Companion Health™

A Platform of Animetric Intelligence Systems LLP

Companion Health™ · Fetch™ · Companion Intelligence Network™

compn.co

ECOSYSTEM

Platform

Fetch™

Developer Documentation

Trust Centre

Companion Intelligence Network™

Roadmap

CONTACT

hello@compn.co

partners@compn.co

developers@compn.co

trust@compn.co

legal@compn.co

Architectural Status Notice. This document describes Companion Health™ target technical architecture, interoperability framework, signal-processing doctrine, and longitudinal intelligence operating model. Descriptions herein should be interpreted as architectural systems specifications and conceptual infrastructure doctrine rather than representations of production deployment status, commercial scale, or operational maturity.

AI & Intelligence. Companion Health™, Fetch™, and the Companion Intelligence Network™ do not make autonomous clinical, veterinary, diagnostic, insurance, underwriting, treatment, research, regulatory, or commercial decisions. All outputs support qualified human review.

Forward-Looking Statements. This document may contain forward-looking statements subject to change without notice; actual outcomes may differ materially.

Data Ownership. Data ownership remains with the originating contributor unless otherwise defined through contractual agreement.

companion | Fetch API TECHNICAL OVERVIEW ARCHITECTURAL SYSTEMS SPEC

● FETCH API · ARCHITECTURAL SYSTEMS SPECIFICATION

The longitudinal *intelligence* layer.

A technical overview of the infrastructure that converts fragmented companion-animal observation into persistent, replayable, interpretive continuity across the ecosystem.

The interoperability surface of a longitudinal inference infrastructure for the pet ecosystem.

01 · TECHNICAL THESIS

Companion Health is a longitudinal inference infrastructure layer for the pet ecosystem.

It synthesises fragmented behavioural, clinical, nutritional, insurance, and research observations into persistent health intelligence over time.

Companion does not replace source systems. It creates interpretive continuity across them.

IDENTITY

Companion Health is biological intelligence infrastructure for the global pet ecosystem. Fetch API is the addressable surface of that infrastructure — the programmable interface through which participants observe, retrieve, and interpret

SCOPE OF THIS DOCUMENT

The architecture underneath Fetch API: the data model, the inference engine, the temporal reconstruction system, the interpretation layer that translates intelligence into consumer-specific outputs, and the integration patterns that

the persistent intelligence the platform synthesises.

connect ecosystem participants to the platform.

02 · ARCHITECTURE OVERVIEW

A nine-layer stack with one cross-cutting orchestration layer.

Each layer carries a single responsibility. Layer boundaries are the contract — what each accepts, what it produces, and what remains its own concern. The Signal Engine is the structural and conceptual centre.

LAYER	RESPONSIBILITY · ACCEPTS → PRODUCES
L9	**Ecosystem Consumers** accepts Briefs, primitives → consumer-specific action
L8	**Fetch API** accepts scoped queries, auth → addressable primitives & Briefs
L7	**Replay Engine** accepts signal lineage → immutable temporal reconstruction
L6	**Risk Synthesis** accepts qualified signals, cohort context → derived intelligence objects
L5	**Confidence Layer** accepts active signals → certainty-qualified signals
L4	**Signal Engine** CENTRE accepts validated observations → persistence-qualified signals
L3	**Validation Layer** accepts raw observations → validated observations
L2	**Ingestion Layer**

	accepts source telemetry streams → normalised observations
L1	**Data Sources** accepts ecosystem activity → raw observation events

Numbered stack

Nine layers form a vertical compositional hierarchy. Telemetry enters at Layer 1, is normalised through Layer 2, validated at Layer 3, and synthesised into signals at Layer 4 — the Signal Engine.

Interpretation Layer

A cross-cutting orchestration layer operates laterally between Layer 8 and Layer 9, composing primitives into consumer-specific Briefs without inventing parallel concepts.

03 · ENGINEERING PRINCIPLES

Five principles govern every design decision.

01 **Longitudinal > Transactional**

Health emerges across time, not in isolated events. The platform reasons over months and years of continuous observation, not single readings.

02 **Interpretation > Storage**

Companion creates meaning from observations rather than accumulating them. Storage is necessary infrastructure; interpretation is the product.

03 **Explainability > Opaqueness**

Every signal carries its reasoning. The platform does not produce risk scores without disclosing the observations that produced them.

04 **Infrastructure > Application**

Companion is ecosystem infrastructure, not a standalone consumer product. Every interface and output is designed to be consumed by other systems.

05

Trust Before Scale

Governance, traceability, and audit precede expansion. The design assumes regulated consumers and adversarial review from the outset.

04 · THE SIGNAL ENGINE

Synthesis is not detection. It is reconstruction.

The engine does not watch for thresholds. It establishes what is normal for a single subject over time, measures sustained deviation against a cohort-adjusted baseline, and reconstructs how a signal emerged. Six operations compose it.

01 · BASELINING

Per-subject baseline

The engine establishes a per-subject behavioural and physiological baseline from longitudinal observation — what is normal for this individual.

02 · DEVIATION

Deviation calculation

Incoming observations are compared against the cohort-adjusted baseline. Deviation magnitude is computed across each axis.

03 · PERSISTENCE

Persistence qualification

Deviation must sustain beyond statistical noise to qualify as a signal.

04 · VELOCITY

Velocity computation

Velocity is the rate and directionality of change in a persisting signal. An accelerating signal compresses the intervention window.

A two-hour deviation is noise; a sustained one is signal.

05 · CONFIDENCE

Confidence modelling

Every signal carries a confidence value between 0.00 and 1.00, modelled against historical validation. Always disclosed; never inferred away.

06 · REPLAY

Replay reconstruction

Once a signal exists, the engine reconstructs its emergence forensically — the day-by-day evolution of the underlying observations.

05 · DATA MODEL & ONTOLOGY

Seven primitives compose every object.

The ontology is the vocabulary of the system — it governs the data model, the API surface, the storage schema, and the consumer interface. Outside it, no terminology is introduced.

Subject	The individual organism whose telemetry is observed, baselined, and synthesised. The atomic identity; all primitives resolve to a Subject.	GET /subjects/{id}

Signal	A behaviourally meaningful deviation from a Subject's baseline, sustained beyond noise tolerance. The unit of intelligence.	GET /subjects/{id}/signals
Persistence	The temporal continuity of a Signal beyond statistical noise. The qualifier that elevates a deviation into a signal.	signal.persistence
Velocity	The rate and directionality of change in a persisting Signal. Determines intervention urgency.	signal.velocity
Confidence	Modelled certainty in a Signal's meaning, 0.00–1.00. Always disclosed alongside the Signal. The platform never asserts; it discloses.	signal.confidence
Replay	The temporal reconstruction of how a Signal emerged and evolved. Forensic, immutable, reviewable.	GET /signals/{id}/replay
Cohort	A statistically grouped population of Subjects providing the	GET /cohorts/{id}

comparative substrate. Cohorts compound — more Subjects, sharper Cohorts.

Signal — canonical schema

The Signal object is the system's unit of currency. Its schema is intentionally narrow — every field load-bearing, every claim disclosed.

Signal		OBJECT · PRIMITIVE
signal_id	string · uuid	Unique identifier for this signal, scoped to tenant.
subject_id	string	The Subject this signal pertains to.
signal_class	enum	Semantic class — renal-trajectory, mobility-stress, behavioural-isolation.
state	enum	Lifecycle: latent · emerging · active · resolved.
persistence	object	Duration of sustained deviation, qualified threshold, current state.
velocity	object	Rate, directionality, deceleration / acceleration profile.
confidence	float · 0–1	Modelled certainty in signal meaning. Always disclosed.
cohort_ref	string · ref	The cohort against which deviation was qualified.
replay_ref	string · ref	Reference to the immutable reconstruction of this signal's emergence.

emerged_at	iso8601	UTC timestamp at which the persistence threshold was crossed.
model_version	string	Inference model that produced this signal. Every signal carries its model lineage.

06 · FETCH API SURFACE

The API exposes the ontology. It does not invent parallel concepts.

Two endpoint families, one worldview: primitive endpoints expose the seven objects directly; Brief endpoints expose consumer-specific compositions. New signal classes are added as values, not new endpoints.

Compositional, not catalogue-driven

New signal classes — renal trajectory, mobility stress, behavioural isolation — are added to the existing contract as new values, not new endpoints.

Confidence-bearing by default

Every response carries confidence for every claim. The contract refuses to expose a signal without disclosing the platform's certainty in it.

Replay-addressable

Every signal exposes a reference to its replay reconstruction. A consumer can always retrieve the temporal lineage that produced it.

Tenant-scoped, model-versioned

Every request resolves within a tenant boundary. Every response carries the model version that produced it.

Canonical Signal response

A Signal retrieval returns the primitive in its canonical shape — narrow, composable, fully disclosed. No diagnosis, no recommendation, no consumer framing.

GET /signals/sig_04A_renal_2026Q2 200 · JSON


```
{
  "signal_id":      "sig_04A_renal_2026Q2",
  "subject_id":     "sub_04A",
  "signal_class":   "renal-trajectory",
  "state":          "active",
  "persistence": {
    "duration_days":  14,
    "qualified_at":   "2026-04-18T11:23:08Z",
    "threshold_days": 6
  },
  "velocity": {
    "profile":       "mixed",
    "components": [
      { "axis": "weight",    "rate": "decelerating" },
      { "axis": "hydration", "rate": "accelerating" }
    ]
  },
  "confidence":     0.81,
  "cohort_ref":     "coh_feline_senior_indoor_v3",
  "replay_ref":     "rep_04A_renal_2026Q2",
  "emerged_at":     "2026-04-18T11:23:08Z",
  "model_version":  "engine-v1.4.0-renal"
}
```


07 · REPLAY ENGINE

Replay turns a predictive system into an evidentiary one.

A signal that cannot be reconstructed cannot be defended. Replay reconstructs the day-by-day emergence of every signal — forensic, immutable, reviewable. Three properties compose it.

01

Immutability

Once a signal has emerged, its reconstruction is permanently fixed. Subsequent observations refine future signals; they do not retroactively alter past ones.

02

Reconstructability

Every signal can be replayed across its temporal window — observation density, deviation accumulation, and the moment persistence qualified.

03

Interpretive lineage

Replay preserves not only what the platform observed but how it interpreted: model version, cohort composition, baseline window, confidence parameters.

08 · INTERPRETATION LAYER

The same signal becomes different intelligence in different consumer hands.

One underlying signal composes into many Briefs — each weighted for the consumer's interpretive frame. A single intelligence layer serves an unbounded number of consumers; Companion never competes with the products that consume it.

01

02

Ecosystem leverage

A single underlying intelligence layer serves an unbounded number of consumer interpretations. The platform does not compete with its consumers; it supplies them.

Non-competitive positioning

Companion never builds the veterinary EMR, the underwriting product, or the caregiver application. It produces the intelligence those products consume.