BROKER CONSOLE · MODULE REFERENCE

Reporting and Business Intelligence

What the business did, in the shape the reader needs it: a board pack a director opens once a week, the worked tables an analyst exports, and the files a trade repository will actually accept.

Self-describing metric catalogue · Period comparison · Power BI feed · CSV and XML export · EMIR trade and position reports · Honest gaps, never zeros

PRODUCT	MODULE	CONTENTS	VERSION
HonorPro Hub	Reporting	3 menus · 8 tabs	1.0 · 24 Sep 2026

Business Intelligence	Dashboard · Finance BI · Trading BI · Clients BI · Partners BI · Logs
Reports & Analytics	Reports — overview, volume, flows and clients, with CSV export
Regulator Reporting	EMIR Reporting — report, submissions and settings

01 THE MODULE

Three readers, three shapes of the same period

Reporting answers one window of trading for three different people. A director wants the whole business on one screen. An analyst wants a table they can sort, page and export. A regulator wants a file in their own schema, validated before it is sent. The module keeps those three apart on purpose: a board tile that can be exported as a filing is a board tile nobody trusts.

3
MENUS
Business Intelligence, Reports & Analytics, Regulator Reporting

8
TABS
Each one an address a person can paste to a colleague

7
CATALOGUE SECTIONS
Trading, clients, money, risk, products, operations, partners

Every figure names its source

The gateway does not publish numbers, it publishes a **catalogue**: each metric declares its kind, which direction is better, how it rolls up across installations, and one sentence naming the read it came from. The console renders whatever the catalogue describes, so a gateway that gains a metric shows it without a console release.

- **kind** — money, percent, count, ratio, duration
- **better** — whether a rise is good news, from the broker's side of the book
- **rollup** — sum, average, max, min, weighted, or none
- **source** — the sentence shown under the figure

A figure this installation cannot compute comes back **not known** with the reason, and is drawn as a dash. It is never drawn as a zero — a zero is an answer, and the wrong one.

One window, compared

Every screen in the module reads the same period control: seven days, thirty, ninety, month to date, quarter to date, year to date, or a custom range. Comparison against the preceding window of equal length is a toggle, not a second screen, and the delta carries the metric's own polarity rather than the sign of the number.

Scope

The Hub reads one installation — this brokerage's own book. The same catalogue is consolidated across every licensed installation in HonorPro 360, where a metric that cannot be added across brokers (a rate, a percentile) is shown per broker and never averaged.

What it never does

Nothing in this module writes to the trading server. Every screen is a read, every export is a file, and the only state it keeps is the broker's own reporting settings and the record of what was filed.

02 MENU · BUSINESS INTELLIGENCE

The board's own view of the period

One screen carrying the whole business, and five cuts of it for the people who only ever look at one part. All six read the same catalogue and the same window, so two tabs can never disagree about the same number.

Dashboard `/bi`

The whole catalogue on one board: every section the gateway declares, drawn as tiles with their deltas, then the charts and breakdowns that section publishes. The header states when it was read, how many figures came back unknown, and refreshes on its own minute.

All sections · Period + compare · Refresh 60s · Export

Finance BI `/bi?section=money`

Deposits, withdrawals and every other balance operation over the window: what came in, what went out, the net, the operations behind each and the currencies they were booked in. The cut a finance lead opens on a Monday.

Money section · Balance operations

Trading BI `/bi?section=trading`

What the closed trades did: volume and notional, long against short, winning and losing trades, holding times, the client's result and the book's, commission and swap. The cut a dealing head reads.

Trading section · Broker-side polarity

Clients BI `/bi?section=clients`

Who is on the book and what happened to them: arrived, funded, traded, stayed, went quiet. Counts, not names — the CRM owns the people, this owns the shape of the population.

Clients section

Partners BI `/bi-partners`

The introducing-broker ledger as a board: partners on the book, who they brought, what those clients traded, and the concentration of the business in the largest few. The join is made on the gateway from the referral code already written onto a client's account at sign-up.

Partners section · Top-1 and top-5 concentration

Logs `/bi?section=operations`

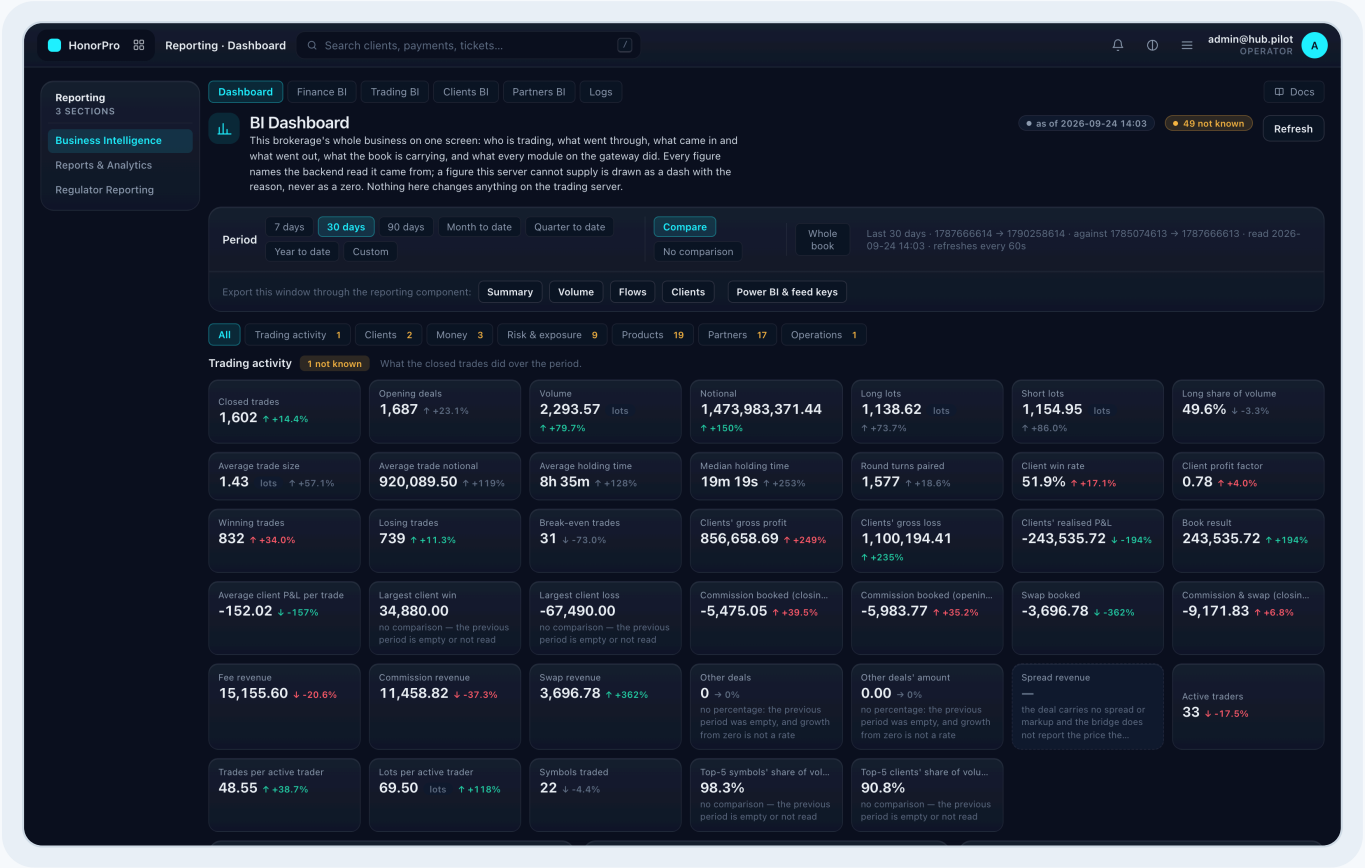
The console's own health rather than the business: the freshness of every source the board is built from, what the feed did, and which reads failed. The tab to open when a number looks wrong before concluding that it is.

Operations section · Source freshness

03 THE BOARD

Forty figures, each naming where it came from

The Dashboard drawn from a live installation. Sections along the top with their own counts, tiles carrying the figure and its move against the previous window, then the charts and breakdowns each section publishes.



Business Intelligence · Dashboard. The header states the read time and how many figures came back unknown; those are drawn as dashes with their reason.

What a tile carries

The figure in the metric's own kind — money in the book's currency, a percentage, a count, a duration — the move against the preceding window of equal length, and the sentence naming the read behind it. The colour of the move follows the metric's own polarity from the broker's side of the book, not the sign of the number.

Export and feed keys

The window can be handed to the reporting component as a summary, volume, flows or clients export, or published as a dataset a BI tool subscribes to. A feed key is scoped to the datasets the broker chooses, shown once on creation and stored only as a hash.

04 MENUS · REPORTS & ANALYTICS · REGULATOR REPORTING

The worked table, and the file that must be accepted

Where Business Intelligence answers "how are we doing", these two answer "show me the rows" and "send this to the repository". Both read the same window as the board, so an analyst and a director are never arguing about different periods.

Commission accrual, payouts and partner tiers live in the CRM rather than on the trading server, so the board returns them **not known** with a sentence saying where they live. They fill in by themselves the day a CRM ledger is registered against the catalogue — no console release, no second screen.

Reports `/reporting`

Four reports over the chosen window — an overview, volume, flows and clients — each a real table with its own grouping, sort, paging and a dimension to break it down by. Every one exports to CSV with a header block naming the window, the filters and the moment it was read, so a file that ends up in a mailbox still says what it is.

Overview · Volume · Flows · Clients · CSV export

EMIR Reporting `/emir`

Trade and position reports built from the broker's own book in the schema a trade repository accepts — **52 fields** on a trade and **26** on a position, each one declaring where its value comes from: the deal, the symbol specification, a derivation, or the broker's own configuration.

Fifteen validation rules run over the whole file before anything can leave. A file that would be rejected is refused here first, naming the rule and the offending rows, rather than being handed over and bounced. Export is BOM CSV in EMIR field order, or ISO 20022 XML where an absent field is an absent element and never an empty one.

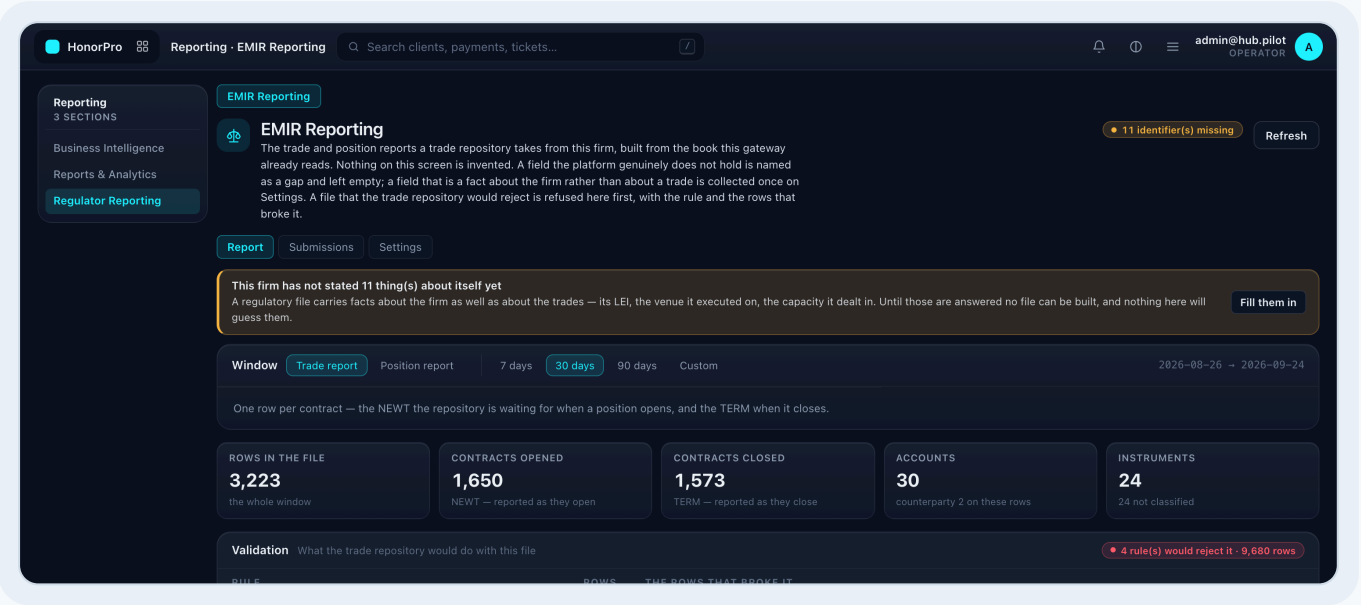
Report · Submissions · Settings · 52 + 26 fields · 15 rules

Every build is recorded: the window, the counts, the identifiers used, a hash of the content, who built it, and every state it passed through — built, submitted, accepted, rejected. **"What did we send on the 14th"** is a question this module can answer.

05 BUILDING A FILING

Refused here, rather than bounced by the repository

The report screen builds the file for a window, counts what went into it, and runs the validation the trade repository would run — before anything can be exported or recorded as filed.



Regulator Reporting · EMIR. This installation's window holds no contracts, and the screen says so — an empty file is a legitimate answer and not the same thing as a failed read.

The broker's own identifiers

LEI, reporting counterparty, clearing threshold, trading capacity, venue, master agreement and the instrument map are collected once on the Settings tab. The LEI is checked against its own control digits before it can be saved. The repository password is never stored — the document holds the name of an environment variable and the screen says whether it resolves.

What is not in the file

Ten fields the regulation asks for are not held anywhere on this platform — the counterparty's own country and sector, a product identifier, a prior UTI, settlement and confirmation timestamps, margin. Each is named on the report screen under its own heading. None is ever defaulted: an invented value in a regulatory file is worse than a gap that is declared.

06 HOW IT CONNECTS

Where the numbers come from

Nothing in this module holds its own copy of the book. Each screen reads the gateway, the gateway reads the trading server, and the one thing the Hub stores is the broker's reporting configuration and the record of what was filed.

TAB	READS	WRITES
Dashboard, and the five cuts	The gateway's BI catalogue and board over the chosen window	Nothing
Partners BI	The same catalogue, partners section — referral codes joined to accounts and their deals	Nothing
Reports	The gateway's reporting component: deals, the open book, the client register	Nothing; CSV is generated per request
EMIR Reporting	The same deal and position reads, plus the symbol specification for notional	The broker's EMIR settings and the submission record

Power BI and feed keys

The board's underlying datasets can be published to a feed a BI tool subscribes to. Keys are minted on the Dashboard, are scoped to the datasets the broker chooses, and can be revoked without touching anything else. A key is shown once, on creation, and stored only as a hash.

Licensing

Business Intelligence and Reports & Analytics are licensed components: a brokerage that has not bought one does not see its menu at all — not greyed out, not drawn. Regulator Reporting rides on the reporting component and needs the broker's own repository credentials before a file can leave the building.

Across an estate, HonorPro 360 draws the union of what every installation's catalogue declares and names the brokers whose gateway does not carry a section — so a consolidated board is never quietly missing a business.