

Trading infrastructure, available as source / IP.

Python · MetaTrader / MetaApi · FX/CFD · Pre-revenue software asset

Final engineering acceptance and broker DEMO qualification are pending. The accepted release and demonstration evidence will be added before transfer.

WHAT FALBI DOES

FALBI brings historical replay, live market-data handling, strategy decisions and account-specific execution into a modular Python system. It includes the TA Trader strategy layer and the surrounding routing, sizing, risk, persistence and broker-state lifecycle components.

WHY A TEAM WOULD EVALUATE IT

A buyer already working with Python and MetaTrader can evaluate an existing foundation instead of starting each component from a blank repository. The relevant question is how the supported modules fit the buyer's intended workflow and what integration work remains.

THE PRIMARY TRANSACTION

Acquisition of the agreed proprietary production source and the owner's transferable IP rights. Documentation, the defined delivery manifest and bounded handover are included in the proposed transaction. Third-party components retain their own terms.

CURRENT COMMERCIAL STAGE

Privately owned and pre-revenue. This is an engineering asset offered for review and transfer. Revenue, customers and a verified trading-profit track record are outside the acquisition proposition.

[Product page and one-page overview ↗](#)

Vladyslav · vvlad0066@gmail.com

From market data to account execution.

The historical and live paths use distinct broker implementations around the documented strategy and execution modules.

Data and strategy

Historical / live market-data handling; warmup and synchronization for the live path. The included TA Trader layer produces trade decisions. Replacing it requires mapping a new strategy into the documented execution contract.

Routing, sizing and risk

Account-specific routing and sizing; configured risk, margin, spread and policy constraints. STANDARD / PROP are account configurations. A PROP configuration does not imply approval by a particular prop firm or a challenge-pass guarantee.

Execution core

ExecutionCore and LiveLogicalBroker manage the live order/position lifecycle through the MetaApi adapter. Supported account models include hedging paths with staged take profits and netting paths with partial-close handling.

Replay path

ReplayEngine and ReplayBroker provide historical evaluation and inspection. A successful replay concerns that dataset and environment; it does not establish live-broker acceptance or future investment performance.

State and recovery

SQLite persistence, journals, broker reconciliation and reconnect/recovery logic support state tracking. Technical review should inspect ownership of execution requests and the handling of broker observations across connection changes.

A specific integration scope. A practical fit decision.

Check these boundaries before discussing a transfer.

Included connectivity	MetaApi-integrated MetaTrader FX/CFD workflows. The buyer supplies compatible broker/account access and the required service account. Seller credentials and personal broker accounts are excluded.
------------------------------	---

Included materials	Agreed production source, integrated TA Trader strategy layer, installation/configuration/operation documentation and a defined release manifest. Internal seller records and private raw evidence are outside the public pack.
---------------------------	---

Buyer engineering	Validate symbol/account configuration, risk settings, service permissions and the deployment environment. Strategy substitutions and new broker adapters require separately scoped engineering work.
--------------------------	--

Outside scope	No documented native FIX, IBKR or exchange adapter; no verified futures/options workflow or rollover; no broker-server plugin, liquidity hub or hosted multi-tenant SaaS service.
----------------------	---

Commercial dependencies	Third-party software and MetaApi/broker services are governed by their own licences, fees and prerequisites. The asset purchase does not transfer a third party's ownership or promise free service access.
--------------------------------	---

Deployment responsibility	The buyer controls its own deployment, account permissions and operational decisions. The proposed handover provides bounded documentation support; ongoing operation and new feature development are separately scoped.
----------------------------------	--

Evidence tied to the code being transferred.

The final review record identifies the accepted release and the controlled demonstration.

Release identity	One accepted release hash, a run identifier and the delivery manifest. The buyer archive must correspond to the release reviewed and demonstrated.
-------------------------	--

Controlled DEMO	Completed warmup and three sequential controlled broker DEMO round trips. Review actual execution observations, request identities and journals; check for duplicate broker mutations and ambiguous terminal state.
------------------------	---

Final broker state	Authoritative final broker observations in the relevant connection generation. No run-owned orders or positions remain; unrelated/manual trades are preserved. Cleanup is confirmed with no unresolved items and a COMPLETED / exit 0 terminal result.
---------------------------	--

Review media	A short recording and screenshots of the actual accepted run, with credentials, account identifiers and private broker information removed. Their release/run identity must match the accompanying record.
---------------------	--

Source inspection	A staged technical review under agreed confidentiality. Review a source sample and relevant evidence before committing. Full delivery follows the written transaction and confirmed funding process.
--------------------------	--

Handover	Production archive and hash, documentation, rights/dependency schedule and the agreed acceptance record. A proposed 14-day, six-hour written handover cap is specified in the indicative terms; final support is fixed in the signed agreement.
-----------------	---