PRD — Crypto News → Structured Signals API

Status: Draft
Version: v1
Last Updated: 2026-01-29
Authors: TBD

---

1. Executive Summary

This product is a **B2B, API-first intelligence layer** that converts unstructured crypto news text into **standardized, machine-readable event objects**.

Given arbitrary crypto-related text (articles, posts, press releases), the system outputs **clean, normalized JSON** that downstream systems can consume deterministically.

The API acts as the missing translation layer between **human-readable news** and **machine logic**, enabling automated trading, alerting, analytics, and agent-based decision-making.

There is no UI and no news aggregation. This is pure infrastructure.

---

2. Problem Statement

2.1 Current State

Crypto news is fundamentally unstructured:

- Articles
- Blog posts
- Tweets
- Press releases

However, every serious crypto system requires **structured inputs**. As a result, teams are forced to:

- Manually tag news
- Maintain brittle regex pipelines
- Build bespoke NLP stacks with inconsistent outputs

This work is duplicated across:

- Trading firms
- Funds
- Analytics platforms
- Research desks

Every mature organization eventually builds this internally.

2.2 Desired Outcome

A **canonical, standalone API** that:

- Transforms raw text into a normalized event schema
- Uses strict enums and versioned fields
- Produces confidence-weighted, queryable outputs
- Can be safely integrated into automated systems

---

3. Goals & Non-Goals

3.1 Goals

ID	Goal	Priority
G1	Convert unstructured crypto news into structured JSON	P0
G2	Provide deterministic, schema-safe outputs	P0
G3	Enable programmatic filtering and querying	P0
G4	Support automated trading, alerts, and agents	P0

G5	Establish a canonical crypto event taxonomy	P1

3.2 Non-Goals

ID	Non-Goal	Rationale
NG1	Consumer-facing UI	API-only product
NG2	News sourcing or aggregation	Input text is user-provided
NG3	Trading execution	Signals only
NG4	Alpha or performance guarantees	Intelligence layer only

---

4. Target Users

4.1 Primary Users

- Quant trading teams
- Crypto hedge funds
- Prop trading desks
- Analytics & data startups
- AI agent developers

4.2 Common Use Cases

- Event-driven trading logic
- Jurisdictional risk alerts
- Research dashboards
- Sentiment and narrative tracking
- Autonomous agent inputs

---

5. Product Scope (v1)

5.1 Core Capability

Given a single block of crypto-related text, return **exactly one canonical event object**.

Multi-event extraction is explicitly out of scope for v1.

5.2 API Endpoint

POST /parse

Request

```
{
  "text": "BlackRock’s Bitcoin ETF saw $400M in inflows after SEC approval."
}
```

Response

```
{
  "event_type": "ETF_INFLOW",
  "topics": ["ETF", "REGULATION"],
  "assets": ["BTC"],
  "entities": ["BlackRock", "SEC"],
  "jurisdiction": "US",
  "sentiment": "positive",
  "impact_score": 0.74,
  "confidence": 0.81,
  "time_horizon": "short_term",
  "schema_version": "v1",
  "model_version": "news-parser-1.0"
}
```

---

6. Event Model

6.1 Event Cardinality

- **Exactly one `event_type` per request**
- If multiple events are detected:
    - The highest-impact primary event is selected

- All others are ignored

---

7. Event Taxonomy (v1)

7.1 Canonical Event Types

Closed-set enum for v1:

- ETF_APPROVAL
- ETF_REJECTION
- ETF_FILING
- ETF_INFLOW
- ETF_OUTFLOW
- ENFORCEMENT_ACTION
- EXCHANGE_HACK
- STABLECOIN_ISSUANCE
- STABLECOIN_DEPEG
- CEX_INFLOW
- CEX_OUTFLOW
- PROTOCOL_UPGRADE
- MINER_SHUTDOWN

No new event types may be added to v1.
Extensions require a new API version.

---

8. Output Schema

8.1 Required Fields

Field	Type	Description
event_type	enum	Canonical event classification
topics	string[]	High-level themes
assets	string[]	Uppercase tickers
entities	string[]	Normalized proper nouns
jurisdiction	enum	Geographic scope
sentiment	enum	positive / neutral / negative
impact_score	float	0.0–1.0 market relevance
confidence	float	0.0–1.0 model confidence

8.2 Optional Fields

Field	Type	Description
market_direction	enum	bullish / bearish / neutral
systemic_risk	boolean	Market-wide implications
retail_relevant	boolean	Consumer-facing relevance
time_horizon	enum	short_term / medium_term / long_term

---

9. Jurisdiction Resolution (v1)

- Jurisdiction is inferred **only when explicitly referenced** (e.g., SEC → US).
- If no explicit signal exists:
    - `jurisdiction = "GLOBAL"`
- No probabilistic guessing in v1.

---

10. Determinism & Reproducibility

10.1 Deterministic Mode

Optional request parameter:

```
{
  "text": "...",
  "deterministic": true
}
```

When enabled:

- Same input text + same model version → same output
- Suitable for backtesting and audits

10.2 Versioning

Every response includes:

- `schema_version`
- `model_version`

---

11. System Architecture (High Level)

```
Client
  |
  | POST /parse
  ▼
API Gateway
  |
  ├─ Text Normalization
  ├─ Event Classification
  ├─ Entity & Asset Extraction
  ├─ Sentiment Analysis
  ├─ Impact Scoring
  |
  ▼
Structured JSON Response
```

---

12. Performance Targets

Metric	Target
p95 latency	< 700ms
Schema validity	100%
API availability	99.9%
Error transparency	Explicit failures only

v1 prioritizes **analysis depth over ultra-low latency**.

---

13. Monetization & Rate Limits

13.1 Pricing Tiers

Tier	Limits	Price
Free	100 requests/day	$0
Pro	10k requests/month	$29
Funds	Unlimited (soft cap)	$199
Enterprise	Custom	Contract

13.2 Enforcement

- API key required
- Per-key rate limiting
- Hard enforcement on Free tier

---

14. Security & Abuse Considerations

- Text-only input (no URLs fetched)

- No code execution
- Rate limiting and abuse detection
- No side effects or state mutation

---

15. Roadmap

v1

- Single-event extraction
- English-only input
- Fixed taxonomy

v2

- Multi-event extraction
- Webhooks / streaming output
- Extended taxonomy

v3

- Historical backfill
- Event correlation
- Cross-event reasoning

---

16. Explicit Out of Scope (v1)

- News aggregation
- Price prediction

- Alpha guarantees
- UI dashboards
- Multi-event outputs

---

17. Success Metrics

Metric	Description
Active API keys	Adoption
Requests per customer	Depth of integration
Paid conversion	Free → Pro
Retention	Stickiness
Automation usage	Used in live systems

---

18. Open Questions (Post-v1)

- Multi-event response format
- Non-English language support
- Streaming vs polling
- Ontology governance process

End of Document