `N CONNECTORS` count; a **connector search** box; **IMPORT / EXPORT / TEMPLATE** buttons; and a live `HH:MM:SS` clock. The search filters the Browse catalog only (not the Connections table).
② **Stats strip**	Four boxes — **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).

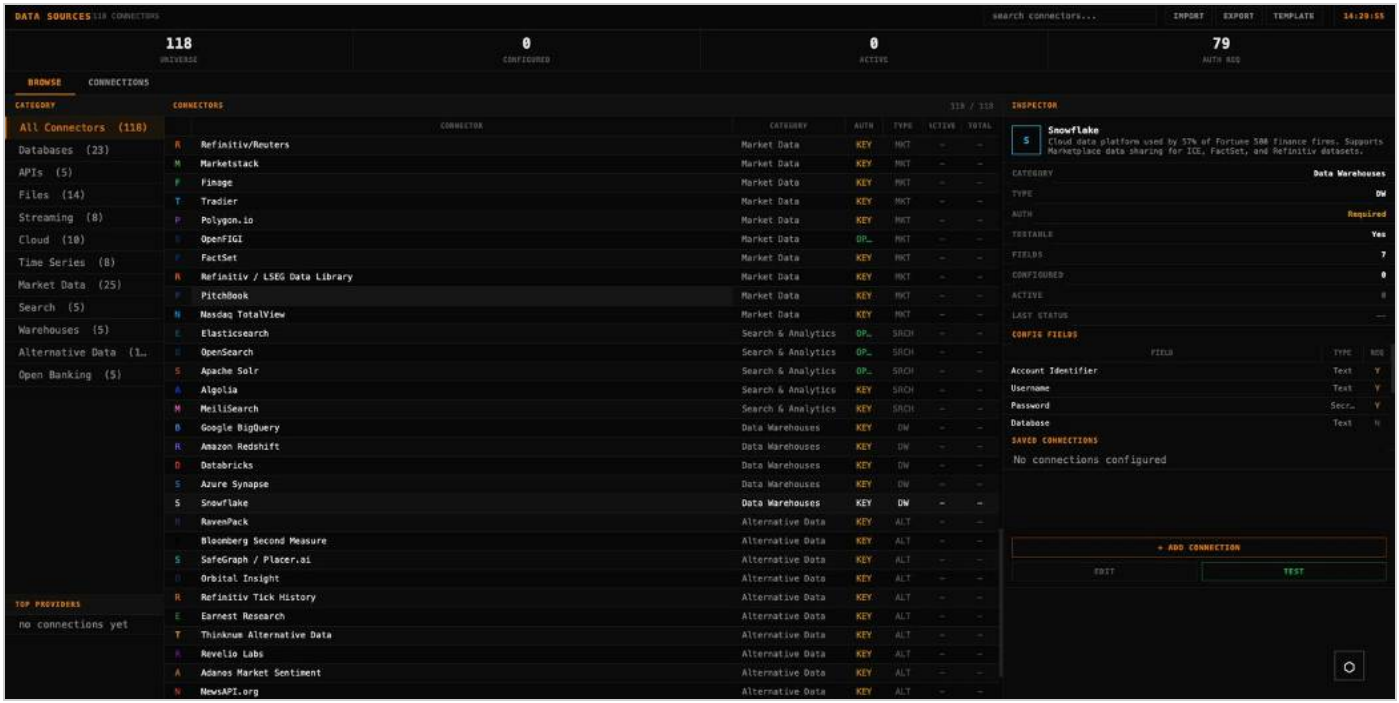
At a glance, in BROWSE you can…

- 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.

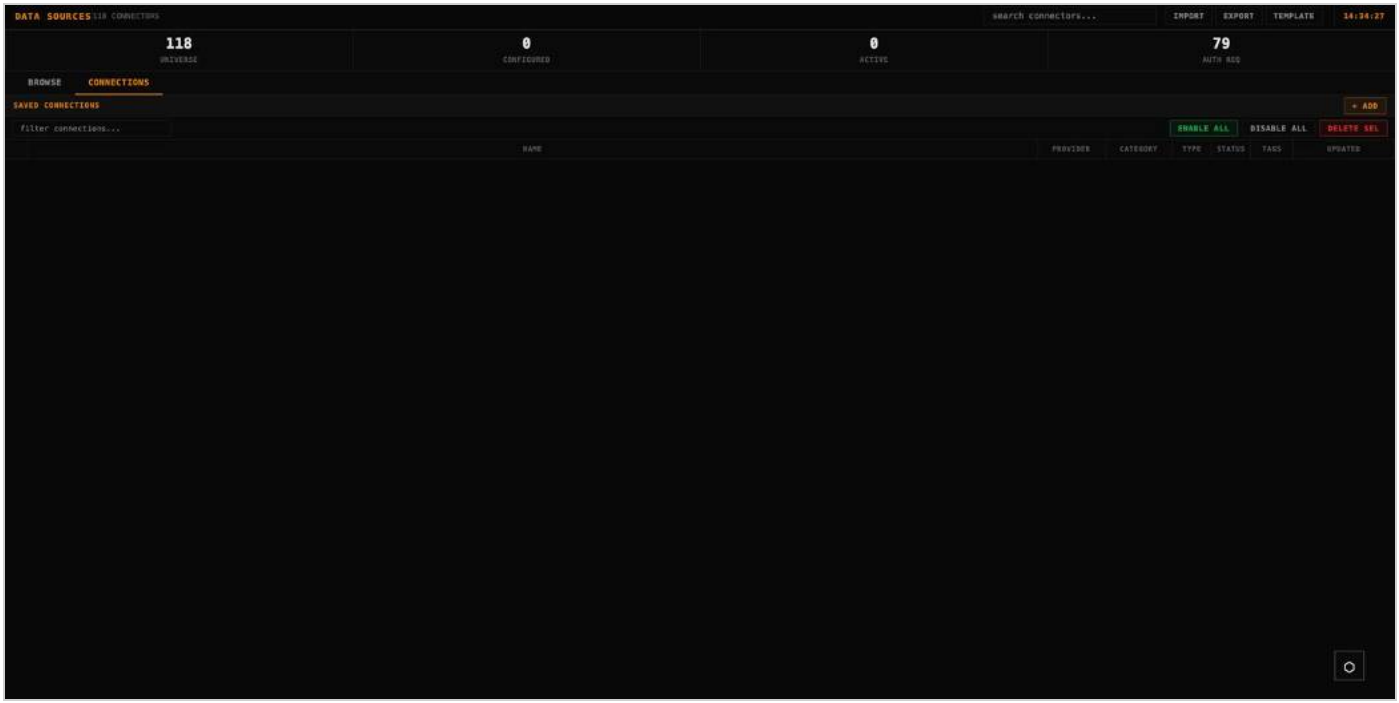
Aspect	Behaviour
What "Test" does	Resolves a provider-specific probe endpoint (a built-in table of each API's cheap health URL, e.g. Binance `/api/v3/ping`) or the saved host/port, then opens a **TCP socket** to it on a worker thread (5 s for a URL, 3 s for host:port).
Result dialog	"Connection successful" (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).

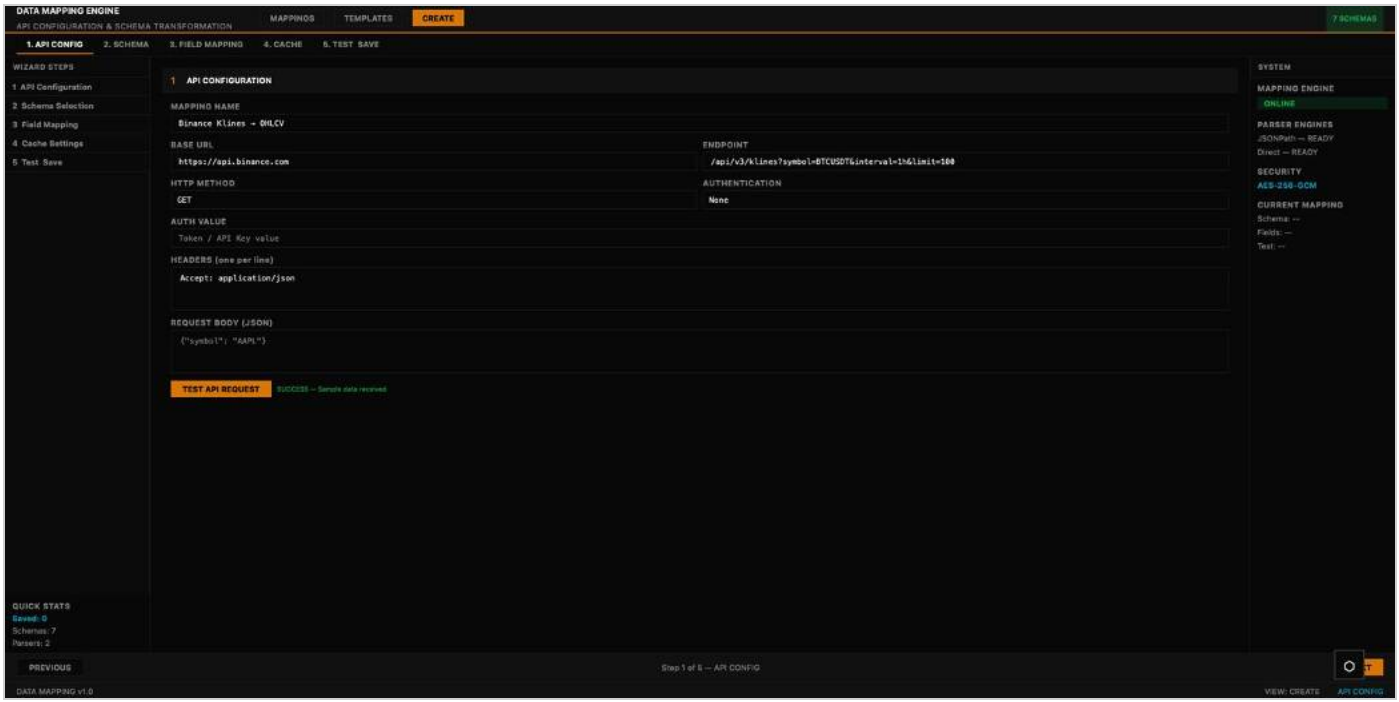
AT A GLANCE, IN DATA MAPPING YOU CAN…

- 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.

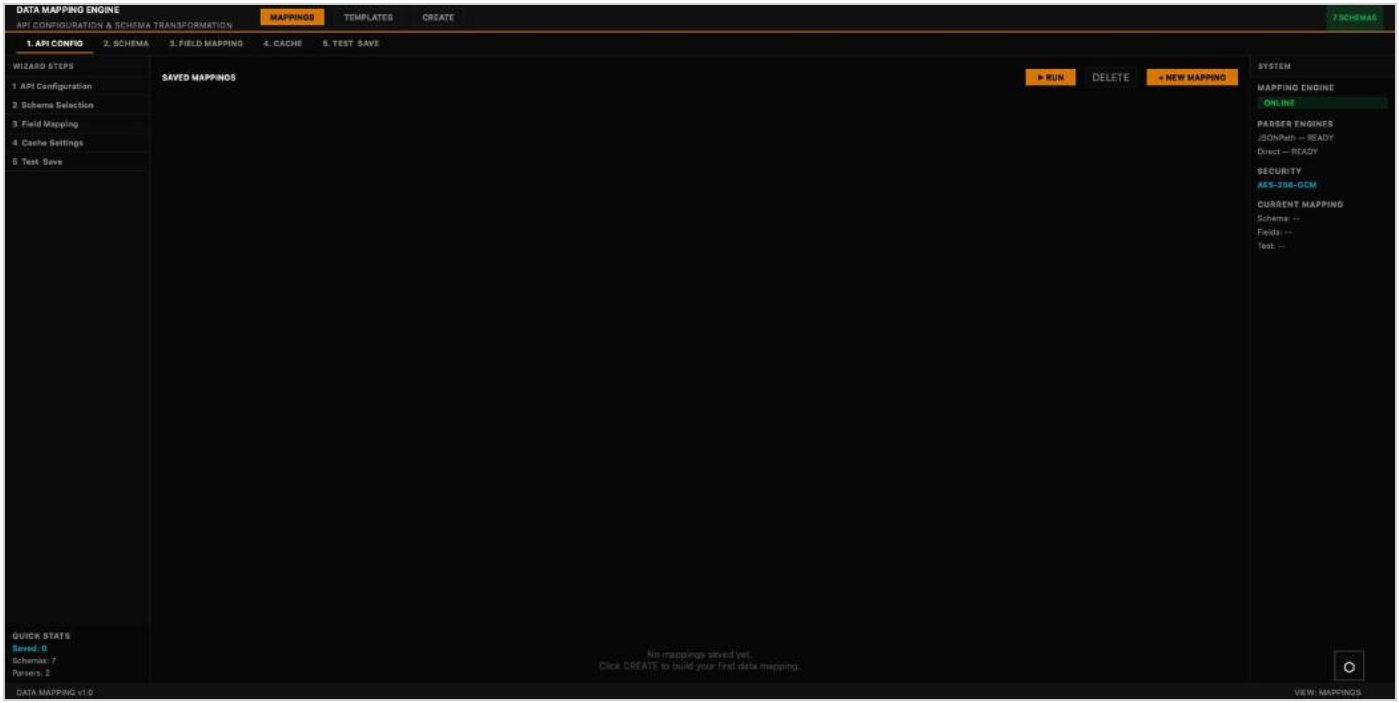
Region	What it shows / does
① **Header + view switcher**	Title **DATA MAPPING ENGINE**, three view buttons **MAPPINGS · TEMPLATES · CREATE** (the active one is amber), and a green **7 SCHEMAS** badge.
② **Step bar**	**1. API CONFIG · 2. SCHEMA · 3. FIELD MAPPING · 4. CACHE · 5. TEST & SAVE**. The active 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.

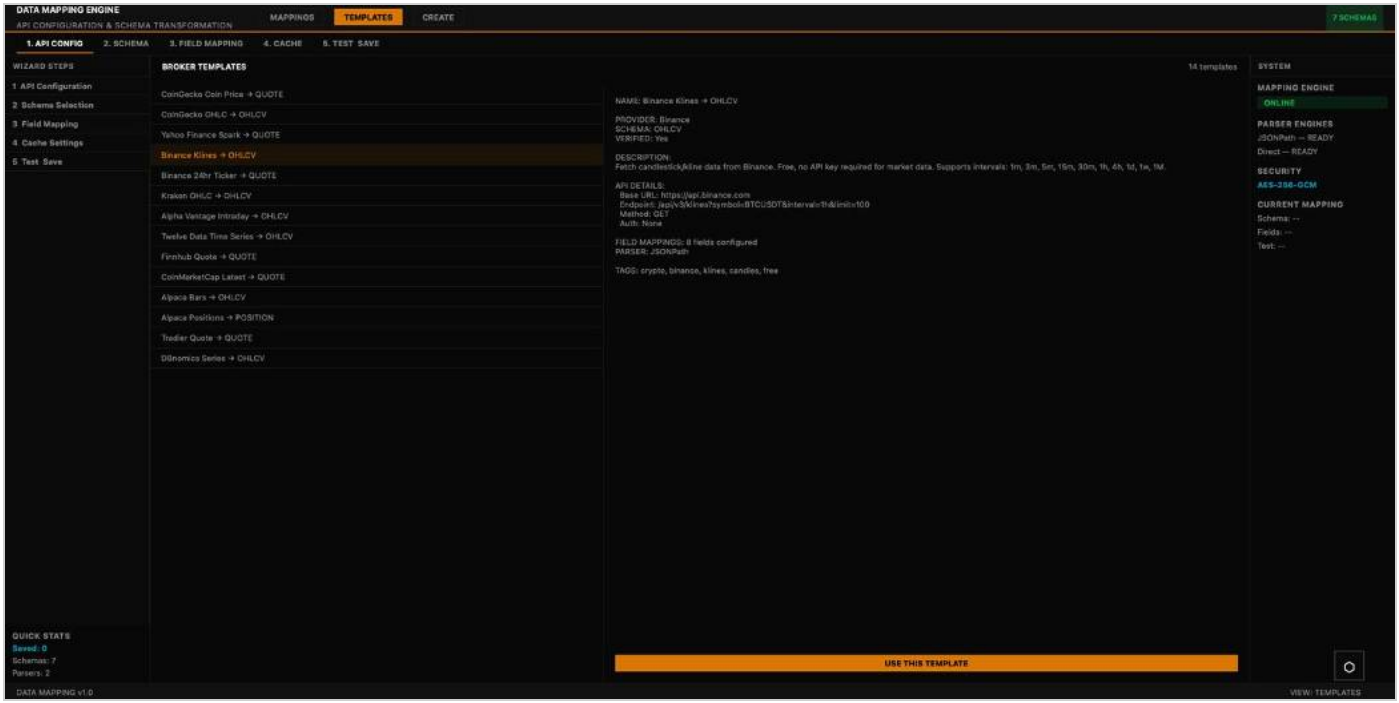
Control	Type	What it does
Saved-mapping list	list	One row per saved mapping, shown as `name — schema`. Select a row to target RUN / DELETE.
▶ RUN	button	Fetches the selected mapping's endpoint live, normalises it, and pops a summary — "Extracted N row(s) for schema X" — with a preview of the first 5 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

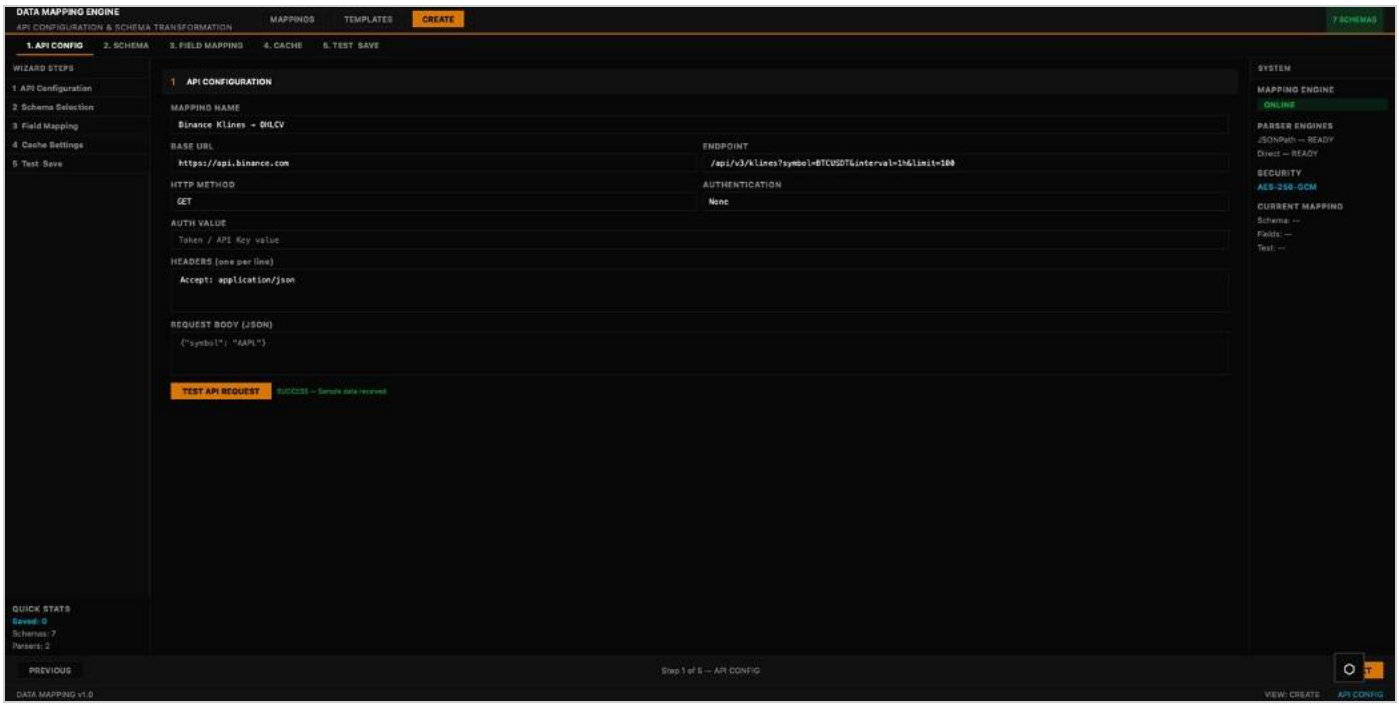
> **Note.** Templates that need a key carry a placeholder like `YOUR_KEY` in the endpoint/headers; **USE THIS TEMPLATE does not fill the Auth Value** — add your own key on Step 1 before saving.

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

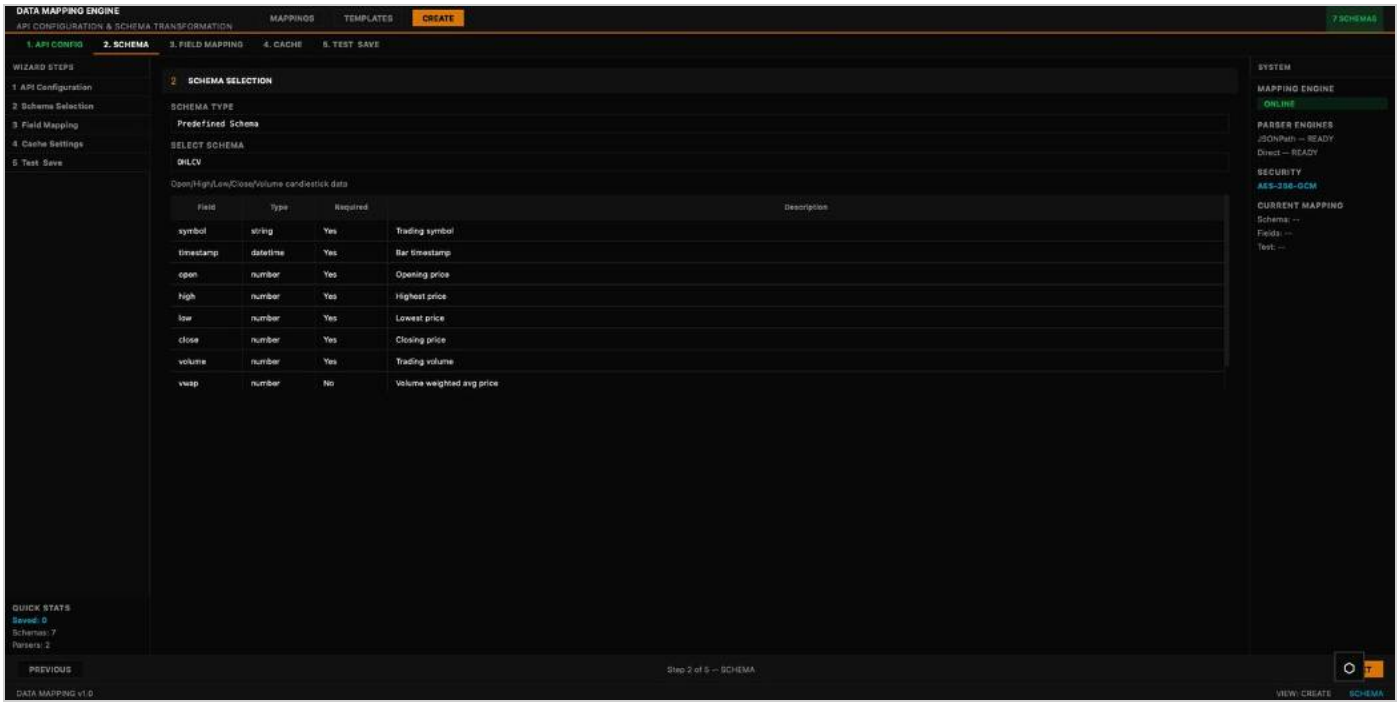


Step 1 after **TEST API REQUEST** — the green "SUCCESS — Sample data received" confirms a live sample was fetched (needed for Steps 3 & 5).

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

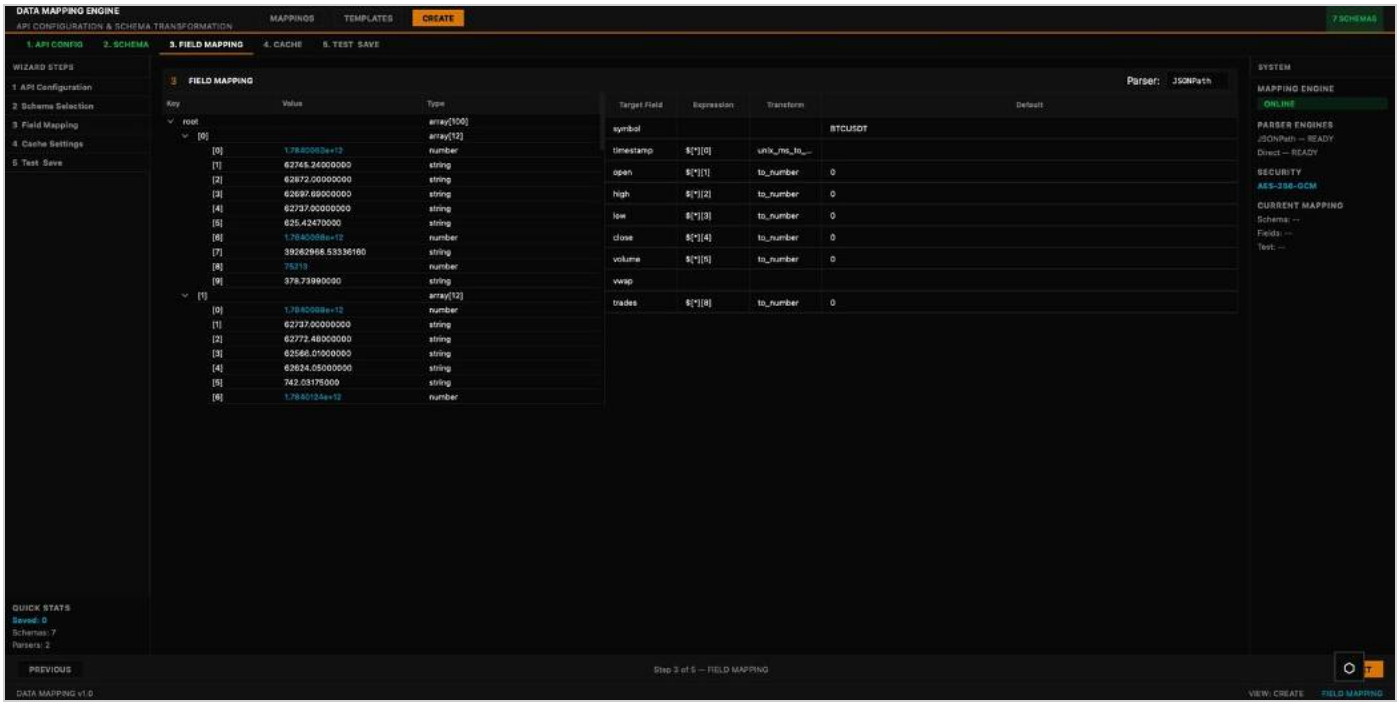


Step 2 — the target schema (OHLCV) and its field table: Field · Type · Required · Description.

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.

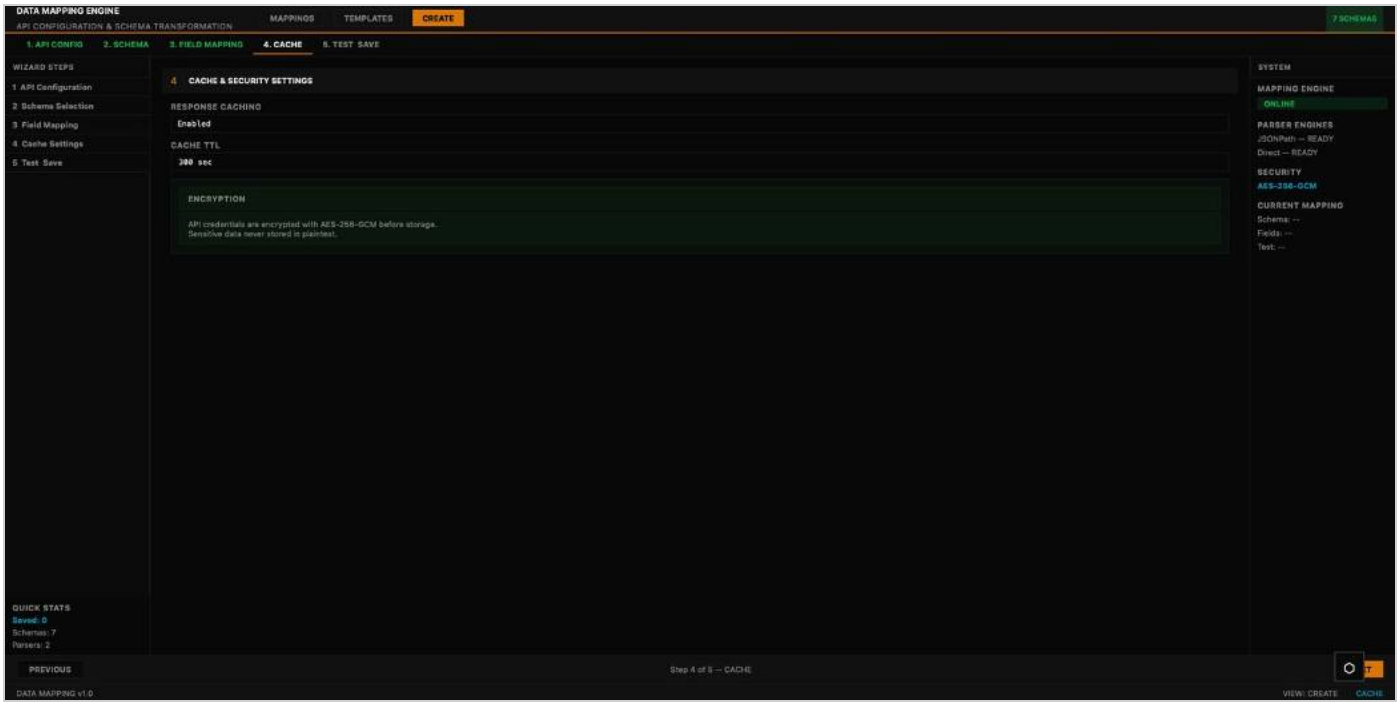
This is the heart of the tab. The **JSON explorer** (left) shows the sample response you fetched in Step 1 as a Key / Value / Type tree — you read the paths off it. The **mapping table** (right) has one row per schema field:

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.

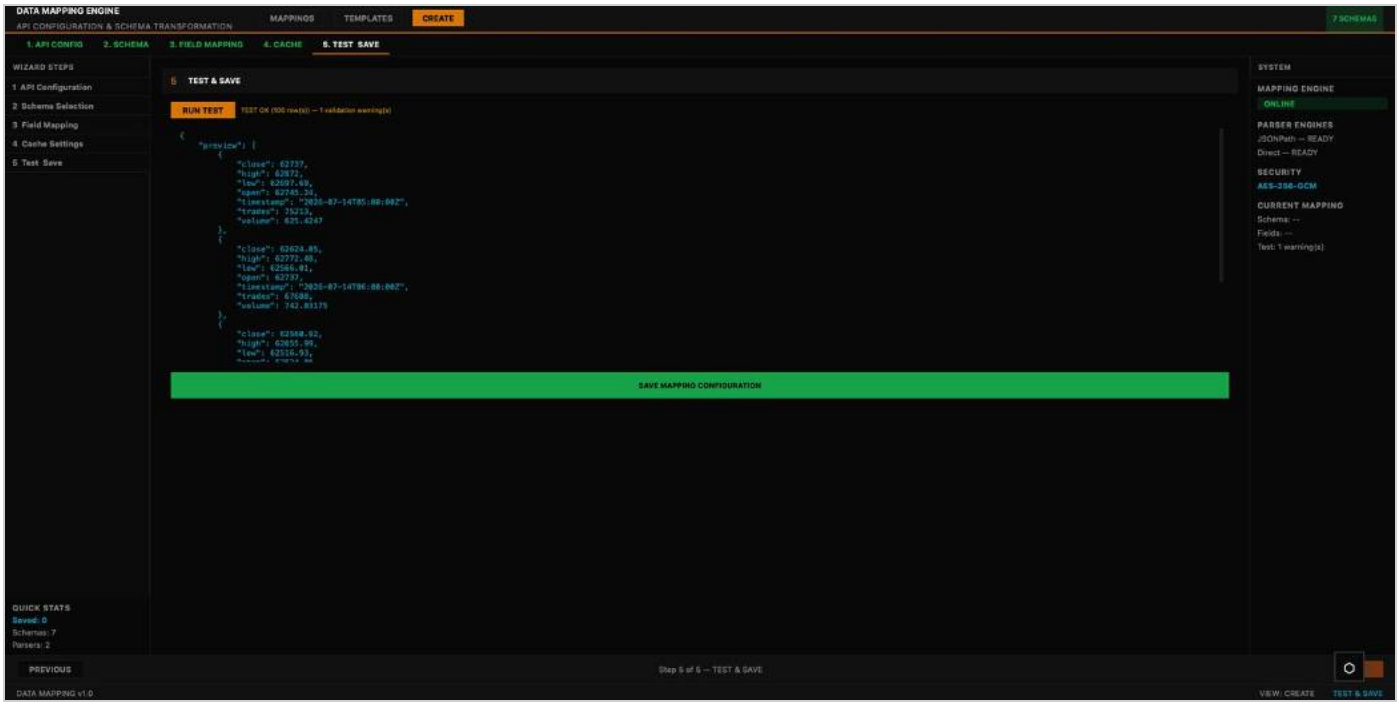
4 Cache & Security



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.

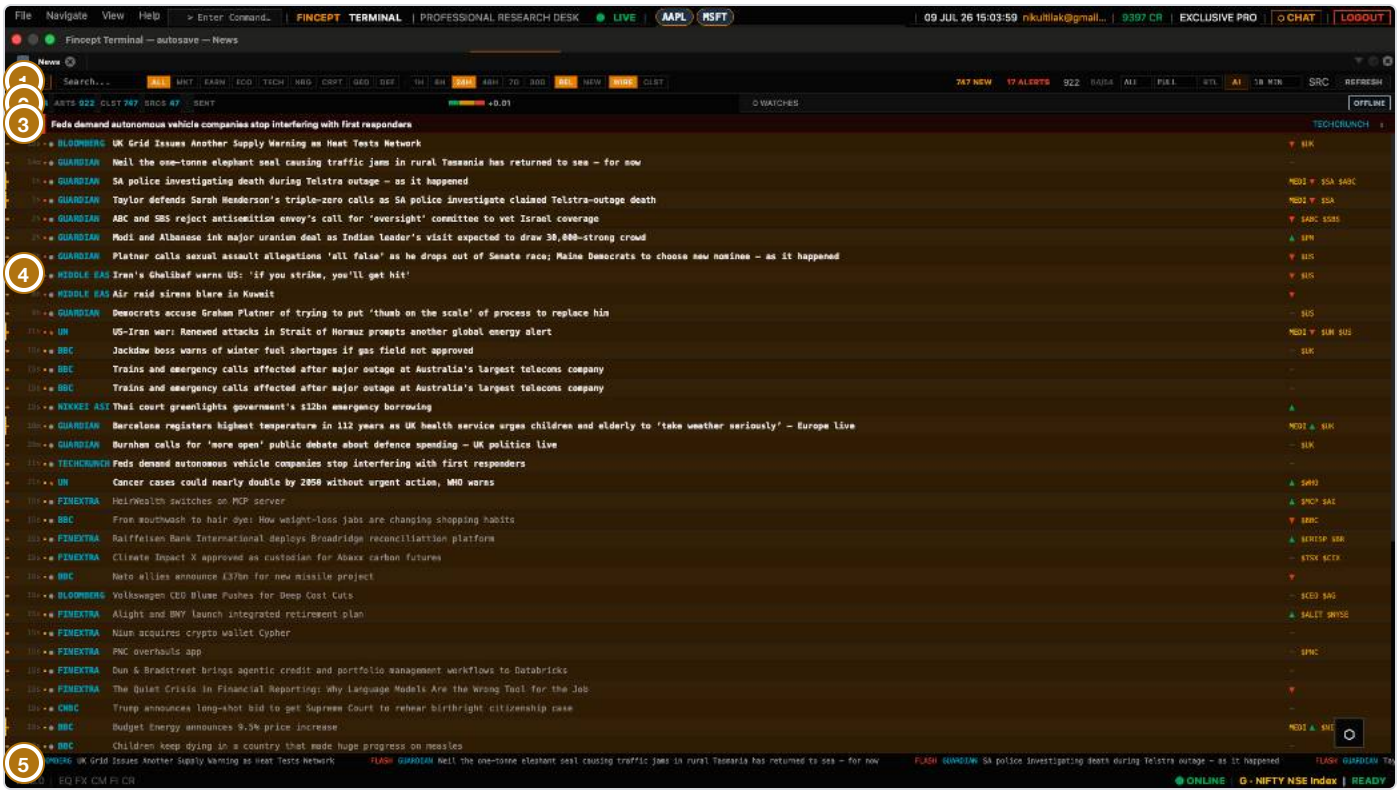
At a glance, you can…

- **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.

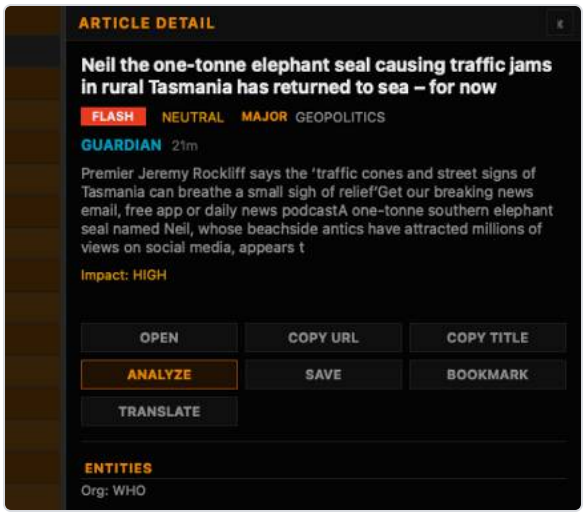
The ticker strip



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.

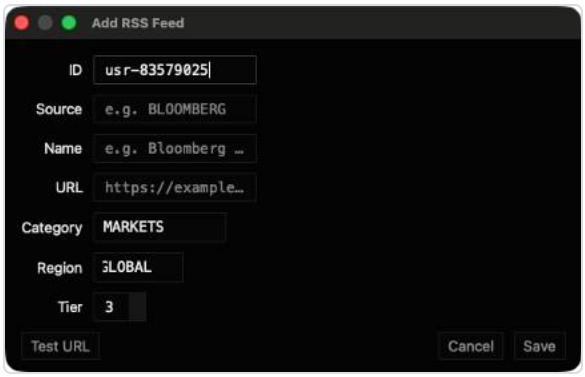


Each feed row: on/off · type (**DEF** built-in / **USR** yours) · source · name · URL · category · region · tier.

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.

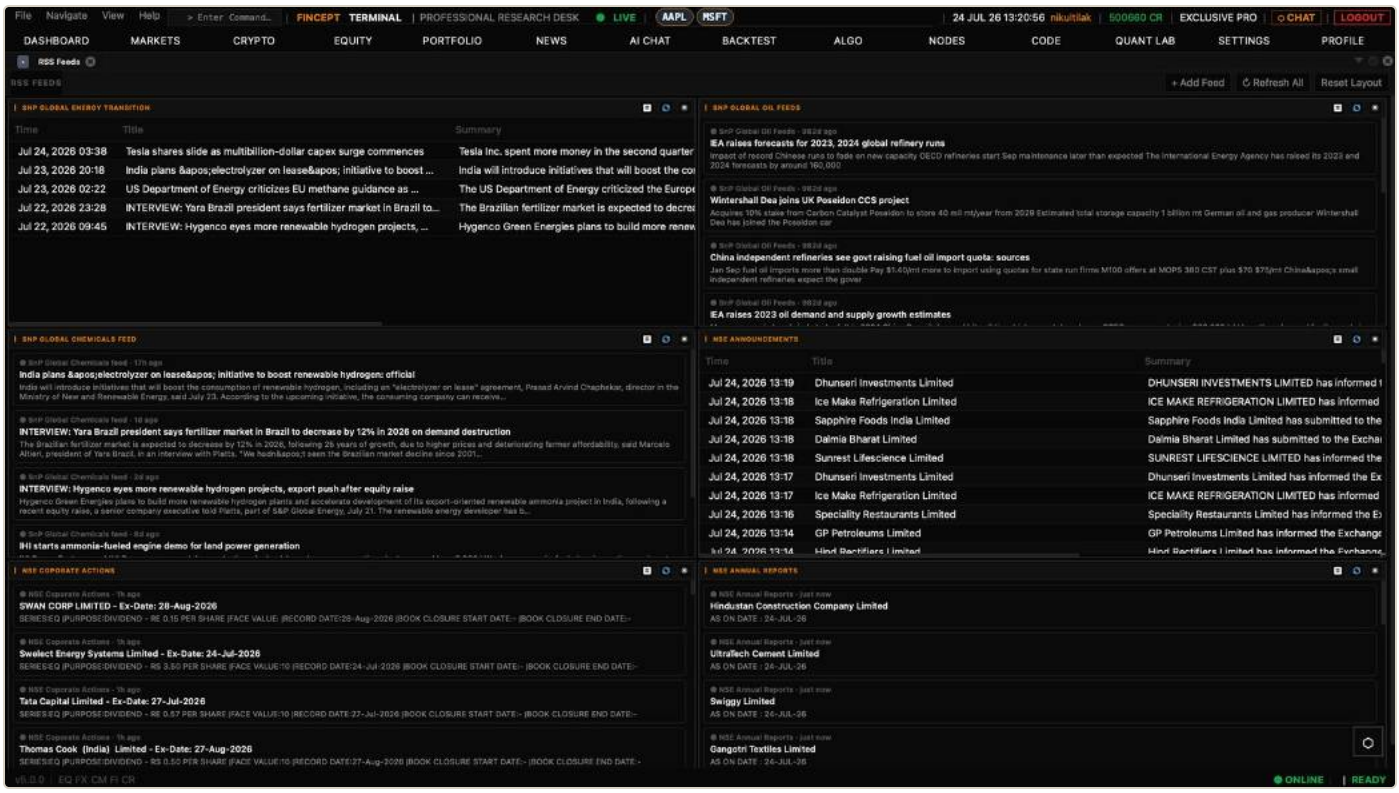


Fig 1. The RSS Feeds board with six live feeds — a mix of card boxes (S&P Global Oil, Chemicals; NSE Corporate Actions, Annual Reports) and table boxes (S&P Global Energy Transition; NSE Announcements).

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.

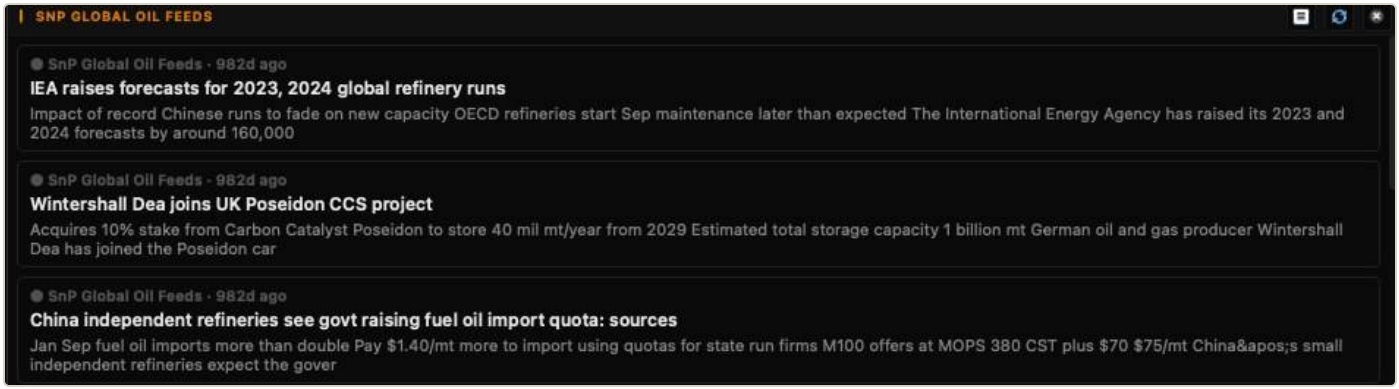


Fig 3. Card view (S&P Global Oil Feeds). Click anywhere on a card to open the article in your default browser.

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.

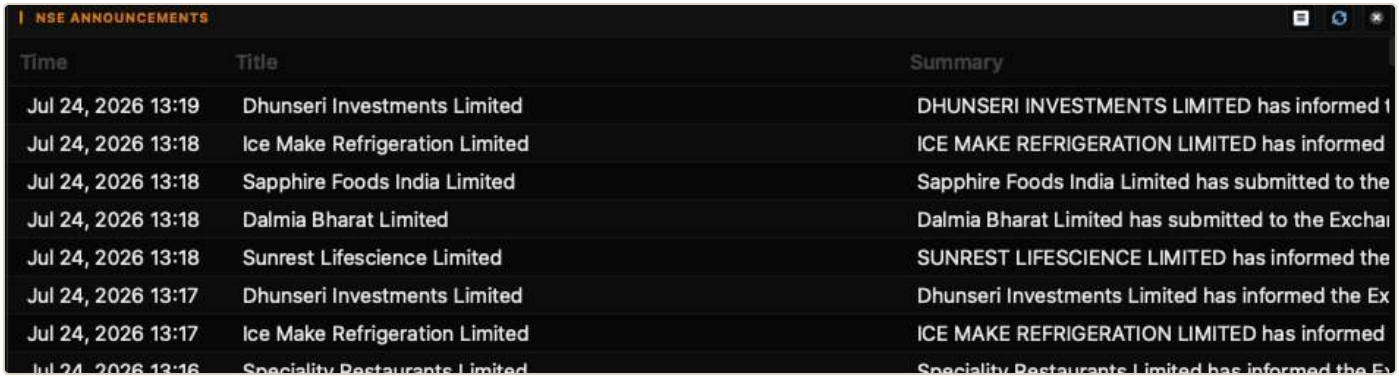


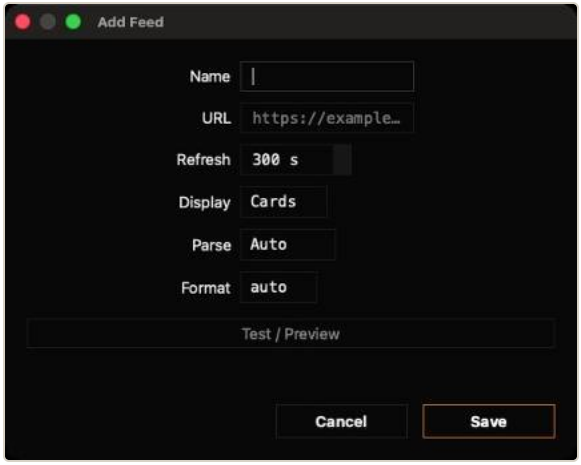
Fig 4. Table view (NSE Announcements) with the full timestamp and Summary column. Wide columns are capped and scroll horizontally, and every cell has a full-text tooltip so nothing is lost to truncation. Double-click a row to open its link.

> **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

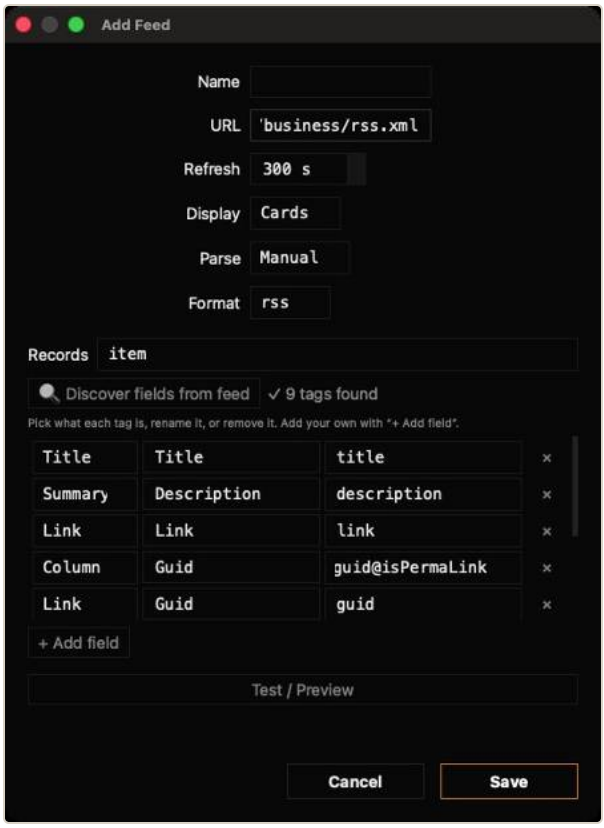
Path	When to use it	What you do
Auto	Standard RSS / Atom / JSON feeds with the usual fields.	Paste the URL, optionally press **Test / Preview**, then **Save**. The app extracts title, summary, link and date for you.
Manual	Unusual XML/JSON, or when you want specific columns.	Set **Parse = Manual**, press **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:

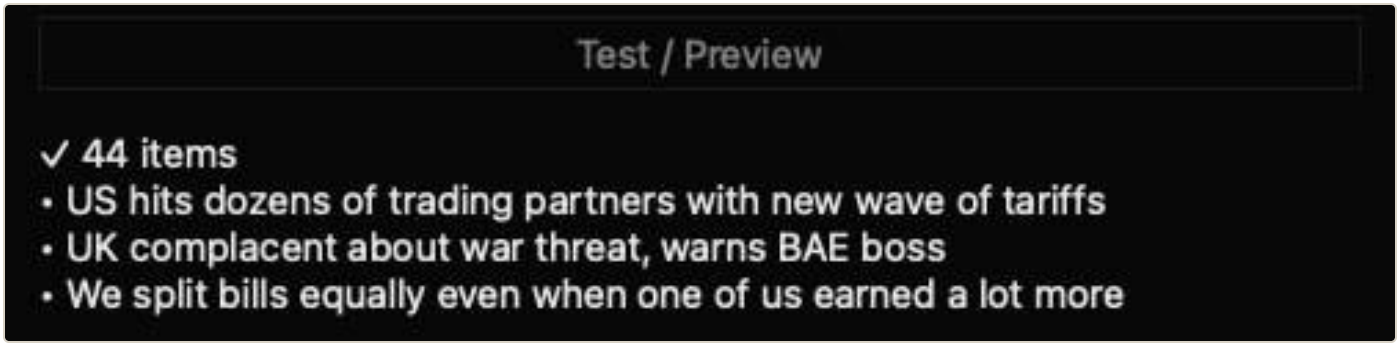


Fig 7. A successful test — **✓ 44 items** and three sample headlines. If the test passes, these items are re-used when you Save so the box paints instantly.

> **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.

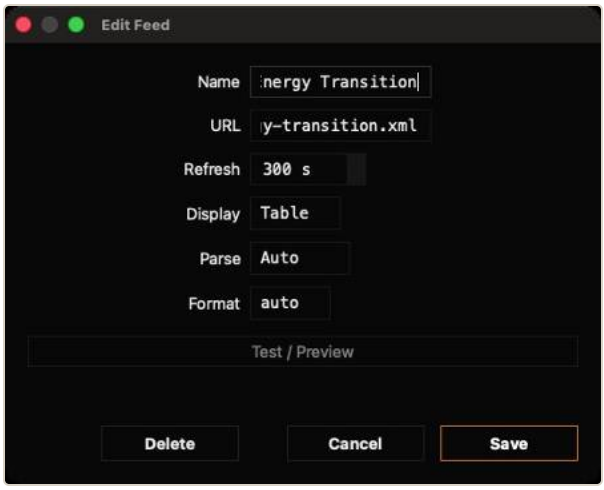
Working with mapping rows

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 — a live world map of geopolitical events, plus humanitarian data, a relations reference and a trade-policy workbench.

`>_ events · HDX data · relations · trade`

Covers · **the event map** · **filters & legend** · **the events table**
HDX data · **relations** · **trade policy** · credits & status

Demonstrated live on the Fincept Terminal Geopolitics tab.

Contents

12 SECTIONS

GETTING ORIENTED

1 What this tab is 3

2 Anatomy of the screen 4

3 Filters & legend 5

MONITORING EVENTS

4 The conflict map 6

5 The events table 8

6 The intelligence sidebar 8

7 HDX Data 11

8 Relations 13

9 Trade 14

REFERENCE

10 Credits — what costs what 16

11 Status & message reference 17

12 Quick reference — how do I… 18

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.

1 · What this tab is

Geopolitics turns a stream of world news into mapped, categorised *events* — who did what, where, and when. You filter by country, city or category; the map, the table and the statistics all describe the slice you pulled.

AT A GLANCE YOU CAN

- **See world events on a map**, colour-coded by category — armed conflict, protests, diplomacy, elections and more.
- **Filter by country, city or category** and reload the whole picture.
- **Read the event list** — headline, place and timestamp, sortable on every column.
- **Inspect one event** — full headline, source and date, with a link out to the original article.
- **Read the shape of the feed** — top categories, and the countries generating the most events.
- **Browse humanitarian datasets** from HDX (the Humanitarian Data Exchange).
- **Consult a conflict reference network** — active crises and the agencies working them.
- **Model trade policy** — tariffs, quotas and subsidies, computed locally.

Opening the tab

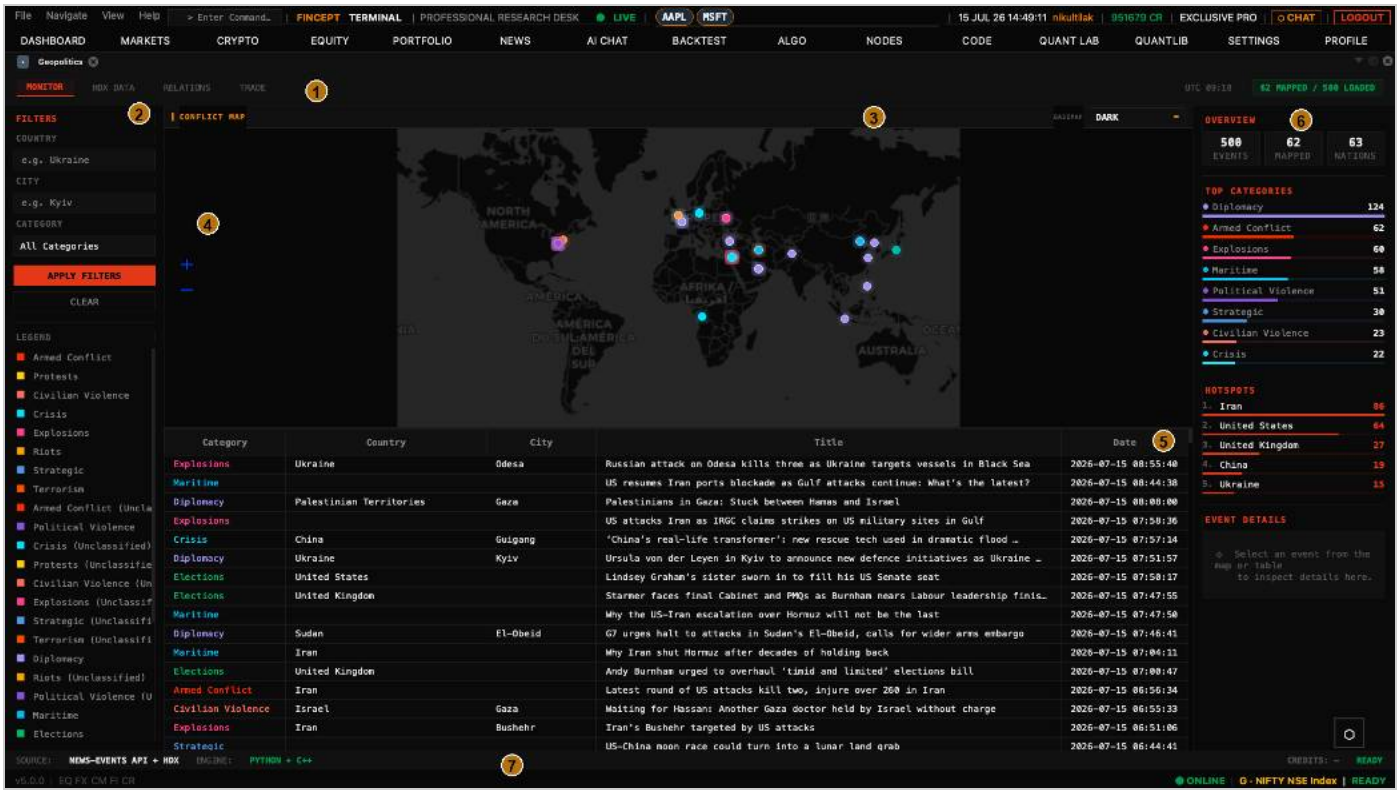
Geopolitics is **not on the top tab bar**. The usual ways in:

ROUTE	HOW
Menu	**Navigate** → **Research & Intelligence** → **Geopolitics**
Command bar	Type `geopolitics` — `geo` and `politics` also work
Component browser	`Ctrl+K` → category **Geopolitics**
Quick switch	**View** → **Quick Switch** → **Geopolitics View** — a 2×2 of Geopolitics, Maritime, Relationship Map and News

> **ENTITLEMENT**
>
> Geopolitics requires **Exclusive+** or higher. On a lower plan the tab still opens, but the content is replaced by a lock card reading `Requires Exclusive+` with an `UPGRADE PLAN` button. Upgrading swaps the real screen in without a restart.

> **THIS IS THE MOST EXPENSIVE TAB IN THE TERMINAL**
>
> Every reload pulls five pages of events at roughly **3 credits a page — about 15 credits**. Opening the tab spends that automatically, and **a timer repeats it every 5 minutes for as long as the tab is on screen**, whether or not you touch anything. Leaving it open for an hour costs on the order of 180 credits. Read **§10** before you settle in.

2 · Anatomy of the screen



The Geopolitics tab, MONITOR view — 500 events loaded, 62 of them mapped.

#	REGION	WHAT IT DOES
1	**View tabs**	Switch between MONITOR, HDX DATA, RELATIONS and TRADE. Also a UTC clock and the mapped/loaded count.
2	**Filter panel**	Country, city and category filters — plus the category legend. Shared by every view.
3	**Map toolbar**	The basemap picker.
4	**Conflict map**	One pin per geolocated event, coloured by category.
5	**Events table**	One row per event: category, place, headline, timestamp.
6	**Intelligence sidebar**	Overview counts, top categories, hotspot countries, and the selected event's detail.
7	**Status bar**	Data source, engine, credit balance, and current state.

The four views



The view tabs, the UTC clock, and the mapped/loaded badge.

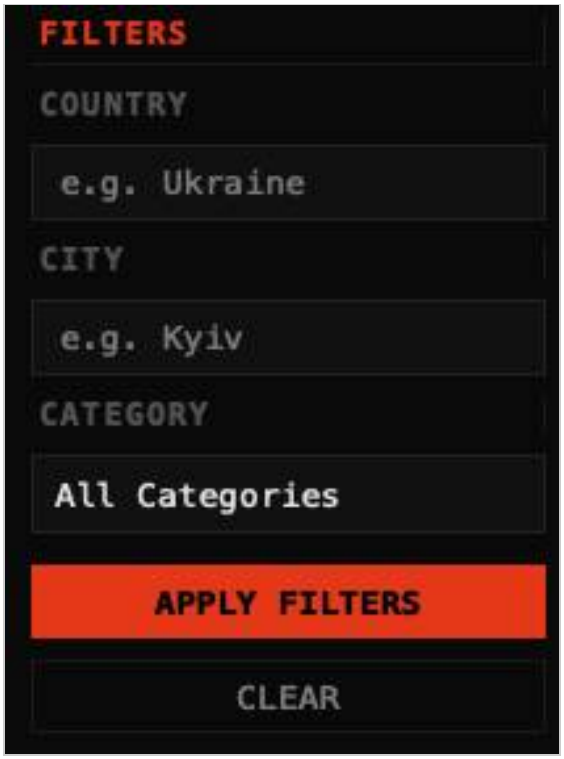
VIEW	CONTENTS
`MONITOR`	The live event map, table and sidebar. This is the tab proper.
`HDX DATA`	Humanitarian dataset search against the Humanitarian Data Exchange.
`RELATIONS`	A fixed reference chart of active conflicts and agencies. **Not live** — see §8.
`TRADE`	A local trade-policy calculator. No network, no credits.

The badge on the right reads `%1 MAPPED / %2 LOADED`. Hover it for the full breakdown (on map, loaded, total in API). The filter panel stays put across all four views; the map and table sit in a splitter you can drag.

MAPPED IS ALWAYS FAR BELOW LOADED

Only about one event in four carries coordinates, so a 500-event pull typically maps 60–120 pins. That is the feed, not a fault. The badge turns **red only when nothing at all could be geolocated** — which still doesn't mean the load failed.

3 · Filters & legend



CONTROL	WHAT IT DOES
`COUNTRY`	Free text, e.g. `Ukraine`. **`Enter` reloads immediately** — **~15 credits**.
`CITY`	Free text, e.g. `Kyiv`. **`Enter` reloads** — **~15 credits**.
`CATEGORY`	Dropdown, `All Categories` plus every category with its API-wide count. Selecting does nothing on its own.
`APPLY FILTERS`	Reloads using all three. **~15 credits.**
`CLEAR`	Empties the fields **and reloads**. **~15 credits.**

CLEAR IS NOT FREE

It looks like a local reset, but it wipes the fields *and then runs a full five-page reload* — the same ~15 credits as `APPLY FILTERS`. So does pressing `Enter` in either text field. There is no way to clear the filters without paying for a reload.

Legend



Every category the feed knows, with its colour and its count. The colours are what you see on the map and in the table's `Category` column.

The legend is **reference only — the rows are not clickable filters**. To narrow by category use the `CATEGORY` dropdown, then `APPLY FILTERS`.

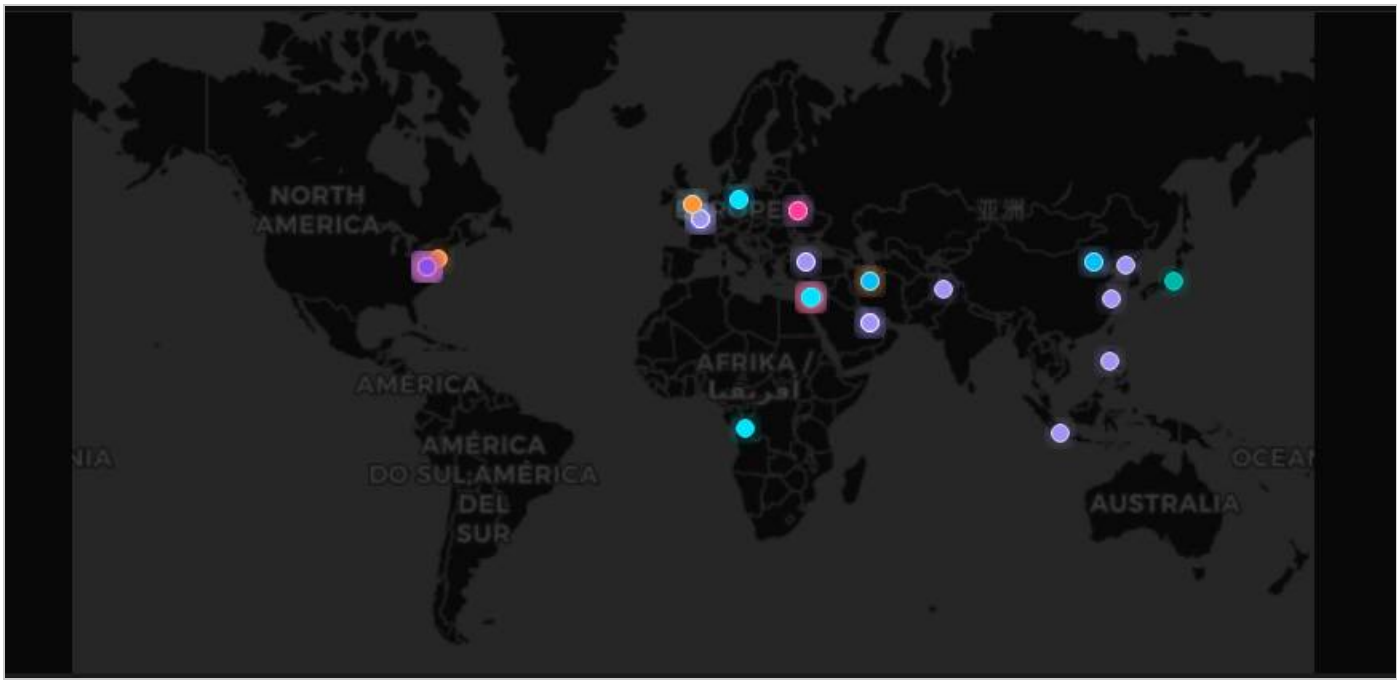
> **WHY "ARMED CONFLICT" AND "ARMED CONFLICT (UNCLASSIFIED)" BOTH EXIST**
>
> The upstream feed tags some events confidently and leaves others unclassified, and both forms reach the legend. They are the **same colour on the map** — the terminal folds the unclassified variant onto its base colour so map, legend and table stay coherent.

> **THE LEGEND'S COUNTS AND THE SIDEBAR'S COUNTS DISAGREE — BOTH ARE RIGHT**
>
> Legend counts are **API-wide totals** across the whole feed. The sidebar's **TOP CATEGORIES** counts only the events *you have loaded*. Two different numbers for the same category on one screen is expected. The legend also refreshes on a slower cycle, so it can lag the events by a few minutes.

4 · The conflict map



The map toolbar — a title and the basemap picker. There are no other map controls.



One pin per geolocated event, coloured by category. Zoom controls sit at the left edge.

INTERACTION	RESULT
Drag	Pan.
Wheel / `+` `−`	Zoom.
Hover a pin	Tooltip: category, city and country.
Click a pin	Selects that event's row in the table and fills **EVENT DETAILS**. Free.

Unlike the Maritime chart, there is **no area drawing here** — no square, no circle. The map re-frames itself to fit the pins after every load, and again shortly after you drag the splitter, so the camera will move on its own.



BASEMAPS

Six tile layers — `SATELLITE`, `DARK`, `OCEAN`, `STREETS`, `LIGHT`, `TERRAIN`. Geopolitics opens on **`DARK`** (Maritime opens on satellite). Switching keeps your pins and camera. The choice is not remembered across restarts.

> **PINS THAT SHARE A LOCATION ARE DELIBERATELY SPREAD APART**
>
> When several events report the same coordinates, the map fans them out in a small spiral so they don't stack into one dot. Those offset pins are **not at their true positions** — they are a legibility device. Read a cluster as "several events here", never as a precise scatter.

5 · The events table



Category	Country	City	Title	Date
Explosions	Ukraine	Odesa	Russian attack on Odesa kills three as Ukraine targets vessels in Black Sea	2026-07-15 08:55:40
Maritime			US resumes Iran ports blockade as Gulf attacks continue: What's the latest?	2026-07-15 08:44:38
Diplomacy	Palestinian Territories	Gaza	Palestinians in Gaza: Stuck between Hamas and Israel	2026-07-15 08:08:00
Explosions			US attacks Iran as IRGC claims strikes on US military sites in Gulf	2026-07-15 07:58:36
Crisis	China	Guigang	'China's real-life transformer': new rescue tech used in dramatic flood …	2026-07-15 07:57:14
Diplomacy	Ukraine	Kyiv	Ursula von der Leyen in Kyiv to announce new defence initiatives as Ukraine …	2026-07-15 07:51:57
Elections	United States		Lindsey Graham's sister sworn in to fill his US Senate seat	2026-07-15 07:50:17
Elections	United Kingdom		Starmer faces final Cabinet and PMQs as Burnham nears Labour leadership finis…	2026-07-15 07:47:55
Maritime			Why the US-Iran escalation over Hormuz will not be the last	2026-07-15 07:47:50
Diplomacy	Sudan	El-Obeid	G7 urges halt to attacks in Sudan's El-Obeid, calls for wider arms embargo	2026-07-15 07:46:41
Maritime	Iran		Why Iran shut Hormuz after decades of holding back	2026-07-15 07:04:11
Elections	United Kingdom		Andy Burnham urged to overhaul 'timid and limited' elections bill	2026-07-15 07:00:47
Armed Conflict	Iran		Latest round of US attacks kill two, injure over 260 in Iran	2026-07-15 06:56:34
Civilian Violence	Israel	Gaza	Waiting for Hassan: Another Gaza doctor held by Israel without charge	2026-07-15 06:55:33
Explosions	Iran	Bushehr	Iran's Bushehr targeted by US attacks	2026-07-15 06:51:06
Strategic			US-China moon race could turn into a lunar land grab	2026-07-15 06:44:41

One row per event. Category is colour-coded to match the map and legend.

COLUMN	MEANING
`Category`	Event type, coloured to match its pin. Unclassified variants are spelled out, e.g. `Armed Conflict (Unclassified)`.
`Country`	As reported. Often blank.
`City`	As reported. Often blank.
`Title`	The headline. Hover for the full text when it is clipped.
`Date`	When the event was extracted, `YYYY-MM-DD HH:MM:SS`.

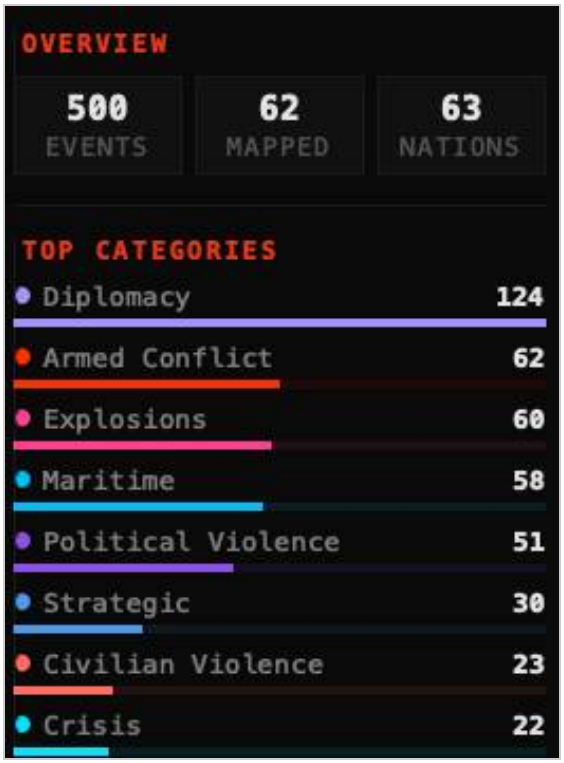
Every column sorts — click a header. Sorting is local to the events already loaded; it never re-queries, so it is free. Rows are not capped: a full pull renders all ~500.

> **SELECTING A ROW IS WHAT DRIVES THE DETAIL PANEL**
>
> Click any row — or arrow-key through the list — and **EVENT DETAILS** rewrites itself to match. It costs nothing, but there is no "back": the previously inspected event is simply gone. Clicking a map pin does the same thing from the other direction, and scrolls the table to that row.

> **READING THE DATA HONESTLY**
>
> These are news articles turned into events, so the seams show. Expect blank `Country` / `City`, headlines that are only loosely "geopolitical", the same story from several outlets, and categories that look arguable. Three quarters of rows have no coordinates and so never appear on the map.

6 · The intelligence sidebar

Four stacked blocks on the right. All of them describe *the events you have loaded*, not the world.



OVERVIEW

`EVENTS` loaded · `MAPPED` pins drawn · `NATIONS` distinct countries present.

TOP CATEGORIES

The **eight** commonest categories in the loaded slice, with bars scaled to the largest. Compare with the legend's totals — see §3.



HOTSPOTS

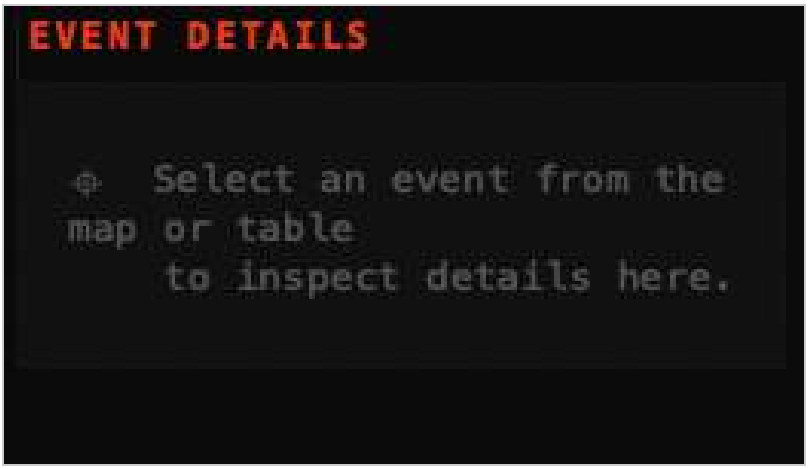
The **top five** countries by event count, ranked. This is a measure of *news volume*, not severity — a heavily reported election out-ranks a quiet conflict.



EVENT DETAILS

The selected event: a category badge, the full headline, country, city, date and source. Blank fields show `—`; a missing headline shows `Untitled event`.

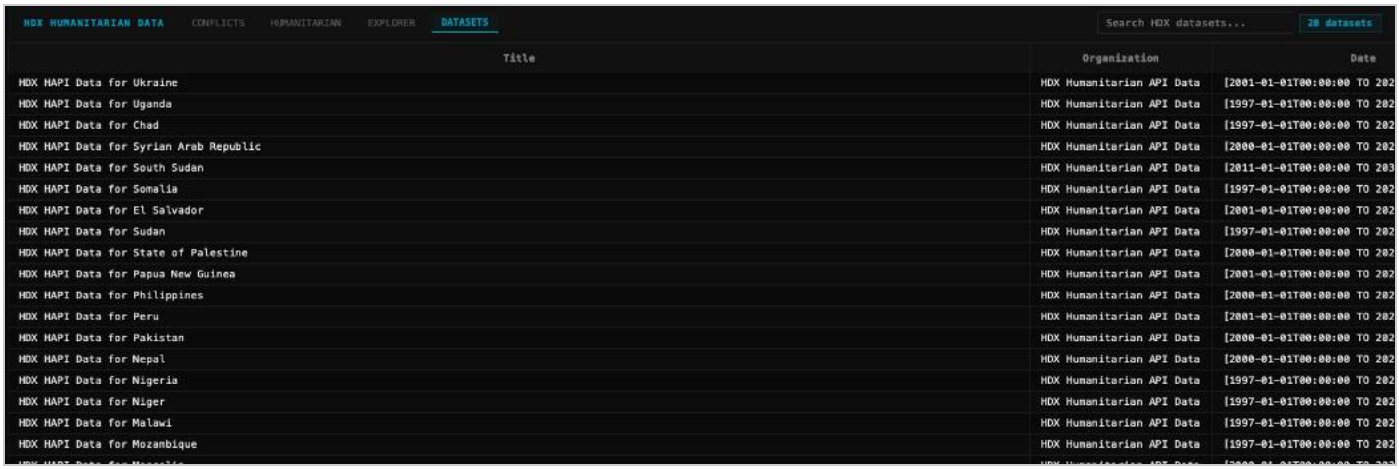
`OPEN SOURCE ↗` opens the original article **in your web browser**. It only appears when the event carries a link.



Before you select anything.

7 · HDX Data

A search across the **Humanitarian Data Exchange** — the UN's open dataset catalogue. This view talks straight to HDX, so it costs **no credits**; it just takes a few seconds.



The DATASETS view — HDX HAPI country data packages.

VIEW	WHAT IT SEARCHES
`CONFLICTS`	Datasets tagged with HDX's conflict-and-peace topic.
`HUMANITARIAN`	Datasets tagged with humanitarian access and aid workers.
`EXPLORER`	Search by country or topic yourself. Opening it runs nothing.
`DATASETS`	A broad standing query — humanitarian, crisis, conflict, displacement.

COLUMN	MEANING
`Title`	Dataset name. Hover for its description.
`Organization`	Who publishes it.
`Date`	The period the data covers.
`Resources`	How many files are attached.
`Tags`	HDX topic tags.

Results are capped at **20** and sorted newest-first. Type in `Search HDX datasets...` and press `Enter` for a free-text search. Selecting a row does nothing — this table is a reading list, not a launcher.



EXPLORER

A `COUNTRY:` dropdown (Ukraine, Gaza, Sudan, Yemen, Syria, Afghanistan, Myanmar, Ethiopia, Haiti, Somalia, Venezuela) and a `TOPIC:` dropdown (conflict, humanitarian, displacement, food security, health, education, refugees), then `SEARCH`. **Country wins if both are set.**

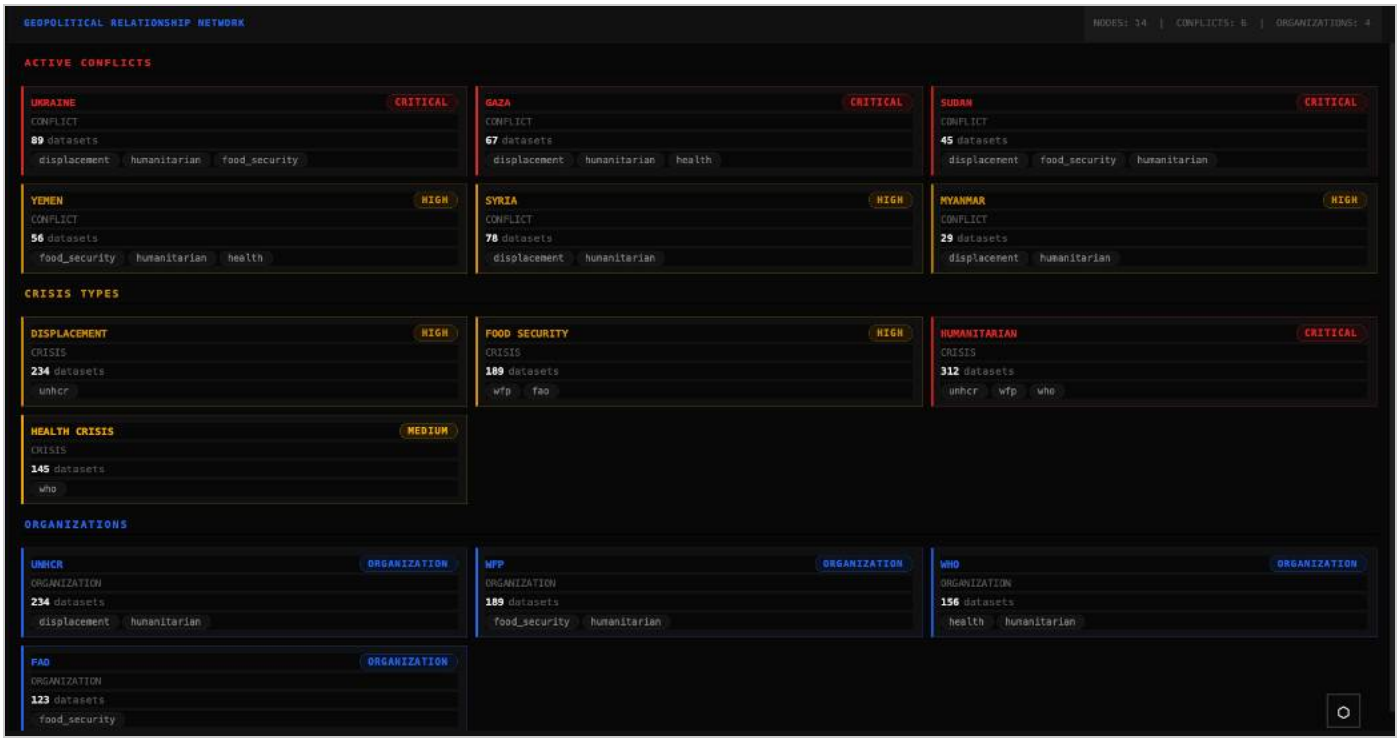
KNOWN LIMITATIONS IN THIS VIEW

- **`CONFLICTS` and `HUMANITARIAN` commonly return `0 datasets`** — their topic queries frequently match nothing in HDX's catalogue. An empty table here is not a network failure.
- **Explorer's country search is unreliable** — it queries HDX using the country's full name where the catalogue expects a three-letter code, so it usually returns nothing. **Use `TOPIC:`, or the free-text search box, instead.**
- `DATASETS` is the view most likely to return results.

A search takes a few seconds and shows `Loading HDX data...` in place of the table.

8 · Relations

> **READ THIS FIRST — RELATIONS IS A FIXED REFERENCE CHART, NOT LIVE DATA**
>
> Every card, every severity badge and every dataset count on this view is **hard-coded into the application**. Nothing here is fetched, nothing refreshes, and nothing responds to your filters. It will look identical tomorrow, next month, and after any world event. Treat it as a printed wall chart that ships with the terminal — **never as a current picture of a crisis**, and never as a source for a number you intend to act on.



The relationship network: active conflicts, crisis types, and organizations.

SECTION	CARDS
`ACTIVE CONFLICTS`	Ukraine, Gaza, Sudan, Yemen, Syria, Myanmar — each with a severity badge and related themes.
`CRISIS TYPES`	Displacement, food security, humanitarian, health — with the agencies that own them.
`ORGANIZATIONS`	UNHCR, WFP, WHO, FAO — and the themes each covers.

Each card shows a severity or type badge (`CRITICAL`, `HIGH`, `MEDIUM`, `ORGANIZATION`), an indicative dataset count, and up to three connection pills, with `+N` when there are more. The header counts — `NODES`, `CONFLICTS`, `ORGANIZATIONS` — are constants.

There is nothing to click. No card opens, no node selects, no detail panel exists. Use it to orient yourself on who works which crisis, then go to **HDX DATA** for the actual datasets.

9 · Trade

A trade-policy calculator. It runs entirely on your machine — **no network, no credits** — and answers textbook questions about tariffs, quotas and trade blocs from numbers you supply.



Pick an analysis, set the inputs, press RUN ANALYSIS.



The four analyses. The dropdown and the tab strip are two views of the same choice.

ANALYSIS	ANSWERS
`Benefits & Costs of Trade`	Efficiency gains, consumer benefits, growth effects and adjustment costs of trade openness.
`Trade Restrictions Analysis`	The economic impact of tariffs, quotas, export subsidies and non-tariff barriers.
`Trading Blocs Analysis`	Trade creation versus diversion for regional blocs and unions.
`Trade Barrier Removal Impact`	FDI, employment, wage and GDP effects of liberalisation.

> **ONLY TWO OF THE FOUR ANALYSES ACTUALLY USE YOUR INPUTS**
>
> **`Benefits & Costs of Trade` and `Trade Restrictions Analysis` work as documented.**
>
> **`Trading Blocs Analysis` and `Trade Barrier Removal Impact` do not.** Both hand your figures to the wrong calculation, which ignores them and substitutes its own defaults. They will return a confident-looking table of numbers that has *nothing to do with what you typed* — and changing the inputs will not change the answer. **Do not rely on those two tabs.** A quick way to confirm it for yourself: run one, change an input substantially, run it again, and watch the results stay identical.

Inputs

BENEFITS & COSTS	DEFAULT
`TRADE VOLUME` — trade as % of GDP (world avg ~60%)	30%
`EXPECTED PRICE REDUCTION` — consumer price fall from openness	5%
`TRADED GOODS SHARE` — share of consumption that is tradeable	30%

RESTRICTIONS	DEFAULT
`TARIFF RATE` — import duty (10% ≈ standard MFN)	10%
`QUOTA VOLUME` — annual import quantity limit	1000 units
`EXPORT SUBSIDY RATE` — subsidy as % of export value	5%
`DEVELOPMENT LEVEL` — developed / developing / middle income	Developed
`INDUSTRY MATURITY` — mature / infant / emerging	Mature

Results appear below the inputs under `ANALYSIS RESULTS` as a `Metric` / `Value` table; the status line reports how many fields came back. Errors surface in a red block. Nothing here is saved — inputs reset when you restart.

10 · Credits — what costs what

One action drives the whole bill: **reloading events**. It pulls five pages at about 3 credits each — call it **15 credits** — and several controls trigger it without saying so.

Spends ~15 credits

ACTION	NOTES
Opening the tab	Automatic on first show. Unavoidable.
The 5-minute auto-refresh	🔴 Fires on its own while the tab is visible. ~180 credits/hour idle.
`APPLY FILTERS`	—
`CLEAR`	Reloads after clearing.
`Enter` in `COUNTRY`	No button press needed.
`Enter` in `CITY`	No button press needed.

A narrow filter costs less — the pull stops early when there are no more pages.

> **THE CREDITS READOUT OFTEN STAYS BLANK**
>
> The status bar is designed to show `CREDITS: N` after each load — green above 100, amber below 100, red below 20. In practice it frequently reads `–` throughout a session, because the service does not always return a balance. **Treat the counts in this section as the authority on cost, not the readout.**

Free

Everything in **HDX DATA** · everything in **RELATIONS** · everything in **TRADE** · the `BASEMAP` picker · panning and zooming · clicking a pin · selecting a row · sorting the table · the `CATEGORY` dropdown · switching views.

> **THE SINGLE MOST IMPORTANT HABIT**
>
> **Don't leave this tab open in the background.** The refresh timer only runs while the tab is on screen — switching to any other tab stops it immediately, and coming back does *not* trigger a fresh pull. Park the terminal on Dashboard, not here.

> **OTHER WAYS TO SPEND LESS**
>
> - **Filter before you browse.** One well-aimed `APPLY FILTERS` beats three exploratory ones.
> - **Sorting and selecting are free.** Re-read what you already pulled rather than reloading.
> - **Use HDX and Trade freely** — they cost nothing.
> - **Repeating a search does not hit a cache.** Every reload pays full price, even an identical one moments later.

> **NOTHING HERE IS DESTRUCTIVE**
>
> The tab never changes server data and never deletes anything. The only thing it writes is your own screen state — the active view and the country/city text. The one outward action is `OPEN SOURCE ↗`, which opens an article in your browser. The cost of a mistake is credits.

11 · Status & message reference



The status bar: source, engine, credits, state.

FIELD	MEANING
`SOURCE`	`NEWS-EVENTS API + HDX` — a fixed label naming the feeds, not a live health check.
`ENGINE`	`PYTHON + C++` — also a fixed label.
`CREDITS`	Your remaining balance, colour-coded, once the service reports one. Commonly stays `–` — see §10.

Status messages

MESSAGE	MEANING
`READY`	Idle, or a load finished.
`LOADING...`	A reload is in flight — five pages, so give it time.
`ERROR`	The request failed. No detail is shown on screen.

Panel messages

MESSAGE	WHERE
`Waiting for events…`	Top categories / hotspots, before the first load.
`No events match the current filters.`	Top categories, when a filter matched nothing.
`No country data.`	Hotspots, when nothing carries a country.
`Select an event from the map or table to inspect details here.`	Event details, nothing selected.
`Untitled event`	Event details, headline missing.
`Loading HDX data...`	HDX, search in flight.
`0 datasets`	HDX, no matches — common on CONFLICTS and HUMANITARIAN.
`Analyzing...`	Trade, calculation running.

TWO SILENCES WORTH KNOWING ABOUT

`ERROR` tells you nothing more. The reason is written only to the application log; the status bar is the whole message.

A partly-failed load still says `READY`. If some of the five pages fail, the tab renders what arrived and reports success. If the map looks thinner than usual, press `APPLY FILTERS` to pull again.

12 · Quick reference: how do I…

HOW DO I…	DO THIS
Open the tab	Command bar → `geo`. Or Navigate → Research & Intelligence → Geopolitics.
See events for one country	Type it in `COUNTRY` → `APPLY FILTERS`. **~15 cr**
See only one kind of event	`CATEGORY` dropdown → `APPLY FILTERS`. **~15 cr**
Reset the filters	`CLEAR` — but it reloads too. **~15 cr**
Read one event in full	Click its row, or its pin. Free.
Open the original article	Select the event → `OPEN SOURCE ↗`. Opens your browser.
Find the newest events	Click the `Date` header to sort. Free.
See which countries dominate	Read `HOTSPOTS` in the sidebar.
Change the map look	`BASEMAP` → one of six. Free.
Find humanitarian datasets	`HDX DATA` → `DATASETS`, or type in the search box. Free.
Model a tariff	`TRADE` → `Trade Restrictions Analysis` → `RUN ANALYSIS`. Free.
Stop spending credits	**Switch to another tab.** The refresh timer stops the moment this one is hidden.

Worth remembering

- **There are no keyboard shortcuts** on this tab, no right-click menus and no double-click actions. `Enter` works in the filter fields and the HDX search box — and in the filter fields **it costs credits**.
- **Only the active view and the country/city text are remembered** across restarts — and a remembered filter silently narrows the next load. Everything else resets: basemap, sorting, selection, HDX view, all Trade inputs.
- **Every number in the sidebar describes your slice**, not the world. The legend's counts describe the whole feed. They will not agree.
- **Mapped will always be much lower than loaded.** Most events have no coordinates.
- **Relations is a fixed chart** and **two of the four Trade analyses ignore your inputs** — see §8 and §9 before trusting either.

EXCLUSIVE · INSTITUTIONAL

FINCEPT TERMINAL · PRODUCT GUIDE

Maritime Intelligence

A complete, screen-by-screen walkthrough of the Maritime tab — live vessel tracking on a world map, by sea area or single hull, with voyage replay and port lookup.

`>_ vessels · sea areas · voyages · ports`

Covers · **loading the fleet** · **the vessel map** · **the vessel table** **search & voyage history** · credits & metering · status reference

Demonstrated live on the Fincept Terminal Maritime tab.

Contents

11 SECTIONS

GETTING ORIENTED

1 What this tab is 3

2 Anatomy of the screen 4

3 Top bar 4

USING THE MAP

4 Left panel — load, draw, intelligence 5

5 Map toolbar & the live vessel map 7

6 The vessel table 9

7 Right panel — search 10

8 Voyage history 13

REFERENCE

9 Credits — what costs what 14

10 Status & message reference 15

11 Quick reference — how do I… 17

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.

1 · What this tab is

Maritime Intelligence answers one question: *what ships are where, and where are they going?* You give it a patch of ocean — or a single IMO number — and it plots the vessels, tabulates their positions, and summarises the fleet it just loaded.

AT A GLANCE YOU CAN

- **Load a global sample** of live vessels — up to 200 pins spread across the world.
- **Search any sea area** — draw a box or circle on the map, type raw coordinates, or pick a named place.
- **Track one ship** by IMO number and drop a pin on its last reported position.
- **Replay a voyage** — plot a vessel's recent positions as a fading trail.
- **Read the fleet at a glance** — how many are loaded, how many are moving, average speed, distinct routes and destination ports.
- **See the busiest trade corridors** in the set you loaded.
- **Find places and ports** — geocode a strait or a harbour and plot it.
- **Switch basemaps** — satellite, dark, ocean, streets, light or terrain.

Opening the tab

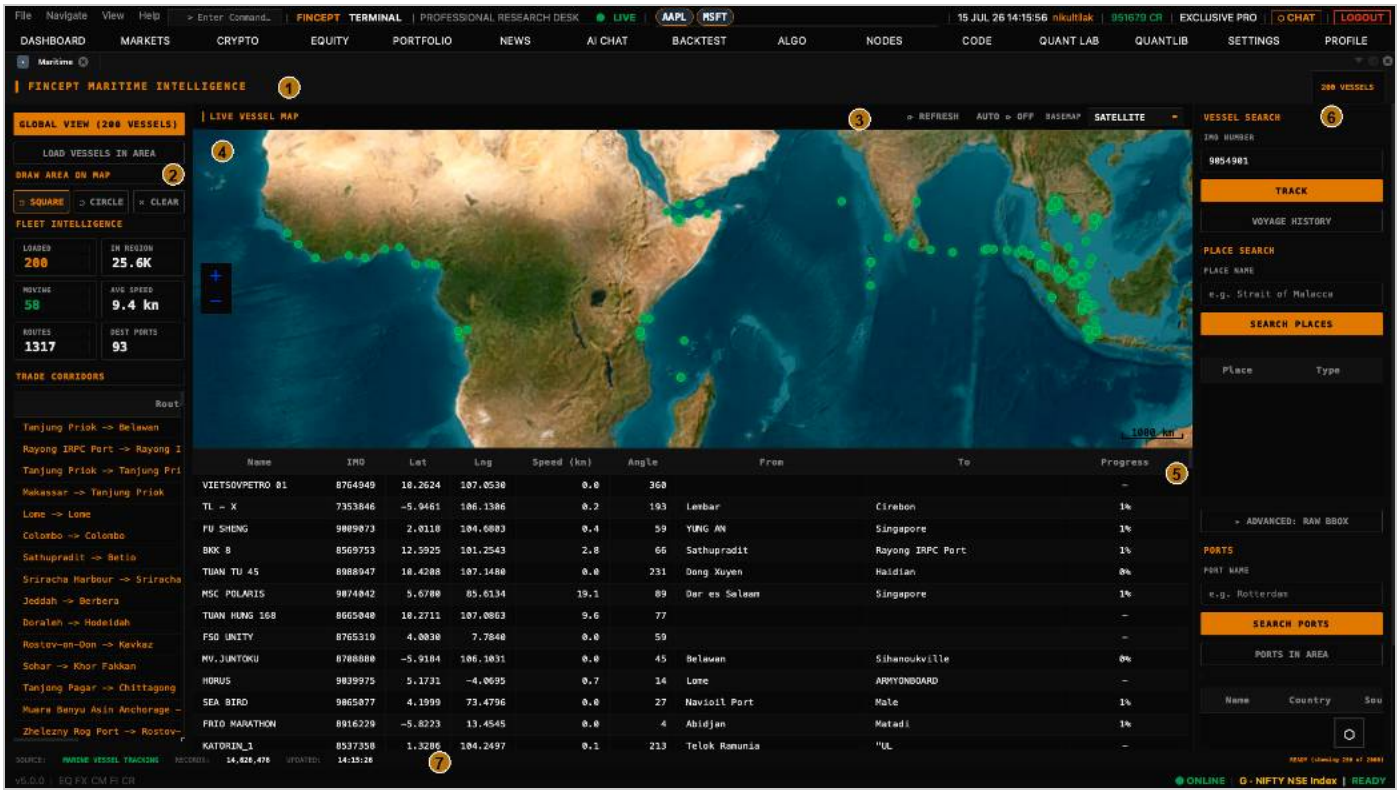
Maritime is **not on the top tab bar**. There are two ways in:

ROUTE	HOW
Menu	**Navigate → Research & Intelligence → Maritime**
Command bar	Click the command bar and type `maritime` — `marine` and `shipping` also work

> **ENTITLEMENT**
> Maritime requires an **Exclusive Pro** plan. On a lower tier the menu entry still appears but is marked with a lock and the screen is replaced by an upgrade panel. Upgrading swaps in the real screen without a restart.

> **OPENING THIS TAB SPENDS CREDITS**
> The first time the tab is shown it automatically loads vessels — an area search that costs **8 credits**. This is unavoidable and happens before you click anything. See **§9** for the full cost model.

2 · Anatomy of the screen



The Maritime tab with a global sample loaded — 200 vessels, satellite basemap.

#	REGION	WHAT IT DOES
1	**Top bar**	Branding, and a live count of the vessels currently plotted.
2	**Left panel**	Load actions, map draw tools, fleet statistics, and trade corridors.
3	**Map toolbar**	Refresh, the auto-refresh toggle, and the basemap picker.
4	**Live vessel map**	The chart itself — pan, zoom, draw areas, click pins.
5	**Vessel table**	One row per loaded vessel: position, speed, heading, route.
6	**Right panel**	Search by IMO, by place name, by raw bounding box, or by port.
7	**Status bar**	Data source, total records, last update, and current operation.

3 · Top bar



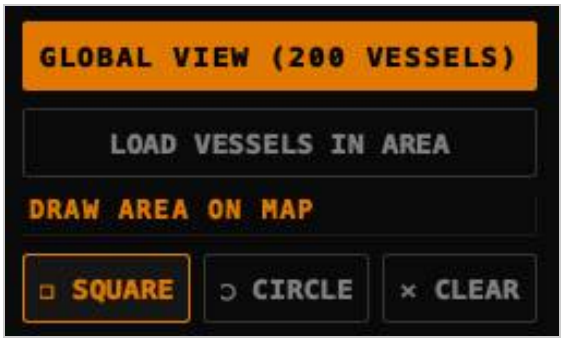
The title, and the live vessel-count badge on the right.

The badge on the right reads `– VESSELS` until something is loaded, then shows the number of vessels *actually plotted on the map* (large numbers are abbreviated, e.g. `1.2K`). This is the same figure as **LOADED** in the left panel — not the size of the region you searched.

The map and the vessel table sit in a **vertical splitter** — drag the divider between them to give either one more room. The right panel scrolls if the window is short.

4 · Left panel — load, draw, intelligence

Load actions & draw tools



CONTROL	WHAT IT DOES
`GLOBAL VIEW`	Loads a worldwide sample. **8 credits.**
`LOAD VESSELS IN AREA`	Searches using the coordinates in **ADVANCED: RAW BBOX**. **8 credits.**
`□ SQUARE`	Arms rectangle drawing. Drag on the map to define a box.
`○ CIRCLE`	Arms circle drawing. Drag from the centre outwards.
`× CLEAR`	Disarms both and removes the shape. Free — and it leaves your loaded vessels on screen.

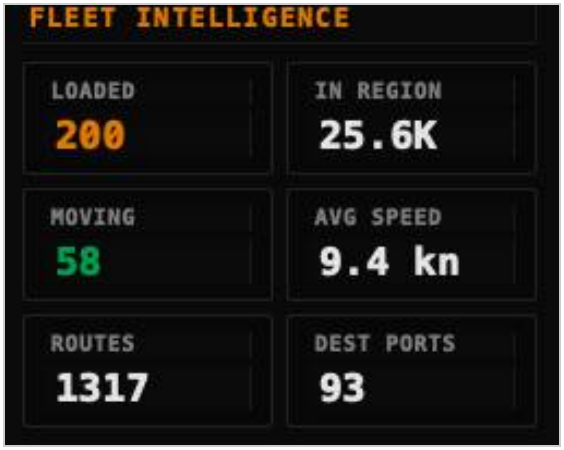
GLOBAL VIEW IS A SPREAD, NOT A "TOP 200"

It asks the API for 2,000 vessels, then bins them into a 10×10 latitude/longitude grid and keeps at most two per cell, up to 200. The API returns busy shipping lanes first, so a naive cut would clump everything into the Malacca Strait. What you see is a *geographic spread*, not the 200 busiest ships — and it is re-sampled on each call, so counts and corridors change between loads.

FINISHING A DRAG RUNS A SEARCH

With a draw tool armed, releasing the mouse immediately fires an area search — **8 credits**, no confirmation. While a tool is armed the cursor becomes a crosshair and *dragging pans nothing*; the wheel still zooms. Accidental micro-shapes are discarded silently (a circle must exceed ~1 km radius, a box ~0.01° on both axes).

Fleet intelligence



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 loaded.
`DEST PORTS`	Distinct destination ports loaded.

All six read `—` until data arrives.

> **LOADED VS IN REGION**
>
> These are deliberately different numbers. **LOADED** is what is drawn (≤200); **IN REGION** is the true population of the searched area. Seeing `200` next to `25.6K` is normal and correct — you are looking at a sample.

Trade corridors



COLUMN	MEANING
`Route`	`<from> -> <to>`.
`Vessels`	How many loaded vessels are on that corridor. **Not** trade value — the API exposes no dollar volume, so the column carries the live count instead.
`Status`	Corridor state. In practice always `ACTIVE`.

Built on your machine by grouping the loaded vessels on *(origin port, destination port)* and sorting by vessel count. Vessels missing either endpoint are skipped. Selecting a row fills the **SELECTED ROUTE** card at the bottom of the right panel.

5 · Map toolbar & the live vessel map



The map toolbar: refresh, auto-refresh toggle, basemap picker.

CONTROL	WHAT IT DOES
`⟳ REFRESH`	Re-runs the current search — the raw bbox if it is valid, otherwise the global sample. **8 credits.**
`AUTO ⟳ OFF`	Toggles a **5-minute** auto-refresh, becoming `AUTO ⟳ 5m`. **Off by default, on purpose** — see below.
`BASEMAP`	Switches the tile layer. Your pins, drawn shapes and camera are preserved.

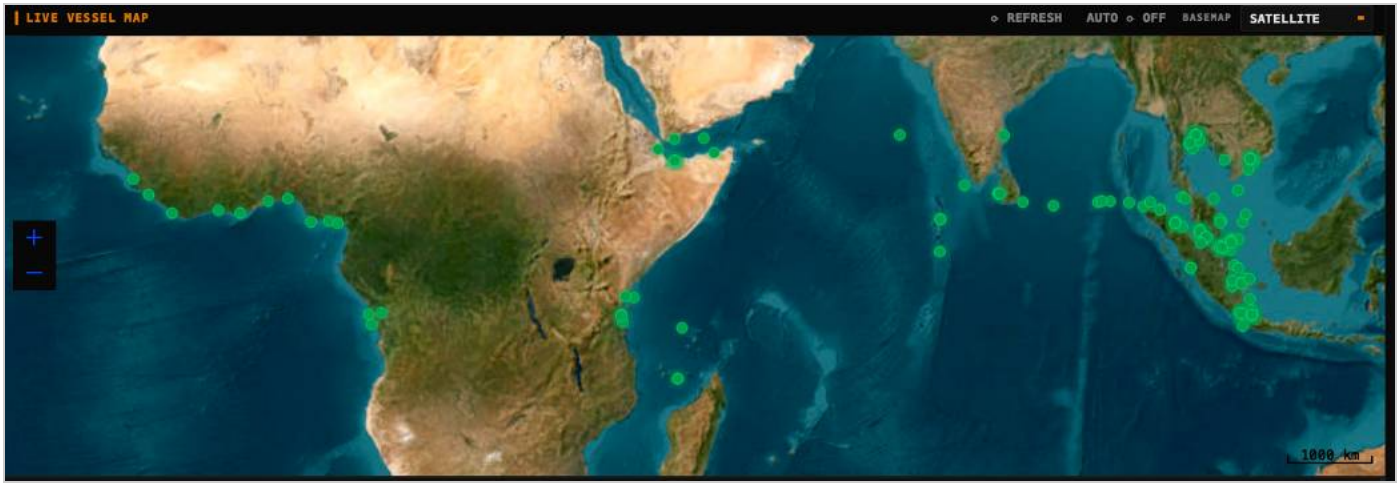
> **AUTO-REFRESH BURNS 8 CREDITS EVERY 5 MINUTES**
>
> While it is on *and the tab is visible*, every tick is a fresh area search. It is opt-in precisely so it never silently drains credits. Leaving the tab stops the timer; it does not resume a load when you come back.



BASEMAPS

Six tile layers — `SATELLITE` (default), `DARK`, `OCEAN`, `STREETS`, `LIGHT`, `TERRAIN`. Tiles are cached to disk (100 MB), so a basemap you have used before redraws instantly and offline. Your choice is *not* remembered across restarts.

The map



A loaded fleet. Zoom controls top-left, scale bar bottom-right.

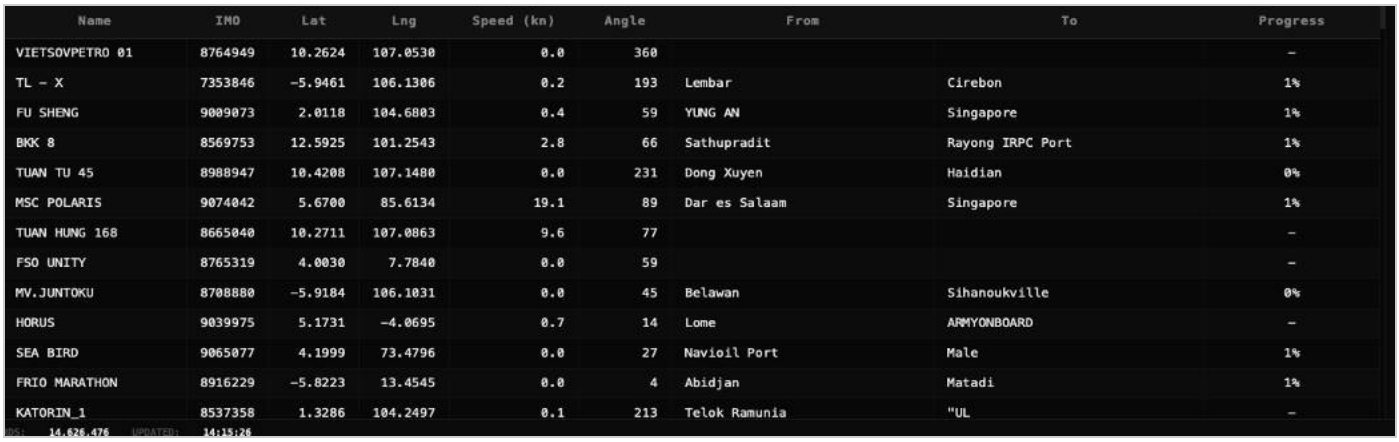
INTERACTION	RESULT
Drag	Pan — unless a draw tool is armed.
Wheel / `+` `−`	Zoom.
Hover a pin	Tooltip: `name (imo)`.
Click a vessel pin	Fills the **SELECTED VESSEL** card, **writes that ship's IMO into the IMO NUMBER field**, and selects + scrolls to its row in the table. Free.

That IMO auto-fill is the intended workflow: click a ship on the map, then press `VOYAGE HISTORY` without typing anything.

PIN COLOURS

PIN	MEANING
Green, medium	A vessel in the loaded fleet. These are the clickable ones.
Amber, large	A ship you tracked by IMO, or a port you plotted.
White, large	The origin or destination port of a voyage you are viewing.
Amber→yellow trail	Voyage history — see **§8**.

6 · The vessel table



Name	IMO	Lat	Lng	Speed (kn)	Angle	From	To	Progress
VIETSOVPETRO 01	8764949	10.2624	107.0530	0.0	360			–
TL – X	7353846	−5.9461	106.1306	0.2	193	Lembar	Cirebon	1%
FU SHENG	9009073	2.0118	104.6803	0.4	59	YUNG AN	Singapore	1%
BKK 8	8569753	12.5925	101.2543	2.8	66	Sathupradit	Rayong IRPC Port	1%
TUAN TU 45	8988947	10.4208	107.1480	0.0	231	Dong Xuyen	Haidian	0%
MSC POLARIS	9074042	5.6700	85.6134	19.1	89	Dar es Salaam	Singapore	1%
TUAN HUNG 168	8665040	10.2711	107.0863	9.6	77			–
FSO UNITY	8765319	4.0030	7.7840	0.0	59			–
MV.JUNTOKU	8708880	−5.9184	106.1031	0.0	45	Belawan	Sihanoukville	0%
HORUS	9039975	5.1731	−4.0695	0.7	14	Lome	ARMYONBOARD	–
SEA BIRD	9065077	4.1999	73.4796	0.0	27	Navioil Port	Male	1%
FRIO MARATHON	8916229	−5.8223	13.4545	0.0	4	Abidjan	Matadi	1%
KATORIN_1	8537358	1.3286	104.2497	0.1	213	Telok Ramunia	"UL	–

One row per loaded vessel. Click any header to sort.

COLUMN	MEANING
`Name`	Vessel name as reported.
`IMO`	IMO number — the ship's permanent hull identifier.
`Lat` / `Lng`	Last known position, decimal degrees to 4 places.
`Speed (kn)`	Speed over ground in knots, 1 decimal.
`Angle`	Heading in degrees.
`From` / `To`	Origin and destination port of the current leg. Often blank — many vessels report no voyage plan.
`Progress`	Voyage completion as a percentage; `–` when the feed gives none.

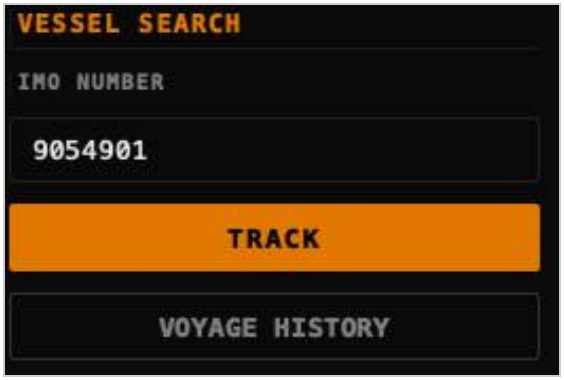
Rows are capped at **200**. If the search matched more, the status bar says so: `READY (showing 200 of 2000)`. Sorting is client-side over the loaded rows only — it does not re-query.

> **READING THE DATA HONESTLY**
>
> This is a live commercial AIS feed, and it is untidy. Expect blank `From` / `To`, destinations that are free text rather than real ports, vessels reporting `0.0` knots at anchor, and the occasional implausible value. The terminal shows what the feed reports, unmodified.

7 · Right panel — search

Four independent tools, stacked. Only one of them spends credits — but it is the easiest to trigger by accident.

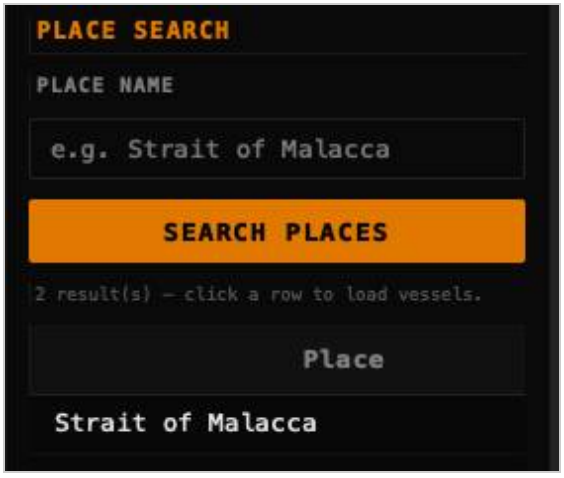
Vessel search



CONTROL	WHAT IT DOES
`IMO NUMBER`	Type a hull number, or let a map pin click fill it. `Enter` tracks.
`TRACK`	Fetches the ship's last known position, drops one amber pin and flies to it.
`VOYAGE HISTORY`	Plots its recent voyage — see **§8**.

> **TRACK CLEARS THE FLEET**
>
> `TRACK` replaces every pin with the one ship you asked for, so nothing is clickable afterwards. Reload an area (or `GLOBAL VIEW`) to get the fleet back — which costs another 8 credits.

Place search



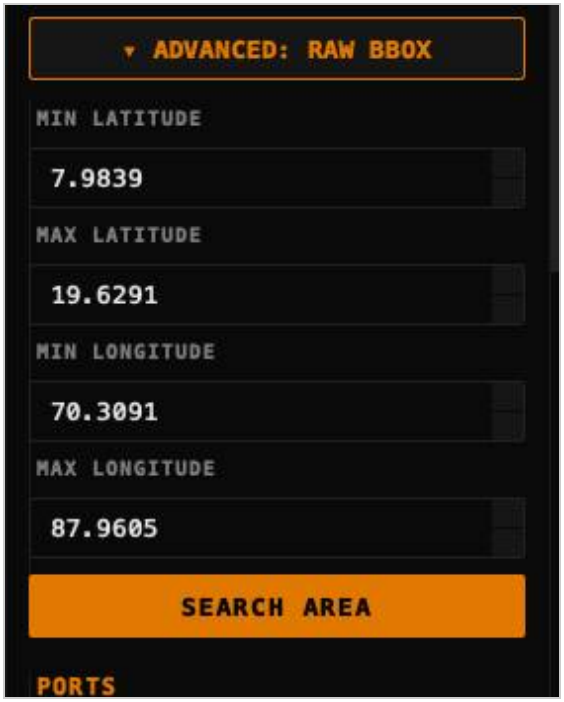
A geocoder: turn "*Strait of Malacca*" into a bounding box. Type three or more characters and suggestions appear after a short pause; or press `SEARCH PLACES` for the full result list. The search itself is **free**. The status line tells you exactly what a click will do.

> **SELECTING A PLACE ROW COSTS 8 CREDITS — WITH NO BUTTON PRESS**
>
> The status line reads "*click a row to load vessels*", and it means it: **merely selecting** a row runs an area search over that place's bounding box. That includes **arrow-keying** through the list, and includes accepting a typeahead suggestion. There is no confirmation step. If you only wanted to see the matches, do not touch the rows.

> **DO NOT SORT THE PLACE OR PORT RESULT TABLES**
>
> Both tables allow header sorting, but a row's *position* is used to look up the underlying result. After sorting, selecting a row acts on **the wrong entry** — loading the wrong sea area (8 credits) or plotting the wrong port. Leave these two tables in the order they arrive.

> **A PLACE IS NOT ALWAYS AN AREA**
>
> Some names geocode to a *point* rather than a region, producing a bounding box only metres across — which then returns no vessels at all. If a place search loads an empty map, open **ADVANCED: RAW BBOX** and look: if the minimum and maximum agree to three decimals, widen them by hand and search again.

Advanced: raw bbox



Type a rectangle directly, then `SEARCH AREA` (**8 credits**). Each maximum must exceed its minimum or the status bar shows `INVALID BBOX` and nothing is sent — no credits spent.

Collapsed by default. These four values are the tab's centre of gravity: they also drive `LOAD VESSELS IN AREA`, `⟳ REFRESH` and `PORTS IN AREA`, and they are rewritten whenever you pick a place.

Two quirks worth knowing: the latitude boxes accept values beyond ±90 without complaint, and typing coordinates alone may not be enough for them to survive a restart — they are saved reliably only when the IMO field is also edited, or at shutdown.

Ports



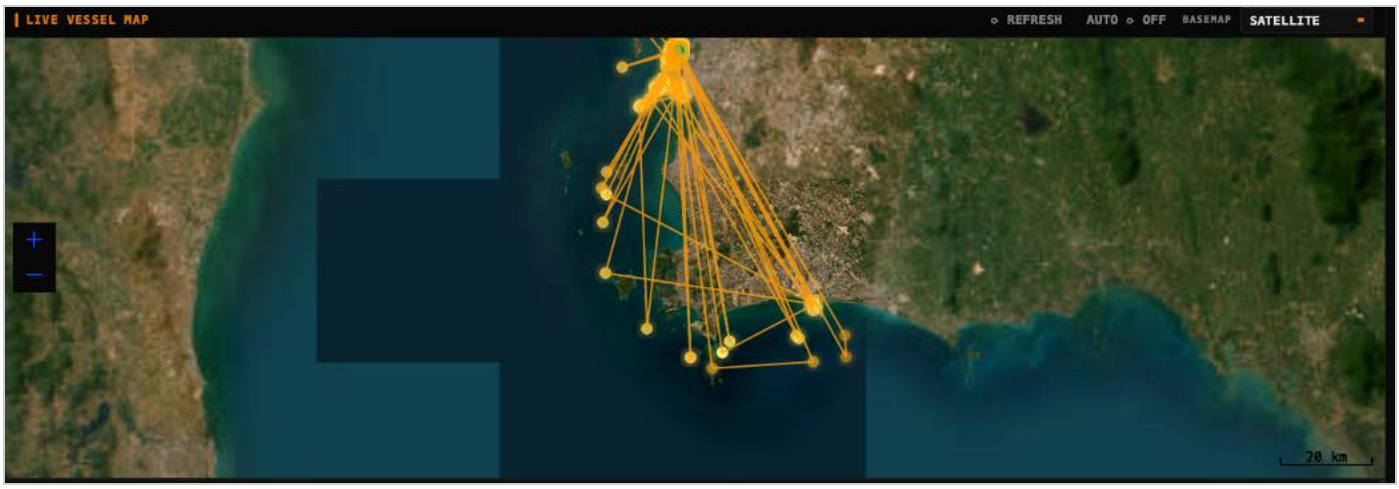
CONTROL	WHAT IT DOES
`PORT NAME`	Type ≥3 characters for suggestions, or `Enter`.
`SEARCH PORTS`	Looks the harbour up by name. Free.
`PORTS IN AREA`	Lists harbours inside the **raw bbox**. Free.
Click a row	Drops an amber pin and flies to it. Free.

COLUMN	MEANING
`Name`	Port name. Hover for its UN/LOCODE where one exists.
`Country`	Country — blank for some sources.
`Source`	Which reference database answered: `Wikidata`, `MarineRegions` or `OSM`.

> **"PORTS IN AREA" IGNORES WHAT YOU CAN SEE**
>
> Despite the name it does **not** use the visible map. It reads the four **ADVANCED: RAW BBOX** values — so it can return harbours from a region nowhere near your current view. If those values are not a valid rectangle it says `Set a valid bbox in AREA SEARCH first.`

8 · Voyage history

Enter an IMO (or click a pin to fill it) and press `VOYAGE HISTORY` to replay where that ship has recently been.



A vessel's recent reported positions, joined into a track.

MARK	MEANING
Bright yellow, largest	The most recent positions.
Faint amber, smallest	The oldest positions — dots shrink and fade as they age.
Green, large	`CURRENT POSITION` — where the ship is now.
White, large	The voyage's origin and destination ports, resolved by name.
Connecting line	The positions joined in the order the feed returned them.

Hover any trail dot for `name — from → to [date]`.

The table switches meaning

Name	IMO	Lat	Lng	Speed (kn)	Angle	From	To	Progress
THAPANEE 9	9054901	13.1379	100.8919	102.3	360			2026-07-15
THAPANEE 9	9054901	13.1380	100.8919	102.3	360			2026-07-14
THAPANEE 9	9054901	13.1377	100.8920	102.3	360			2026-07-13
THAPANEE 9	9054901	13.1378	100.8919	102.3	360			2026-07-13
THAPANEE 9	9054901	13.1373	100.8922	102.3	360			2026-07-12
THAPANEE 9	9054901	13.1365	100.8917	102.3	360			2026-07-11
THAPANEE 9	9054901	13.1373	100.8921	102.3	360			2026-07-10
THAPANEE 9	9054901	13.1377	100.8920	102.3	360			2026-07-10

History mode: one row per reported position, vessel name in amber.

> **THE "PROGRESS" COLUMN IS A DATE HERE**
>
> In history mode each row is a *position report*, and the last column holds that report's **date** (`YYYY-MM-DD`) — but the header still reads `Progress`. Read it as "Date". Rows are newest first.

The status bar confirms the result: `HISTORY: <vessel> (<n> positions)`, or `NO HISTORY` if the feed has nothing. Like `TRACK`, this replaces the fleet on the map. A ship sitting at a mooring produces a tight knot of dots rather than a line — that is the ship being stationary, not an error.

9 · Credits — what costs what

Vessel data is metered. Everything else on this tab — places, ports, basemaps — is free. One number matters: **an area search costs 8 credits**.

Spends 8 credits

ACTION	NOTES
Opening the tab	Automatic on first show. Unavoidable.
`GLOBAL VIEW (200 VESSELS)`	—
`LOAD VESSELS IN AREA`	—
`⟳ REFRESH`	—
`SEARCH AREA`	—
`AUTO ⟳` while on	**Every 5 minutes**, for as long as the tab is visible.
Selecting a **place** result row	Including by arrow key. No button press needed.
Accepting a place suggestion	From the typeahead dropdown.
Finishing a **square/circle drag**	Fires on mouse release.

Metered, cost not published

ACTION	NOTES
`TRACK`	Single-vessel position lookup.
`VOYAGE HISTORY`	Position history for one vessel.

Free

`SEARCH PLACES` · `SEARCH PORTS` · `PORTS IN AREA` · clicking a *port* row · `× CLEAR` · `ADVANCED: RAW BBOX` · the `BASEMAP` picker · panning and zooming · clicking a vessel pin · selecting a trade corridor · arming (but not dragging) a draw tool.

HOW TO KEEP THE BILL DOWN

- **Repeating the identical search within 60 seconds is free** — it replays from cache.
- **Leave `AUTO ⟳` off** unless you are actively watching a region.
- **Look, don't select.** Reading the place results costs nothing; clicking a row costs 8.
- **Returning to the tab is free.** The automatic load happens only on the first show.
- **Let a load finish.** A global sample can take 30–60 seconds; clicking again is another 8.
- Port and place lookups cache for **7 days**.

NOTHING HERE IS DESTRUCTIVE

This tab never changes server-side data, never deletes anything, and never opens a browser. The only thing it writes is your own screen state — the IMO field and the four bbox values. The cost of a mistake is credits, not data.

10 · Status & message reference



The status bar: source, record count, last update — and current operation on the right.

FIELD	MEANING
`SOURCE`	The upstream module. Green when the service reports healthy, amber otherwise.
`RECORDS`	Total vessel records in the provider's database.
`UPDATED`	Clock time of your last successful load.

Status messages

MESSAGE	MEANING
`READY`	Idle, or a load finished within the 200-row cap.
`READY (showing 200 of 2000)`	More matched than can be drawn. You are seeing a sample.
`LOADING GLOBAL...`	Global sample in flight.
`SEARCHING...`	Area search in flight.
`TRACKING...`	Single-vessel lookup in flight.
`LOADING HISTORY...`	Voyage history in flight.
`HISTORY: <name> (<n> positions)`	History plotted.
`NO HISTORY`	The feed has no positions for that IMO.
`INVALID BBOX`	A maximum is not greater than its minimum. Nothing sent — no credits spent.
`ERROR`	The request failed. Check the network badge in the top bar.

On the map



A load in progress. The map dims and names the operation.

The overlay reads `LOADING GLOBAL VESSELS`, `SEARCHING AREA`, `TRACKING VESSEL` or `LOADING HISTORY`. A global load can take **30–60 seconds** — normal; let it finish rather than clicking again.

In the search panels

MESSAGE	WHERE
`Searching…`	Place / port lookup in flight.
`No places found.` · `No ports found.`	No match.
`N result(s) — click a row to load vessels.`	Place results. Remember: a click here costs 8 credits.
`N port(s) — click a row to plot.`	Port results. Free.
`Search failed: …` · `Lookup failed: …`	The reference database did not answer.
`Set a valid bbox in AREA SEARCH first.`	`PORTS IN AREA` with an unusable rectangle.

11 · Quick reference: how do I…

HOW DO I…	DO THIS
Open the tab	Command bar → type `maritime`. Or Navigate → Research & Intelligence → Maritime.
See ships worldwide	`GLOBAL VIEW (200 VESSELS)`. **8 cr**
See ships in one sea area	Arm `□ SQUARE`, drag a box on the map. Releasing searches. **8 cr**
Search a named sea area	Type it in `PLACE NAME` → `SEARCH PLACES` → click the right row. **8 cr on the click**
Search exact coordinates	Expand `ADVANCED: RAW BBOX`, fill the four boxes, `SEARCH AREA`. **8 cr**
Find one ship	Type its IMO → `TRACK`.
See where a ship has been	Click its pin (fills the IMO) → `VOYAGE HISTORY`.
Get the fleet back after TRACK	`GLOBAL VIEW` or re-run your area search. **8 cr**
Find a port	Type it in `PORT NAME` → `SEARCH PORTS` → click a row to plot. Free.
List harbours in a region	Set the raw bbox, then `PORTS IN AREA`. Free.
Change the map look	`BASEMAP` → pick one of six. Free, and pins are kept.
Keep a region updating	Load it, then switch `AUTO ⟳` on. **8 cr / 5 min**
Stop drawing on the map	`× CLEAR`. Your vessels stay put.
See more of the map	Drag the divider between the map and the table downwards.
Check a corridor	Click a row in `TRADE CORRIDORS`; details fill the SELECTED ROUTE card.
Avoid spending credits	Read results without selecting place rows; leave `AUTO ⟳` off; repeat a search within 60 s to hit the cache.

Worth remembering

- **There are no keyboard shortcuts** on this tab, no right-click menus, and no double-click actions. `Enter` works in the three text fields.
- **Only the IMO field and the four bbox values are remembered** across restarts. Basemap, splitter position, the `AUTO ⟳` toggle, drawn shapes and table sorting all reset.
- **Every count on this tab describes what you loaded**, not the ocean. Reload and the sample — and therefore the corridors and the statistics — will differ.
- **The map always renders**, even before data arrives. A grey tile reading "*Map data not available*" comes from the tile provider at a zoom it does not cover — zoom out.

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**.

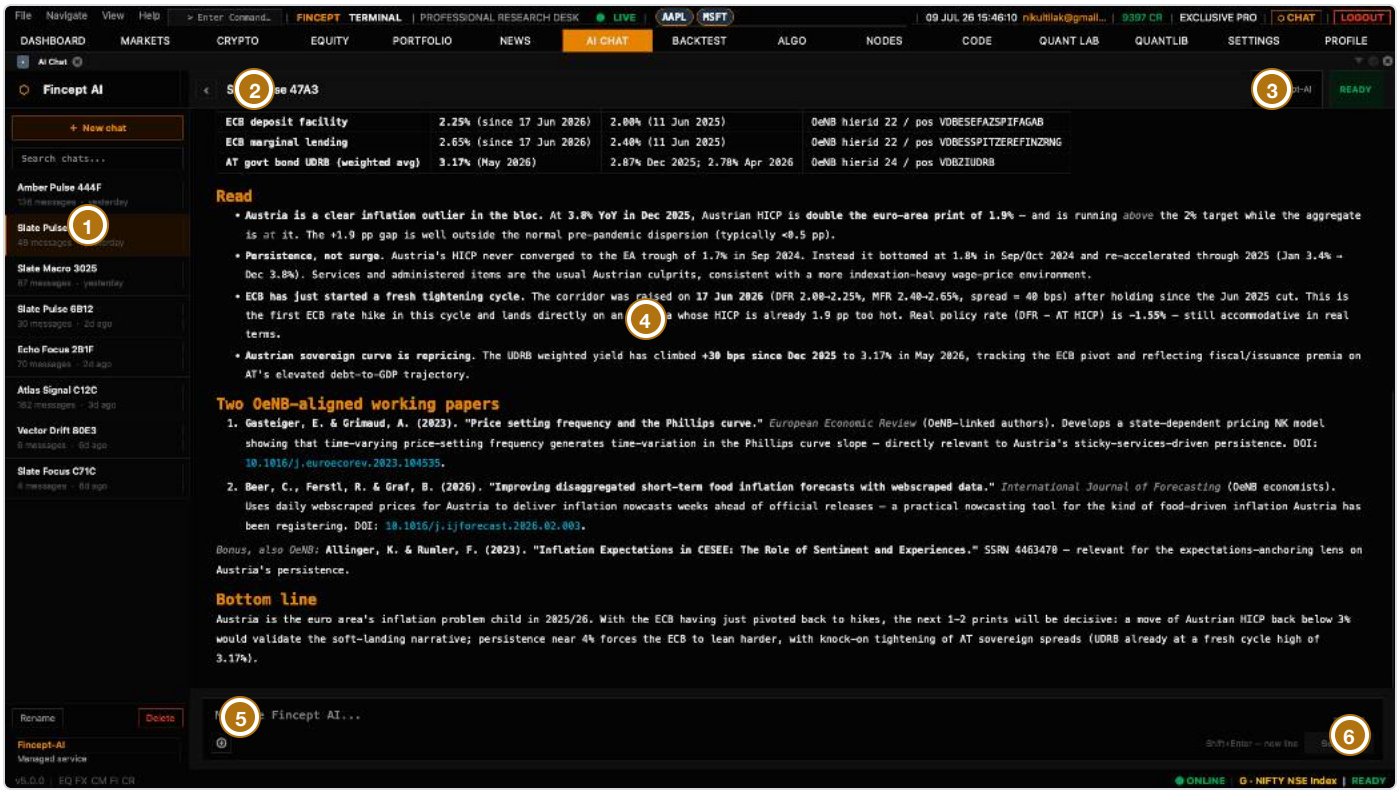
At a glance, you can…

- **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

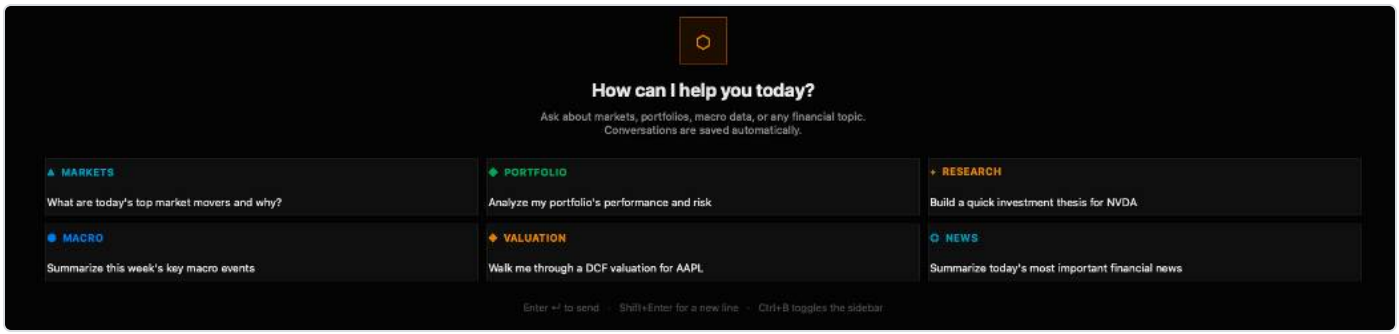
A conversation sidebar on the left, the chat itself on the right, and a floating helper bubble.



#	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.

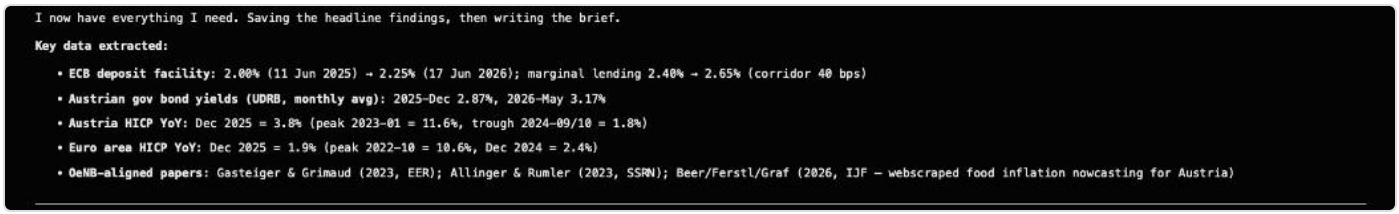
Element	What it means
Your message	Right-aligned amber bubble labelled **You**, with the time.
AI answer	Full-width, led by the `⬡ Fincept AI` name. Streams in live as it's written.
🔧 **Tool strip**	A small line on an answer showing the tools it used, e.g. `get_quote · get_history ×2 · +3 more`.
✦ **Reasoning**	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.



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.

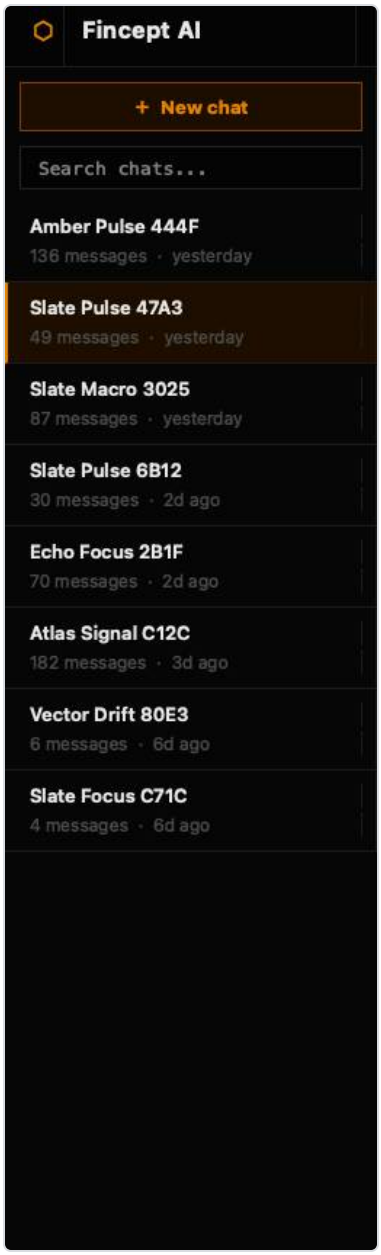
Approvals — you're always in control

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.

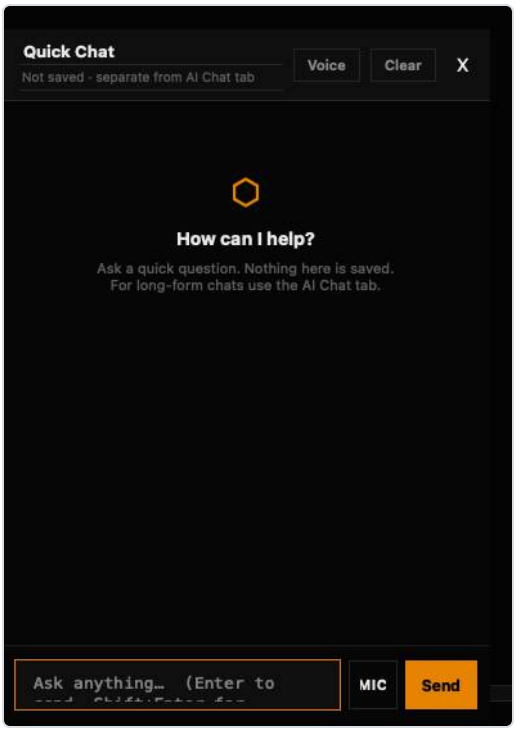
> Your history syncs to your account, so your conversations follow you. Switching chats mid-answer is blocked until the current reply finishes, so nothing gets cut off.

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.

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.



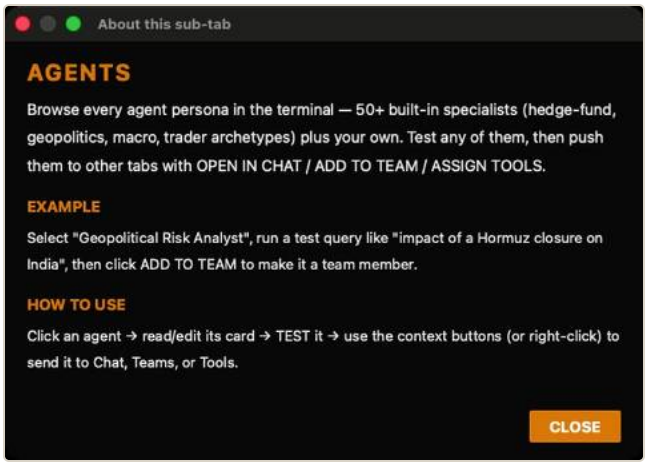
The Agent Studio navigation bar — title, nine sub-tab buttons, the ⓘ INFO help button and the live agent-count chip.

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

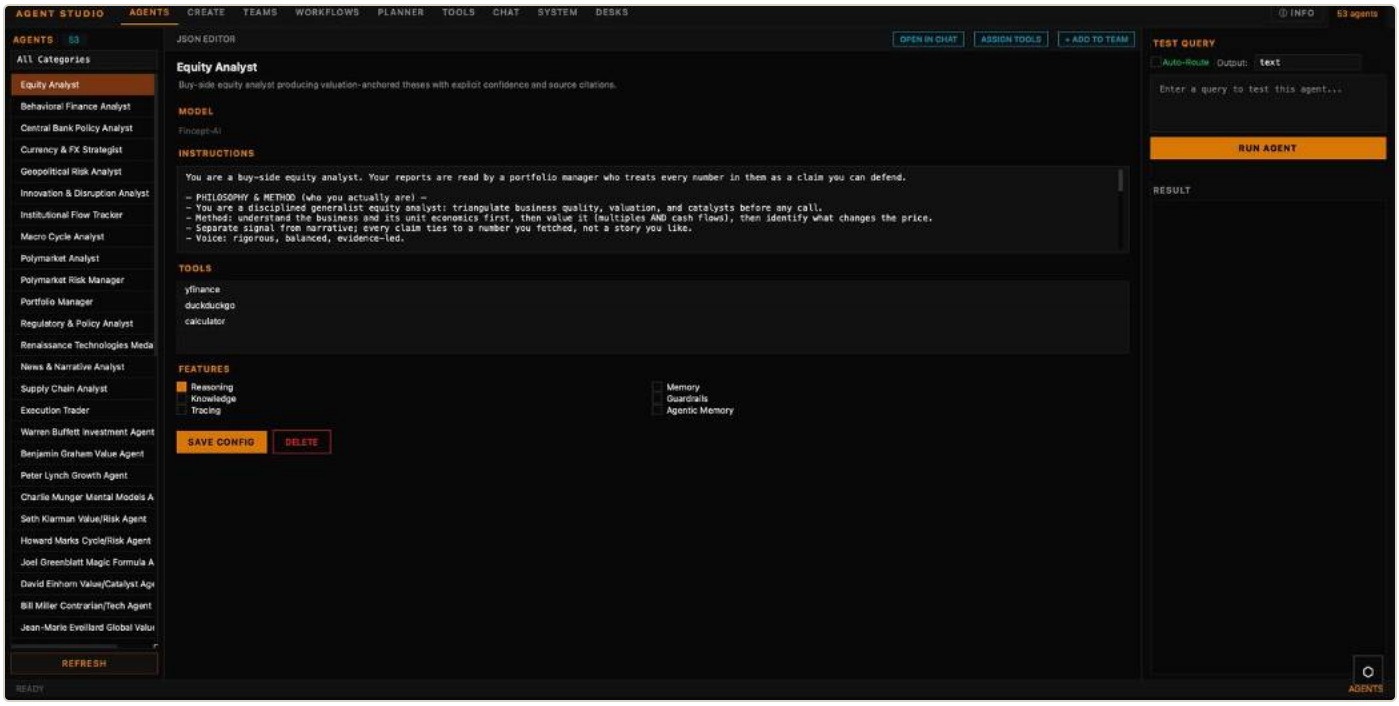
Every sub-tab ships its own contextual help. The card names the tab, explains what it does, gives a concrete worked example (here: “run a test query like *impact of a Hormuz closure on India*, then click ADD TO TEAM”), and a short how-to. It is the fastest way to orient a new analyst without leaving 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

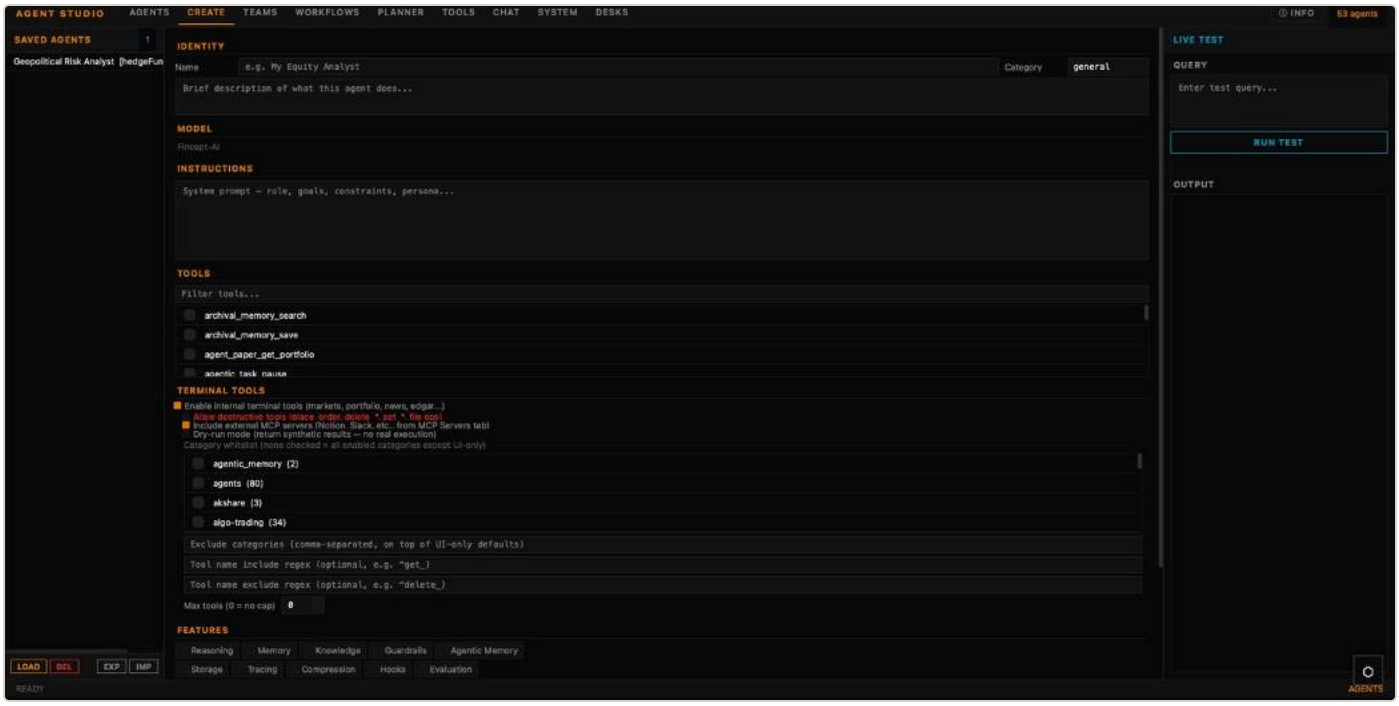
Control	Type	What it does
Auto-Route	checkbox	When on, an intent router picks the best-matching persona for the query instead of the selected one, and shows a `Routed → agent (intent, confidence%)` line.
Output	dropdown	Response shape: `text` (free-form, streamed) or a structured schema — `InvestmentReport`, `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.

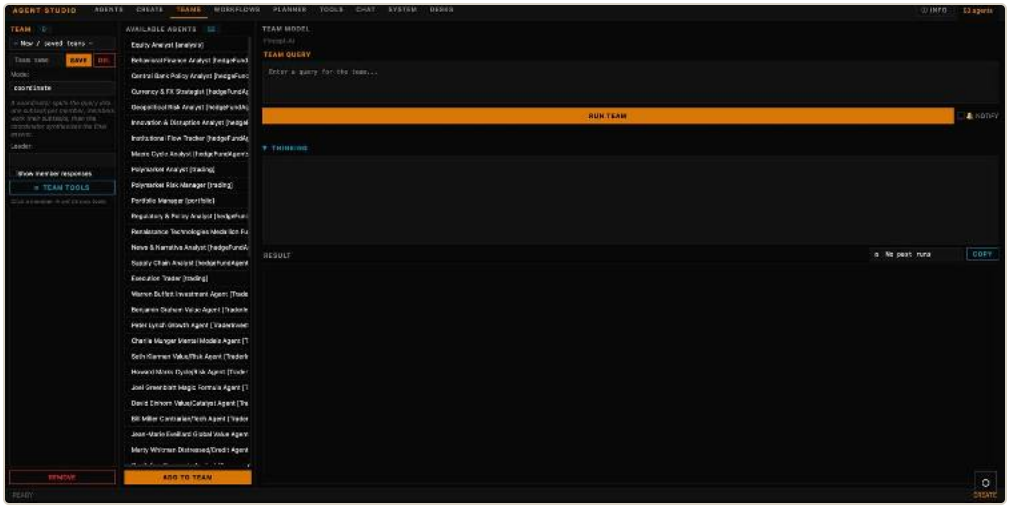
> **Save & reuse. SAVE AGENT** writes to your cloud settings blob and the agent immediately appears under SAVED AGENTS and across the terminal. The left rail also offers **LOAD / DEL / EXP / IMP** — export a config to JSON and import it back (Import assigns a fresh id). **LIVE TEST** dry-runs the in-progress 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.

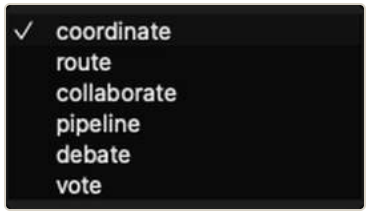
4 TEAMS

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.

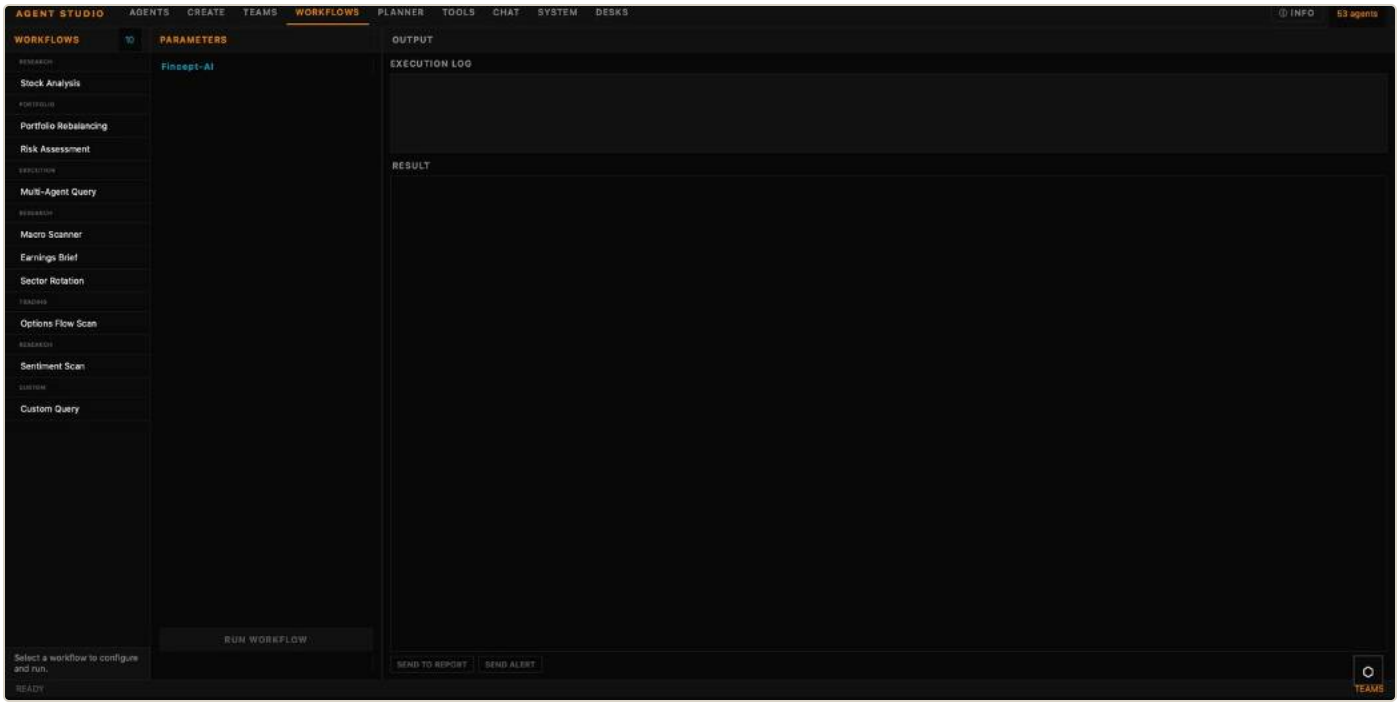
Show member responses	Include each member's own answer alongside the synthesis.
⚙ TEAM TOOLS · click a member	Scope MCP tool categories (or individual tools) for the whole team, and additively per-member.
ADD TO TEAM · REMOVE	Add the selected catalog agent / remove the selected member.
TEAM QUERY · RUN 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.

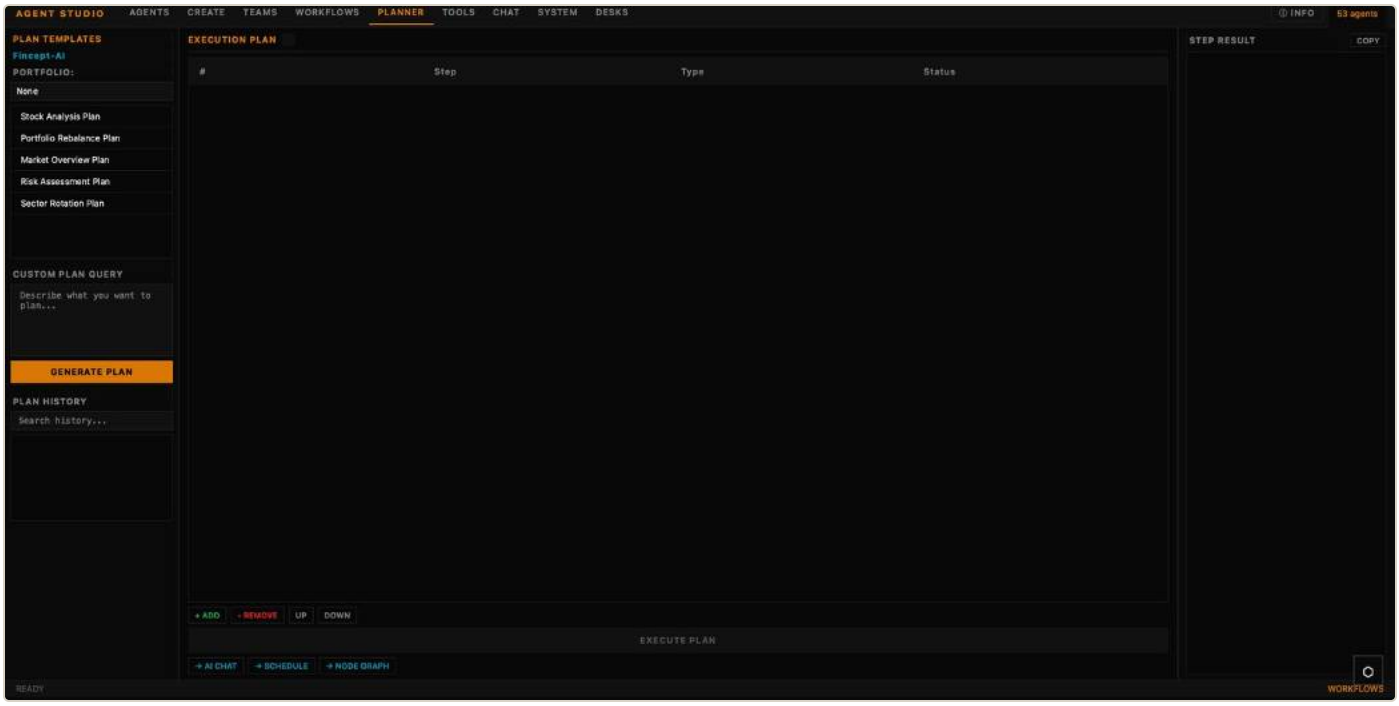
Route the result onward. After a successful run, **SEND TO REPORT** appends the result to your Report Builder document (as a heading + text block), and **SEND ALERT** dispatches it through your configured notification channels.

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

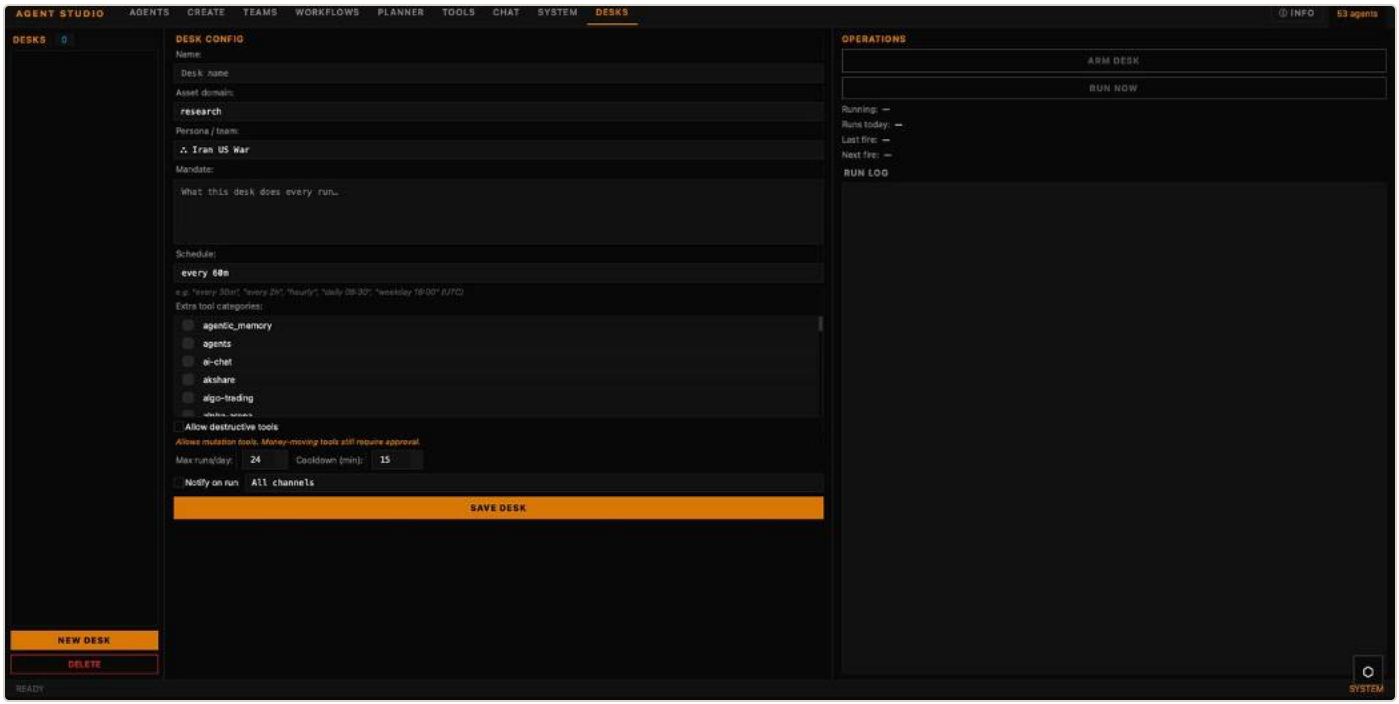
→ **AI CHAT**	Creates a new AI Chat session seeded with the plan as its task state.
→ **SCHEDULE**	Registers the plan as a recurring task (e.g. `daily 09:30`, `weekday 16:00`, UTC).
→ **NODE GRAPH**	Saves the plan as a Node-Editor workflow — one agent node per step, chained.

> **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

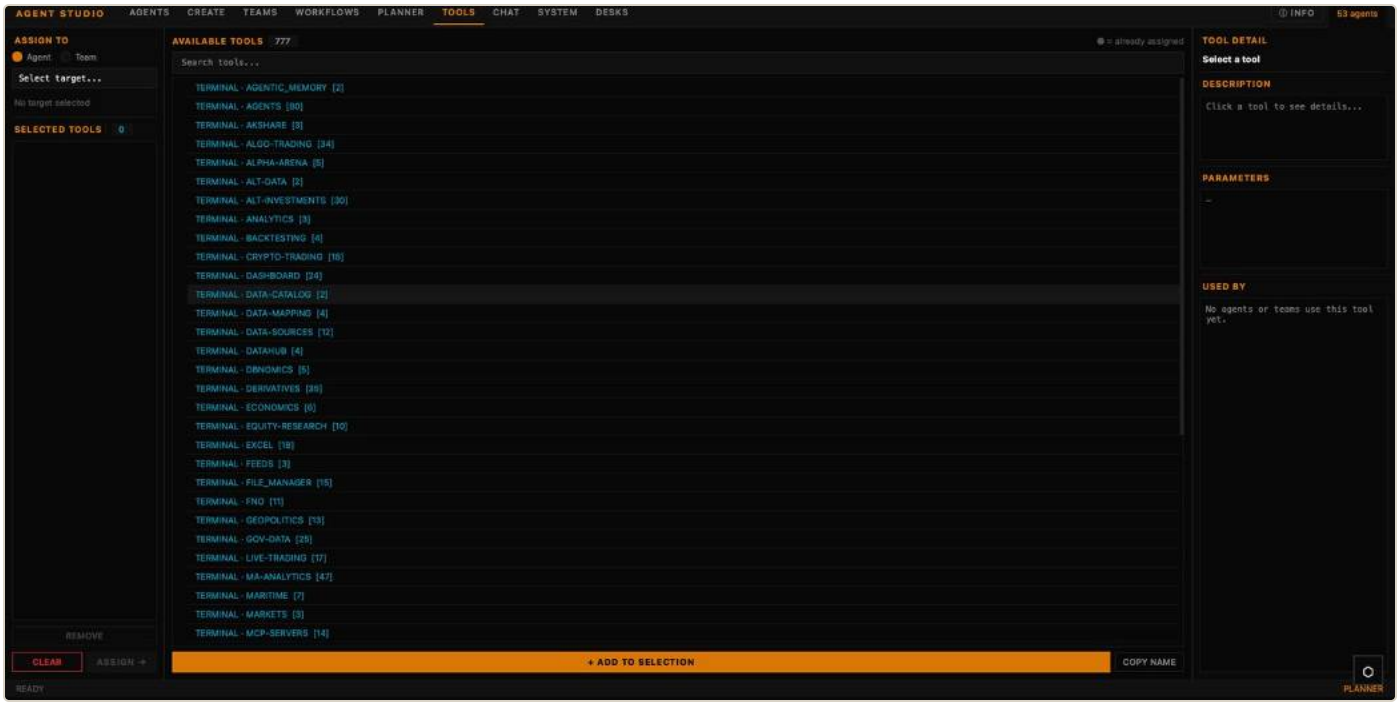
Control	What it does
ARM DESK	Toggles the desk armed/disarmed; an armed desk fires on its schedule.
RUN NOW	Fires immediately (suppressed if a cooldown, daily cap, or in-flight run applies).
Running · Runs today · Last fire · Next fire	Live status, refreshed every few seconds while the tab is open.
RUN 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

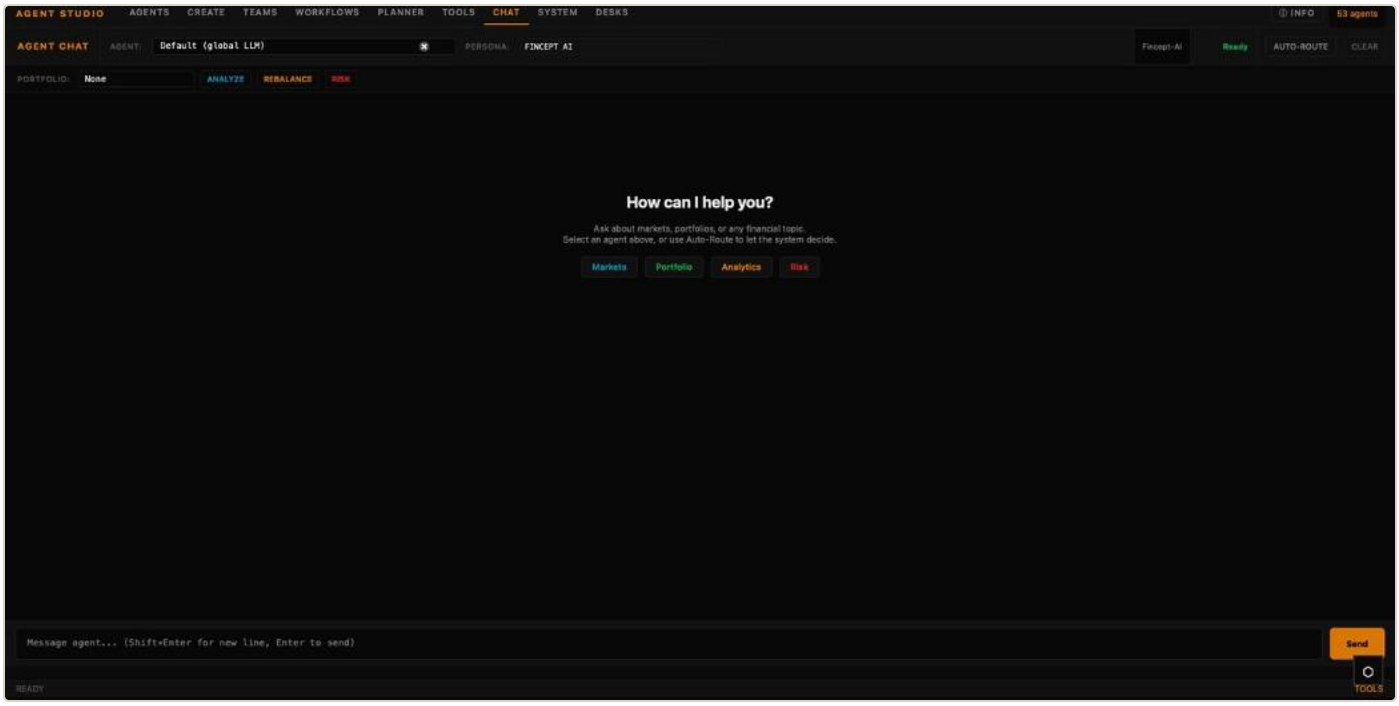
Control	What it does
ASSIGN TO · Agent / Team	Choose whether you're assigning to an agent or a team, then pick the target.
AVAILABLE TOOLS · Search	Live filter over tool name and description; matching categories auto-expand.
TERMINAL · CATEGORY [n]	The terminal's own MCP tools, grouped by category (cyan). 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,

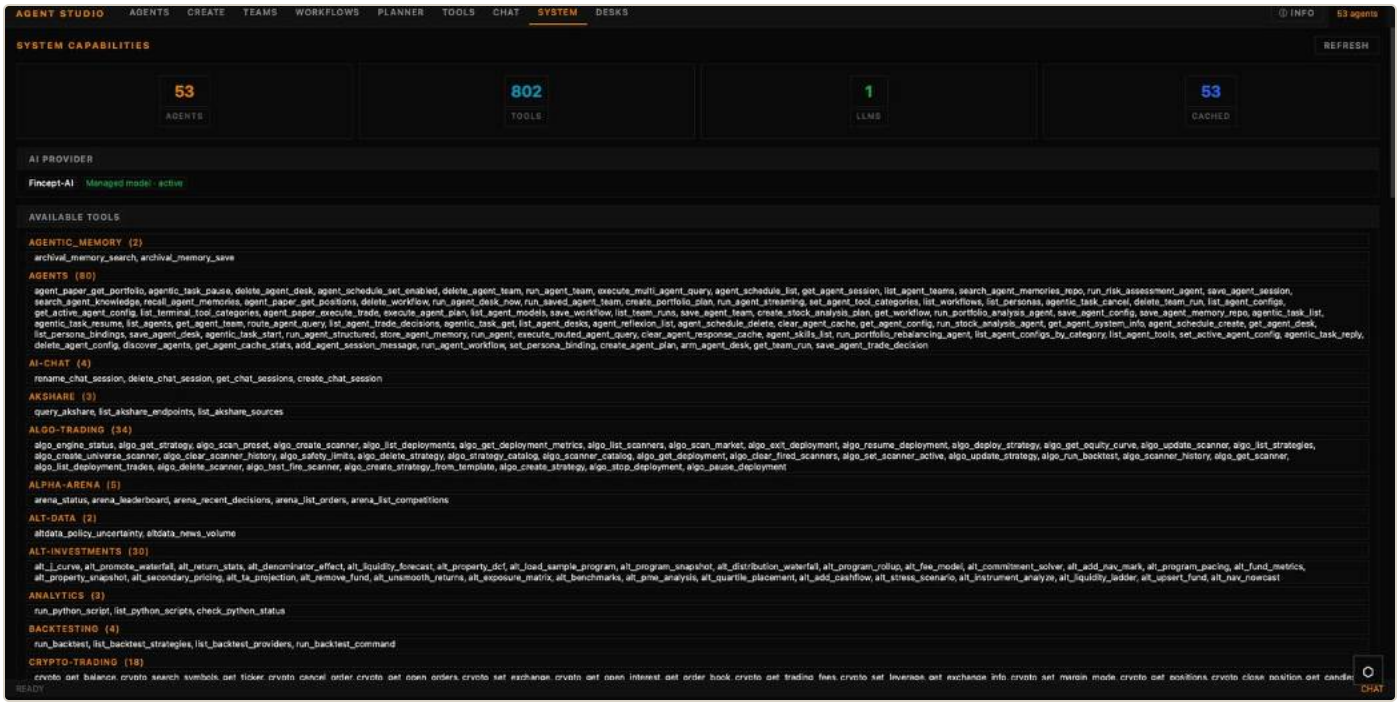
turning a query into a durable background task (§13).

> **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.

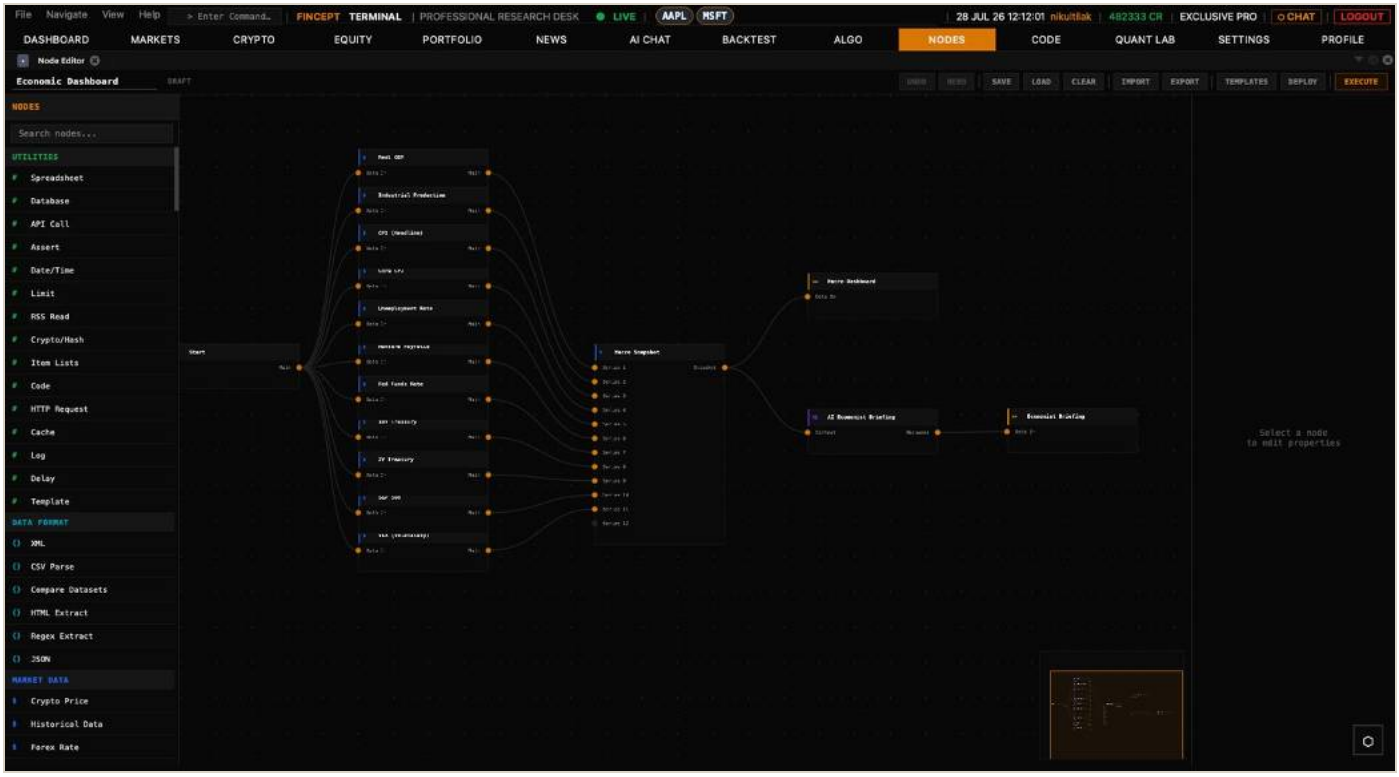


Fig 2 — The NODES tab holding a 17-node Economic Dashboard workflow.

#	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.

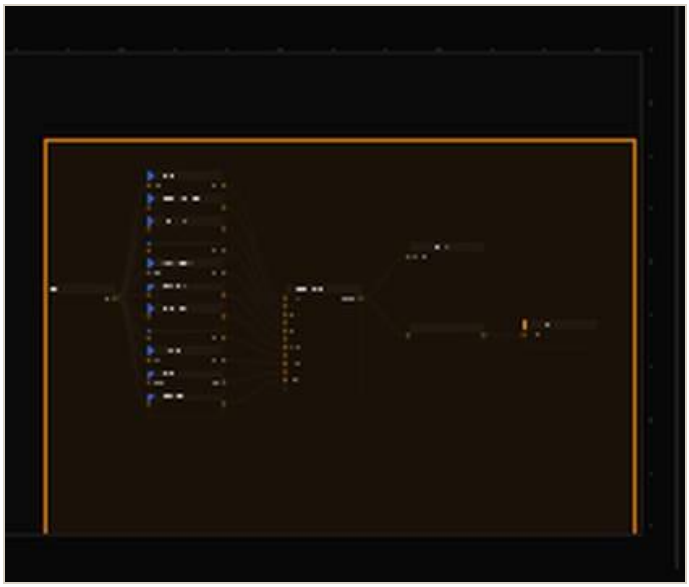


Fig 3 — The minimap: the amber rectangle is your viewport.



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

PALETTE · 04

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.

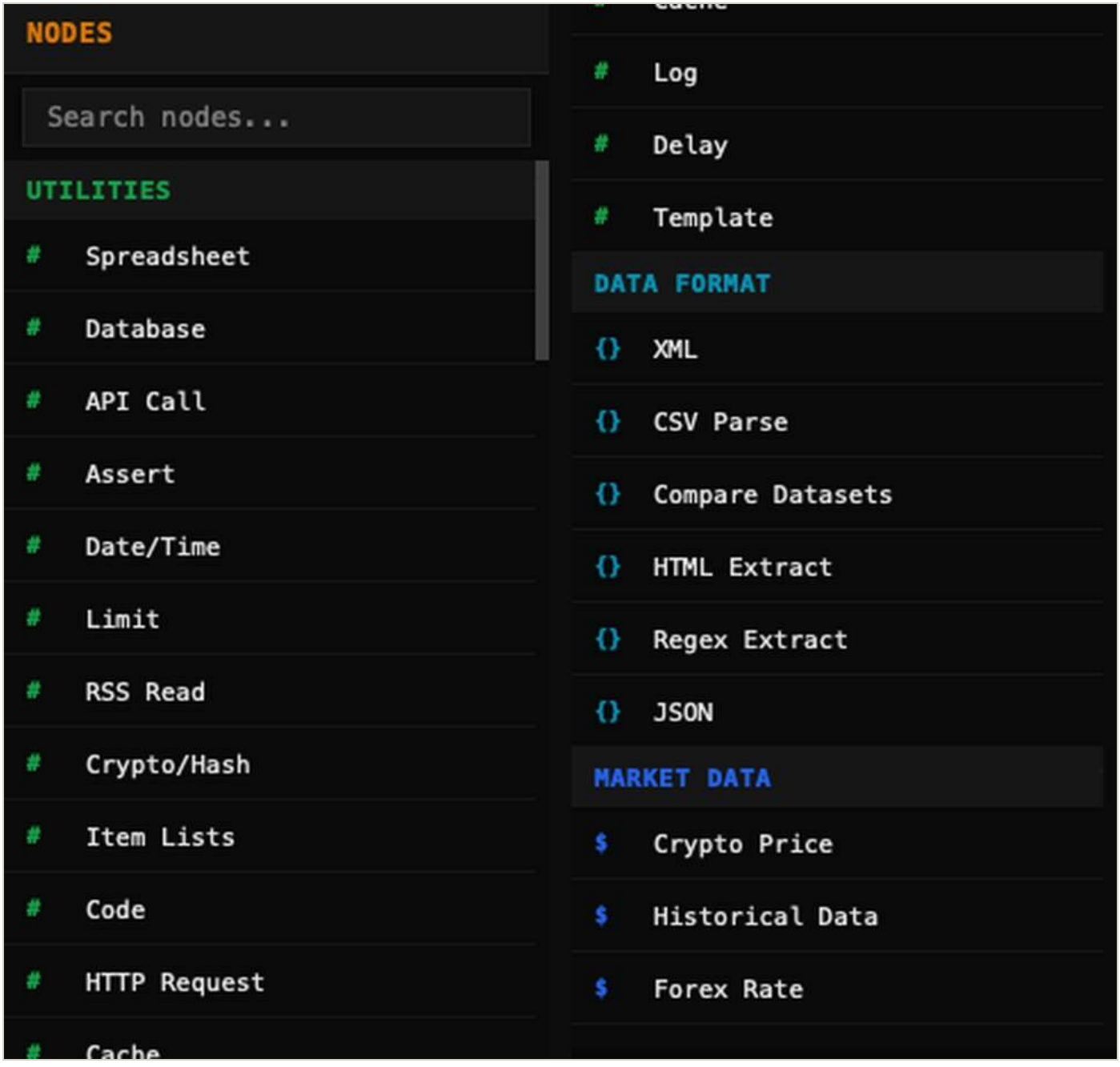


Fig 5 — The palette, shown here as two columns for legibility: the search box, then UTILITIES (15), DATA FORMAT (6), MARKET DATA (14) and the rest.

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.

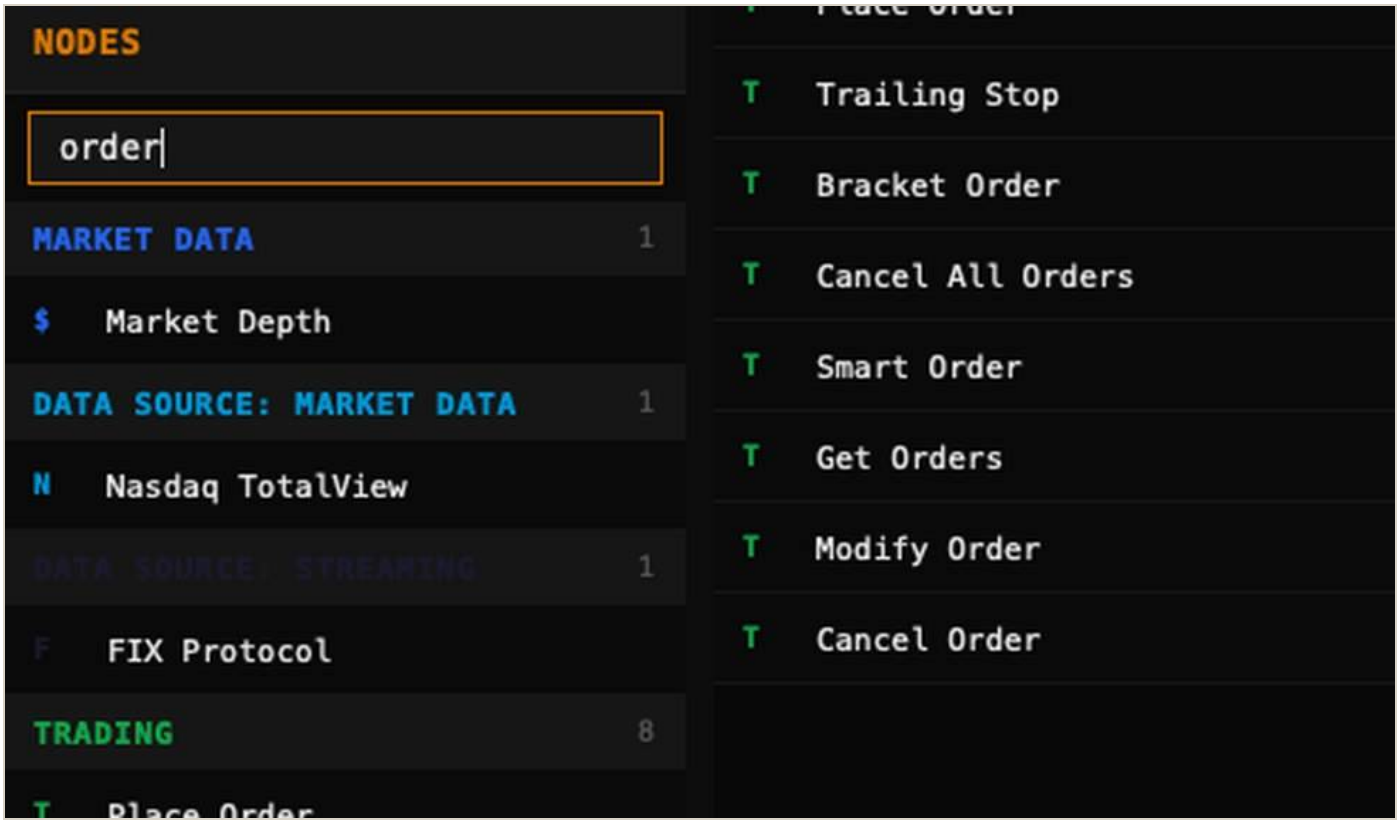


Fig 6 — Searching `order`: one Market Data hit, two auto-generated Data Source nodes, and the eight Trading nodes that place or manage orders.

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

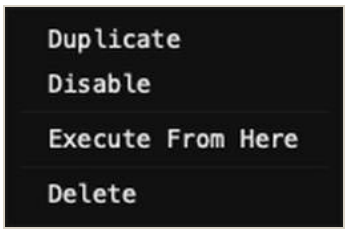


Fig 9 — Right-clicking a node.

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

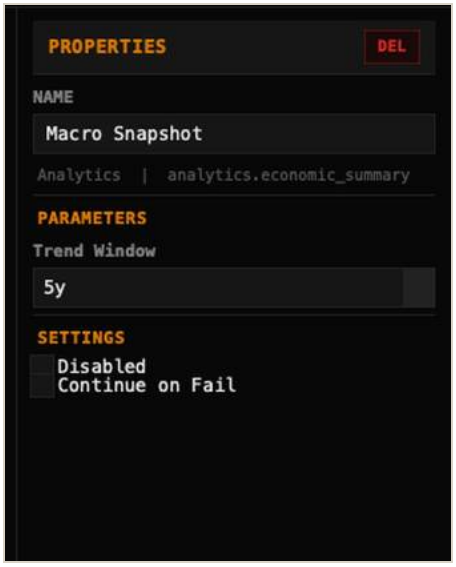


Fig 10 — The panel for the selected analytics node.

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

ENGINE · 08

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

RESULTS · 09

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

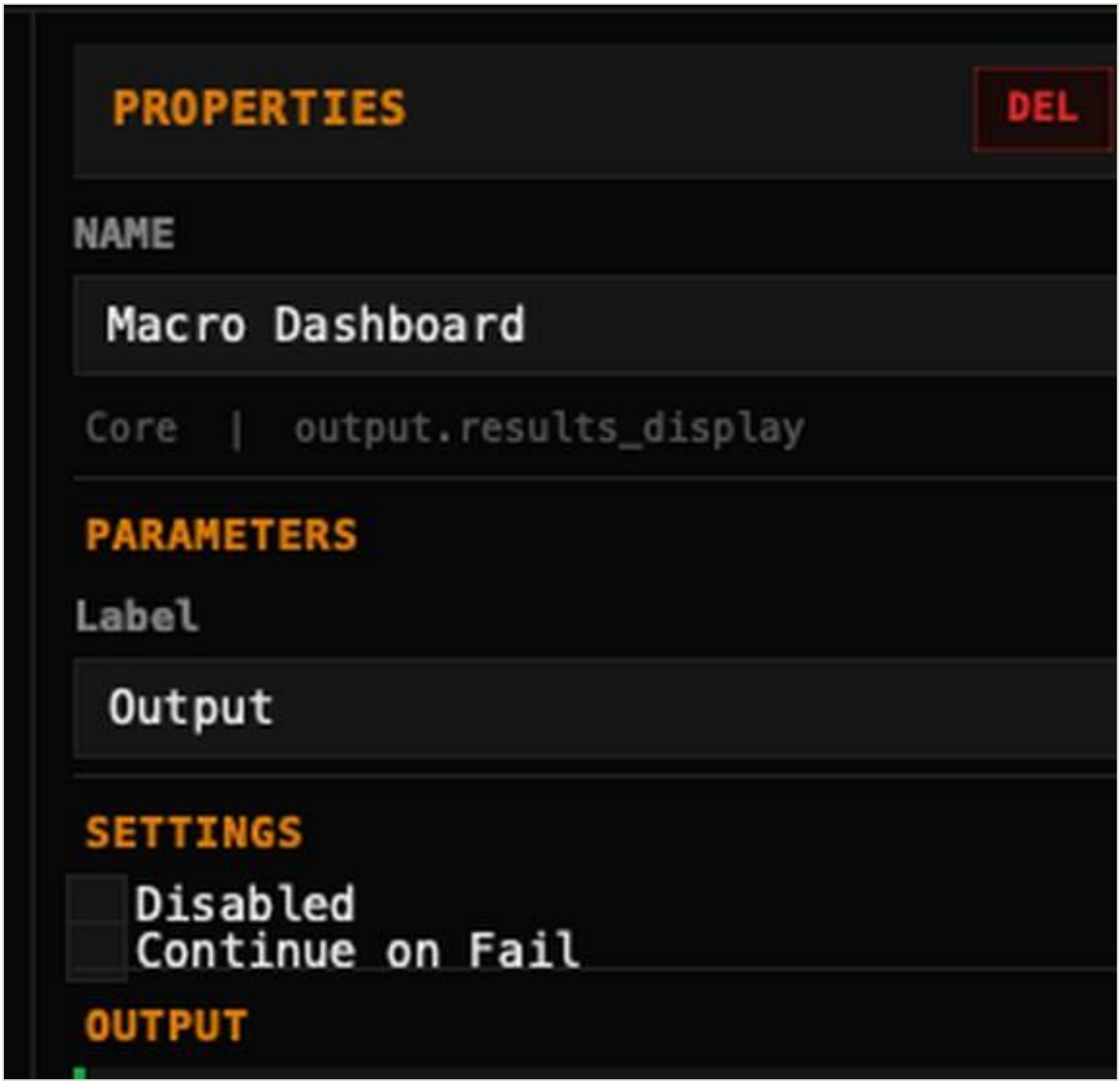


Fig 12 — The same result in the properties panel: selecting a node after a run adds an **OUTPUT** section showing what it returned and how long it took.

> **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.

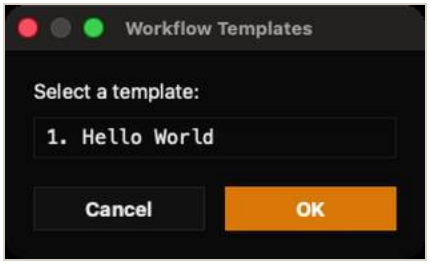


Fig 13 — The template picker. The list is a dropdown; all fifteen are tabulated opposite.

#	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

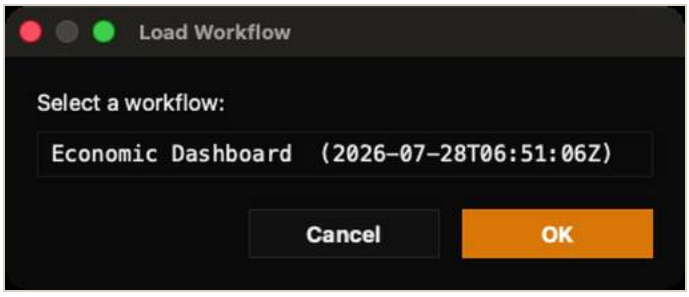


Fig 14 — The load picker: name plus the last-updated time, in UTC.

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**.

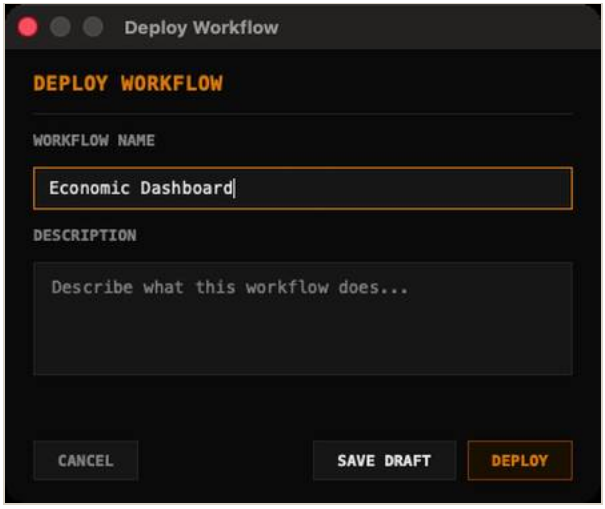


Fig 16 — The deploy dialog: name, description, and three choices.

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.

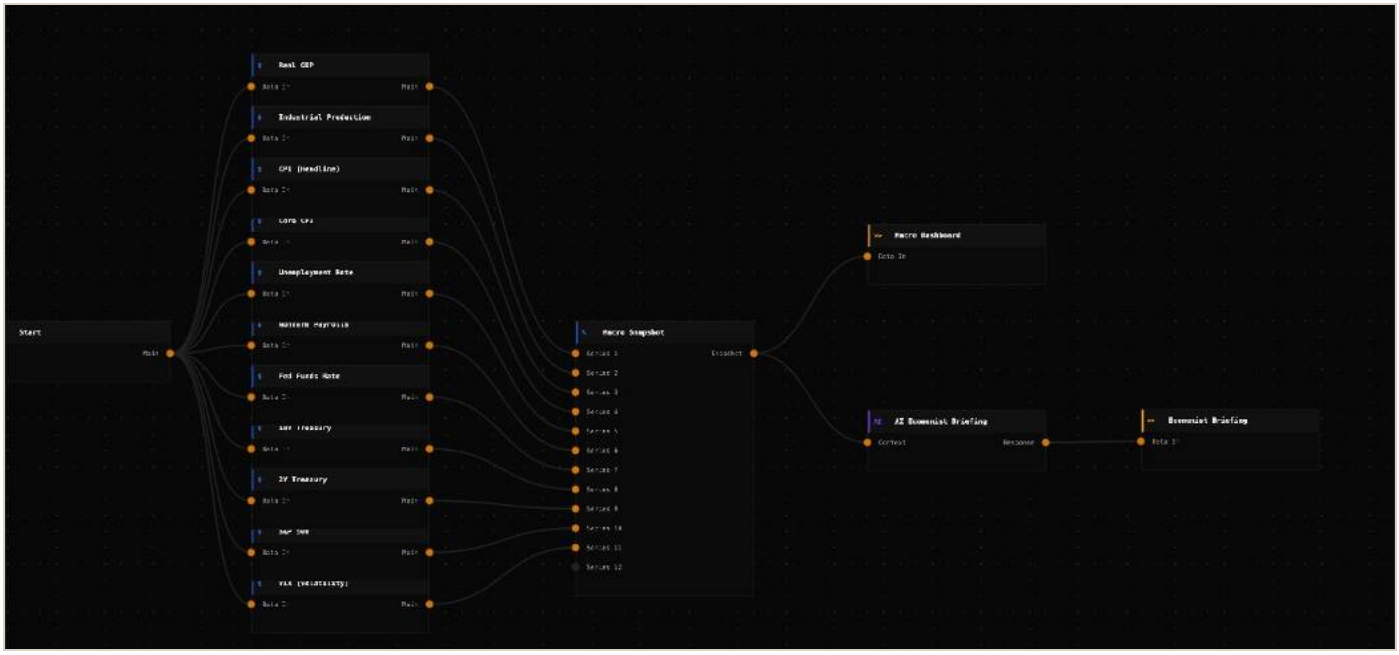


Fig 17 — Economic Dashboard: one trigger, twelve series fetches, one analytics node, then a dashboard and an AI briefing.

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

A complete, screen-by-screen walkthrough of the MCP Servers tab — where you extend the terminal's AI agent with external Model Context Protocol tool servers and enable or disable its ~790 built-in tools.

`>_ marketplace · installed · tools`

Covers · **the 3-view tab** · **MARKETPLACE** · **INSTALLED** · **TOOLS**
adding a custom server · **~790 built-in tools** · transport & security · 14 agent tools

Demonstrated live on the Fincept Terminal MCP Servers tab.

Contents

9 SECTIONS

GETTING ORIENTED

Overview — what the MCP Servers tab is 3

1 Anatomy — the header, three views & status bar 4

THE THREE VIEWS

2 MARKETPLACE — the curated catalog 5

3 INSTALLED — manage your servers (cards, logs, remove) 7

4 TOOLS — enable / disable individual tools 9

ADDING, CONNECTING & REFERENCE

5 Adding a server — step by step (with dialogs) 11

6 How servers connect — model & transport 12

7 Persistence & security 13

8 Driving this tab from the AI agent (14 tools) 13

9 Quick reference & important notes 14

ABOUT THIS GUIDE

Every screenshot in this document is a real capture of the running terminal, including its live server dialogs, cards and tool lists. Control labels are quoted exactly as they appear on screen.

What the MCP Servers tab is

OVERVIEW

Model Context Protocol (MCP) is the open standard by which the terminal's AI agent reaches tools, and this tab is where you manage those tools. It does two jobs: **add and run external MCP servers** — small helper programs (fetch a URL, query a database, run git) that expose extra tools to the agent — and **browse and toggle the terminal's own ~790 built-in tools**.

There are two sources of tools. Internal tools (~790, compiled into the terminal — markets, portfolio, trading, economics, derivatives, agents…) appear under one synthetic, always-available server called Fincept Terminal. External tools come from separate MCP programs you add here, each launched as a local process that speaks MCP to the terminal while it runs. Both feed one unified catalogue that AI Chat, agents/teams and the node editor all draw from — so a server you add here immediately becomes usable by the agent.

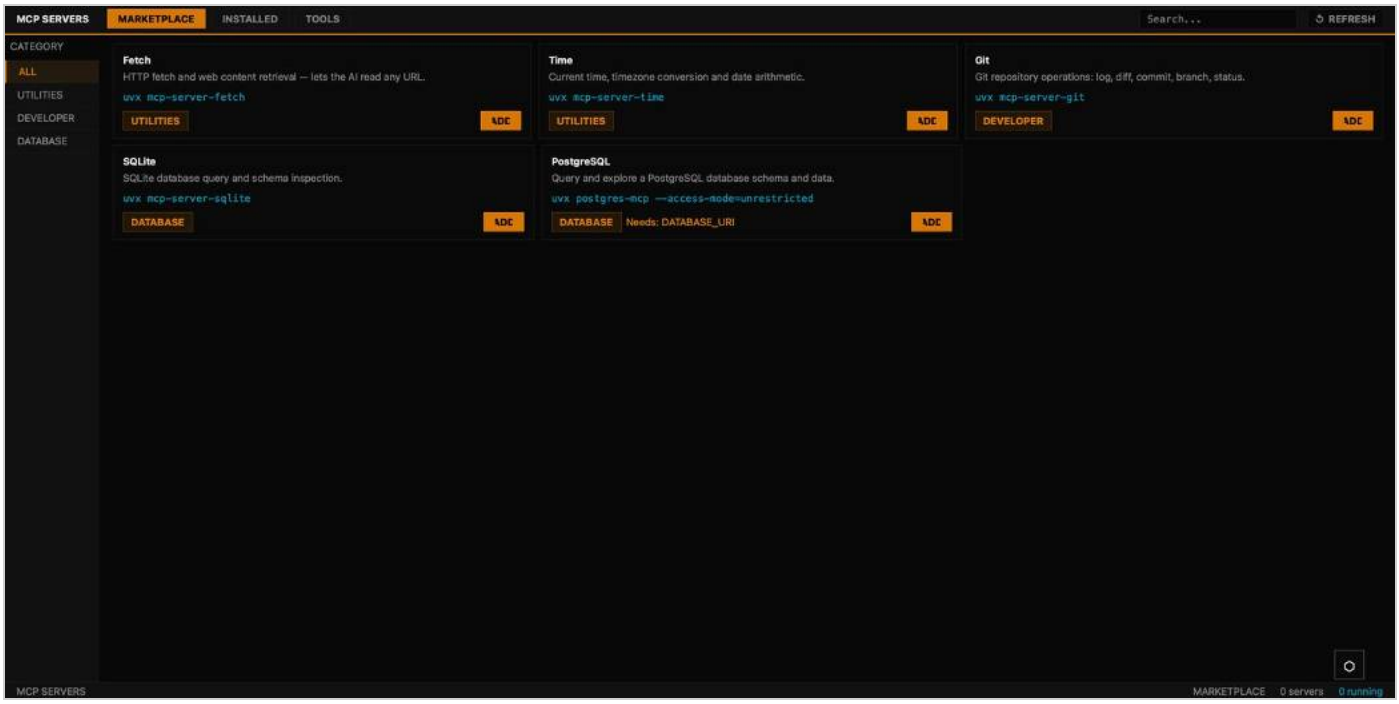
AT A GLANCE, IN THE MCP SERVERS TAB YOU CAN…

- Browse a curated **Marketplace** of ready-to-run servers and one-click ADD them (a pre-filled config dialog).
- Add a **custom MCP server** of your own — command, arguments, environment variables, category and auto-start.
- **Start / stop** each server with a single ENABLE toggle, read its live status, and view its logs.
- Remove a server (with confirmation).
- Browse every tool — internal + external — and **enable / disable** individual built-in tools for the AI agent.
- Search across all three views; the whole configuration roams with your account.

1 Anatomy of the tab

ORIENTATION

Open the tab from the command bar (type `mcp` → **MCP Servers**) or **Navigate ▸ Tools ▸ MCP Servers**. The screen is one header, a switchable body of three views, and a status bar.



The MCP Servers tab on the Marketplace view. ① Title ② View switcher (Marketplace / Installed / Tools) ③ Search ④ Refresh ⑤ View body ⑥ Status bar (view · servers · running).

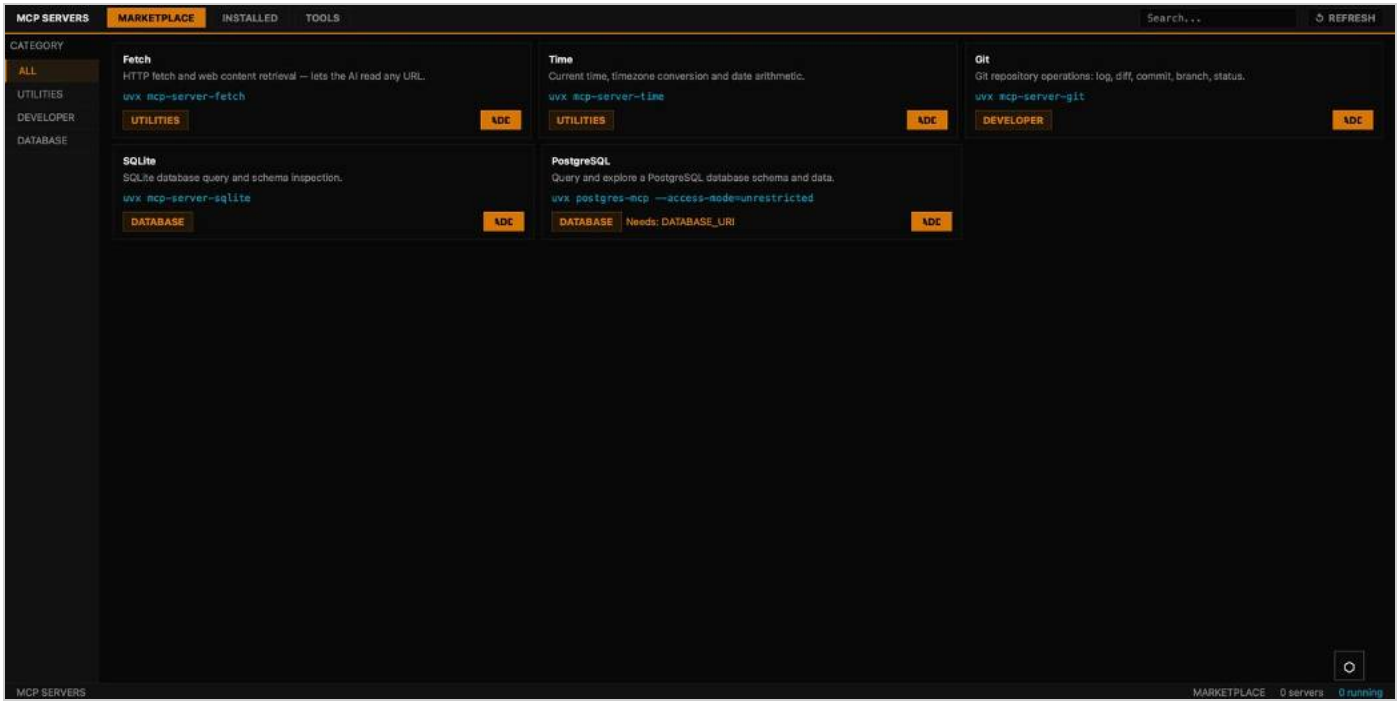
Element	What it does
① **MCP SERVERS** title	Static screen label.
② **View switcher** — MARKETPLACE · INSTALLED · TOOLS	Three buttons switch the body between the catalog, your installed servers, and the full tool list. The active one is amber. Your active view is remembered.
③ **Search** box	Live filter. In Marketplace & Installed it filters cards by name / description / category; in Tools it hides non-matching rows (matches tool name, server or description).
④ ↺ **REFRESH**	Re-reads the server registry and rebuilds the active view.
⑥ **Status bar**	`MCP SERVERS` · the active view name · **N servers** (total installed) · **N running** (cyan).

> **The Marketplace and Installed views are only for EXTERNAL servers.** The terminal's ~790 built-in tools are never shown as installable or removable — they appear only in the **Tools** view (under server "Fincept Terminal"), where they can be individually toggled.

2 MARKETPLACE — the catalog

VIEW 1

A curated catalog of preset external servers you can add in one click. A category sidebar filters the cards; each card is a ready-made configuration.



Marketplace — category sidebar (ALL / UTILITIES / DEVELOPER / DATABASE) and server cards, each with its launch command, category badge, any required env keys, and an ADD button.

Category sidebar & the 5 catalog servers

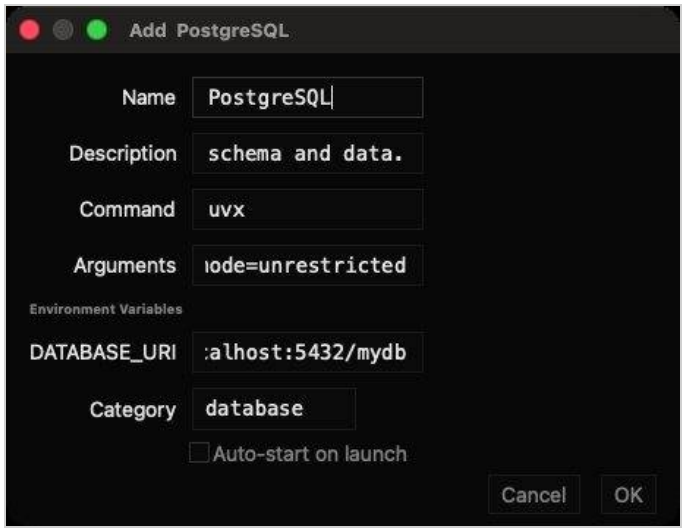
The sidebar has four filters: **ALL · UTILITIES · DEVELOPER · DATABASE**. The catalog ships with five servers:

Server	Category	Launch command	Needs
Fetch	utilities	`uvx mcp-server-fetch` — HTTP fetch / web content for the AI	—
Time	utilities	`uvx mcp-server-time` — time, timezone & date arithmetic	—
Git	developer	`uvx mcp-server-git` — repo log / diff / commit / branch / status	—
SQLite	database	`uvx mcp-server-sqlite` — SQLite query & schema	—
PostgreSQL	database	`uvx postgres-mcp --access-mode=unrestricted`	`DATABASE_URI`

A catalog card & its ADD button

On the card	Shows / does
Name · Description	The server's name and a one-line purpose.
Command line (cyan)	The exact `command + args` that will run it.
Category badge (amber)	UTILITIES / DEVELOPER / DATABASE.
"Needs: …" (amber)	Any required environment keys (e.g. `DATABASE_URI`).

On the card	Shows / does
ADD button	Opens a pre-filled config dialog (see §5). Once added, the card instead shows a green ✓ **ADDED** badge.



Clicking ADD on a card opens a pre-filled dialog. For servers that need secrets (here PostgreSQL) a field appears per required env key with a sample value.

At a glance, in MARKETPLACE you can…

- Filter the catalog by category or search; read each server's command and any required env keys.
- One-click ADD a preset server into a pre-filled config dialog; already-added servers show ✓ ADDED.

3 INSTALLED — manage your servers

VIEW 2

One full-width card per server you've added, each with inline status and controls, plus a sticky button to add a custom server. When empty it reads "No servers installed yet."



An installed server (Time) in the running state — the ● **RUNNING** status pill, the ● **ENABLED** toggle (top-right), the launch command & category, and the LOGS / REMOVE buttons. Status bar shows "1 running".

Anatomy of a server card

Control	What it shows / does
Name + status pill	The server name and its live state: ● **RUNNING** (green) · ⟳ **STARTING** · ● **ERROR** (red) · ○ **STOPPED** (red).
Enable toggle (top-right)	**This is the start/stop control.** Reads ● **ENABLED** (running), ⟳ **STARTING…** (disabled, transitional) or ○ **DISABLED** (stopped). Click when stopped → launches the server on a background thread (can take 60s+ on first run); click when running → stops it.
Description	The server's blurb (if any).
Command · Category	The `command + args` (monospace) and its category label.
LOGS button	Toggles an inline log panel under the card (see below).
REMOVE button (red)	Deletes the server — confirmation required.
+ ADD CUSTOM MCP SERVER	Sticky amber button at the bottom — opens the custom-server dialog (§5).

> **Starting a server can fail.** If the process can't launch or the MCP handshake fails, a dialog titled **"Server Failed to Start"** appears with the error, and the card shows ● **ERROR**. A healthy server is health-checked every 60 seconds and auto-restarted up to 3 times before being marked in error.

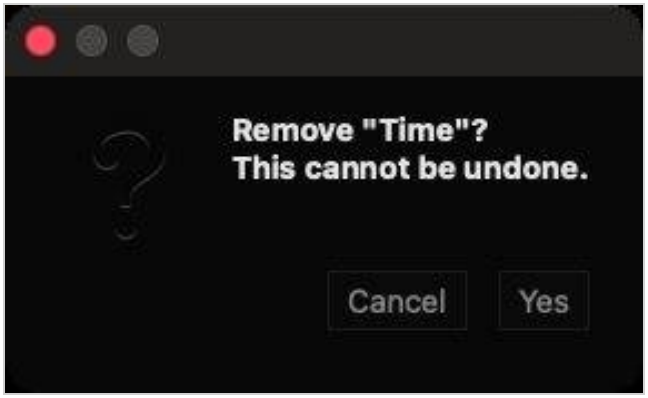
Logs



The LOGS panel expands inline beneath the card. Before a server produces output it reads "No output yet. Start the server to see logs here." — here shown while the server is still starting.

The panel shows the server's captured `stdout` / `stderr`, up to the last **500 lines**, auto-scrolled to the bottom. Logs are per-session and in-memory only — they are not persisted and there is no export.

Removing a server



The REMOVE confirmation.

REMOVE opens a confirmation — **Remove "<name>"? This cannot be undone.** with **Cancel** / **Yes**. On Yes the server is stopped (if running) and deleted from your configuration; the status bar's server count drops accordingly.

Auto-start note: a server's "auto-start on launch" flag can only be set when you *create* it (a checkbox in the add dialogs). There is no per-card auto-start toggle — to change it later, remove and re-add the server.

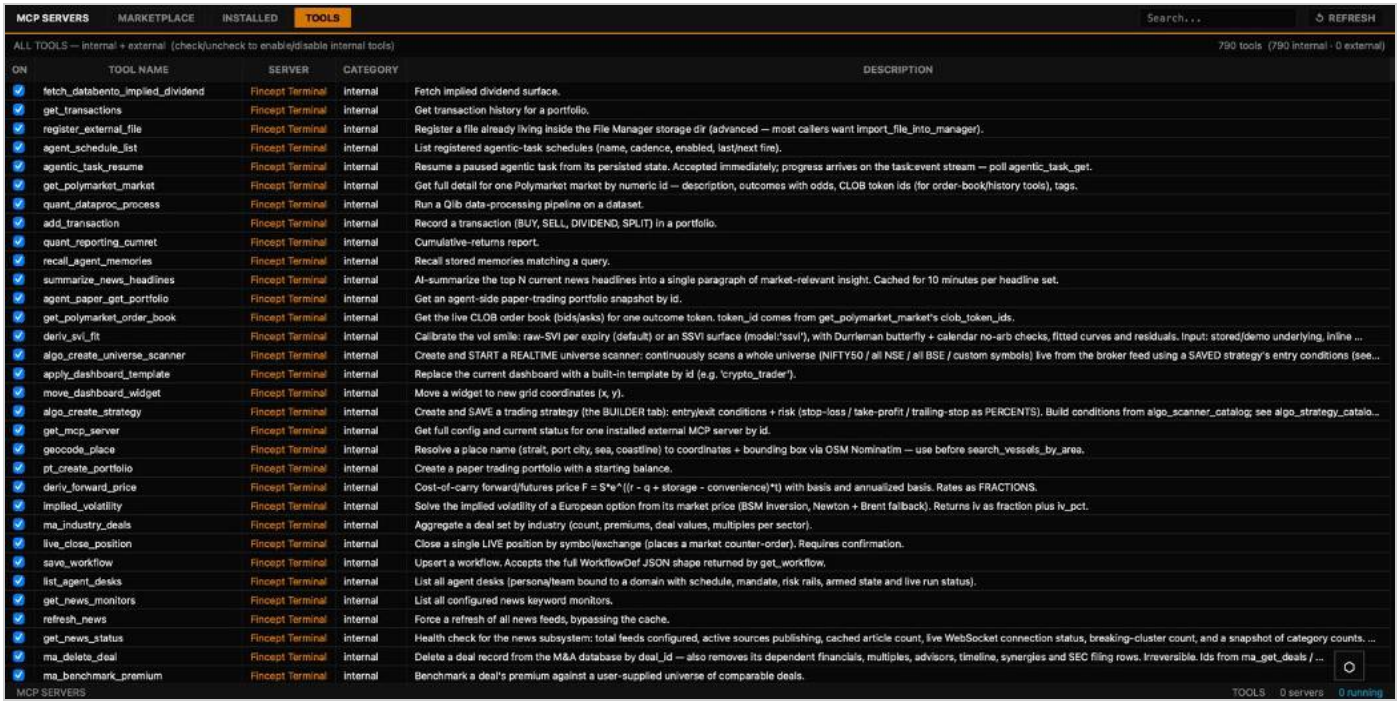
At a glance, in INSTALLED you can…

- See each server's live status; start or stop it with the ENABLE toggle; read its logs (last 500 lines).
- Add a fully custom server; remove any server (with confirmation).

4 TOOLS — enable / disable tools

VIEW 3

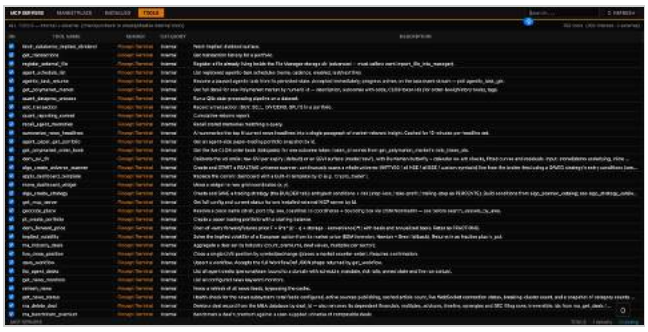
One searchable table of every tool available to the AI agent — internal (built-in) plus external (from running servers) — where you can switch individual built-in tools on or off.



The Tools view — columns ON · TOOL NAME · SERVER · CATEGORY · DESCRIPTION. The count (top-right) reads "790 tools (790 internal · 0 external)" with no external servers running.

Column	Meaning
ON	Checkbox. For **internal** tools it is a working enable/disable switch. For **external** tools it is a read-only "on" indicator (external tools are always on while their server runs and cannot be toggled here).
TOOL NAME	The tool's identifier.
SERVER	**Fincept Terminal** (amber) for internal tools, or the external server's name (cyan).
CATEGORY	`internal` or `external`.
DESCRIPTION	What the tool does. The count label reads `N tools (X internal · Y external)`.

What disabling a tool does. Unchecking a built-in tool immediately removes it from what the AI model can see *and* hard-blocks it at call time (a disabled tool refuses to run even if the model tries). The effect is process-wide — it applies to AI Chat, agents/teams and the node editor for the session.



With the Time server running, the count rises to "792 tools (790 internal · 2 external)".

Running an external server contributes its tools here automatically — the example above added 2 external tools the moment the Time server connected. Stop or remove the server and they disappear.

Note: internal-tool on/off state is session-scoped — it is held in memory and resets when the terminal restarts. (Plan-tier gating also disables tools whose category is above your subscription tier.)

At a glance, in TOOLS you can…

- Search the full internal + external tool list; read each tool's server, category and description.
- Enable or disable any built-in tool for the AI agent (session-scoped); see external tools appear as their servers run.

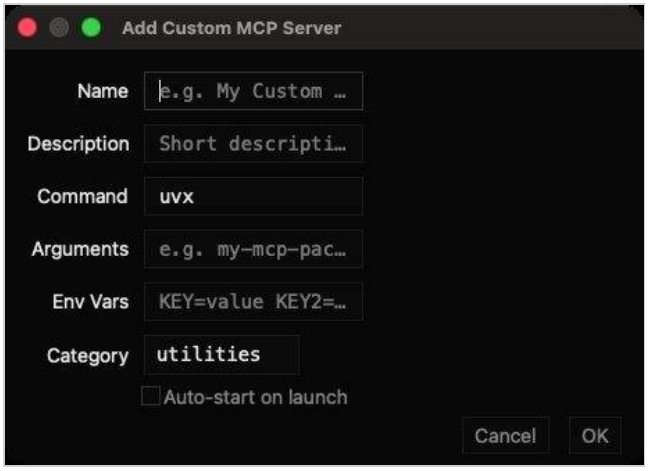
5 Adding a server — step by step

WALKTHROUGH

A) From the Marketplace (fastest)

1. Open **MARKETPLACE**, optionally filter by category or search.
2. Click **ADD** on a card. A dialog titled **"Add <Name>"** opens, pre-filled with the name, description, command and arguments.
3. Fill any **required env keys** (e.g. `DATABASE_URI` for PostgreSQL — the sample value is pre-selected so typing replaces it).
4. Optionally tick **Auto-start on launch**, then click **OK**. The server is saved and you land on **INSTALLED**. (Adding only registers it — it does not start it.)

B) A fully custom server



The Add Custom MCP Server dialog.

Field	Enter
Name	Display name (**required**).
Description	Optional blurb.
Command	Executable — defaults to `uvx` (also `npx`, `python` …).
Arguments	Space-separated args (e.g. `my-mcp-package --flag value`).
Env Vars	Space-separated `KEY=value` pairs.
Category	utilities / developer / database.
Auto-start on launch	Launch this server at app start.

Click **OK** to save (only **Name** is required), then start it from its card. Then flip the ENABLE toggle to launch it.

> **Two limits to know.** (1) In the custom dialog the **Env Vars** field is space-separated, so an env *value* that itself contains a space can't be entered there — use a Marketplace preset (one field per key) or keep values space-free. (2) Saving with an empty **Name** silently discards the dialog with no error message.

6 How servers connect

UNDER THE HOOD

Each external server is a locally-launched program the terminal talks to over a standard protocol. Knowing the model helps you configure and troubleshoot one.

Transport — stdio only

External MCP servers here are **local processes** spoken to via **JSON-RPC 2.0 over stdio** (standard in/out). The terminal spawns `command + args`, injects your env vars over the system environment, performs the MCP `initialize` handshake, then lists and calls the server's tools. There is **no SSE or HTTP/URL transport** — a server must be locally spawnable. (A `uvx …` command is transparently run through the terminal's bundled `uv` tool runner, which downloads the package on first use — hence a start can take up to a minute the first time.)

A server's configuration (the fields behind a card)

Field	Meaning
`id`	Stable key derived from the name (lower-cased, spaces→underscores).
`name · description · category`	Display + grouping.
`command · args`	The executable and its arguments.
`env`	Extra environment variables (merged over the system env) — where secrets like `DATABASE_URI` live.
`enabled · auto_start`	Whether it's active, and whether it launches at app startup (only `enabled && auto_start` auto-launch).
`status`	Live process state — Stopped / Starting / Running / Error (runtime; reset to Stopped on load).

Lifecycle

ENABLE (start) → **Starting** → handshake → **Running** (tools cached & exposed to the agent), or **Error** on failure. A concurrent-start guard prevents double launches. Health `ping` every 60s; on failure it auto-restarts up to 3× then is marked Error.

7 Persistence & security

DATA

- **Your server list roams with your account.** The whole configuration — each server's command, args, env, category, enabled and auto-start flags — is stored server-side against your account (synced through the Exclusive API), so it follows you across machines.
- **Internal tool on/off is session-only.** Enabling/disabling built-in tools is held in memory and resets on restart (it is not synced).

> **Security — MCP environment variables are NOT stored in the OS keychain.** Unlike broker API credentials (which are kept in the system keychain), any secrets you put in a server's env — database URIs, API keys — are saved inside your synced settings. Treat MCP env vars as account settings, not vault secrets: prefer least-privilege credentials (e.g. a read-only DB role) and avoid pasting long-lived production secrets.

8 Driving this tab from the AI agent

AGENTIC

The terminal's AI agent can manage this tab itself through 14 tools in the `mcp-servers` category. Mutating/lifecycle tools are marked destructive and ask for your confirmation.

- **Inspect:** list_mcp_servers · get_mcp_server · get_mcp_server_logs · list_external_mcp_tools · list_mcp_marketplace
- **Add / remove:** install_mcp_server_from_marketplace · add_mcp_server · remove_mcp_server
- **Configure:** set_mcp_server_enabled · set_mcp_server_autostart
- **Lifecycle:** start_mcp_server · stop_mcp_server · restart_mcp_server
- **Use:** call_external_mcp_tool (invoke a tool on a running server)

So you can simply ask the agent, e.g., "install the Git MCP server and start it," and approve the action — the same operations documented in this guide, performed for you.

9 Quick reference & important notes

HOW DO I...

I want to…	Do this
Add a ready-made server	MARKETPLACE ▸ **ADD** on a card ▸ fill any env keys ▸ OK.
Add my own server	INSTALLED ▸ **ADD CUSTOM MCP SERVER** ▸ set command/args/env ▸ OK.
Start / stop a server	INSTALLED ▸ the **ENABLED / DISABLED** toggle on its card.
See why a server won't start	Read the **"Server Failed to Start"** dialog, then open **LOGS** on the card.
Auto-launch a server at startup	Tick **Auto-start on launch** when adding it (can't be changed later without re-adding).
Turn off a built-in tool for the AI	TOOLS ▸ uncheck its **ON** box (session-scoped).
Delete a server	INSTALLED ▸ **REMOVE** ▸ confirm **Yes**.

Important notes for institutional users

- **Two tool sources.** ~790 built-in tools (server "Fincept Terminal") are always present and toggleable in TOOLS; external MCP servers are what you add/start in Marketplace & Installed.
- **stdio only.** External servers must be locally-spawnable processes (JSON-RPC over stdio); there is no remote-URL/SSE/HTTP transport.
- **First start is slow.** A `uvx` server downloads its package on first launch — allow up to ~60 seconds; the toggle shows ↻ STARTING… meanwhile.
- **Env vars are account settings, not keychain secrets** — use least-privilege credentials (see §7).
- **Internal-tool toggles reset on restart;** external-tool checkboxes are display-only.
- **Config roams** with your account across machines.

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.

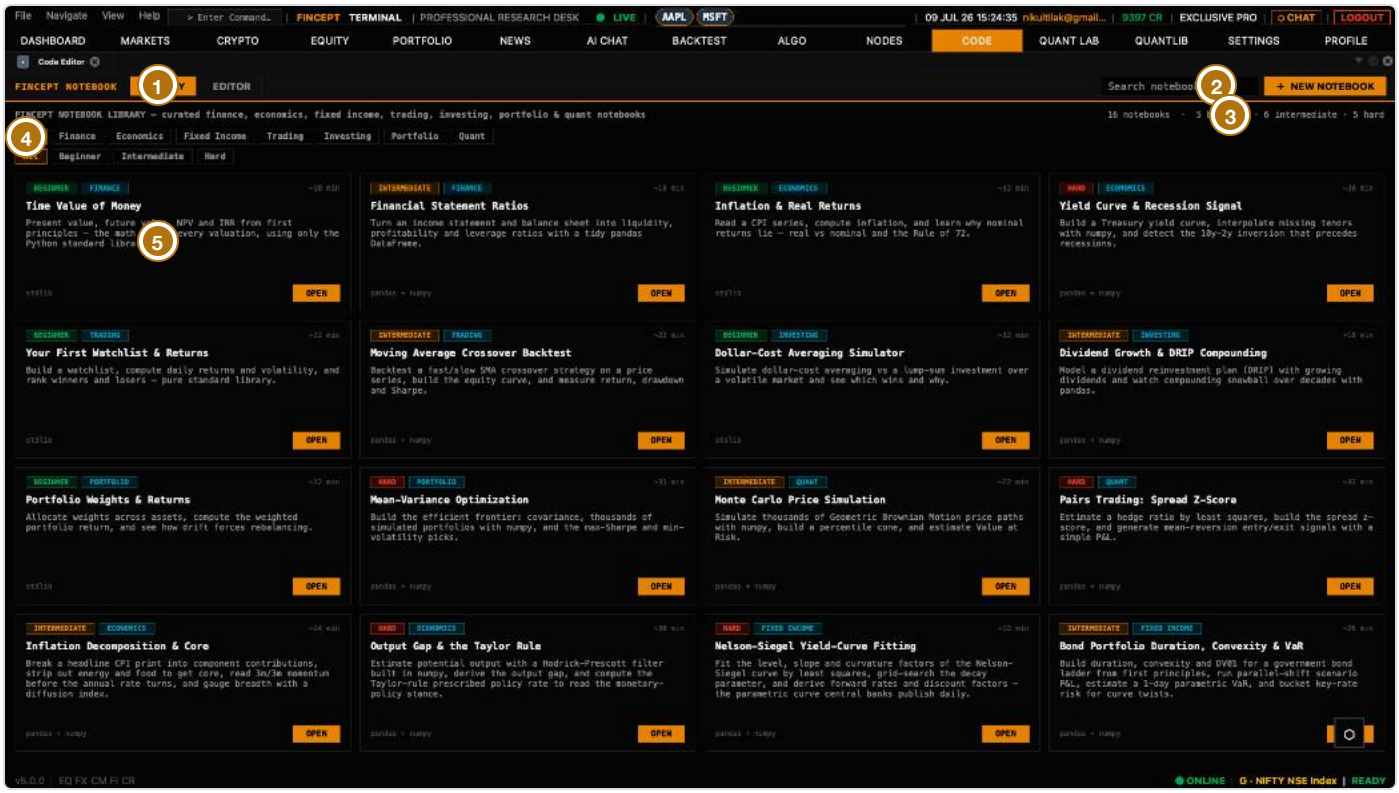
At a glance, you can…

- 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.

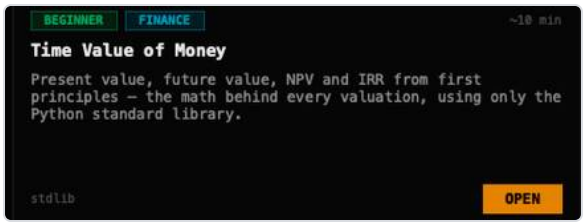


Search & new notebook.

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.

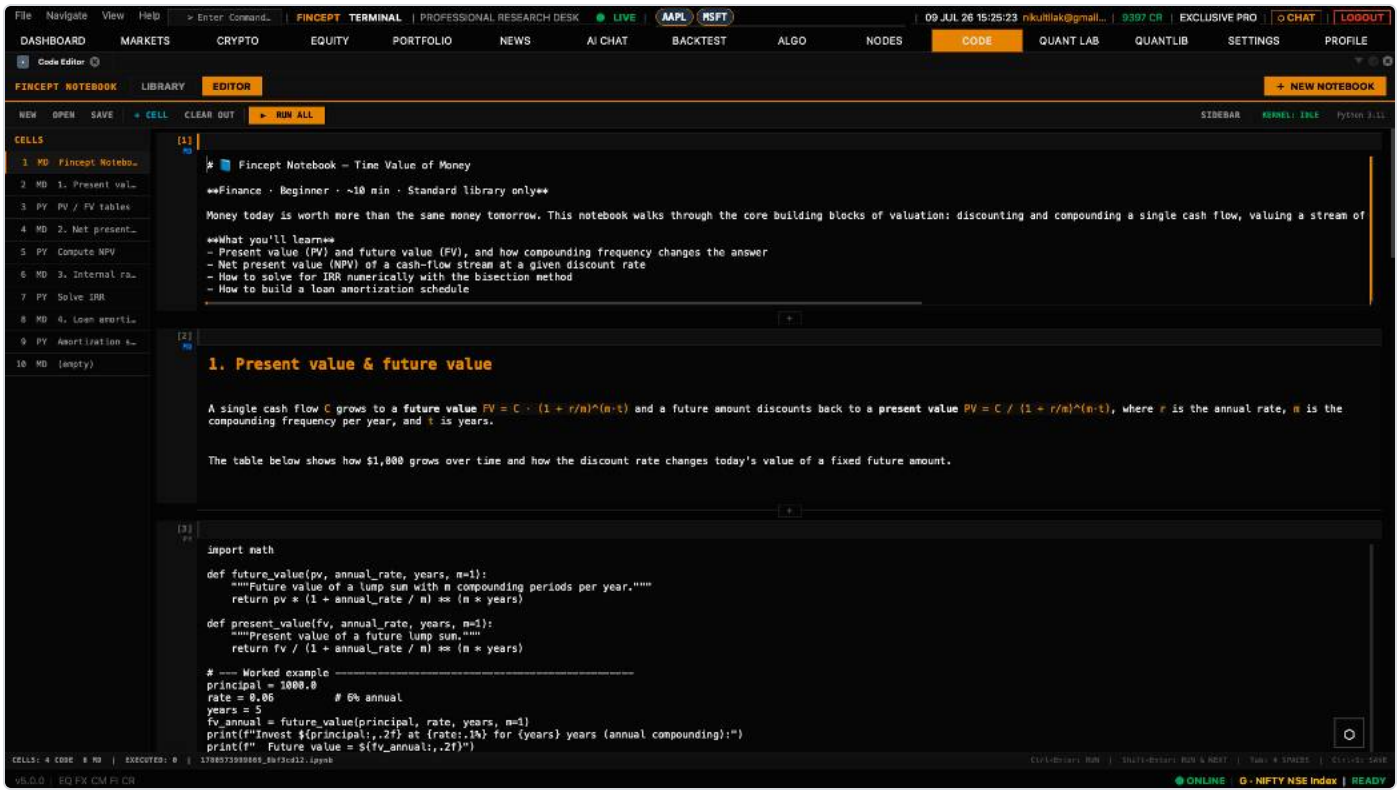


Filtering to `Economics` narrows the grid to just those notebooks.

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.

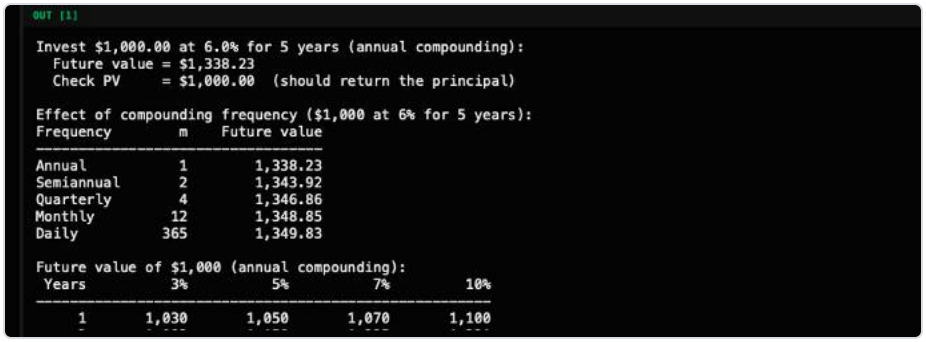


Hover a cell for its controls: **RUN** · **TYPE** (code⇄Markdown) · **UP** / **DN** (reorder) · **DEL**. The gutter shows the run count, e.g. `[1] PY`.

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



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

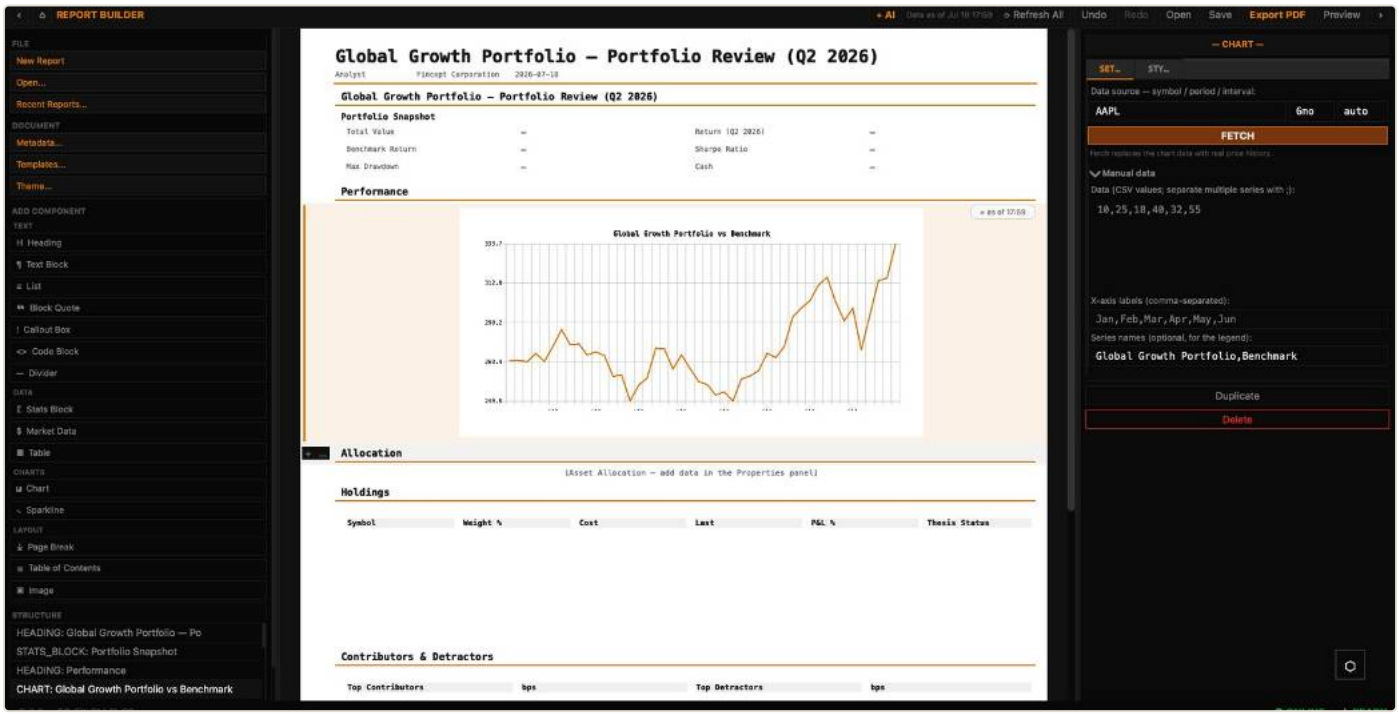
WHERE TO FIND IT

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

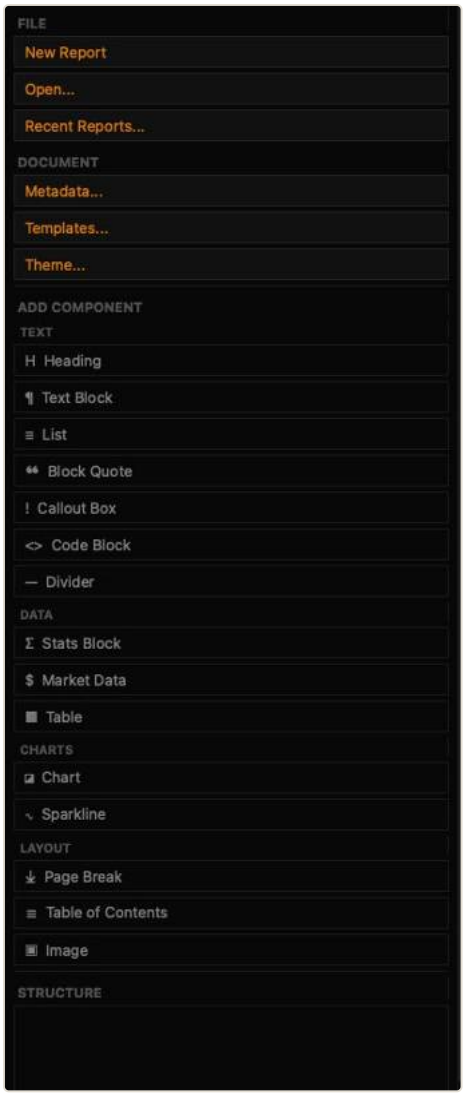
ANATOMY · 4 REGIONS

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

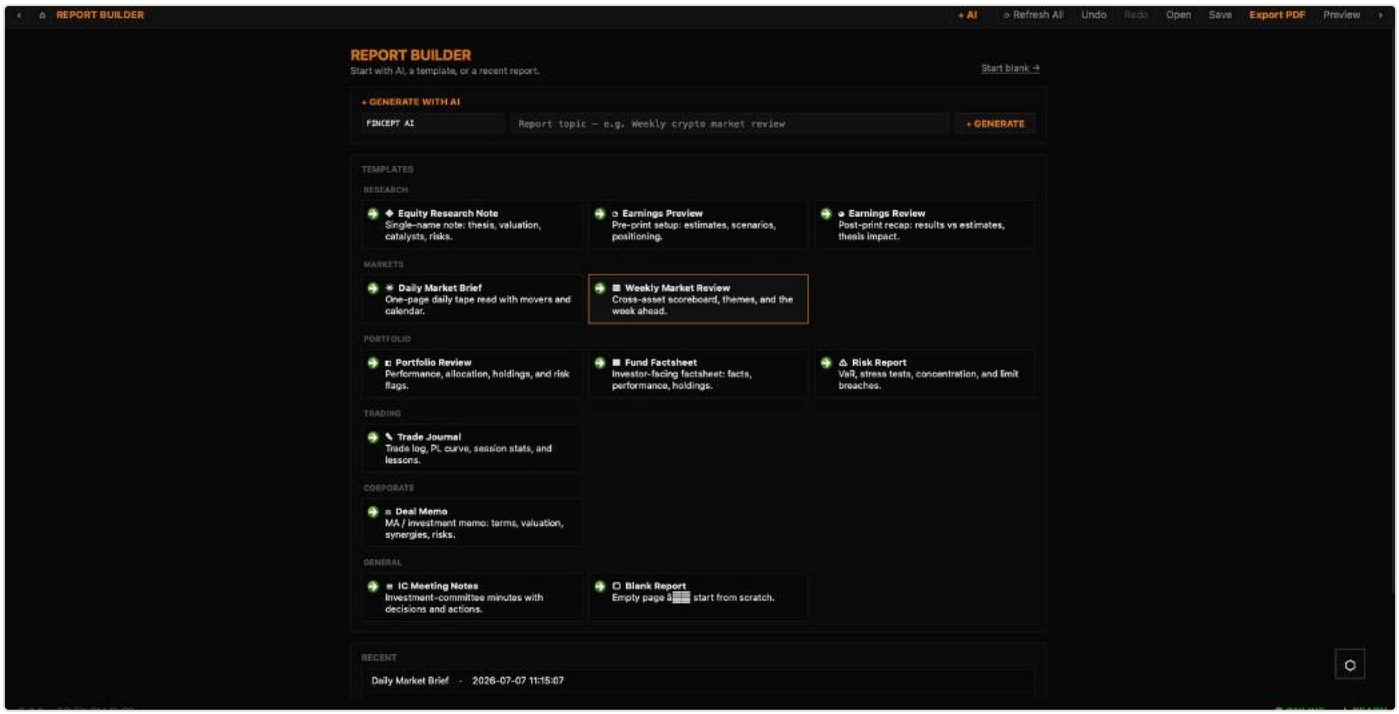
FILE — New Report (confirms first), Open… (`.fincept` / `.json`), Recent Reports… (your stored library).
DOCUMENT — Metadata… (title, header/footer, branding), Templates… (jumps to the Home gallery), Theme… (four colour presets, live preview).
ADD COMPONENT — the same 15 blocks the slash menu offers, 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.	—

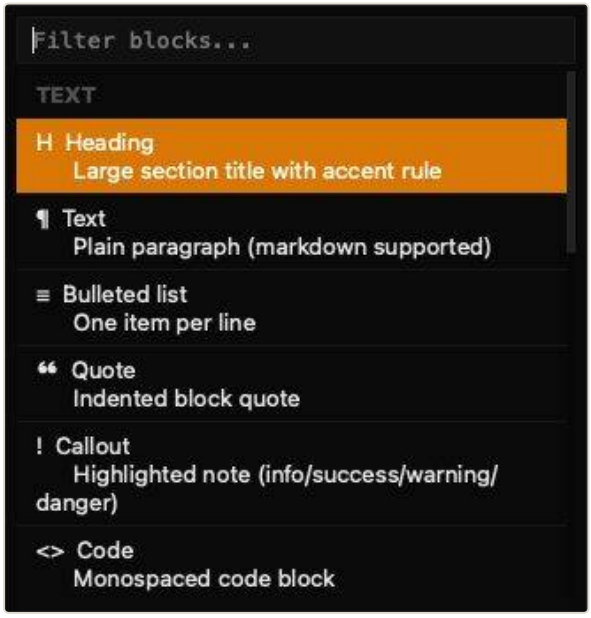
LEGACY NAMES STILL WORK

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

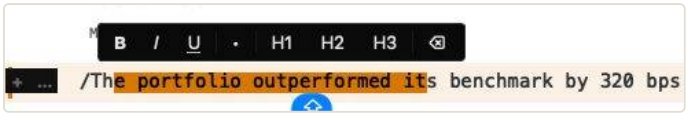
Type `/` to open the **slash menu** — start typing to filter ("chart", "call"…). Or hover a block's left edge and click the `[+]` handle to insert below. The rail's ADD COMPONENT group does the same thing by category.

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

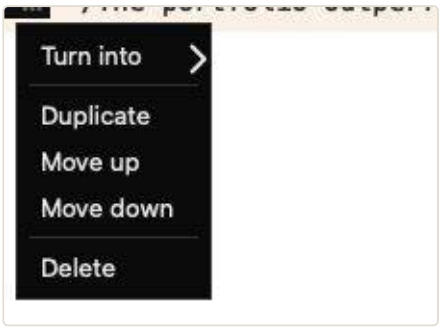
Select text in any editable block and a floating toolbar appears: **B** bold, **I** italic, **U** underline, • turn into a bulleted list, **H1/H2/H3** turn into a heading, and ⌫ to strip all formatting marks.



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.

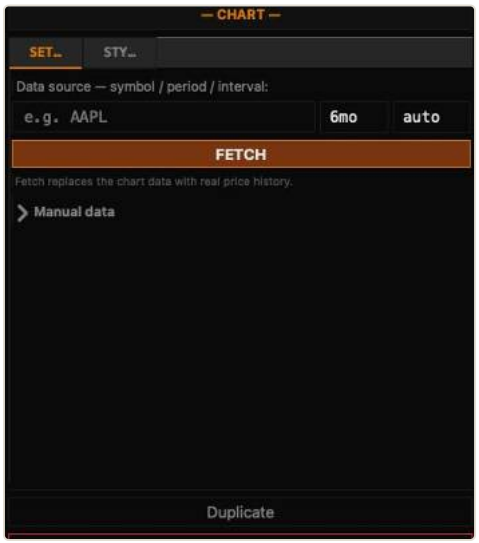


Chart · SETUP. Bind a symbol, choose a period & interval, and **FETCH** real price history — or open **Manual data** to enter CSV.

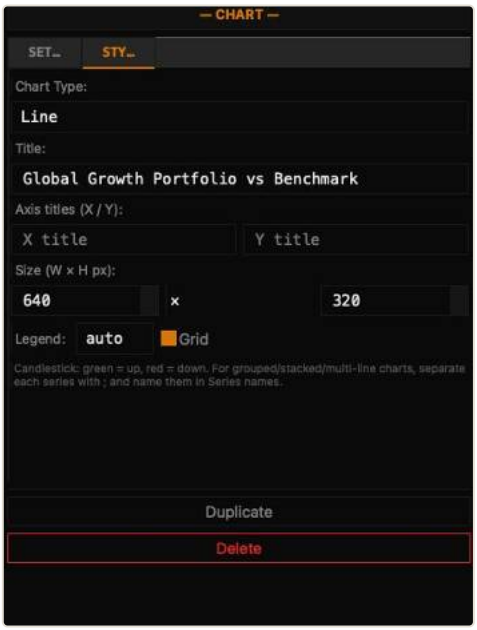


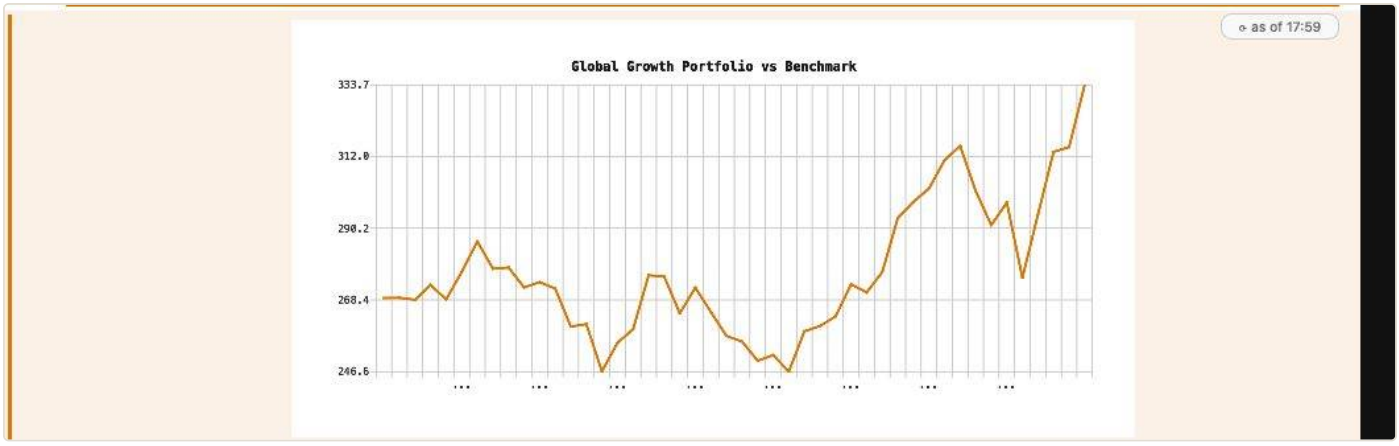
Chart · STYLE. Pick one of 10 chart types, set the title, X/Y axis labels, width × height, legend mode and grid.

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.

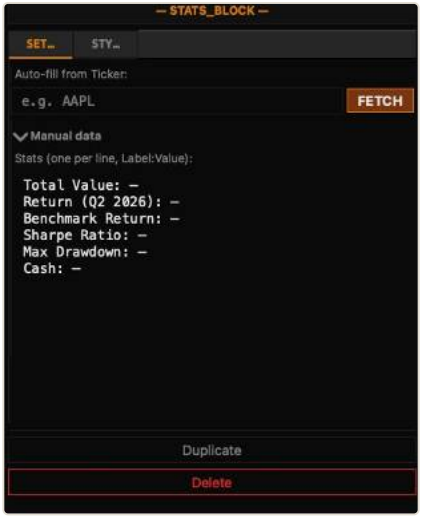


A fetched price chart on the page, with its live "⟳ as of 17:59" refresh chip. The chart renders identically here and in the exported PDF.

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.

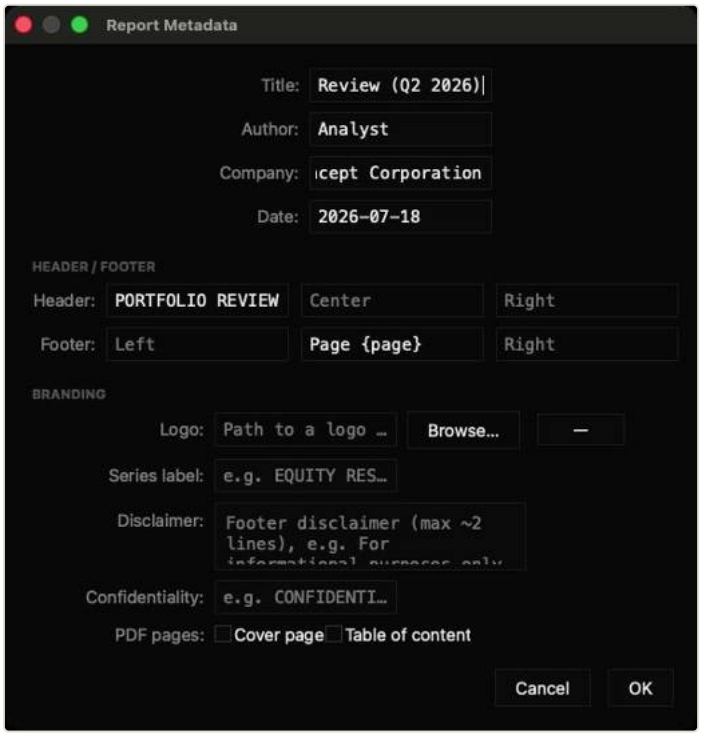
HONEST NOTES ON DATA

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.

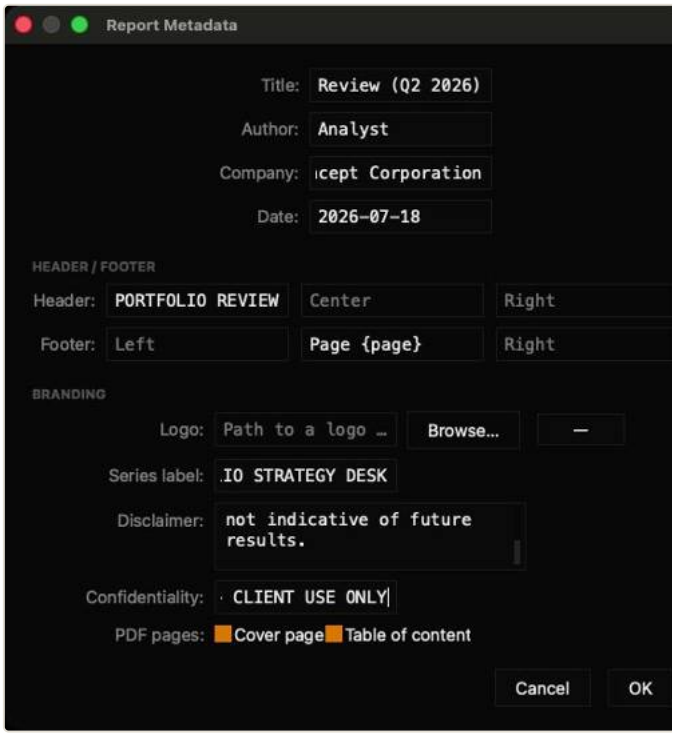
7 Document setup — metadata, branding & themes

BRANDING · 4 THEMES

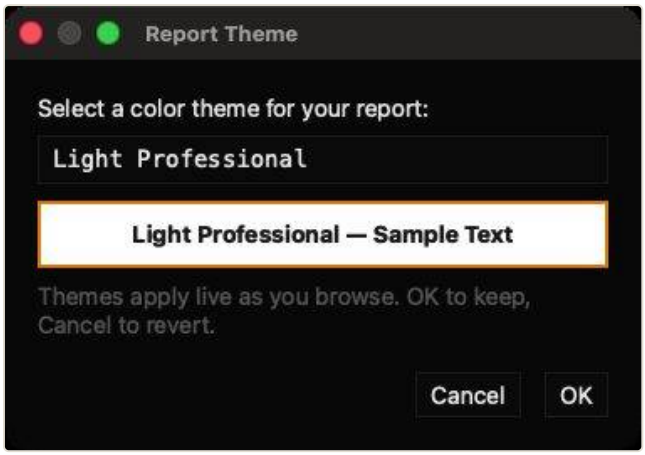
Two small dialogs control how the finished PDF looks. **Metadata…** sets the title block, running header/footer and the institutional furniture — logo, series label, disclaimer, confidentiality line, and whether to add a cover page and contents. **Theme…** recolours the whole report.



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.

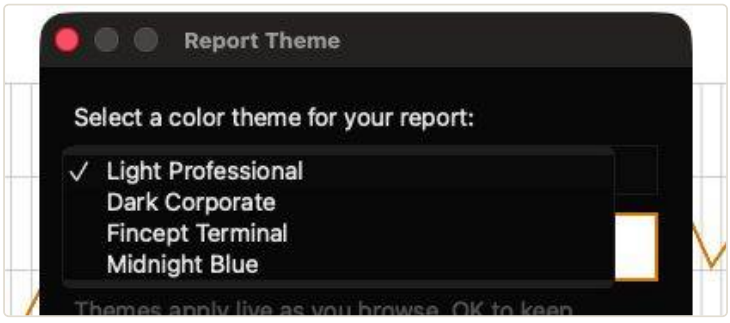


Filled in. A series label, disclaimer and confidentiality line are set, and both cover and contents are switched on — these drive the PDF's front matter and every page's header/footer.



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.

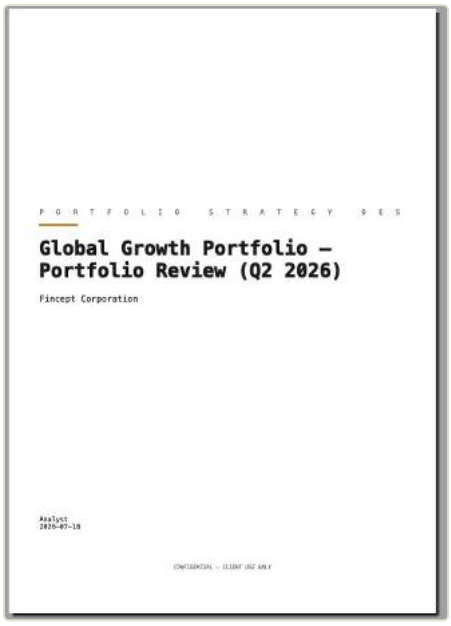
Light Professional is the default and the right choice for a printed client document. The others recolour headings, rules, table headers and callouts to a house palette.

> **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.

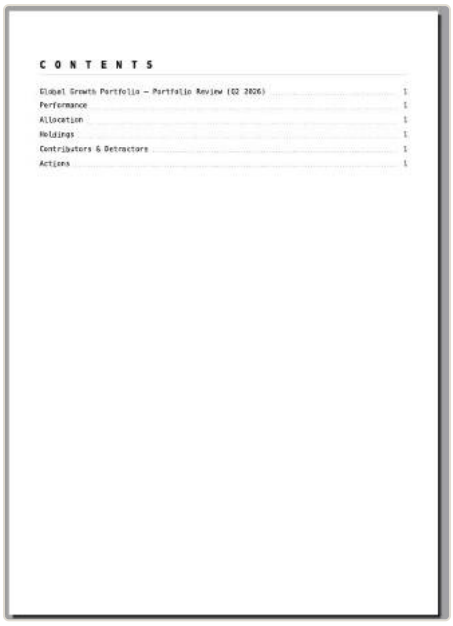
8 Export & the PDF

PREVIEW == EXPORT

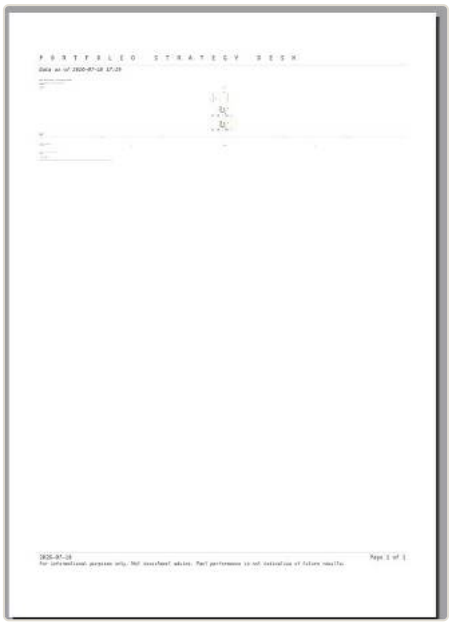
Report Builder has a single rendering path, so **Preview** shows exactly what **Export PDF** writes — cover, contents, running header and footer included. With branding set (§7), the tab produces a genuinely client-ready A4 document.



Cover page. Series label in letter-spaced caps, the report title, company, author & date, and the confidentiality line at the foot.

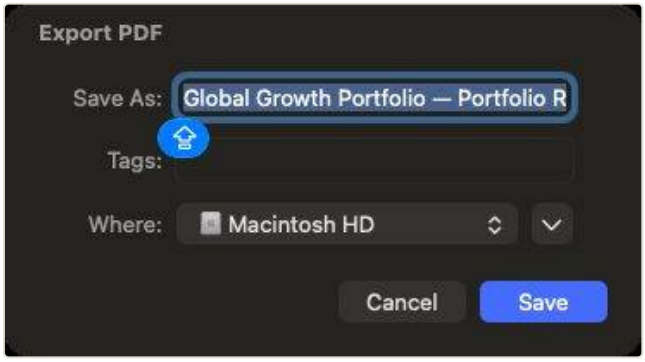


Contents. Auto-built from every Heading, with dotted leaders and page numbers.



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.

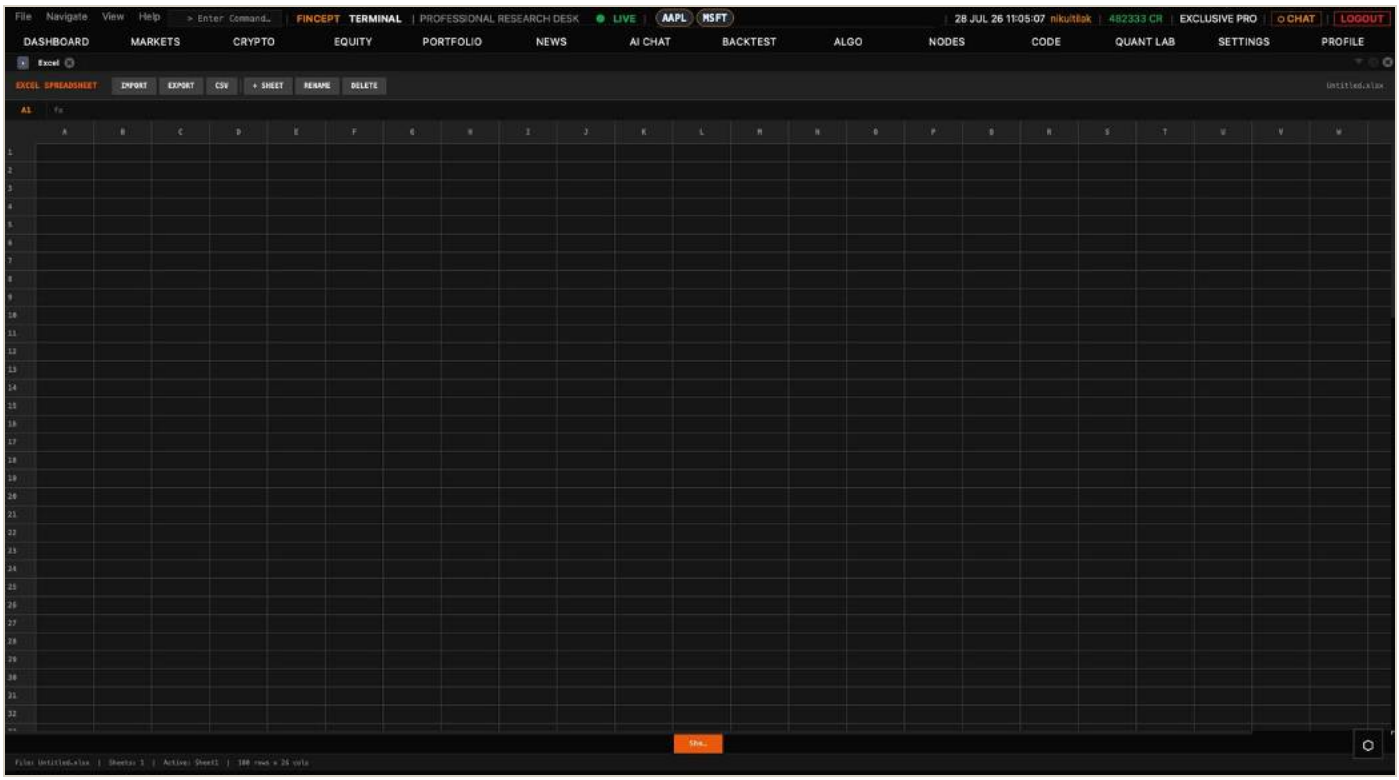


Fig 1 — The Excel tab on open, with all five regions visible.

#	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.

```
File: Untitled.xlsx  |  Sheets: 2  |  Active: Sheet2  |  100 rows x 26 cols
```

Fig 2 — The status bar answers four questions at a glance, and updates on every sheet add, delete, rename and switch.

03 The toolbar

TOOLBAR · 03

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**

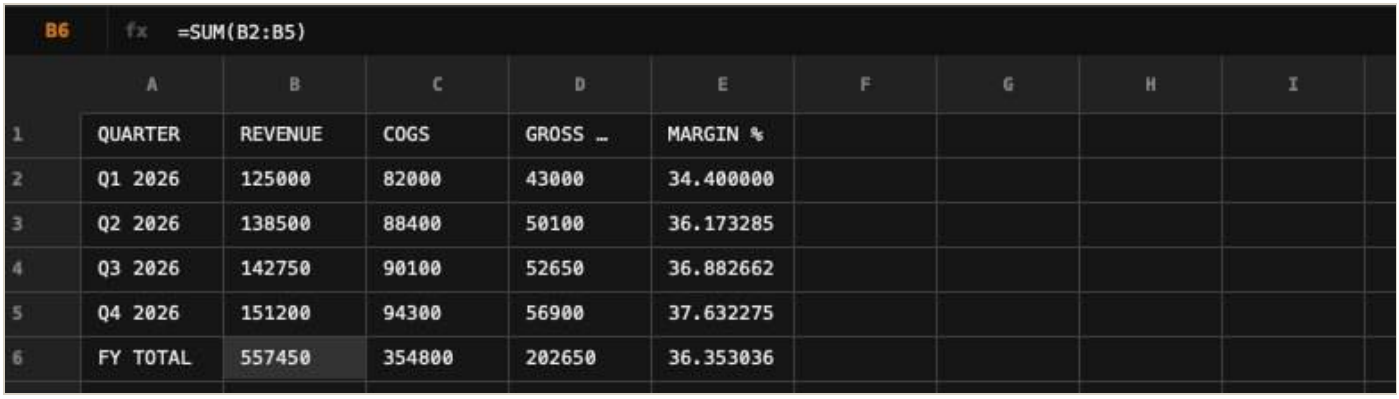


Fig 4 — The reference box reads **B6**, the **fx** field shows the raw formula `=SUM(B2:B5)`, and the cell itself shows the result, 557450.

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

LIMITS · 06

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.

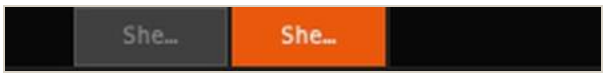


Fig 6 — Two sheet tabs, the active one amber. At the default width the labels clip to a few characters, so keep sheet names short and read the full name from the status bar.

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

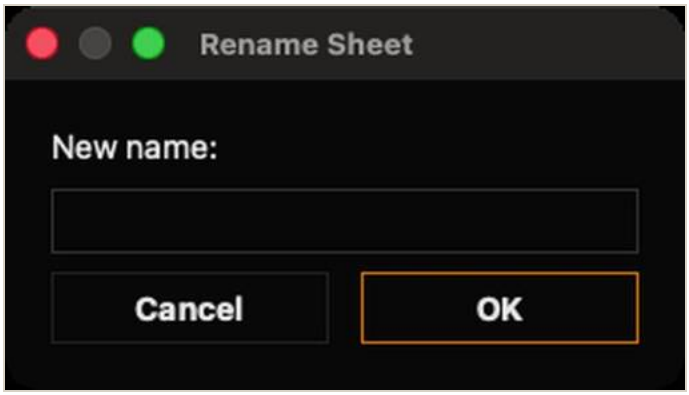


Fig 7 — **RENAME** opens a small dialog pre-filled with the current name.

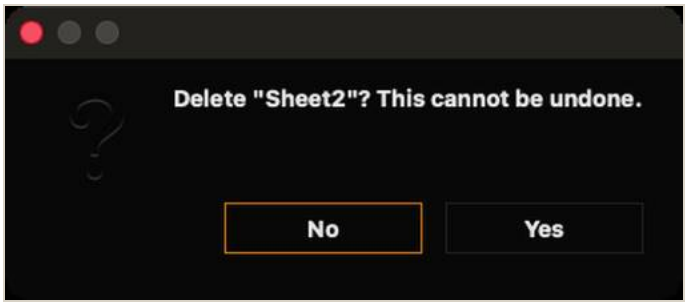


Fig 8 — **DELETE** always confirms, and defaults to **No**.

> 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

CELL MENU · 08

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

FILE IO · 09

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

PERSISTENCE · 10

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.

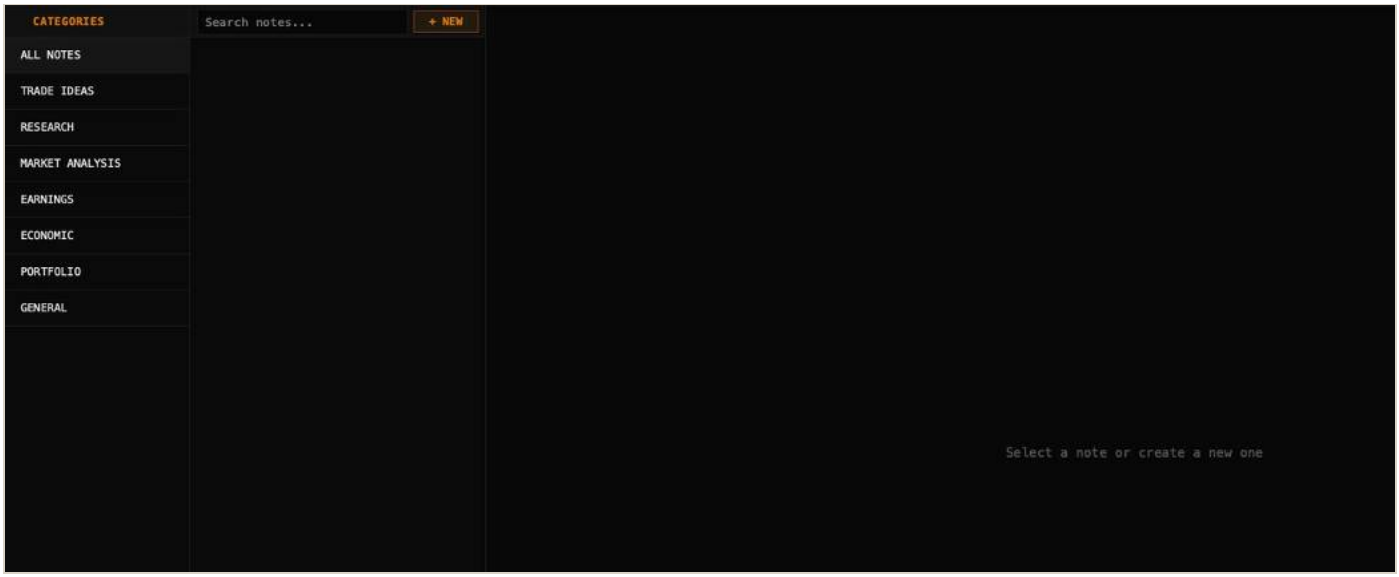


Fig 1 — A fresh journal. The eight categories are always present; the middle and right panels fill in as you write.

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.

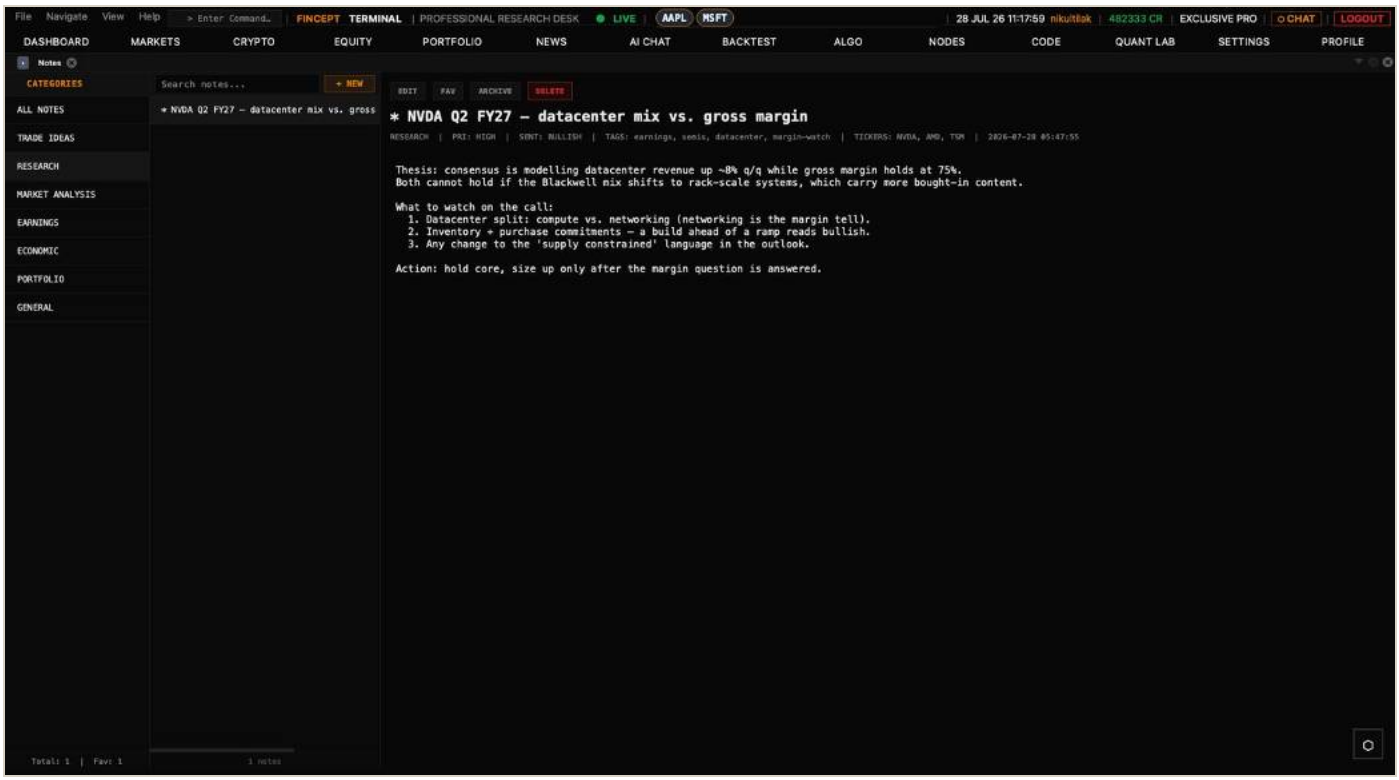


Fig 2 — The Notes tab with one note selected. Left: categories. Middle: the notes in the selected category. Right: the note in view mode.

#	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

SIDEBAR · 03

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.

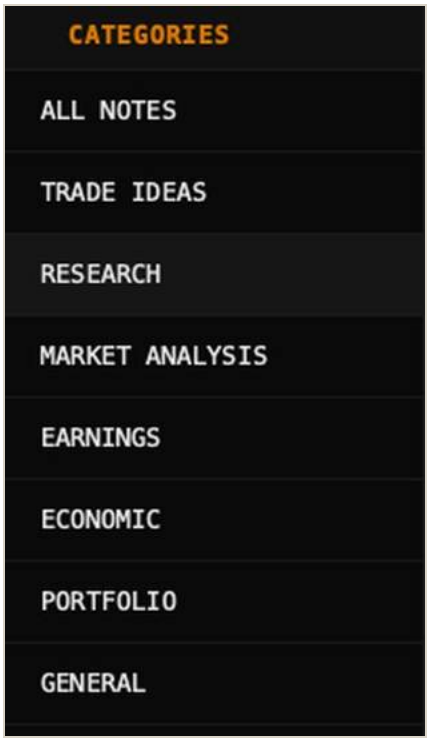


Fig 3 — The category sidebar.

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.

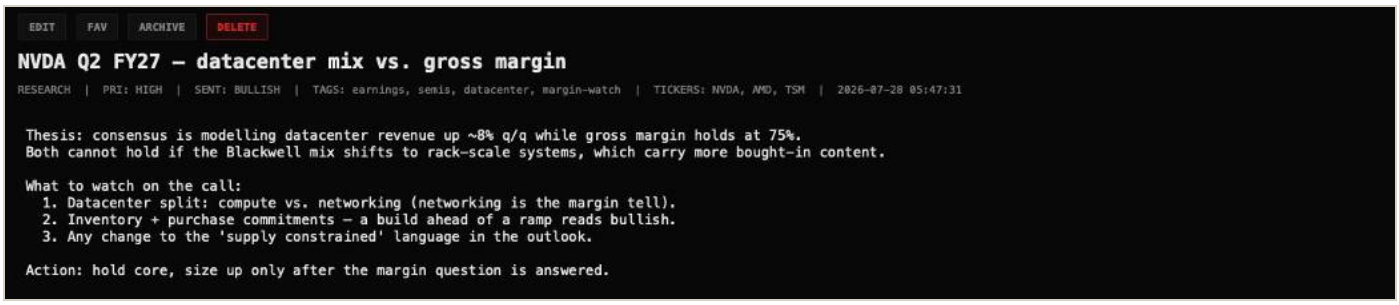


Fig 6 — View mode. The * in the title marks a favourite; the metadata line carries category, priority, sentiment, tags, tickers and the last-updated timestamp.

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

EDIT MODE · 06

The editor is one screen: a toolbar, the title, two rows of metadata, and the body.

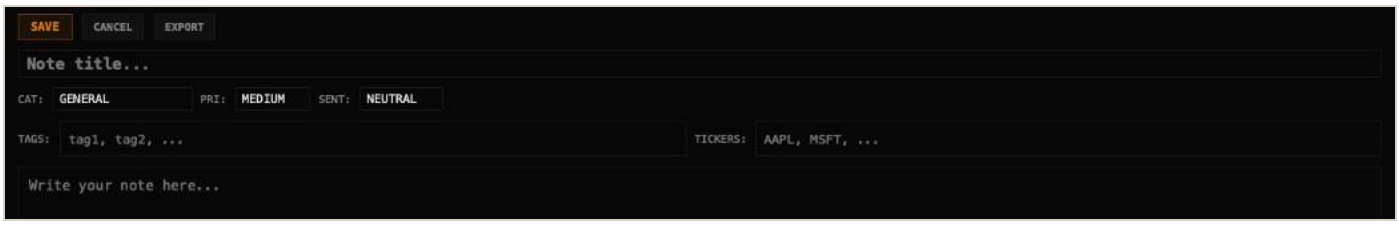


Fig 7 — A blank editor from **+ NEW**. Priority defaults to MEDIUM, sentiment to NEUTRAL, and the category to whichever one you were standing in.

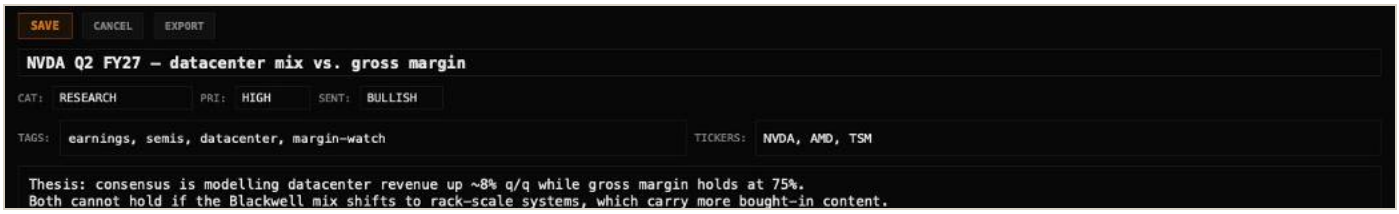


Fig 8 — The same editor filled in: a RESEARCH note, HIGH priority, BULLISH, tagged, and tickered NVDA, AMD, TSM.

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

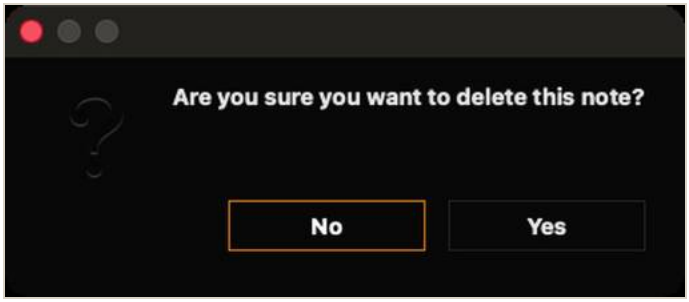


Fig 10 — The delete confirmation defaults to **No**.



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

AGENT TOOLS · 11

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.

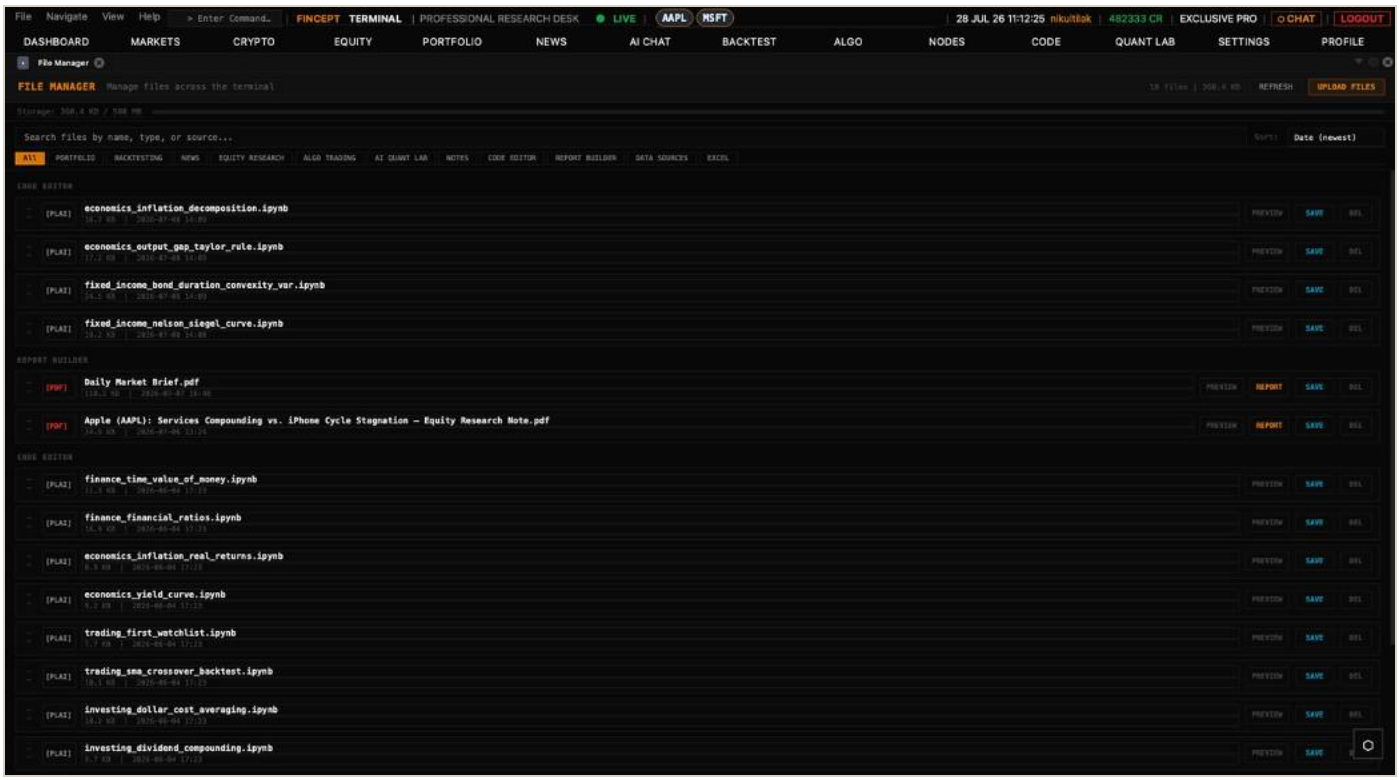


Fig 2 — The File Manager on a working index of 18 files, grouped by the screen that produced each one.

#	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

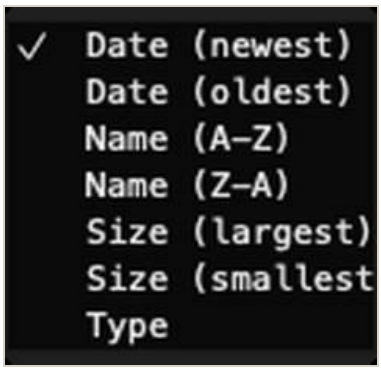


Fig 5 — All seven sort options.

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

FILE CARD · 05

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.

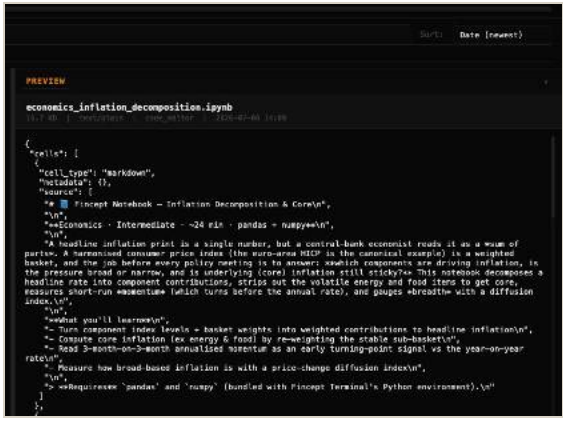


Fig 8 — Previewing a notebook. The metadata line reads *16.7 KB | text/plain | code_editor | 2026-07-08 14:09*.

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.

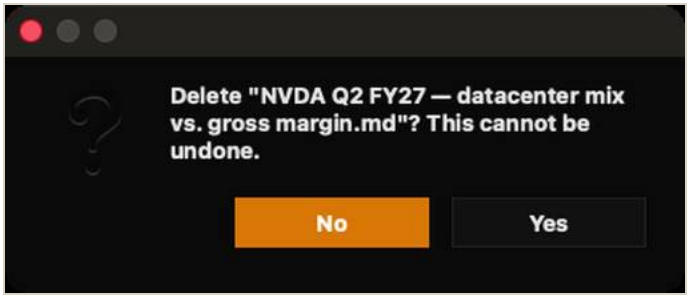


Fig 10 — The single-file confirmation names the file.

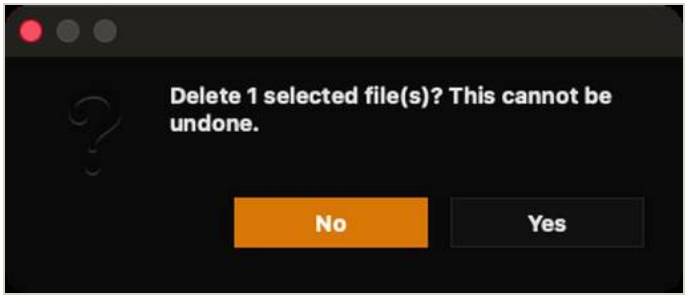


Fig 11 — The bulk confirmation counts them. Both default to **No**.

> **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

SOURCES · 08

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 · 8 profile · 15 sub-tabs · 27 credential slots`

Covers · **General, Appearance, Notifications, Keybindings** · **Voice, Logging, Developer**
Profiles · Credentials · Security · Data Sources · MCP Servers
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 Voice — speech-to-text 10

09 Voice — speech, keys and the wake trigger 11

10 Logging — levels and the log file 12

11 Logging — shipping logs to the backend 13

12 Developer 14

PROFILE — YOUR ACCOUNT AND CONNECTIONS

13 Profiles 15

14 Credentials 16

15 Security 17

16 Data Sources 18

17 MCP Servers 19

18 Python Env 20

19 Storage — disk usage and categories 21

20 Storage — files, state and the SQL console 22

21 Cloud 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.

01 What the Settings screen is

ORIENTATION · 01

Settings configures the terminal itself — not one form, but fifteen 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, tool servers, the Python environment and cloud sync.

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
- Configure speech-to-text and text-to-speech
- Arm a clap-to-talk wake trigger
- Tune log levels per subsystem
- Ship five log streams to the backend
- Hold 27 third-party API keys
- Set a PIN, auto-lock and lock-on-minimize
- Manage data connections and MCP tool servers
- Repair the embedded Python environment
- See disk usage and clear cached data
- Run SQL against the local databases
- Choose which of 10 domains sync to the cloud

> **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 two exceptions are called out where they occur.

02 Anatomy — the rail and the two scopes

ANATOMY · 02

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.

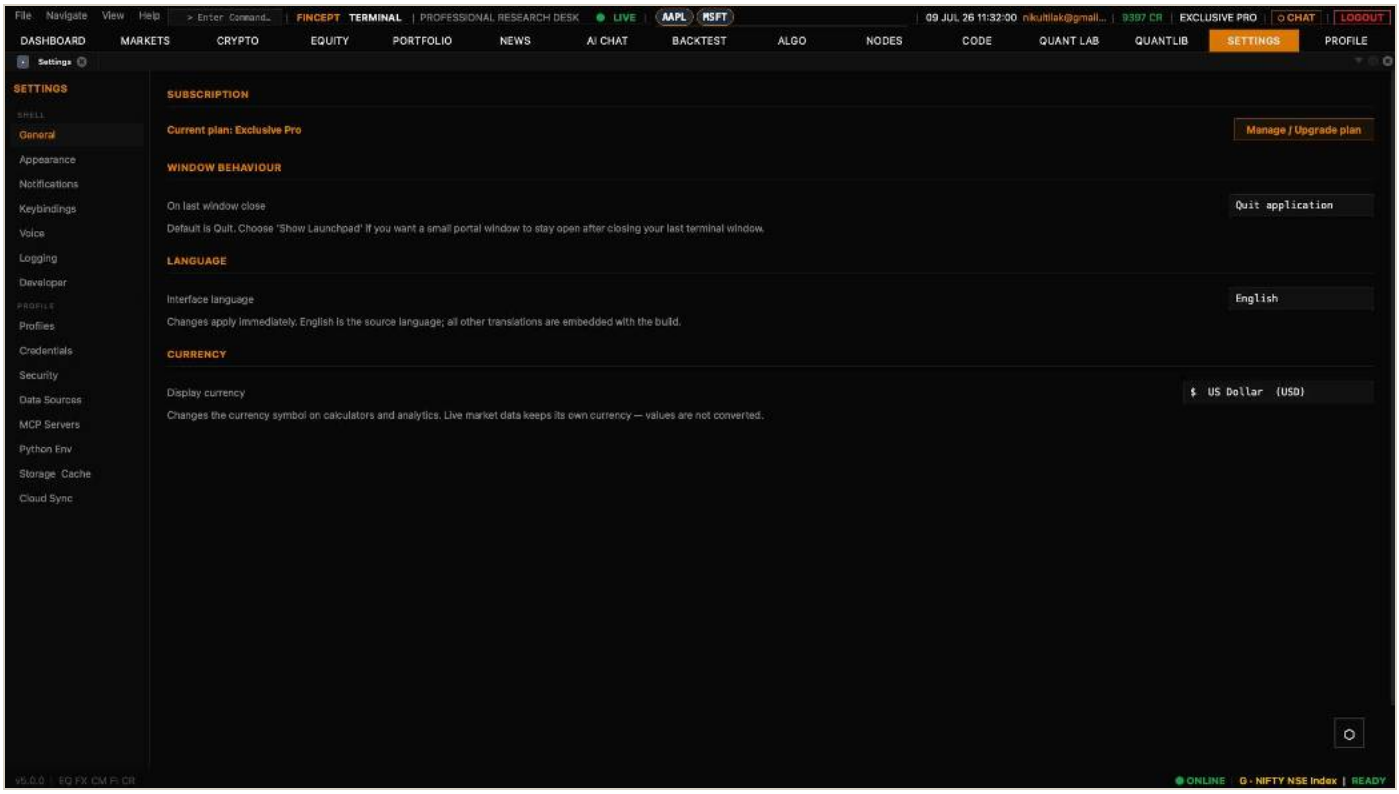


Fig 1 — The Settings screen on open. The rail lists SHELL (7 entries) then PROFILE (8); General is selected by default.

Region	What it does
`Rail header`	Reads SETTINGS. Fixed width, always visible.
`SHELL`	General, Appearance, Notifications, Keybindings, Voice, Logging, Developer.
`PROFILE`	Profiles, Credentials, Security, Data Sources, MCP Servers, Python Env, Storage & Cache, Cloud Sync.
`Panel`	The selected sub-tab — a scrolling stack of titled cards.

> Each panel is built from the same parts: a card title, then rows of *label — control* with an explanatory line underneath. Where this guide quotes that line, it is the text the screen itself shows.

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.

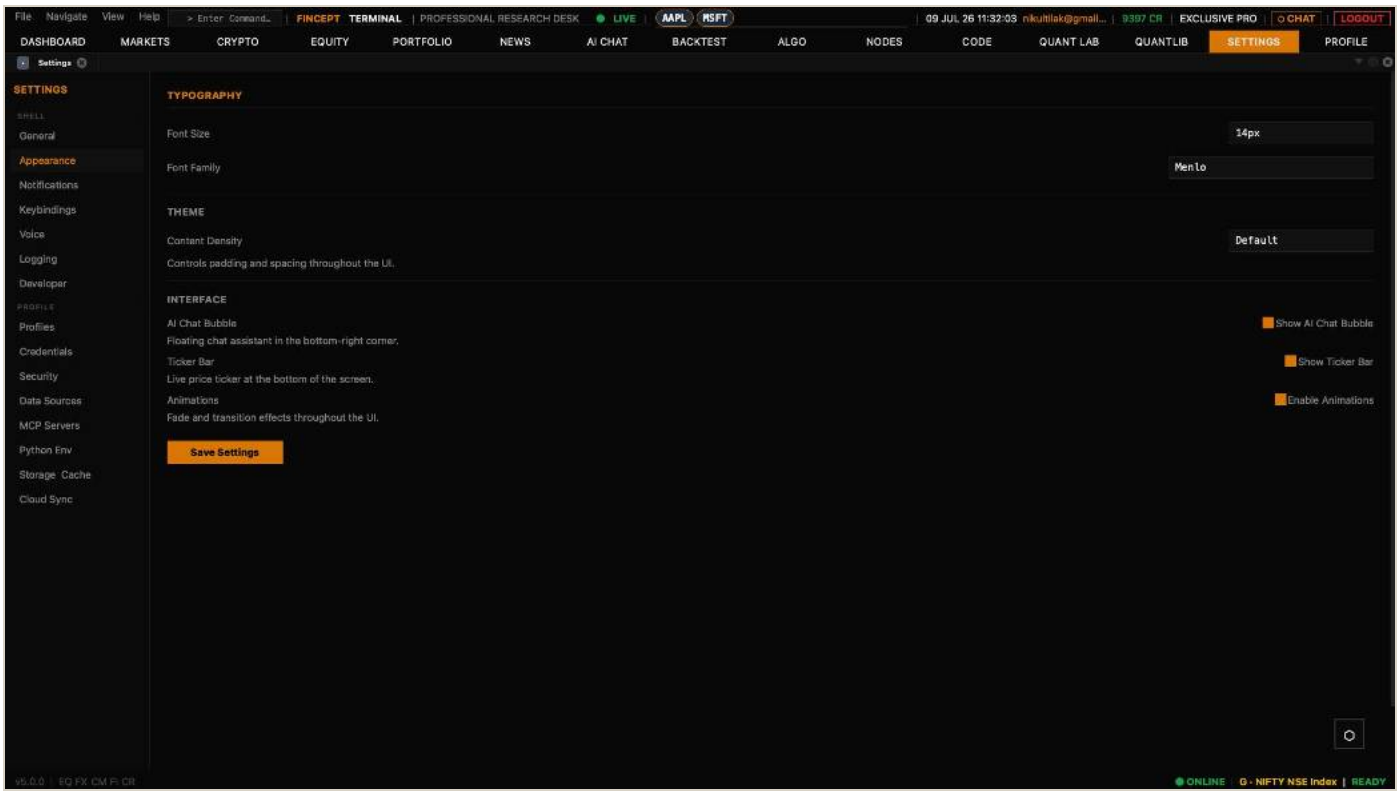


Fig 2 — Appearance. Each card is a stack of labelled rows with the control on the right.

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

PROVIDERS · 05

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.

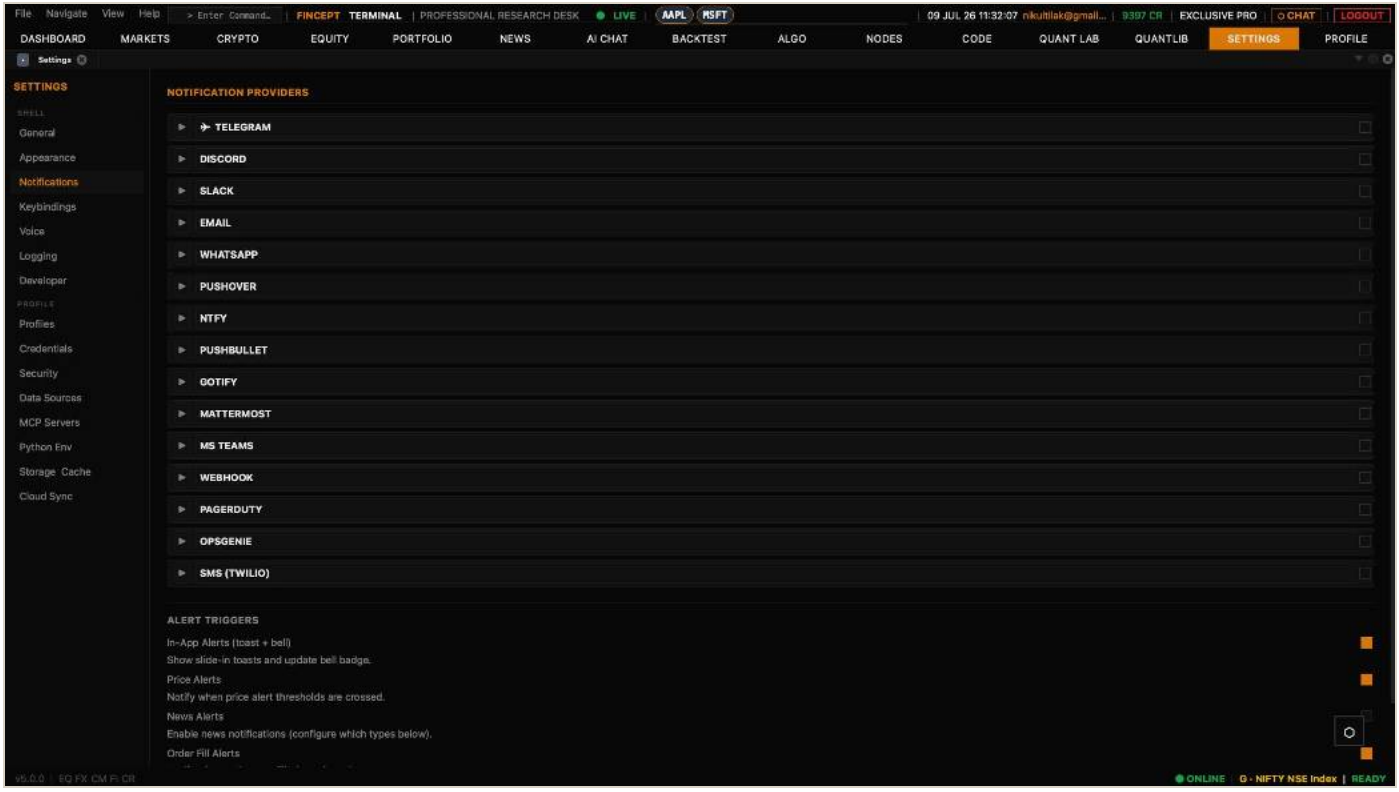


Fig 3 — Notifications. Providers expand to reveal only the fields that provider needs.

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

TRIGGERS · 06

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.

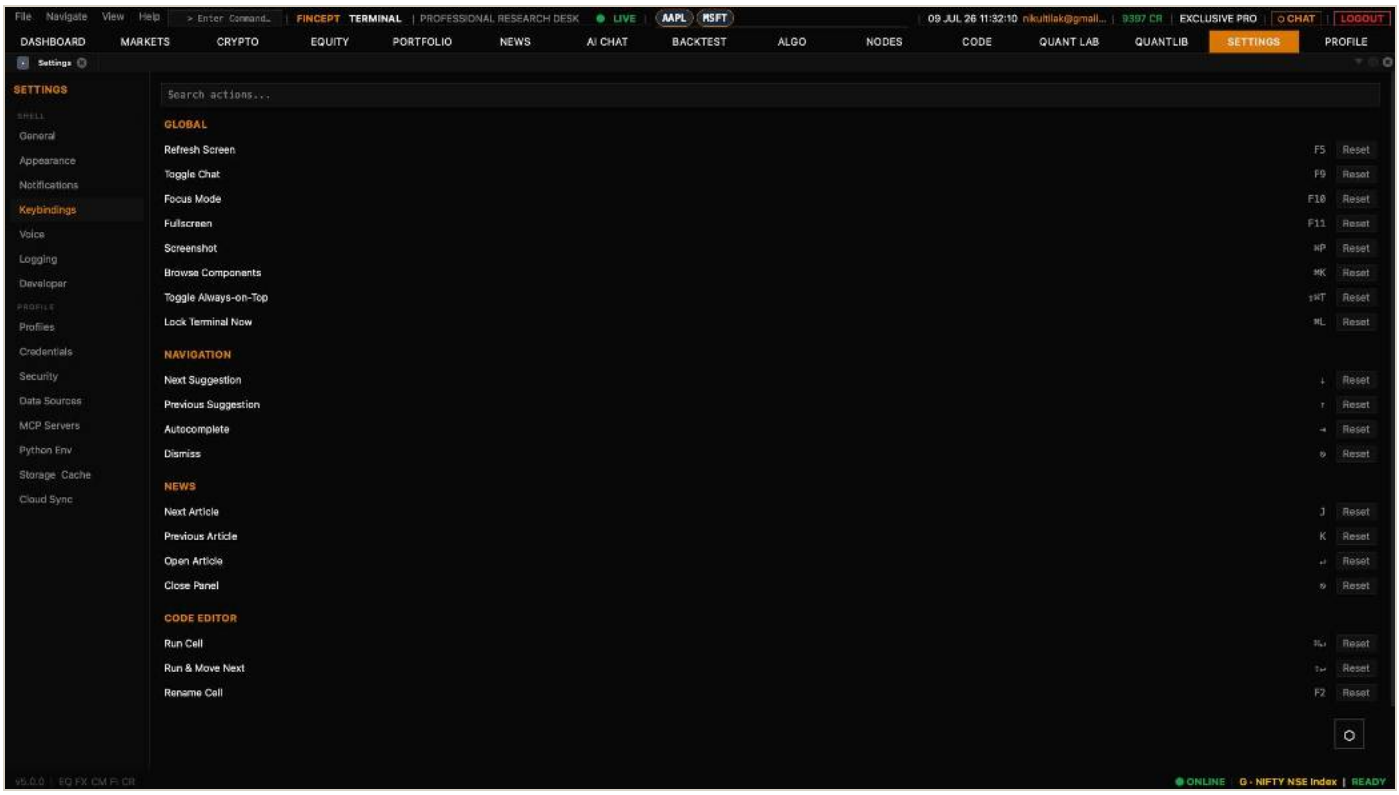


Fig 4 — Keybindings. One row per action; the current combination sits on the right with a Reset beside it.

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 Voice — speech-to-text

SPEECH IN · 08

The sub-tab covers both directions and says so at the top: *pick a provider for each direction independently*. This page is the microphone half.

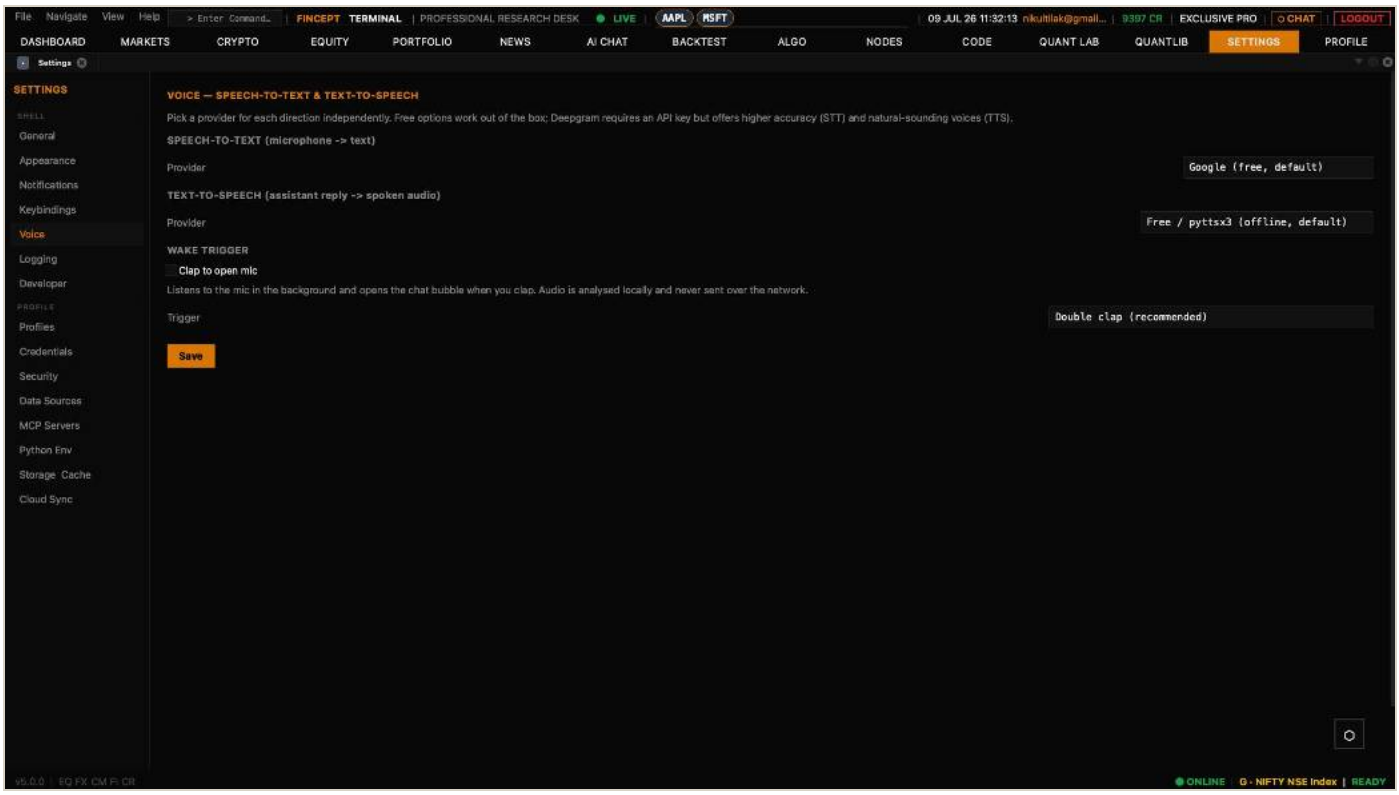


Fig 5 — Voice. SPEECH-TO-TEXT above, TEXT-TO-SPEECH below, shared Deepgram credentials and the wake trigger at the foot.

Control	Options / meaning
`Provider`	Google (free, default) · Deepgram (API key required)
`STT Model`	nova-3 (recommended) · enhanced · base
`Language`	English (auto) · English (US) · English (UK) · Multilingual (nova-3 only)
`Key terms`	Comma-separated recognition hints — e.g. `AAPL, BTCUSD, Nifty`
`Mic gain`	Raise if the microphone is quiet.
`Mic device`	Substring match on the device name — e.g. `Headset`. Blank uses the system default.

> **Key terms are worth filling in.** Tickers and index names are exactly the vocabulary general speech models get wrong; listing the ones you say often biases recognition toward them.

> Multilingual is only available on the **nova-3** model. Selecting it against *enhanced* or *base* will not give you multilingual recognition.

09 Voice — speech, keys and the wake trigger

SPEECH OUT · 09

The output half, the shared credentials card, and the hands-free trigger.

Text-to-speech	Options
`Provider`	Free / pyttsx3 (offline, default) · Deepgram Aura-2 (API key required)
`Voice`	Nine Aura-2 voices — Thalia, Helena, Andromeda, Aurora, Luna (female) and Apollo, Orion, Arcas, Zeus (male), all American

Deepgram credentials

One card, headed **DEEPGRAM CREDENTIALS (shared by STT + TTS)** — a single **API Key** field serves both directions. **Show** / **Hide** reveals the value and **Test Deepgram key** checks it before you depend on it.

Wake trigger	Options
`Clap to open mic`	Listens to the microphone in the background and opens the chat bubble when it hears the trigger.
`Trigger`	Double clap (recommended) · Single clap (more reactive, more false positives)

> Pressing **Save** reloads both the speech and text-to-speech services and restarts the clap detector, so a new provider, key, voice or trigger takes effect without a restart.

> **The wake trigger holds the microphone open.** Clap-to-open listens continuously in the background. Single clap is more reactive and, as the screen itself says, produces more false positives.

10 Logging — levels and the log file

Six levels throughout: `Trace`, `Debug`, `Info`, `Warn`, `Error`, `Fatal`.

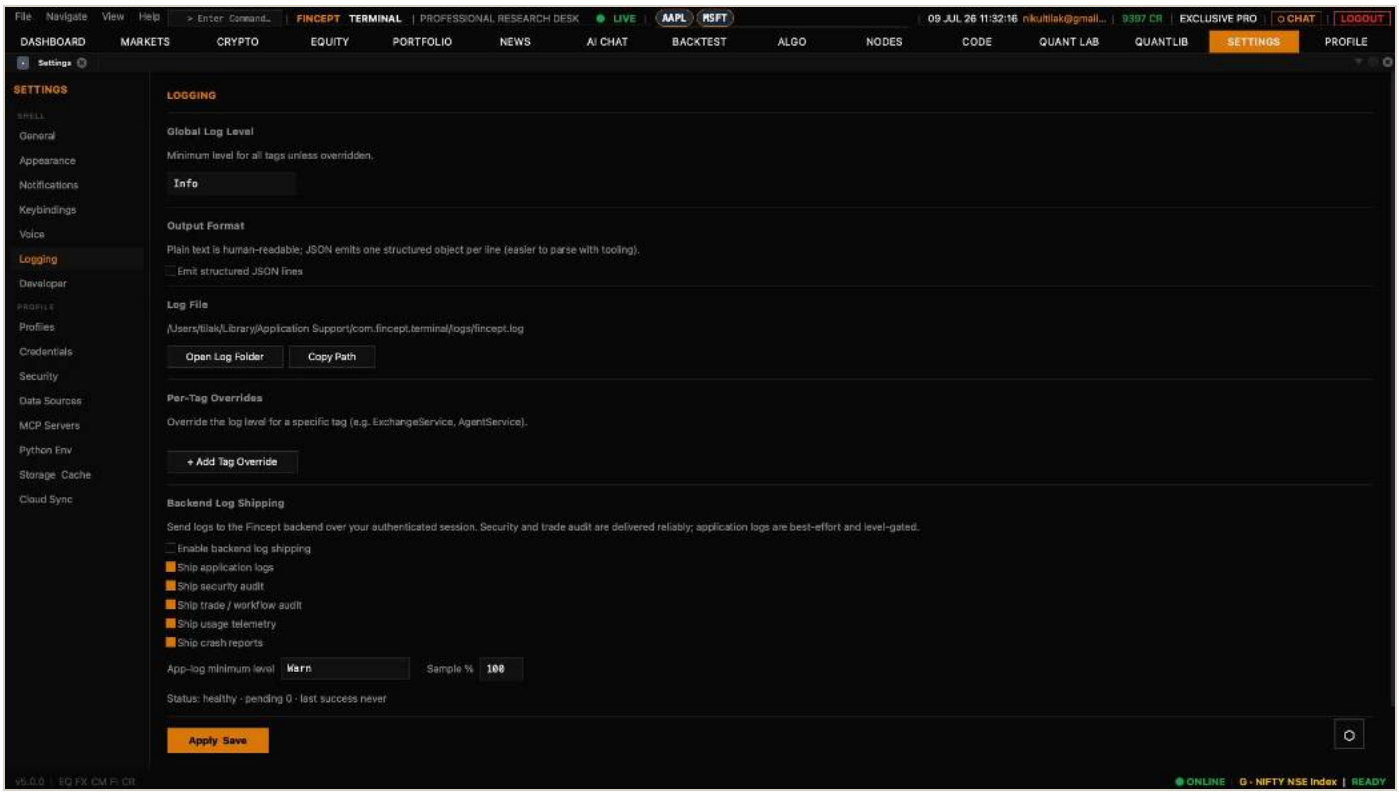


Fig 6 — Logging. Global level and format at the top, per-tag overrides in the middle, backend shipping at the foot.

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.

11 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.

12 Developer

The smallest sub-tab, and the only one whose switches are labelled experimental.

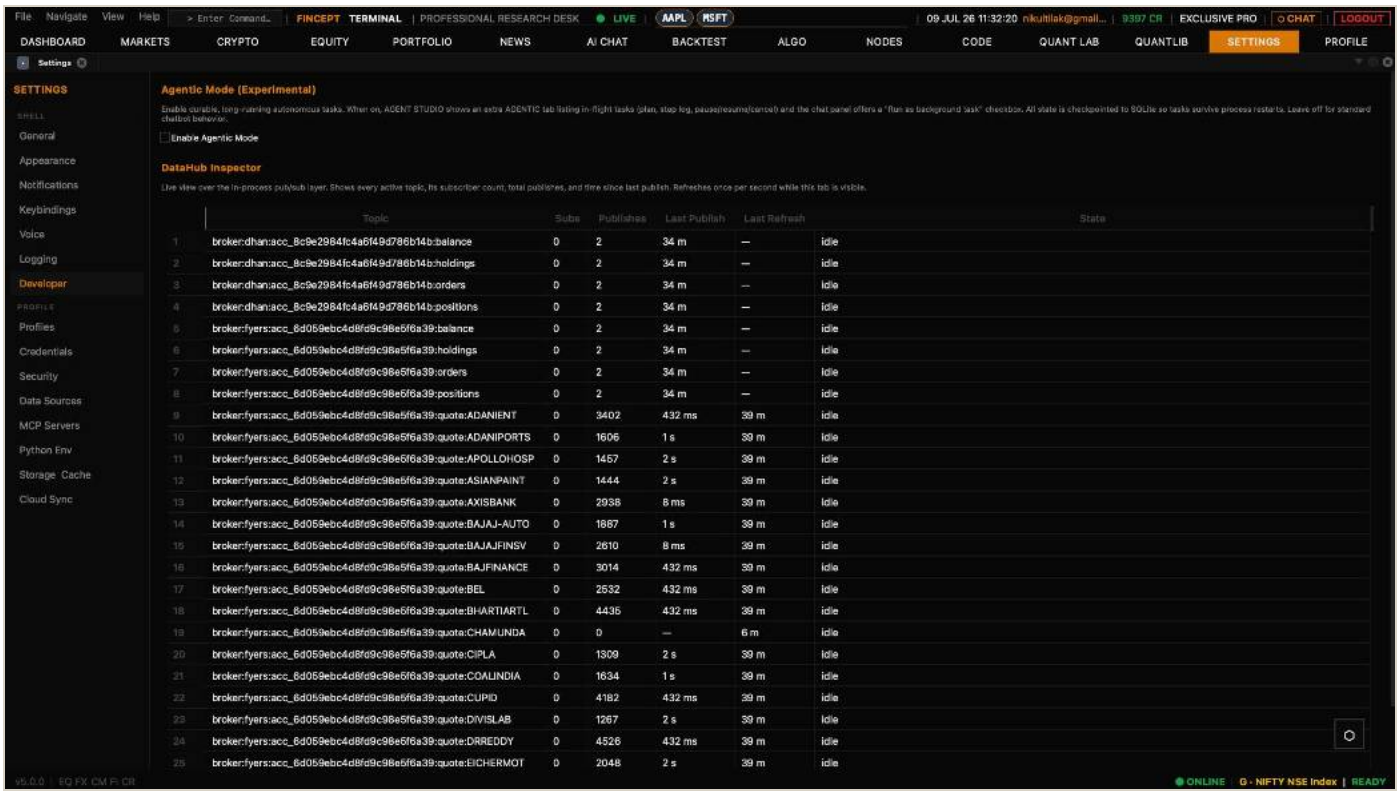


Fig 7 — Developer. Two entries: the Agentic Mode switch and the DataHub Inspector.

Control	What it does
`Agentic Mode (Experimental)`	**Enable Agentic Mode** turns on the experimental agentic path in AI Chat, where the assistant plans and executes a multi-step tool run rather than answering a single turn.
`DataHub Inspector`	Opens an inspector over the internal data hub — the fastest way to tell whether an empty panel is a data problem or a rendering one.

> **Experimental means experimental.** Agentic Mode is a developer switch, not a supported feature, and it changes how AI Chat executes. Leave it off unless you are deliberately testing it.

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.

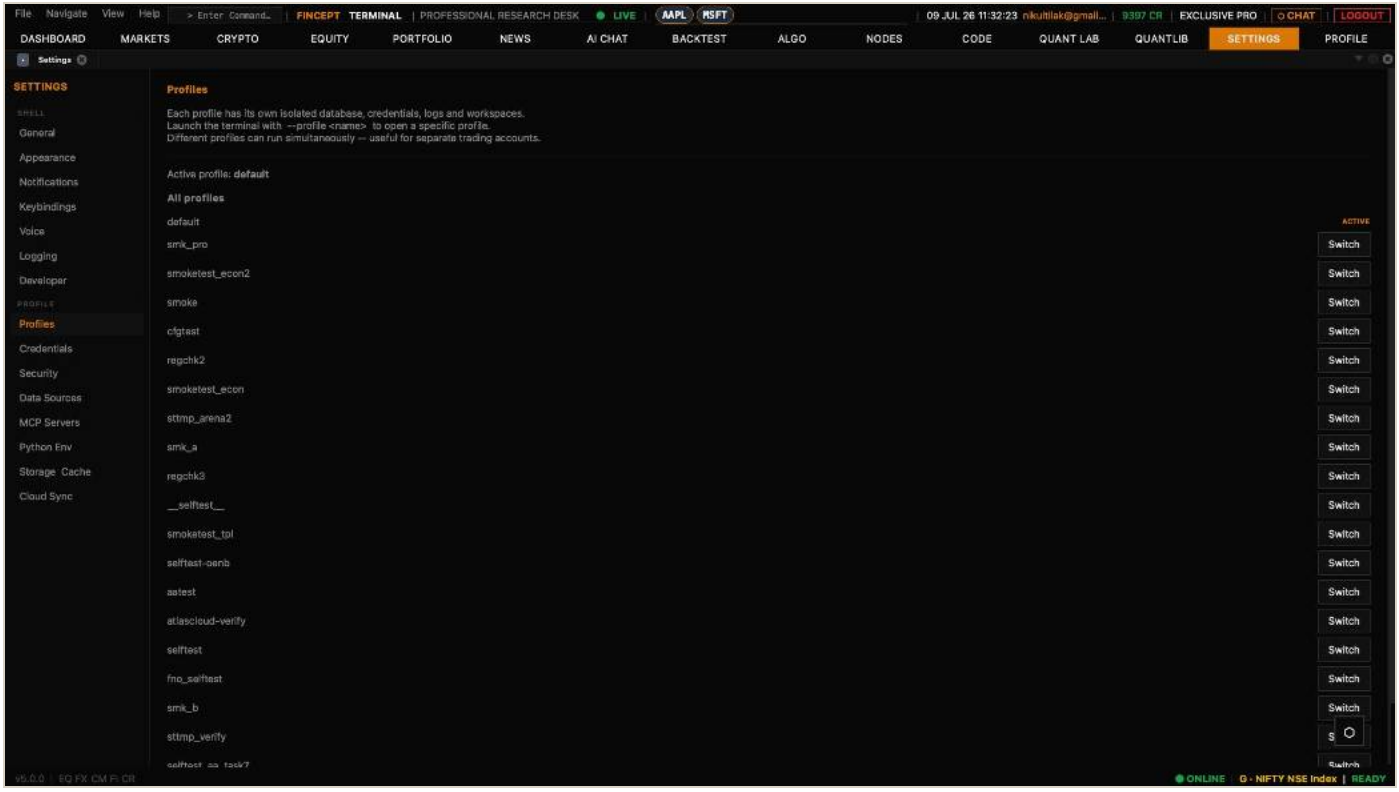


Fig 8 — Profiles. The active profile is badged ACTIVE; every other row carries a Switch button.

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-seven 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.

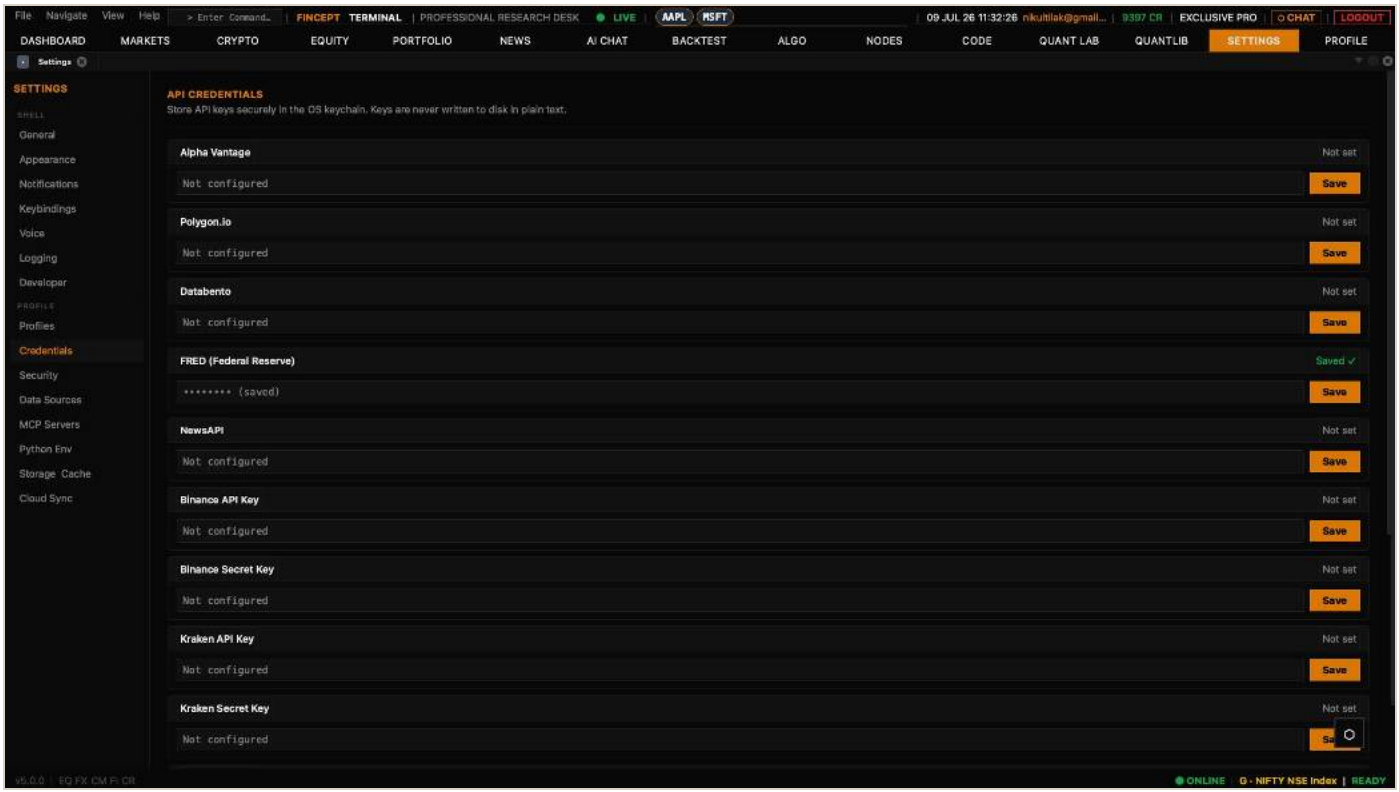


Fig 9 — Credentials. One row per provider, saved values masked.

Group	Keys
`Market data`	Alpha Vantage · Polygon.io · Databento · IEX Cloud · Finnhub · Tiingo · Quandl · Nasdaq Data Link
`Economics & government`	FRED · BEA · BLS · US Census · EIA · UN Comtrade · WTO · Trading Economics · Congress.gov
`News`	NewsAPI
`Crypto venues`	Binance (key + secret) · Kraken (key + secret)
`Prediction markets`	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.

15 Security

A six-digit PIN guards the terminal; an idle timer and a minimize hook can re-lock it.

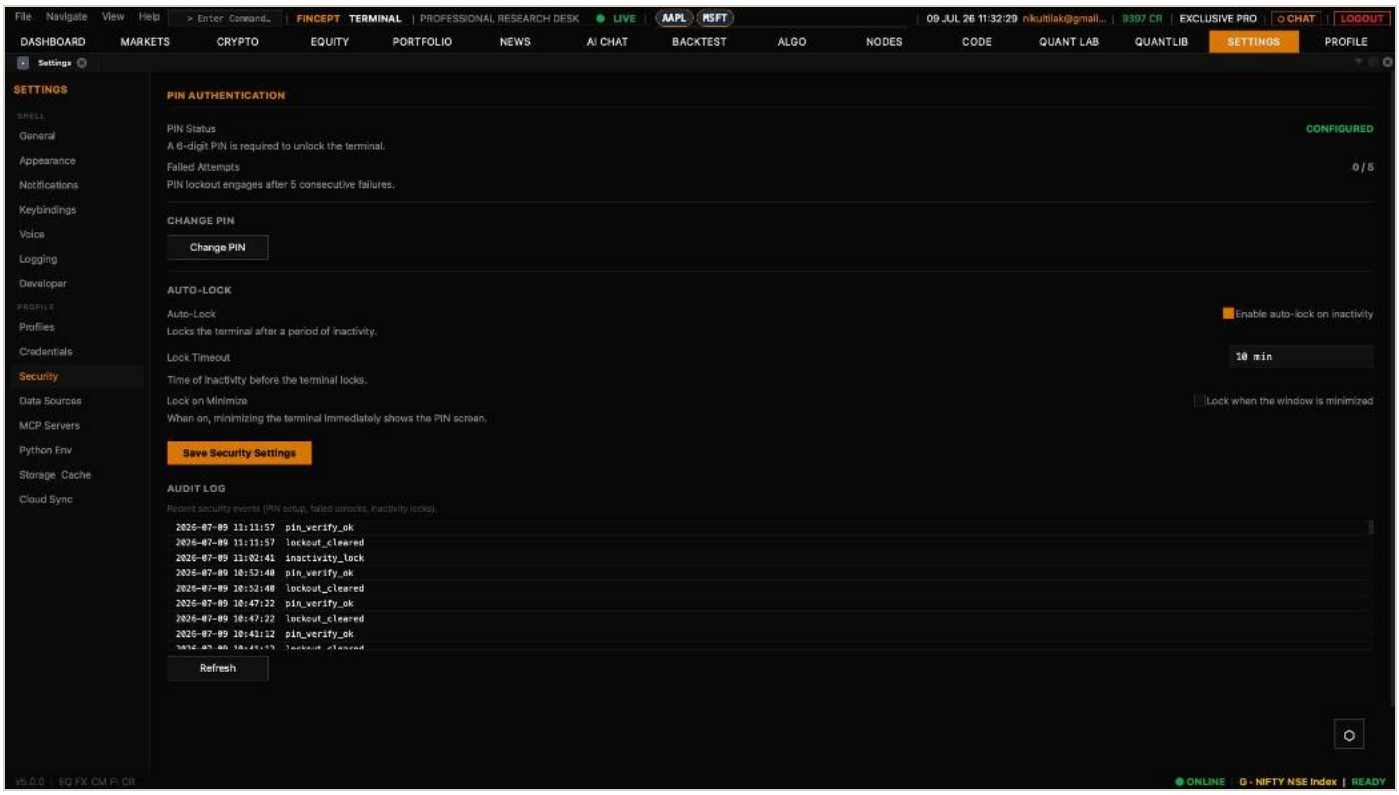


Fig 10 — Security. PIN status and failed attempts, the change form, then auto-lock.

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 minutes.
`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.

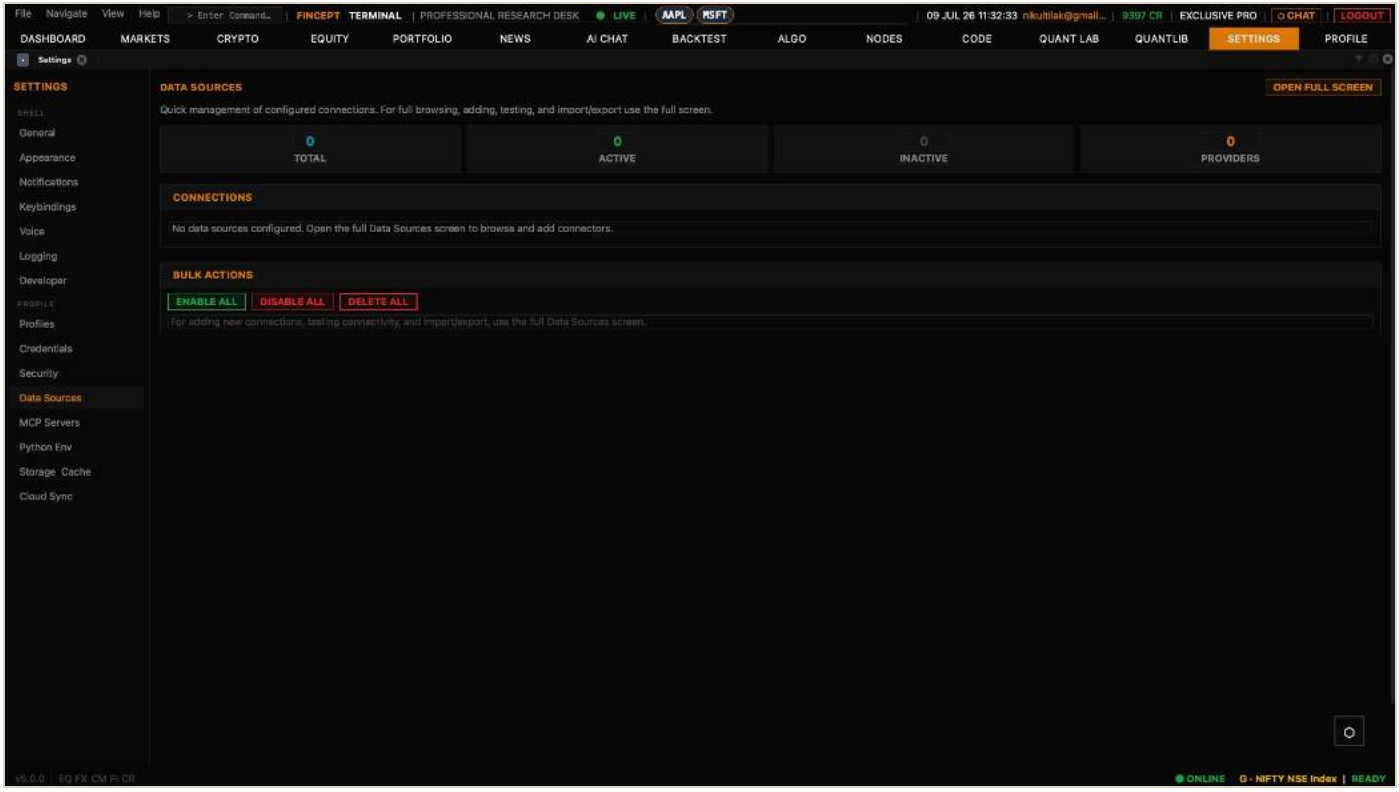


Fig 11 — Data Sources. Counters across the top, the connections table below, bulk actions at the foot.

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 MCP Servers

The terminal exposes its own capabilities to the AI agent as MCP tools, and can additionally run external MCP servers. Both are managed here.

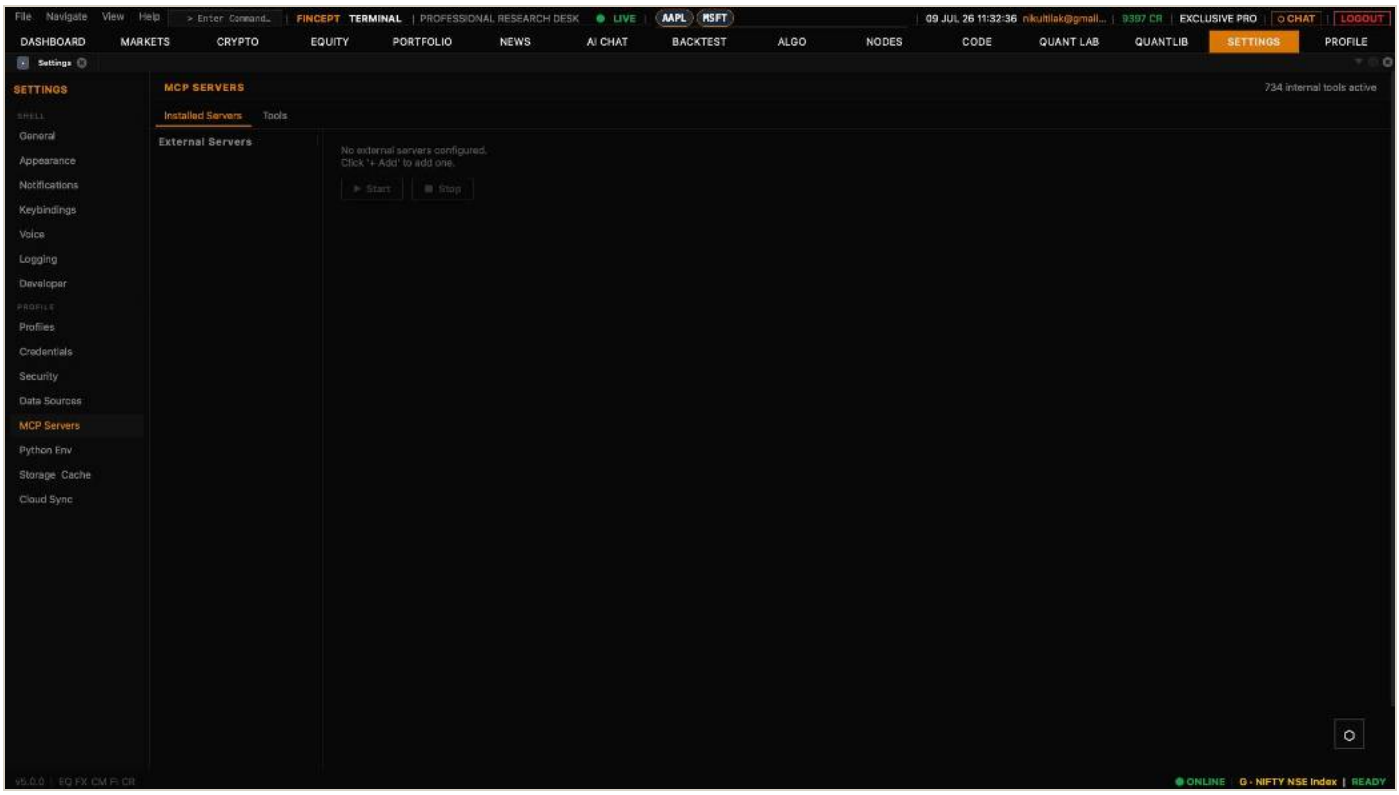


Fig 12 — MCP Servers. Installed servers on the left, the selected server's detail and its tool table on the right.

Region	What it does
`Installed Servers`	The internal server, reporting *N internal tools active*, plus any external ones. Each is badged *internal* or *external*.
`External Servers`	**+ Add** opens the dialog; **Remove** deletes the entry. Empty until you add one.
`▶ Start / ■ Stop`	Runs or halts the selected external server; the row reports *Running* or *Error*.
`Tools table`	Tool Name, Server, Category and Description for everything currently exposed to the agent.

Adding an external server

Name (e.g. *My Database Server*) and **Command** (`npx`, `uvx`, `python`) are required — the dialog refuses to save without both. **Arguments** (`-y @modelcontextprotocol/server-postgres`), **Environment** as space-separated `KEY=VALUE` pairs, a **Category** label and **Auto-start on launch** are optional.

> Changes are picked up without a restart — the screen listens for registry events and refreshes in place, so a newly added server's tools appear to the agent straight away.

18 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.

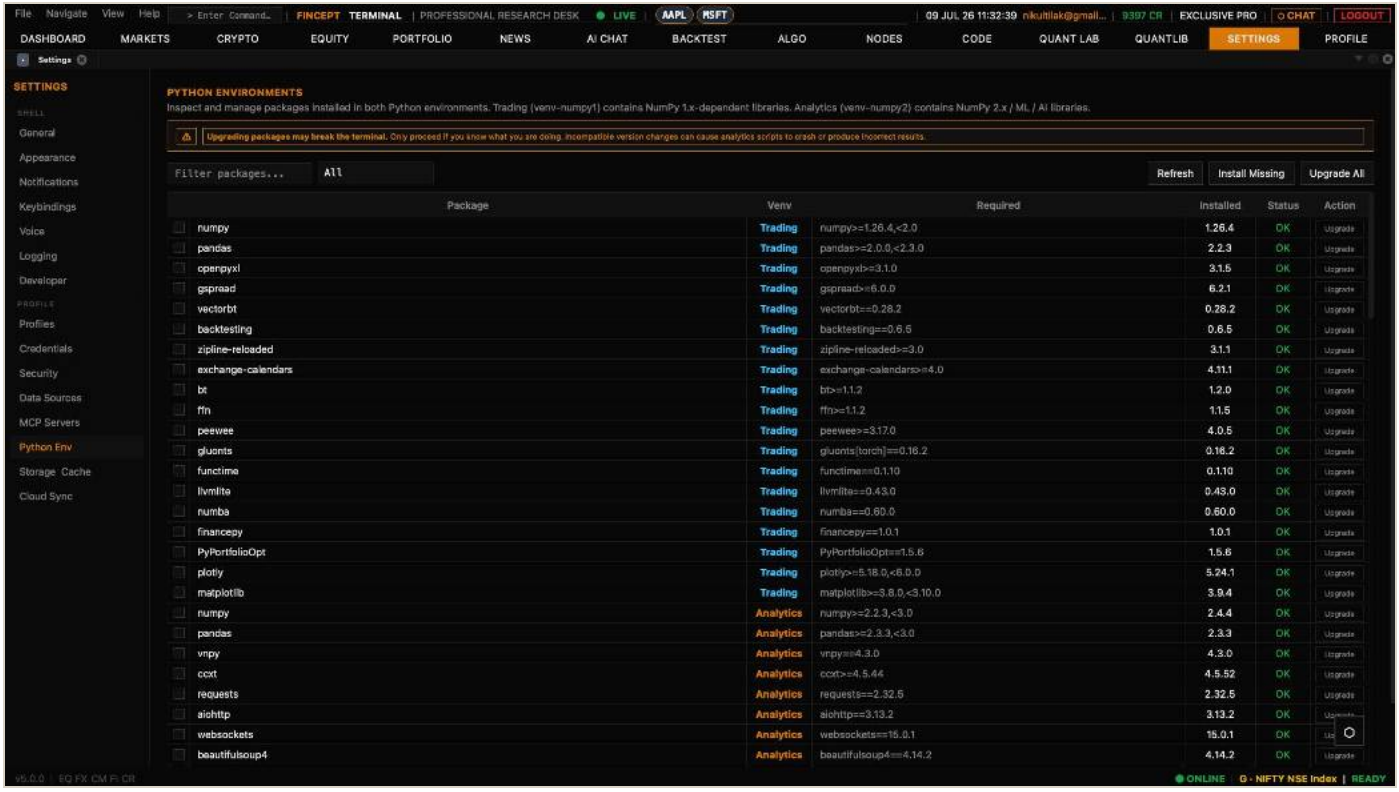


Fig 13 — Python Env. Required against installed versions, with a status and an action per row.

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.

19 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.

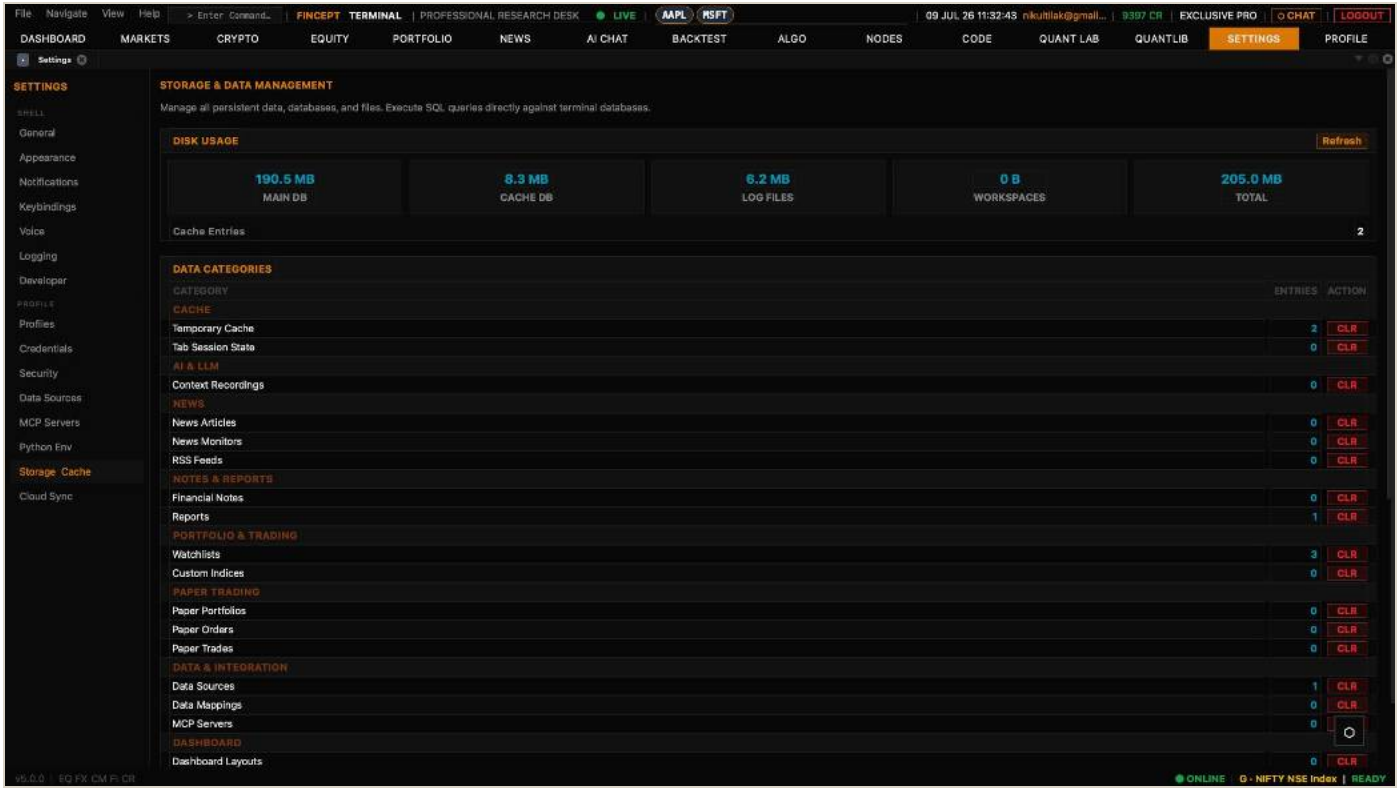


Fig 14 — Storage & Cache. DISK USAGE at the top, per-category counts with a CLR button on each row.

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.

20 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.

21 Cloud Sync

Mirrors chosen domains to your Fincept account so another machine picks them up. Your local copy stays the working copy; the cloud is the mirror.

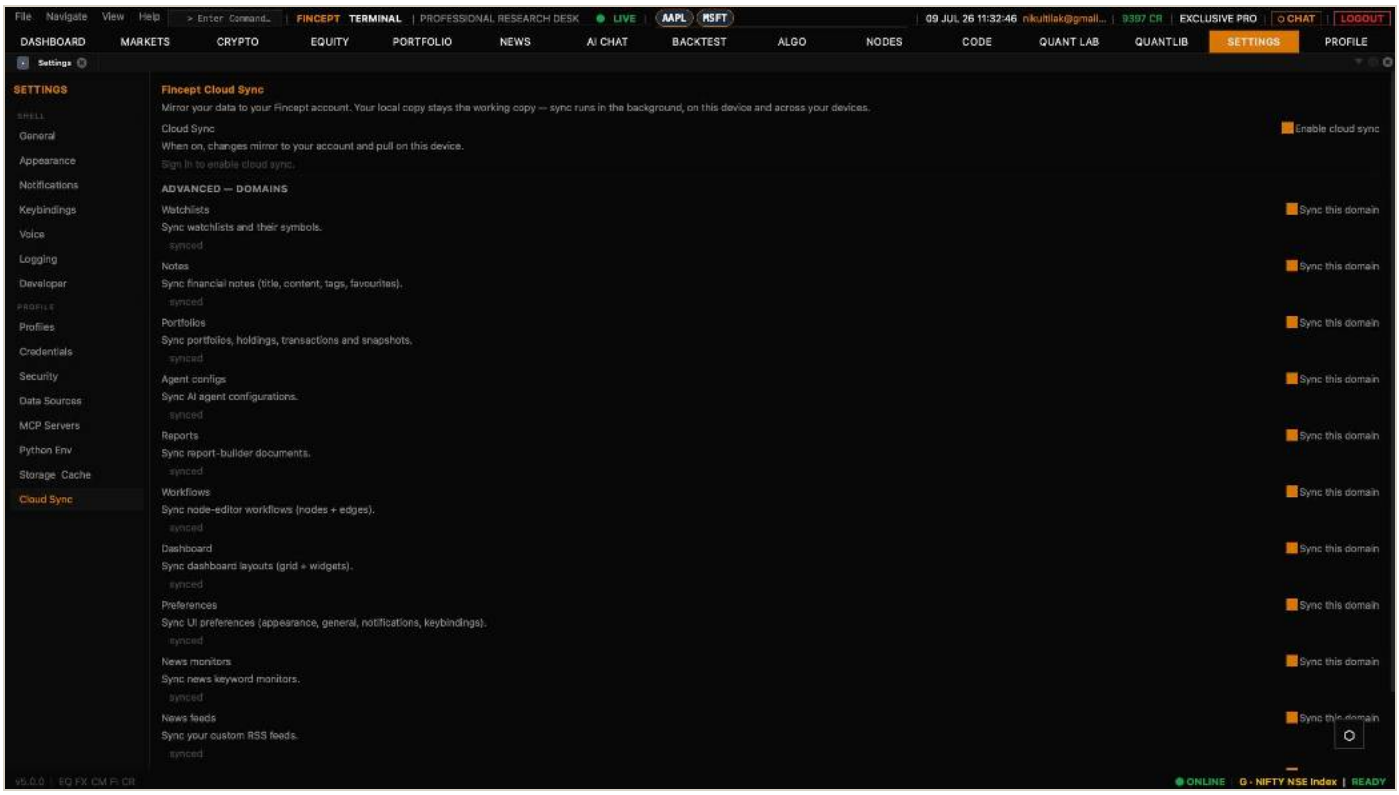


Fig 15 — Cloud Sync. The master switch above, per-domain rows beneath.

The ten synced domains

- **Watchlists** — lists and their symbols
- **Notes** — title, content, tags, favourites
- **Portfolios** — holdings, transactions, snapshots
- **Agent configs** — AI agent configurations
- **Reports** — report-builder documents
- **Workflows** — node-editor nodes and edges
- **Dashboard** — layouts, grid and widgets
- **Preferences** — appearance, general, alerts, keys
- **News monitors** — keyword monitors
- **News feeds** — your custom RSS feeds

Enable cloud sync is the master switch and needs you signed in. Each domain then has its own **Sync this domain** tick and a **Refresh now** action.

> **First sync asks who wins.** With data on both sides you choose **Upload & merge** (keeps your local items, recommended) or **Use cloud**. Sync spends credits and pauses when the balance empties.

22 Where each setting is stored

Three storage tiers sit behind this screen. 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 instrument cache.	The installation.
`OS keychain`	Broker API credentials, by design.	The machine's keychain.
`Account (server)`	Preferences, agent configs, portfolios, notes, watchlists and the other synced domains.	Your login — if the domain is ticked in Cloud Sync.

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.
- **Broker credentials never travel.** They are re-entered per machine on purpose; only account *metadata* leaves the device.
- **An unticked domain stays local** even though it is an account-tier domain. The tier decides where it *can* live; Cloud Sync decides whether it does.
- **Preferences are themselves a synced domain.** Tick *Preferences* and appearance, general, notification and keybinding settings follow the account too.

> 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 once cloud sync is on.

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
Talk to the terminal hands-free	Voice › Clap to open mic › Double clap
Debug one noisy subsystem	Logging › Per-Tag Overrides › + Add Tag Override
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 › Enable › tick the domains

Honest notes

- **Display currency does not convert.** It changes the symbol on calculators and analytics; live market data keeps its own currency.
- **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.
- **Cloud sync spends credits** and pauses 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 fifteen 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.

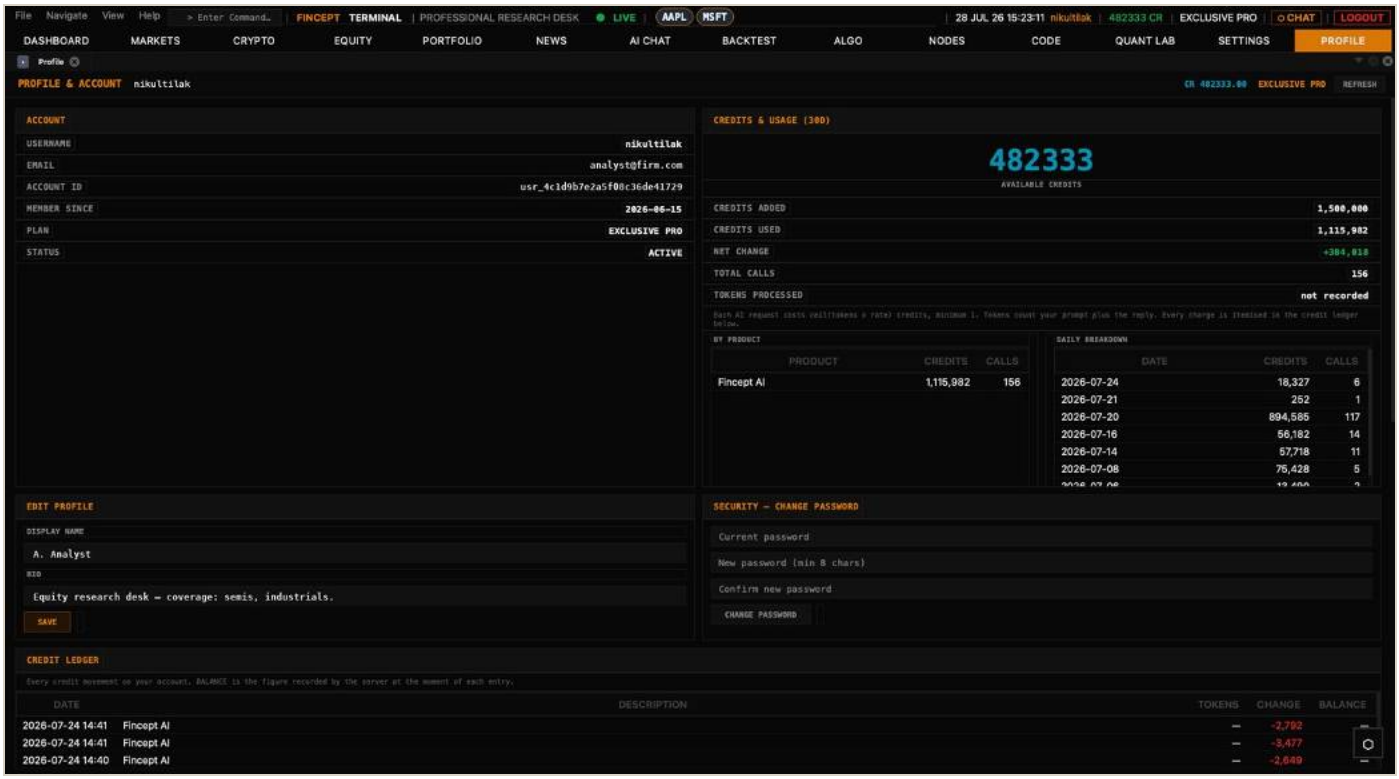


Fig 2 — The top of the page: ACCOUNT and CREDITS & USAGE side by side, then EDIT PROFILE and SECURITY, with the CREDIT LEDGER beginning underneath.

#	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

HEADER · 03

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.

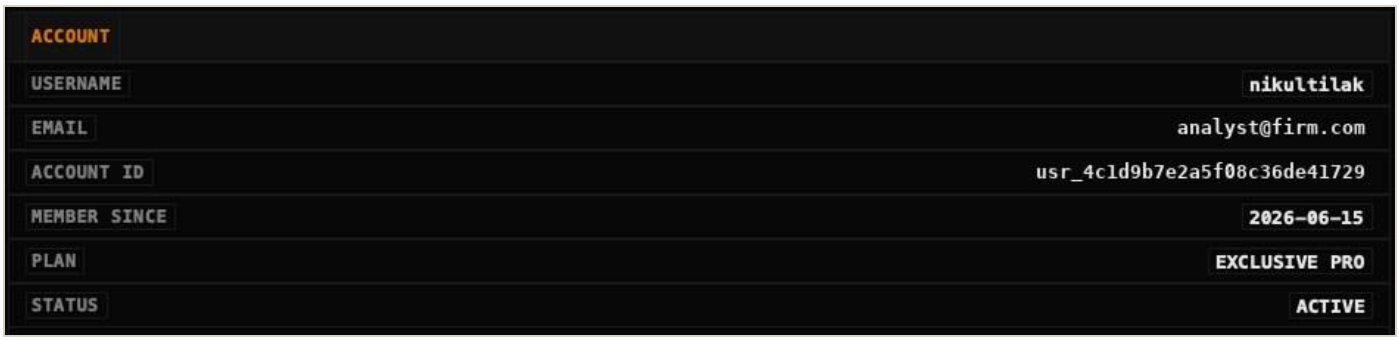


Fig 4 — The ACCOUNT panel. (Email and account id are replaced with placeholders in this guide.)

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.



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

LEDGER · 07

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.

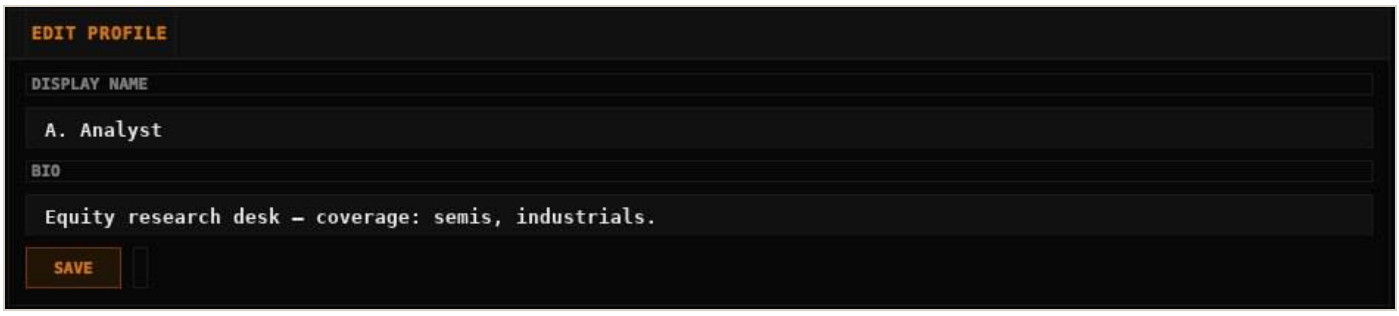


Fig 8 — The EDIT PROFILE panel. (The values shown are placeholders.)

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.

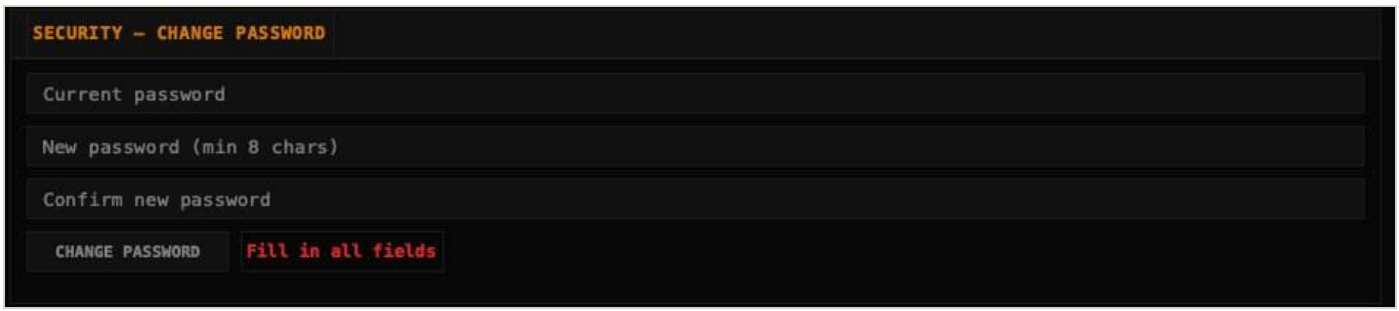


Fig 9 — The SECURITY panel, showing the validation message that appears when the fields are empty.

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.

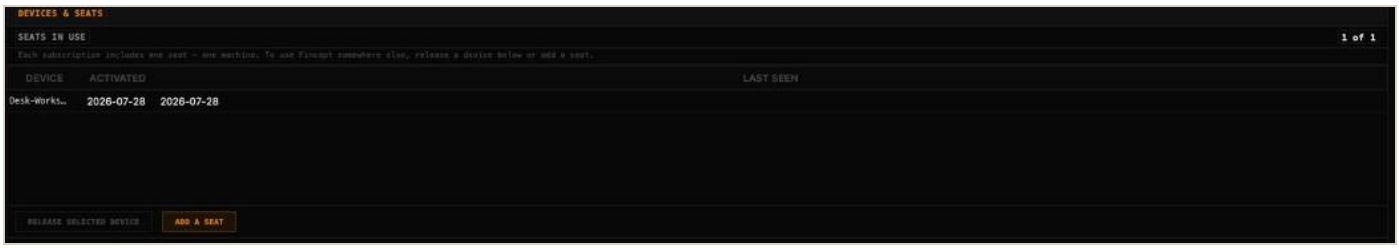


Fig 11 — DEVICES & SEATS: the seat count, the roster, and the two actions. (The machine name is a placeholder in this guide.)

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:

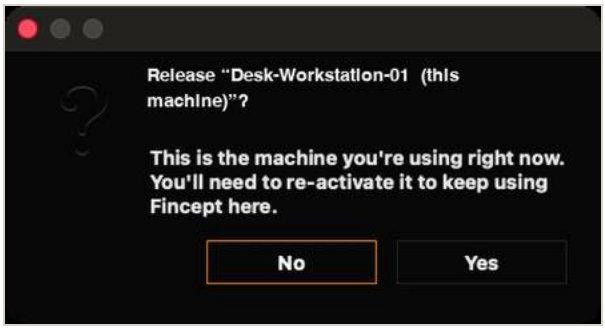


Fig 12 — Releasing the machine you are using right now says so plainly. Releasing any other machine warns that it will need to re-activate.

- 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

SESSION · 12

Logging out

A single red **LOGOUT** button sits at the foot of the page, well away from everything else. It always confirms.

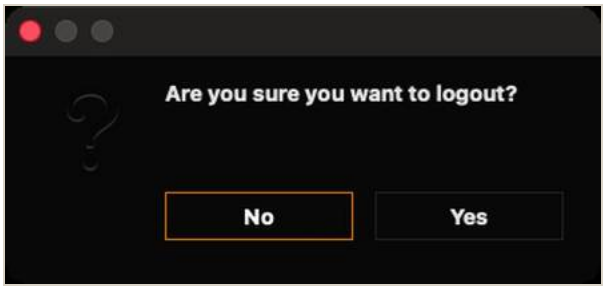


Fig 13 — The logout confirmation, defaulting to **No**.

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.

41	472	700	6
GUIDES	SECTIONS	PAGES	PARTS

Part I Getting Started — 4 guides
The terminal, end to end — what it is and how it is laid out.

Part II Markets & Trading — 6 guides
Live prices, order entry and automated execution.

Part III Research & Analytics — 9 guides
The analyst desks — valuation, risk, portfolios and strategy.

Part IV Data & Intelligence — 11 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 July 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.