

Signa AI Signal Feed

Institutional Data-as-a-Service Product Brief

A Multi-Agent Quantitative Signal Pipeline for Hedge Funds, Quant Desks, and Systematic Asset Managers

Confidential — For Institutional Distribution Only

getsigna.ai / institutional@getsigna.ai / Effective Date: June 2026

Table of Contents

1. [Executive Summary](#)
 2. [Data Coverage and Ticker Universe](#)
 3. [Signal Taxonomy and Data Dictionary](#)
 4. [Delivery Methods and Infrastructure](#)
 5. [Institutional Use Cases](#)
 6. [Data Quality and Methodology](#)
 7. [Performance Metrics and Backtesting Availability](#)
 8. [Pricing Tiers](#)
 9. [Integration Examples](#)
 10. [Compliance and Data Governance](#)
-

1. Executive Summary

The alternative data market has reached a structural inflection point. With institutional adoption exceeding 90% and serious hedge fund data budgets scaling toward \$1 million per \$1 billion AUM in the first year of deployment, the question for data sourcing teams is no longer whether to consume alternative signals but which providers deliver genuinely differentiated, production-ready intelligence.

Signa (getsigna.ai) is a multi-agent AI trading signal platform that applies an ensemble of specialized technical analysis agents to generate consensus-scored, directional trade signals across equities, cryptocurrencies, and commodities. Where single-model providers produce brittle, overfit outputs, Signa's pipeline requires independent agreement across multiple agents before a signal is published — a design principle borrowed from the same

ensemble methodology that underpins the most robust quantitative strategies at institutional funds.

The Signa institutional Data-as-a-Service offering delivers this signal pipeline directly into client infrastructure. Every signal record includes a complete execution specification: direction, entry price, stop-loss, take-profit target, timeframe, a normalized confidence score, and a full agent-level breakdown. The feed covers 500+ tickers across three asset classes, updates in real time, and is accessible via REST API, event-driven webhooks, or bulk cloud delivery for historical research.

The core value proposition for institutional clients rests on three pillars:

Ensemble Consensus. Signa's pipeline runs a suite of named, independent agents — stage-scanner, trend-engine, bollinger-band, and others — simultaneously against each ticker. A signal achieves maximum confidence only when five or more agents reach directional agreement. This credibility-weighted architecture eliminates the inflated scores that plague single-agent systems and provides a transparent, auditable confidence framework that quant teams can incorporate directly into their factor models.

Execution-Ready Output. Signa signals are not directional opinions. Each record includes precise, dynamically calculated entry, stop, and target levels derived from current volatility and technical structure, enabling direct ingestion into execution management systems without additional pre-processing.

Institutional Delivery Infrastructure. The feed is designed for production quantitative environments. REST API endpoints support programmatic on-demand queries; webhooks provide sub-second signal delivery for automated strategies; and bulk S3 or Snowflake delivery supports historical backtesting and factor research workflows.

2. Data Coverage and Ticker Universe

Signa's signal feed provides cross-asset coverage across three primary asset classes, with the breadth and update cadence appropriate for both systematic intraday strategies and longer-horizon position-taking.

Asset Class Coverage

Asset Class	Ticker Universe	Coverage Detail	Update Frequency
US Equities	500+ tickers	Large-cap and mid-cap stocks across all major sectors; S&P 500 and Russell 1000	Real-time during market hours; End-of-Day batch available

		constituents prioritized	
Cryptocurrencies	Top 50 by market capitalization	BTC, ETH, SOL, and all major liquid tokens; 24/7 coverage including weekends	Real-time, continuous
Commodities	Major contracts	Gold, Silver, Crude Oil (WTI/Brent), Natural Gas, Copper	Real-time during exchange hours; End-of-Day batch available

The ticker universe is maintained dynamically. Tickers are added when they meet minimum liquidity thresholds and agent coverage criteria; tickers are retired when liquidity degrades below acceptable levels. Enterprise clients may request custom ticker universes, including small-cap equities, additional commodity contracts, or specific cryptocurrency pairs, subject to liquidity review.

Historical Data Depth

Point-in-time historical signal data is available from platform inception, enabling clients to construct multi-year backtests without look-ahead bias. All historical records preserve the exact signal state at the time of publication, including the agent consensus count and confidence score that were available to a hypothetical consumer at that moment.

3. Signal Taxonomy and Data Dictionary

Signa's data schema is organized as a three-layer hierarchy: a top-level aggregated recommendation, a regime-level engine direction, and a trade-level execution direction. This structure allows clients to consume the data at the level of granularity appropriate for their use case — from a simple directional overlay to a full agent-level decomposition.

3.1 Core Signal Hierarchy

The three primary directional fields form the backbone of every signal record.

Field	Description	Possible Values
<code>signa.action</code>	The final, aggregated trade recommendation produced by the consensus engine. This is the top-level signal for systematic strategies.	<code>BUY</code> , <code>SELL</code> , <code>HOLD</code>

engine.direction	The prevailing market regime as identified by the trend-engine agent. Represents the macro-directional context in which the signal was generated.	BULLISH , BEARISH , NEUTRAL
data.direction	The specific execution direction for the trade. Provides the most granular instruction for position management.	LONG , SHORT , WAIT

The relationship between these three fields is intentionally hierarchical. A `signal.action` of `BUY` will always be accompanied by an `engine.direction` of `BULLISH` and a `data.direction` of `LONG`. Divergences between layers (e.g., a `BULLISH` engine direction but a `HOLD` action) indicate that the consensus threshold was not met, providing a useful signal of regime uncertainty.

3.2 Execution Parameters

Every signal record includes a complete set of dynamically calculated execution parameters. These levels are derived from the current price structure, volatility regime, and technical support/resistance analysis performed by the agent pipeline.

Field	Type	Description
levels.entry	float	The recommended price at which to initiate the position. Calculated to represent an optimal risk/reward entry within the current technical structure.
levels.stop	float	The recommended stop-loss price. Positioned beyond the nearest structural support or resistance level to minimize noise-driven exits.
levels.target	float	The primary take-profit price target. Derived from the next significant resistance (for longs) or support (for shorts) level.

timeframe	string	The expected trade duration. Values include Intraday , Swing (2–10 days), and Position (weeks to months).
-----------	--------	---

The implied risk/reward ratio (distance from entry to target divided by distance from entry to stop) is available as a derived field in the API response and is consistently used as a secondary filter in Signa's signal publication logic. Signals with a risk/reward ratio below 1.5:1 are suppressed by default.

3.3 Confidence and Consensus Metrics

The confidence and consensus fields are the most differentiated aspect of Signa's data product relative to single-model alternatives. They provide a transparent, auditable measure of signal quality that clients can use to tier their own execution logic.

Field	Type	Description
confidence_score	integer (0–100)	A normalized composite score representing the aggregate conviction of the consensus engine. Scores above 85 indicate high-conviction, multi-agent agreement.
consensus_count	integer	The number of independent agents that contributed a directional output to this signal. Maximum value is the total number of active agents in the pipeline.
agent_breakdown	array	A structured array containing the individual output of each agent (name, direction, and sub-score), enabling clients to build custom weighting schemes.

<code>credibility_weight</code>	<code>float</code> (0.0–1.0)	The credibility multiplier applied to the raw consensus score, derived from the <code>consensus_count</code> . A single-agent signal receives a weight of approximately 0.60; five-agent consensus receives 1.0.
---------------------------------	------------------------------	--

3.4 Metadata Fields

Field	Type	Description
<code>signal_id</code>	<code>string</code>	A globally unique identifier for the signal record, suitable for idempotent processing in downstream systems.
<code>ticker</code>	<code>string</code>	The ticker symbol of the underlying instrument.
<code>asset_class</code>	<code>string</code>	One of <code>equity</code> , <code>crypto</code> , or <code>commodity</code> .
<code>generated_at</code>	<code>ISO 8601 timestamp</code>	The UTC timestamp at which the signal was published to the feed.
<code>is_latest</code>	<code>boolean</code>	Indicates whether this record is the most recent signal for the given ticker. Used to filter stale records in bulk queries.

4. Delivery Methods and Infrastructure

Signa's delivery infrastructure is designed to meet the latency, reliability, and integration requirements of institutional quantitative environments. Three primary delivery mechanisms are available, and Enterprise clients may combine them within a single subscription.

4.1 REST API (On-Demand Pull)

The Signa REST API provides authenticated, on-demand access to the full signal feed. It is the primary integration point for portfolio management systems, research platforms, and discretionary dashboards.

Base URL: `https://api.getsigna.ai/v1`

Authentication: Bearer token authentication using scoped API keys. Keys are issued per client, per environment (production/sandbox), and can be scoped to specific asset classes or endpoints.

Key Endpoints:

Endpoint	Method	Description
<code>/signals/latest</code>	GET	Returns the most recent signal for all tickers, with optional filters for asset class, minimum consensus count, and minimum confidence score.
<code>/signals/{ticker}</code>	GET	Returns the full signal history for a specific ticker, with pagination and date-range filtering.
<code>/signals/batch</code>	POST	Accepts a list of tickers and returns the latest signal for each in a single request. Optimized for portfolio-level queries.
<code>/portfolio/{type}</code>	GET	Returns the current composition of Signa's curated portfolios (<code>aggressive</code> , <code>conservative</code> , <code>thematic_etf</code>).
<code>/performance</code>	GET	Returns historical performance metrics for signals, filterable by agent, asset class, consensus threshold, and timeframe.

Rate Limits: Professional API tier: 500 requests per minute. Enterprise tier: custom limits negotiated per contract.

4.2 Real-Time Webhooks (Event-Driven Push)

For systematic strategies requiring immediate signal delivery, Signal provides an event-driven webhook infrastructure. When a new signal is published or an existing signal is updated, Signal performs an HTTP POST to the client's registered endpoint within milliseconds of publication.

Webhooks are appropriate for automated execution workflows where latency between signal generation and order submission is a material factor. Each webhook delivery includes:

- A unique `event_id` for idempotent processing and deduplication.
- An HMAC-SHA256 signature in the `X-Signal-Signature` header, enabling clients to verify the authenticity and integrity of every payload.
- Automatic retry logic with exponential backoff for failed deliveries (non-2xx responses), with a maximum of five retry attempts over a 30-minute window.
- A delivery receipt log accessible via the API for audit and reconciliation purposes.

4.3 Bulk Historical and Cloud Delivery

For quantitative research, factor construction, and backtesting workflows, Signal delivers point-in-time historical datasets via cloud-native channels. Available delivery targets include:

- **Amazon S3:** Scheduled delivery of daily or weekly snapshot files in Parquet or CSV format to a client-designated S3 bucket.
- **Snowflake Data Marketplace:** Direct data sharing to a client's Snowflake account, enabling zero-copy access without ETL overhead.
- **Secure FTP (SFTP):** Available for clients with existing SFTP-based data ingestion pipelines.

All historical datasets are structured as point-in-time records, meaning each row reflects the exact signal state at the moment of publication, with no retroactive revisions. This property is essential for preventing look-ahead bias in backtesting environments.

5. Institutional Use Cases

Signal's signal feed is designed to be consumed by sophisticated market participants across a range of quantitative and fundamental investment applications. The following use cases represent the most common deployment patterns observed among institutional clients.

5.1 Systematic Alpha Generation

Quantitative funds can integrate `signal.action` and `confidence_score` directly into their factor models or execution engines. The confidence score functions as a continuous signal strength variable, enabling clients to size positions proportionally to conviction rather than treating all signals as binary. Filtering on `consensus_count >= 5` provides a high-conviction subset suitable for strategies with strict drawdown constraints.

A typical implementation involves subscribing to the real-time webhook feed, filtering for signals above a minimum confidence threshold, and routing qualifying signals to an execution management system (EMS) for automated order generation. The `levels.entry`, `levels.stop`, and `levels.target` fields provide the parameters required to construct limit orders and bracket orders without additional pre-processing.

5.2 Risk Overlay and Regime Hedging

The `engine.direction` field provides a real-time regime classification that can be used independently of the trade-level `signal.action`. Portfolio managers can monitor the aggregate distribution of `engine.direction` values across the equity universe as a market breadth indicator, using a shift from predominantly `BULLISH` to predominantly `BEARISH` readings as a trigger to reduce gross exposure, increase cash allocation, or add systematic hedges.

This application is particularly valuable for multi-strategy funds that need a rapid, data-driven mechanism to adjust portfolio beta in response to deteriorating market conditions, without relying on lagging macroeconomic indicators.

5.3 Portfolio Construction Signals

Signal's curated portfolio endpoints (`/portfolio/aggressive`, `/portfolio/conservative`, `/portfolio/thematic_etf`) provide pre-constructed, signal-weighted portfolio recommendations that can be used as starting points for portfolio construction or as benchmarks against which to evaluate proprietary models. Each portfolio is rebalanced dynamically as the underlying signal landscape evolves, with full rebalancing history available via the API.

5.4 Discretionary Idea Generation and Thesis Validation

Fundamental portfolio managers can use the Signal feed as a quantitative screen to surface high-probability entry points for names already in their research pipeline, or to identify technically strong candidates that warrant fundamental investigation. The `agent_breakdown` field provides a transparent decomposition of the signal that allows a discretionary analyst to understand which technical factors are driving the recommendation, rather than treating the output as a black box.

5.5 Cross-Asset Macro Research

The combination of equity, cryptocurrency, and commodity signals within a single, normalized data schema enables cross-asset correlation analysis that would require multiple vendor relationships to replicate. Researchers can query the feed to identify periods of synchronized directional agreement across asset classes, which may serve as leading indicators of macro regime shifts.

6. Data Quality and Methodology

Signal's methodology is grounded in the principle that ensemble models produce more reliable, generalizable outputs than any single algorithm. This section describes the architecture of the signal pipeline and the specific quality controls applied at each stage.

6.1 The Multi-Agent Pipeline

The Signal pipeline operates a suite of named, independent technical analysis agents. Each agent is specialized for a distinct aspect of market behavior:

Agent	Specialization
stage-scanner	Identifies the current stage of the market cycle (accumulation, markup, distribution, markdown) using volume-price analysis.
trend-engine	Classifies the prevailing trend direction and strength using multi-timeframe moving average analysis.
bollinger-band	Detects volatility breakouts and mean-reversion setups using Bollinger Band width and position analysis.
Additional agents	Further specialized models covering momentum, relative strength, support/resistance, and volume profile analysis.

Each agent operates independently on the same underlying market data, producing its own directional output and sub-score. The outputs are then aggregated by the consensus engine, which applies a weighted voting mechanism to produce the final `signal.action` and `confidence_score`.

6.2 Credibility Discounting for Low-Agent-Count Signals

A critical design feature of Signa's scoring methodology is the credibility discount applied to signals generated by a small number of agents. Without this control, a single agent in strong agreement with itself would produce an artificially high consensus score, creating the false impression of multi-model conviction.

Signa applies the following credibility multiplier to the raw consensus score:

Plain Text

```
credibility_weight = min(1.0, 0.50 + (consensus_count × 0.10))
```

Under this formula, a signal generated by a single agent receives a credibility weight of approximately 0.60, capping its maximum confidence score at around 69/100. A signal with five or more agents in agreement achieves a credibility weight of 1.0 and is eligible for the full confidence score range. This design ensures that the `confidence_score` field is a meaningful, comparable measure of conviction across all signals in the feed, regardless of the number of agents that happened to cover a given ticker at a given moment.

6.3 The Verification Ladder

All signals pass through a four-stage verification process before publication:

Stage 1 — Raw Agent Output. Each agent in the pipeline independently processes the latest market data for the target ticker and produces a preliminary directional output and sub-score.

Stage 2 — Consensus Aggregation. The consensus engine collects all agent outputs for the ticker. Conflicting directional outputs (e.g., one agent bullish, another bearish) are flagged. If the net directional agreement falls below the minimum publication threshold, the signal is suppressed and the ticker is assigned a `HOLD / WAIT` state.

Stage 3 — Risk Parameter Validation. For signals that pass the consensus threshold, the execution parameters (`levels.entry` , `levels.stop` , `levels.target`) are dynamically calculated based on the current volatility regime and technical structure. Signals where the implied risk/reward ratio falls below 1.5:1 are suppressed.

Stage 4 — Publication and Delivery. The validated signal is assigned a unique `signal_id` , timestamped, scored with the credibility-weighted `confidence_score` , and published to the API and webhook infrastructure simultaneously.

6.4 Data Integrity and Revision Policy

Signa operates a strict no-retroactive-revision policy for published signals. Once a signal record is published to the feed, its content is immutable. If market conditions change and a

new signal is generated for the same ticker, a new record is published with a new `signal_id` and timestamp, and the previous record's `is_latest` flag is set to `false`. This policy ensures that historical datasets are suitable for point-in-time backtesting without survivorship or revision bias.

7. Performance Metrics and Backtesting Availability

Institutional due diligence requires rigorous, independently verifiable evidence of signal efficacy. Sigma provides a comprehensive set of tools and data products to support this evaluation.

7.1 Historical Signal Performance

The `/performance` API endpoint provides aggregated performance statistics for the signal feed, calculated on a rolling and cumulative basis. Available metrics include:

Metric	Description
Win Rate	Percentage of signals where the price reached the <code>levels.target</code> before the <code>levels.stop</code> . Reported by asset class, timeframe, and consensus threshold.
Average Risk/Reward Realized	The average ratio of realized gain to realized loss across all closed signals.
Signal Frequency	The number of signals published per day/week, segmented by asset class and direction.
Confidence Score Calibration	A calibration table showing the empirical win rate for each decile of <code>confidence_score</code> , enabling clients to validate the predictive value of the score.
Agent-Level Attribution	Performance attribution broken down by individual agent, allowing clients to identify which agents contribute the most predictive value for their target asset class.

7.2 Backtesting Data Availability

Enterprise clients receive access to the full historical signal archive in bulk format, structured for direct ingestion into backtesting frameworks such as Backtrader, Zipline, or

proprietary systems. The dataset includes:

- All published signals from platform inception, with complete field values as they existed at the time of publication.
- A separate `market_data_snapshot` table containing the OHLCV data that was available to the agents at the time of signal generation, enabling clients to reconstruct the exact information set used by the pipeline.
- A `signal_outcome` table tracking whether each signal's target or stop was subsequently hit, with the timestamp of the outcome event.

7.3 Trial Data for Due Diligence

Prospective Enterprise clients are offered a complimentary 90-day historical data sample covering a representative cross-section of the ticker universe. This sample is provided under a standard evaluation agreement and is intended to support internal backtesting and statistical validation prior to contract execution. The trial dataset carries the same data quality guarantees and point-in-time integrity as the production feed.

8. Pricing Tiers

Signa's pricing is structured to serve the full spectrum of institutional market participants, from individual quantitative researchers to multi-strategy hedge funds requiring bulk data delivery and dedicated support.

Professional API

Designed for individual quantitative researchers, family offices, and boutique proprietary trading firms.

Feature	Detail
Monthly Price	\$199 / month
API Access	Full REST API access to all signal endpoints
Webhooks	Real-time webhook delivery, up to 3 registered endpoints
Asset Classes	All (equities, crypto, commodities)
Historical Data	12-month rolling history via API
Rate Limits	500 requests per minute
Support	Email support, standard SLA

Enterprise / Institutional

Designed for hedge funds, large asset managers, multi-strategy platforms, and data aggregators requiring bulk data, cloud delivery, and dedicated technical support.

Feature	Detail
Pricing	Custom, based on AUM tier, delivery method, and usage volume
API Access	Uncapped API access with dedicated infrastructure
Webhooks	Unlimited registered endpoints; dedicated webhook delivery cluster
Bulk Delivery	Amazon S3, Snowflake Data Marketplace, SFTP
Historical Data	Full archive from platform inception
Custom Universe	Custom ticker universe available upon request
SLA	99.9% API uptime guarantee; defined latency SLAs for webhook delivery
Support	Dedicated Technical Account Manager; priority incident response
Compliance	Full DDQ package, data provenance documentation, and contractual representations and warranties
Trial	90-day historical data sample for pre-contract evaluation

Pricing inquiries for the Enterprise tier should be directed to institutional@getsigna.ai. Signa is prepared to engage with data sourcing teams under NDA prior to sharing detailed pricing and methodology documentation.

9. Integration Examples

The following examples illustrate common integration patterns for institutional clients. Production implementations should incorporate appropriate error handling, retry logic, and logging.

9.1 Python REST API Client

The following snippet demonstrates authenticated access to the Signa REST API, fetching the latest high-conviction signals filtered to a minimum consensus count of five agents.

Python

```
import os
import requests
from typing import List, Dict, Any

API_KEY: str = os.environ["SIGNA_API_KEY"] # Never hardcode credentials
BASE_URL: str = "https://api.getsigna.ai/v1"

def get_high_conviction_signals(
    min_consensus: int = 5,
    min_confidence: int = 80,
    asset_class: str = "equity"
) -> List[Dict[str, Any]]:
    """
    Fetch the latest Signa signals filtered by consensus count and confidence score.

    Args:
        min_consensus: Minimum number of agents required to have reached consensus.
        min_confidence: Minimum confidence score (0-100).
        asset_class: One of 'equity', 'crypto', 'commodity'.

    Returns:
        A list of signal records matching the specified criteria.

    Raises:
        requests.HTTPError: On non-2xx API responses.
    """
    headers = {
        "Authorization": f"Bearer {API_KEY}",
        "Accept": "application/json",
    }
    params = {
        "min_consensus": min_consensus,
        "min_confidence": min_confidence,
        "asset_class": asset_class,
        "is_latest": True,
    }
```



```

response = requests.get(
    f"{BASE_URL}/signals/latest",
    headers=headers,
    params=params,
    timeout=10,
)
response.raise_for_status()

payload = response.json()
return payload.get("data", [])

if __name__ == "__main__":
    signals = get_high_conviction_signals()
    for signal in signals:
        ticker = signal["ticker"]
        action = signal["signal_action"]
        score = signal["confidence_score"]
        entry = signal["levels"]["entry"]
        stop = signal["levels"]["stop"]
        target = signal["levels"]["target"]
        print(
            f"{ticker:6s} | {action:4s} | Score: {score:3d} | "
            f"Entry: {entry:.2f} | Stop: {stop:.2f} | Target: {target:.2f}"
        )

```

9.2 Webhook Payload Schema

The following JSON object represents a complete webhook payload delivered by Signa upon publication of a new signal. Clients should validate the `X-Signa-Signature` header against an HMAC-SHA256 digest of the raw request body using their registered webhook secret before processing the payload.

JSON

```

{
  "event_id": "evt_a1b2c3d4e5f6a7b8c9d0e1f2",
  "timestamp": "2026-06-15T14:30:00.000Z",
  "type": "signal.generated",
  "schema_version": "2.0",
  "data": {
    "signal_id": "sig_9f8e7d6c5b4a3210fedcba98",
    "ticker": "AAPL",
    "asset_class": "equity",
    "generated_at": "2026-06-15T14:29:58.412Z",
    "is_latest": true,

```



```

"signa_action": "BUY",
"engine_direction": "BULLISH",
"data_direction": "LONG",
"levels": {
    "entry": 185.50,
    "stop": 179.00,
    "target": 198.00,
    "risk_reward_ratio": 1.92
},
"confidence_score": 92,
"consensus_count": 6,
"credibility_weight": 1.0,
"timeframe": "Swing",
"agent_breakdown": [
    { "agent": "stage-scanner",    "direction": "BULLISH", "sub_score": 88 },
    { "agent": "trend-engine",    "direction": "BULLISH", "sub_score": 95 },
    { "agent": "bollinger-band",  "direction": "BULLISH", "sub_score": 91 },
    { "agent": "momentum",        "direction": "BULLISH", "sub_score": 87 },
    { "agent": "relative-strength", "direction": "BULLISH", "sub_score": 93 },
    { "agent": "volume-profile",  "direction": "BULLISH", "sub_score": 90 }
]
}

```

9.3 Webhook Signature Verification (Python)

Python

```

import hashlib
import hmac
import os

def verify_signa_webhook(
    raw_body: bytes,
    signature_header: str,
    webhook_secret: str
) -> bool:
    """
    Verify the HMAC-SHA256 signature of an incoming Signa webhook payload.

    Args:
        raw_body: The raw, unparsed request body bytes.
        signature_header: The value of the X-Signa-Signature header.
        webhook_secret: The shared secret registered with Signa for this endpoint.

    Returns:
        True if the signature is valid; False otherwise.
    """

```

```
#####
expected_signature = hmac.new(
    key=webhook_secret.encode("utf-8"),
    msg=raw_body,
    digestmod=hashlib.sha256,
).hexdigest()

return hmac.compare_digest(expected_signature, signature_header)
```

10. Compliance and Data Governance

Institutional data sourcing teams operate under rigorous regulatory and fiduciary obligations. Signal has designed its data governance framework to meet the expectations of registered investment advisers, hedge fund compliance officers, and institutional risk committees.

10.1 Data Sourcing and MNPI Compliance

Signal's signal pipeline operates exclusively on publicly available market data: exchange-reported price and volume data, publicly accessible order book information, and publicly available technical indicators derived therefrom. Signal does not ingest, process, or distribute any information that could constitute Material Non-Public Information (MNPI) as defined under Section 10(b) of the Securities Exchange Act of 1934 and SEC Rule 10b-5.

This design is intentional and fundamental to the product architecture. Clients can consume Signal signals with confidence that they are not exposed to the MNPI-related risks that have drawn SEC scrutiny to certain categories of alternative data providers. Full documentation of data sourcing methodology is provided as part of the standard Due Diligence Questionnaire (DDQ) package.

10.2 Due Diligence Questionnaire (DDQ)

Signal maintains a comprehensive, pre-prepared DDQ aligned with the expectations of institutional compliance teams and the guidance published by the Financial Information Services Division (FISD). The DDQ covers:

- Data collection methodology and provenance documentation.
- Legal rights to collect, process, and redistribute the underlying data.
- Privacy policy and data handling procedures.
- Information security controls and incident response procedures.
- Employee trading restrictions with respect to covered securities.

- Escalation procedures for compliance issues or suspected data integrity events.

The DDQ is available to prospective clients under NDA upon request. Signal is prepared to participate in follow-up due diligence calls with client compliance and technology teams.

10.3 Information Security

Signal's data delivery infrastructure is designed to meet institutional information security standards:

- **Encryption in Transit:** All API communications are secured via TLS 1.3. Webhook deliveries are made exclusively over HTTPS.
- **Authentication:** API access is governed by scoped bearer tokens. Tokens can be restricted to specific endpoints, asset classes, or IP address ranges. Token rotation is supported without service interruption.
- **Webhook Integrity:** All webhook payloads are signed with HMAC-SHA256 using a per-endpoint shared secret, enabling clients to verify payload authenticity and detect tampering.
- **Access Logging:** All API access events are logged with timestamp, endpoint, client identifier, and response code. Logs are available to Enterprise clients via a dedicated audit endpoint.

10.4 Service Level Agreements

Enterprise contracts include formal SLA commitments covering:

SLA Metric	Commitment
API Availability	99.9% monthly uptime (excluding scheduled maintenance windows, notified 48 hours in advance)
Webhook Delivery Latency	P95 delivery within 500ms of signal publication
Historical Data Delivery	Bulk S3/Snowflake snapshots delivered within the contracted schedule window
Incident Notification	Client notification within 30 minutes of any incident affecting data availability or integrity
Data Integrity	Zero tolerance for retroactive revision of published signal records

10.5 Contractual Protections

Signa offers institutional clients contractual representations and warranties regarding data provenance, MNPI status, and legal rights to distribute the data. Enterprise contracts include:

- Representations that the data does not contain MNPI to Signa's knowledge.
- Notice obligations requiring Signa to promptly inform clients of any compliance events or data integrity issues.
- Prohibition on auto-renewal without affirmative client consent, accommodating institutional annual due diligence cycles.
- Mutual confidentiality provisions protecting both client trading strategies and Signa's proprietary methodology.

10.6 Employee Trading Restrictions

Signa maintains and enforces an internal policy restricting employees from trading in securities covered by the signal feed in a manner that could exploit advance knowledge of signal publications. This policy is documented in the DDQ and is subject to annual compliance training for all relevant personnel.

Appendix: Glossary of Key Terms

Term	Definition
Agent	A specialized, independent AI model within the Signa pipeline, trained to identify a specific class of technical pattern or market condition.
Consensus Count	The number of independent agents that produced a directional output for a given signal. Higher counts indicate broader agreement across the pipeline.
Credibility Weight	A mathematical discount applied to the raw consensus score to penalize signals generated by a small number of agents. Prevents inflated confidence scores from low-coverage tickers.
DaaS	Data-as-a-Service. A delivery model in which data is provided to clients via API or cloud

	delivery channels rather than as a one-time file transfer.
DDQ	Due Diligence Questionnaire. A structured compliance document used by institutional data buyers to evaluate data vendors.
MNPI	Material Non-Public Information. Information that has not been publicly disclosed and that a reasonable investor would consider important in making an investment decision. Trading on MNPI is prohibited under US securities law.
Point-in-Time Data	Historical data that reflects the state of information as it existed at a specific moment in time, without retroactive revision. Essential for preventing look-ahead bias in backtesting.
Signal	A complete, published record from the Signa pipeline containing a directional recommendation, execution parameters, confidence score, and agent breakdown for a specific ticker at a specific moment.

For inquiries regarding institutional access, trial data, pricing, or to request a formal Due Diligence Questionnaire package, please contact:

institutional@getsigna.ai

getsigna.ai

This document is intended for institutional distribution only. It does not constitute investment advice, a solicitation to trade, or a guarantee of future performance. Past signal performance is not indicative of future results. All institutional clients are encouraged to conduct independent due diligence and consult their own legal and compliance counsel before integrating third-party data feeds into their investment processes.