

Canonical Taxonomy and Object Model

Purpose and scope

This document defines the canonical nouns, object boundaries, cardinalities, state models, provenance laws, and execution semantics of the Kinetic intelligence platform before interface and workflow design.

NORMATIVE LANGUAGE

MUST and **MUST NOT** are invariants. **SHOULD** is the recommended default. **MAY** is an allowed extension. Terms marked Deferred are outside the v1 universal intelligence-production contract.

ANCHOR**Subject** anchors every Action Run, Operation Run, Signal, and File.

SOURCE	RECORD	RESULT	OPERATION	SIGNAL	FILE
where Kinetic looks	what Kinetic retrieves	what an Action returns	how Results are analyzed	what becomes meaningful	what a person keeps

ACQUISITION	Source · Source Rights · Record · Citation
MACHINE	Action · Action Run · Endpoint · Result
INTELLIGENCE	Operation · Operation Run · Signal
HUMAN	File · File Version · Share · Export
CONTINUITY	Watch Config · Watch Run · Active File
ACCESS	Account · Member · Role · Capability · Entitlement · API Key · Service Account
DEFERRED	Case · Matter · Mandate · Actions Taken · Outcomes

Complete term register

TERM	LAYER	NORMATIVE DEFINITION	TYPE	CARDINALITY	VISIBILITY
Account	Access	Tenant, billing, entitlement, policy, ownership, and audit root.	First-class resource	1:N Members; 1:N Capabilities via Entitlements; 1:N API Keys; 1:N Files	Customer + system
Member	Access	Authenticated human identity associated with an Account through a role.	First-class resource	N:1 Account in v1; may become N:M for multi-account membership	Customer + system
Role	Access	Named permission set assigned to a Member within an Account.	Enum / policy	Owner, Admin, Member, Viewer	Customer + system
API Key	Access	Revocable Account-scoped credential created by an authorized Member.	Secret-bearing resource	N:1 Account; optional N:1 Service Account	Developer + system
Service Account	Access	Optional named non-human identity used by software or infrastructure.	First-class resource	N:1 Account; 1:N API Keys	Developer + system
Machine	Access	Plain-language term for the runtime caller presenting an API Key. Not necessarily persisted.	Runtime actor	Authenticated through API Key	Developer
Capability	Access	Entitlement and discovery grouping containing related Operations and Endpoints.	Catalog resource	M:N Accounts via Entitlements; 1:N Operations; 1:N Endpoints	Customer + system
Entitlement	Access	Effective grant allowing an Account to discover or invoke a Capability, Operation, or Endpoint under stated limits.	Policy resource	N:1 Account; references granted object; time and usage bounded	System
Source	Acquisition	Provider, registry, chain, court, feed, archive, database, or supplied collection Kinetic is authorized to access.	Catalog resource	1:N Source Rights; 1:N Records	Internal
Source Rights	Acquisition	Permitted-purpose and downstream-use contract governing a Source.	Policy resource	N:1 Source; constrains Action, Result, API, MCP, File, export, retention	Internal
Record	Acquisition	Exact retrieved filing, transaction, page, docket event, list entry, document, or observation.	Immutable resource	N:1 Source; M:N Results through Citations	System + inspectable
Citation	Provenance	Exact typed link from a Result or Signal to a supporting Record.	Relationship resource	N:1 Record; N:1 cited Result or Signal	System + customer
Subject	Core	Primary person, company, wallet, domain, asset, document set, event, or question under examination.	First-class resource	1:N Action Runs; 1:N Operation Runs; 1:N Files; N:M related Subjects	Customer + system
Action	Machine	Atomic, versioned engine task that retrieves, normalizes, resolves, or computes.	Versioned definition	1:N Action Runs; may use N Sources; exposed by 0:1 Endpoint	Internal
Action Run	Machine	One execution of an Action against a frozen input and Subject context.	Append-only event/resource	N:1 Action version; N:1 Subject; produces exactly 1 Result	System + inspectable

TERM	LAYER	NORMATIVE DEFINITION	TYPE	CARDINALITY	VISIBILITY
Endpoint	Machine	Entitled machine-access contract exposing an Action through API or MCP.	Versioned interface	Maps to one Action contract; invoked by N Action Runs	Developer + advanced customer
Result	Machine	Immutable canonical structured return from one Action Run.	Immutable resource	1:1 Action Run; M:N Records through Citations; consumed by N Operation Runs	Developer + system
Operation	Intelligence	Versioned analytical recipe that interprets one or more Results for a defined purpose.	Versioned definition	1:N Operation Runs; declares required Actions/Endpoints and Signal contracts	Customer + system
Operation Run	Intelligence	One reproducible execution of an Operation over pinned Results and rule versions.	Append-only resource	N:1 Operation version; N:1 Subject; consumes N Results; produces N Signals	System + inspectable
Signal	Intelligence	Discrete typed intelligence statement produced by an Operation from Results.	Immutable resource	N:1 Operation Run; M:N Results and Records; may enter N File Versions	Customer + system
File	Human	Persistent human intelligence object organizing Signals and supporting Results around a primary Subject.	First-class mutable identity	N:1 Account; N:1 primary Subject; 1:N File Versions; 0:1 Watch Config	Customer
File Version	Human	Immutable snapshot of a File at a point in time, pinning exact Signals, Results, structure, and manifest.	Immutable resource	N:1 File; references N Signals and Results	Customer + system
Watch Config	Continuity	Settings that make a File Active: triggers, cadence, conditions, pinned Operations, and notification policy.	Mutable configuration	0:1 per File in v1; creates N Watch Runs	Customer + system
Watch Run	Continuity	One append-only evaluation of an Active File under its Watch Config.	Append-only resource	N:1 Watch Config; invokes N Operations/Endpoints; yields File Version or No Change	System + inspectable
Share	Human	Revocable permission or signed reference allowing another identity to view a File or File Version.	Access resource	N:1 File or File Version; has scope, expiry, revocation	Customer + system
Export	Human	Rendered or machine-readable edition produced from a pinned File Version.	Derived immutable artifact	N:1 File Version; PDF, DOCX, JSON, signed web edition	Customer + system
Case	Deferred	Future enforcement container organizing multiple Files. Not a stage in intelligence production.	Deferred first-class resource	1:N Files or N:M if later required	Deferred

Source

An authorized location or provider from which Kinetic may retrieve Records.

INPUTS / REFERENCES

Source Rights, credentials, source class, jurisdiction, permitted purpose.

OUTPUTS / RELATIONS

Records. Availability and rights constraints inherited downstream.

INVARIANTS

- A Source is a provider or collection, not an individual retrieved item.
- No Source enters runtime before rights review.
- A desktop subscription without downstream display or derivative rights is not a protocol Source.

NOT THIS

- Not a Citation.
- Not a Record.
- Not a customer-facing guarantee of completeness.

```
source {  
  id, class, jurisdiction, status,  
  rights_policy_id, freshness_policy,  
  failure_policy, recall_policy  
}
```

Record

The exact source-native item Kinetic retrieved and preserved as an observation.

INPUTS / REFERENCES

Source, retrieval event, locator, observed_at, content or normalized reference.

OUTPUTS / RELATIONS

Citations into Results and Signals.

INVARIANTS

- Immutable after acquisition. Corrections produce a new Record.
- Carries Source, observation time, content hash, rights tags, and retention constraints.
- Provider-native conflict is preserved, never silently reconciled.

NOT THIS

- Not the full Result.
- Not automatically Evidence.
- Not Kinetic analysis.

```
record {
  id, source_id, source_locator,
  observed_at, effective_at?, content_hash,
  rights_tags[], retention_policy,
  payload_ref
}
```

Subject

INPUTS / REFERENCES

Account context, primary identifier(s), subject type, supplied context.

INVARIANTS

- Every Action Run and Operation Run identifies a Subject.
- Subject identity is not silently collapsed with a related entity.
- Cross-tenant Actor resolution remains internal and never leaks through Subject.

OUTPUTS / RELATIONS

Action Runs, Operation Runs, Signals, Files, related-Subject edges.

NOT THIS

- Not necessarily a person.
- Not an Actor assertion.
- Not a Case.

```
subject {  
  id, account_id, type,  
  display_name, identifiers[],  
  supplied_context, created_at  
}
```

Action / Endpoint / Result

The atomic machine sequence. An Action performs one task. Its Endpoint is the callable contract. An Action Run returns one Result.

INPUTS / REFERENCES

Subject, typed input, Action version, entitled Account, Sources, frozen execution context.

OUTPUTS / RELATIONS

Exactly one immutable canonical Result per completed Action Run.

INVARIANTS

- Action definitions are versioned.
- Endpoint access fails closed if unavailable or not entitled.
- A Result has an immutable ID, hash, manifest, coverage statement, conflicts, gaps, and Citations.
- Retries do not create contradictory successful Results without lineage.

NOT THIS

- Action is not Capability.
- Endpoint is not the Result.
- Result is not provider-raw JSON.
- Result is not a Signal.

```
action_run {
  id, action_id, action_version,
  subject_id, account_id, input_hash,
  started_at, completed_at, status
}
result {
  id, action_run_id, schema_version,
  objects[], relationships[], collections[],
  citations[], conflicts[], gaps[], coverage,
  rights, manifest, content_hash
}
```


Operation / Signal

The analytical sequence. An Operation interprets pinned Results using versioned rules. An Operation Run emits typed Signals.

INPUTS / REFERENCES

Subject, Operation version, pinned Result IDs, frozen rule set, graph snapshot, model version if applicable.

OUTPUTS / RELATIONS

Zero or more immutable Signals, plus explicit No Signal or Insufficient Coverage state.

INVARIANTS

- Same frozen inputs and versions produce byte-equivalent typed outputs.
- Signals retain direct lineage to Results and Records.
- Signal class is explicit: observation, derived fact, scored inference, contradiction, change, connection, or gap.
- Competing explanations and limits remain visible.

NOT THIS

- Operation is not an Action bundle sold as a Capability.
- Signal is not narrative prose.
- Signal is not advice, verdict, prediction, or opaque risk score.

```
operation_run {
  id, operation_id, operation_version,
  subject_id, result_ids[], rule_version,
  graph_snapshot, status, manifest
}
signal {
  id, operation_run_id, class, statement,
  support_result_ids[], citation_ids[],
  confidence?, competing_explanations[],
  limits[], effective_at, content_hash
}
```

File / File Version

The persistent human intelligence object and its immutable historical snapshots.

INPUTS / REFERENCES

Account, primary Subject, selected Signals, supporting Results, structure, annotations, presentation policy.

OUTPUTS / RELATIONS

File Versions, Shares, Exports, Watch Config.

INVARIANTS

- File identity persists while versions append.
- Every File Version pins exact Signals and Results.
- A File may accumulate outputs from multiple Operation Runs.
- Presentation cannot remove required provenance, uncertainty, conflicts, or stated limits.

NOT THIS

- Not one Operation export forever.
- Not a Case.
- Not silently overwritten by Watch.

```
file {
  id, account_id, primary_subject_id,
  title, lifecycle_status, watch_mode,
  current_version_id, created_at
}
file_version {
  id, file_id, sequence,
  signal_ids[], result_ids[], structure,
  annotations[], manifest, content_hash,
  created_by, created_at
}
```

Watch / Active File

Continuity behavior that reruns pinned intelligence work and appends change to a File. Active is the customer-facing derived state when Watch is on.

INPUTS / REFERENCES

File, Watch Config, trigger, pinned Operation/Endpoint versions, notification policy.

OUTPUTS / RELATIONS

New File Version, No Change event, or Attention state.

INVARIANTS

- Watch mode is independent from File lifecycle status.
- Every Watch Run is auditable and idempotent.
- No Change is a durable outcome.
- A change never mutates a prior File Version.

NOT THIS

- Active is not a permanent File type.
- Watch is not primary navigation.
- Scheduled polling is not the only activation mode.

```
watch_config {
  file_id, mode, triggers[], cadence?,
  conditions[], pinned_operations[],
  notification_policy, next_evaluation_at
}
watch_run {
  id, file_id, config_version,
  trigger_type, started_at, outcome,
  new_file_version_id?, no_change_reason?
}
```

Capability / Entitlement

Capability is a customer discovery and access grouping.
Entitlement is the effective Account grant.

INPUTS / REFERENCES

Catalog definitions, Account, offer, rights constraints, capacity and usage policy.

OUTPUTS / RELATIONS

Visible Operations and Endpoints; authorization decision at invocation.

INVARIANTS

- Capability never means Action.
- Complete access does not imply unlimited compute or licensed-source use.
- Global Endpoint OFF blocks new invocation without deleting historical Results or Files.

NOT THIS

- Not an execution step.
- Not a Result family.
- Not a role.

```
capability {  
  id, name, operation_ids[], endpoint_ids[],  
  availability, display_policy  
}  
entitlement {  
  id, account_id, grant_type, grant_id,  
  starts_at, ends_at?, usage_policy,  
  rights_constraints, status  
}
```

Account / Member / Machine Access

INPUTS / REFERENCES

Verified identity, Account policy, role assignment, credential issuance.

OUTPUTS / RELATIONS

Authorized human sessions and machine requests with complete audit lineage.

INVARIANTS

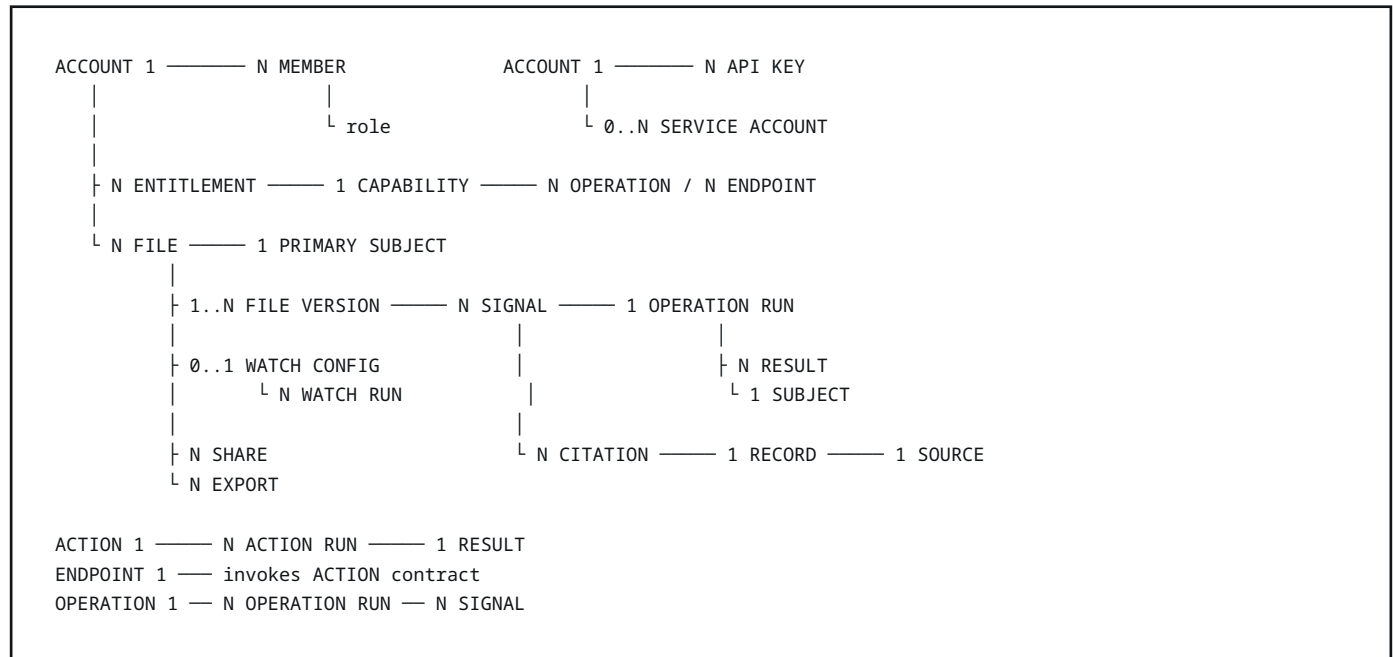
- An Account has Members; it does not belong to one generic Operator.
- API Keys belong to the Account and survive Member offboarding.
- Issuing Member is retained in audit history.
- Each key or Service Account has scopes, spend caps, concurrency caps, expiry and revocation.

NOT THIS

- Operator is not the generic user noun.
- Machine need not be a persisted object.
- A shared API key is not a Member identity.

```
account { id, name, status, policy }
member { id, identity_id, account_id, role, status }
service_account { id, account_id, name, scopes[] }
api_key { id, account_id, service_account_id?,
  issued_by_member_id, scopes[], expires_at?, revoked_at? }
```

Canonical relationship graph

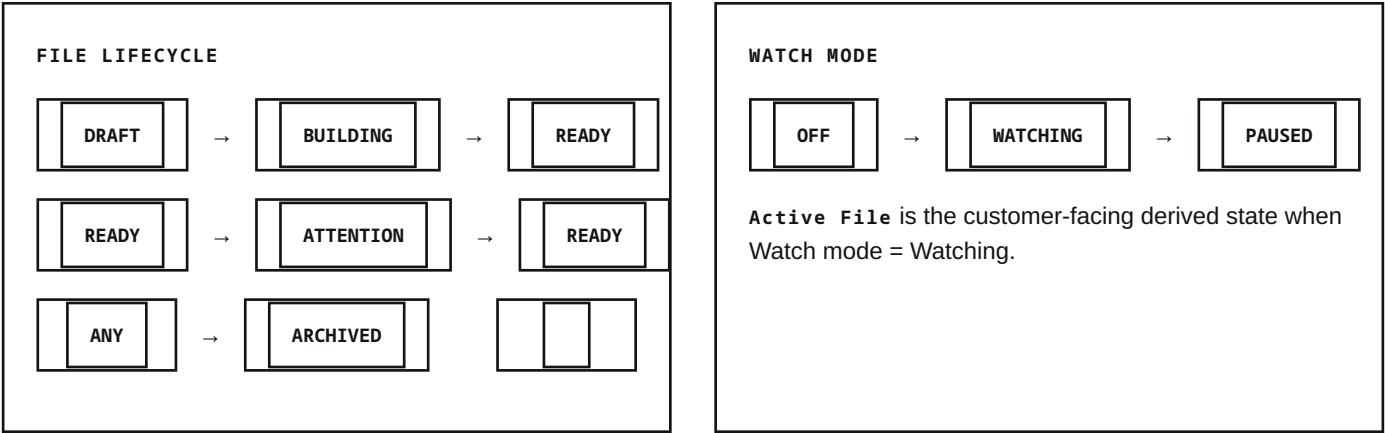


CARDINALITY DECISIONS

1. A completed Action Run produces exactly one canonical Result.
2. An Operation Run consumes one or more pinned Results and produces zero or more Signals.
3. A File may accumulate Signals from multiple Operation Runs across versions.
4. A File has exactly one primary Subject in v1 and may reference related Subjects.
5. Watch is optional. When enabled, one Watch Config governs many append-only Watch Runs.
6. Case is deferred and is never inserted into the production chain.

File state and continuity

Lifecycle status describes the condition of the File. Watch mode describes whether Kinetic is evaluating it for change. They **MUST** remain separate.



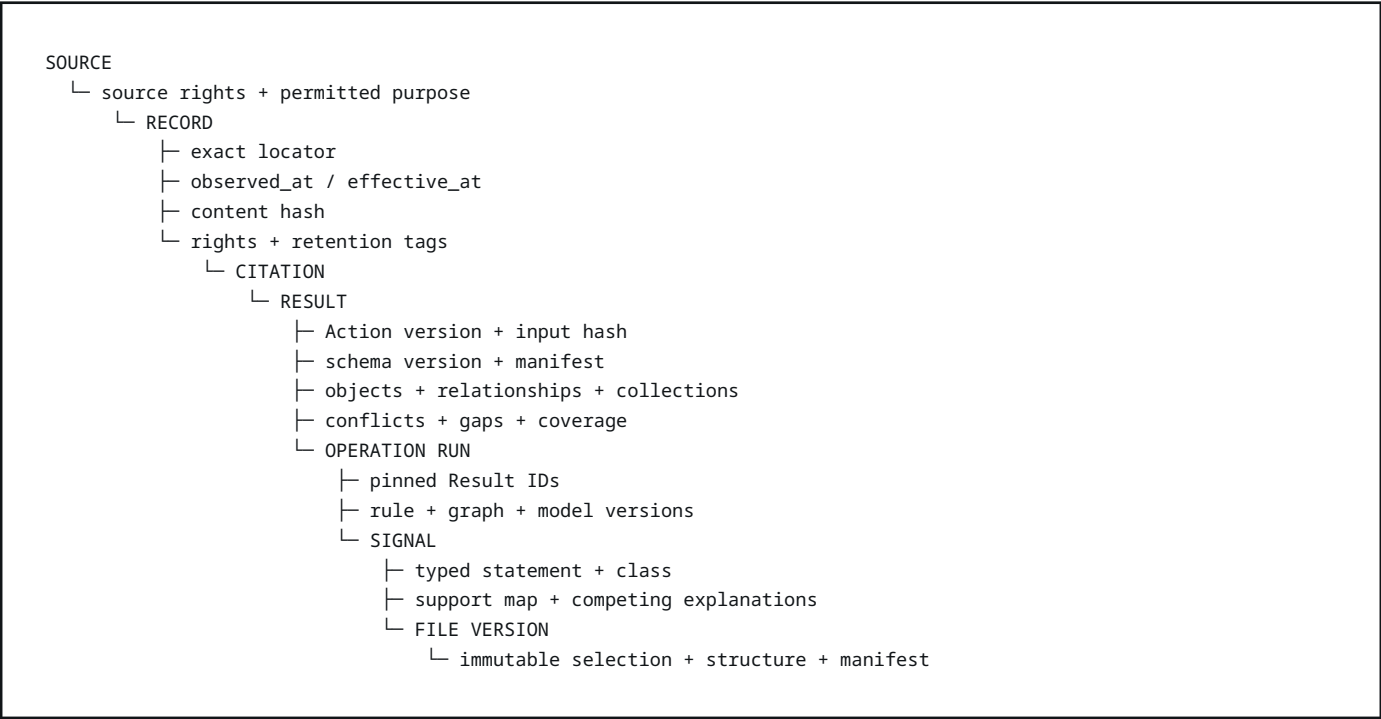
ACTIVATION TRIGGER CLASSES

SCHEDULE	Daily, weekly, monthly, or supported custom cadence.
RECORD CHANGE	New filing, transaction, docket event, list entry, domain state, or other source event.
CONDITION	Amount, match, relationship, severity, contradiction, or other declared threshold.
ON DEMAND	Authorized Member requests evaluation now.

OUTPUT LAW

Every Watch Run **MUST** produce a new File Version, an explicit No Change event, or an inspectable failure. It **MUST NOT** silently overwrite prior state.

Scientific lineage chain



REPRODUCIBILITY STANDARD

Same frozen inputs, Action and Operation versions, source snapshot, rule set, graph snapshot, and model version MUST produce byte-equivalent typed outputs. Reproducible does not mean certain or complete. Missing data, conflict, uncertainty, and coverage limits remain explicit.

EPISTEMIC CLASSES

OBSERVATION	Direct canonical fact represented in a Record or Result.
DERIVED FACT	Deterministic transformation of observations.
SCORED INFERENCE	Probabilistic assessment with deterministic calculation and full lineage.
CONTRADICTION	Incompatible supported claims preserved together.
GAP	Declared missing, unavailable, or insufficient coverage.

Platform laws

-
- 01 Every Action Run and Operation Run MUST bind to an Account, Subject, entitlement decision, and versioned definition.
-
- 02 Every completed Action Run MUST return exactly one canonical Result or an explicit failure. Partial completion MUST be typed.
-
- 03 Every Result MUST retain immutable identity, schema version, content hash, manifest, coverage, conflicts, gaps, rights, and citations.
-
- 04 Every Signal MUST trace to an Operation Run, supporting Results, and exact cited Records.
-
- 05 No generative narrative MAY introduce a claim absent from typed Results or Signals.
-
- 06 Kinetic MUST state what it found and how it found it. It MUST NOT issue advice, recommendations, legal conclusions, or opaque verdicts in the truth path.
-
- 07 File Versions MUST be append-only and MUST pin exact Signal and Result versions.
-
- 08 Watch MUST create a File Version, No Change event, or inspectable failure. Silent overwrite is prohibited.
-
- 09 Endpoint authorization MUST fail closed. Global OFF blocks new invocation without deleting historical objects.
-
- 10 Tenant data MUST NOT improve another tenant's Result without explicit contribution and derivative rights.
-
- 11 Source rights MUST propagate through Result delivery, API/MCP eligibility, File sharing, Export, retention, and recall.
-
- 12 API Keys MUST belong to Accounts, not individual Members. Issuance and revocation remain audited.
-
- 13 Capability MUST remain an access and discovery grouping. It MUST NOT become a synonym for Action.
-
- 14 Operator MUST remain reserved for named products and actual operating roles, not generic customer identity.
-
- 15 Case MUST remain outside the universal intelligence-production chain until separately specified.

End-to-end object trace

ACCOUNT	Northstar Advisory
MEMBER	Avery Chen · Admin
SUBJECT	Northstar Trading Ltd. · company
SOURCES	Company registry · court docket · DNS history
RECORDS	Two ownership filings · four docket events · six historical DNS observations
ACTIONS	entity.ownership_history · court.docket_events · domain.history
RESULTS	Three immutable canonical JSON returns with 12 Records and full manifests
OPERATION	Deep Screen · version 2.3
SIGNAL	The named owner changed, but the same signatory and domain administrator persisted across both changes.
FILE	Northstar Trading Ltd. · version 07 · Signal plus supporting Results and Citations
ACTIVE	Record-change trigger for ownership filing, docket event, or DNS state change

INTERPRETATION BOUNDARY

The Signal identifies supported continuity. It does not assert hidden ownership, intent, control, liability, or recommended action.

Questions for the UI and workflow phase

Q01 · DEFAULT ACTIVATION

Which default Watch trigger and cadence applies by Subject type and Operation?

Q02 · FILE COMPOSITION

Does every Operation Run create a draft File, or may Signals remain unfiled until selected?

Q03 · RELATED SUBJECTS

When does a related person, wallet, domain, or company become a first-class linked Subject rather than a Result object?

Q04 · SHARING

Is Share a live revocable pointer, a frozen File Version, or both? What are default expiry and download rights?

Q05 · NO SIGNAL

How should a complete Operation Run with adequate coverage but no qualifying Signal appear to a human?

Q06 · CAPABILITY PACKAGING

Which Capabilities are commercial entitlements versus discovery-only catalog groupings?

Q07 · COMPLETE RESULT DELIVERY

Exact thresholds and semantics for summary, standard, complete, pagination, streaming, and resource handles.

Q08 · CASE ATTACHMENT

Deferred: whether File-to-Case is 1:N or N:M and whether attachment pins a File Version.

NEXT METHOD

Represent these objects and laws in the interface. Validate the relationships visually. Then specify workflows, permissions, failure states, and user interactions. Freeze view-model contracts only after the object model survives that review.