ruby2 — a discovery-layer protocol for homeless services

Methodology summary, May 2026. Master version (audience-tuned variants per `FRAMING.md` Section 17E).
Potential names:
PEARL(*Provenance-Enriched Aid Resource Locator*): the protocol
SOCKS(*Survival Oriented Community Knowledge System*): the app

---

1. Executive Summary

ruby2 is a protocol — a methodology and a partially-built engine — for an accuracy-first directory of homeless-service organizations. The thesis: **the discovery layer is upstream of the placement and prevention layers everyone else is measuring.** A 95% accurate listing for a homeless person is a 100% experiential failure: the wrong day, the wrong address, or the wrong intake hour means the person walks an hour to a closed door. Most homeless-services directories today don't measure their own failure rate. ruby2 measures its own — and that measurement is the methodology's central artifact.

Built by a protocol architect with 10+ years of homeless-services navigation, across 30+ US states, as the source corpus and modern AI / LLM-infrastructure capability as the operational layer. The combination is market-rare: a builder operating at the intersection of lived expertise and the technical infrastructure required to translate that expertise into a scalable, auditable system.

What makes ruby2 methodologically rigorous is the same thing that, to an execution-focused observer, looks like weakness: the repository documents its own failures — gate-fired audits, mixed and negative findings from production-vendor evaluations, an honest accounting of where the engine works and where it doesn't. Most directory projects ship without ever discovering their false-positive rate. ruby2 knows the number, knows where it comes from, has it documented, and has a methodology designed to drive it down.

Current state (May 2026): the first three stages of the discovery pipeline are shipped end-to-end; stages four through seven are planned or design-shaped. The seeker-facing surface is the gap. **The ask:** research-grade funding partners (foundations, academic-collaboration funders, impact investors, and methodology-adoption pathways at the County and State levels).

---

2. The Problem

2.1 The discovery gap

Most homeless-services directories don't measure their own accuracy.

A directory of soup-kitchen schedules can publish 500 listings without ever asking, "what fraction of these listings are correct on any given Tuesday?" The methodology for *finding* organizations is rarely separated from the methodology for *verifying their current information.* Update cycles are typically annual or worse. Most directories don't carry per-record freshness signals visible to the person reading them.

- *'ruby2' is a reference to the project's predecessor: www.rubyslist.org. rubyslist was born from a collection of spreadsheets hand-curated and validated in-person. The burden of manually verifying and updating the site's schedules is exactly 'the problem' described above*

In LA specifically, this gap manifests in load-bearing infrastructure. The County's CareConnect platform: IBM Connect360-based, launched as part of the JCOD (Justice, Care and Opportunities Department) initiative — publicly reported **0 individuals served from 29 referrals over 22 months** in its first published outcomes window. The referral plumbing exists; the throughput is documented as approximately zero. 211 LA, the County's referral hub, has well-documented accuracy gaps across seeker-side records that frontline advocates routinely correct in the field. LAHSA's Greater Los Angeles Homeless Count produces a Point-in-Time snapshot of the seeker population (75,312 in 2024; 72,308 in 2025) but does not audit ongoing directory accuracy across the producer-side organizations the County's response routes seekers toward.

LA's homelessness-data infrastructure is dense — CHAMP, HMIS, IBHIS, ORCHID, HODA, the LAHSA PIT count, CES, CareConnect, 211 LA — integrated where possible into the County Chief Data Office's CWIH lakehouse, all of it producer-side, oriented toward institutional throughput, placements, beds, behavioral-health utilization, and survey snapshots, with no instrument in the inventory built to serve the seeker at the moment of discovery.

2.2 Why 95% accurate equals 100% experiential failure

For a homeless person, finding services is rarely a research task. It is a same-day, sometimes same-hour task, often performed on a low-battery phone with intermittent connectivity, sometimes by someone in active crisis. A schedule listing that says a meal program runs Tuesday at noon, when in fact it moved to Thursday three months ago, is not a 95% accurate listing in any meaningful sense. The person walks to a closed door, loses an hour, loses some trust in the system, and may not try again that week. The downstream cost of one bad referral compounds: missed services lead to missed downstream services, lead to missed appointments, lead to missed housing placements.

This is why ruby2's accuracy bar is **100% or nothing.** "Good enough for v1" is not an acceptable framing in a domain where a single wrong field can cost a person a day's worth of forward progress.

In ruby2's three locked pilot regions, the failure modes look different. **Long Beach** runs its own Continuum of Care under a single mayor with a consolidated political reception layer — a standalone-jurisdiction discovery problem. The **San Gabriel Valley** spans 31 cities and six producer-side language ecosystems (Spanish, Korean, Armenian, Chinese, Vietnamese, Tagalog) — multi-jurisdiction multilingual operation where a seeker's primary language determines which subset of orgs are even legible to them. **Antelope Valley** is geographically isolated over the mountains from the LA basin, has a thin service-infrastructure ecosystem, weather-extreme operating conditions, and a producer-side served-population that has more than doubled in recent years — underserved-region capability with its own accuracy-decay dynamics.

2.3 What gets measured today: the placement-throughput layer

LA County's accountability architecture is, as of December 2025, organized around **112 performance measures** approved by the Executive Committee for Regional Homeless Alignment (ECRHA) on December 12, 2025. The measures are organized across four categories (System Goals, System Component Performance Indicators, Contract Performance Indicators, Required Services) and three programmatic areas (Permanent Supportive Housing: 35 measures; Interim Housing: 33; Outreach: 44). They align with the three Measure A goals: increase the number of people moving from encampments into permanent housing; reduce the number of people with mental illness and substance use disorders who experience homelessness; increase the number of people permanently leaving homelessness.

This architecture is operationalized through a stack of new and established entities: the **Department of Homeless Services and Housing (HSH)** launched January 1, 2026; the **LA County Affordable Housing Solutions Agency (LACAHSA)** receives 40% of the half-cent Measure A sales tax revenue (~~$385M FY 2025-26 reference figure); the **LA County Development Authority (LACDA)** — established 1982 — administers Section 8 vouchers, public housing, and homeless-incentive programs at the unincorporated-areas layer~~ ($32.3M of the 40% slice). Together with the **Board of Supervisors** sitting as both BoS and as LACDA Commissioners, this stack governs over $1 billion in annual Measure A revenue.

The 112 measures count throughput through this stack: people moved, services delivered, beds filled, recovery outcomes verified.

2.4 What doesn't get measured: the seeker-side discovery layer

What these producer-side measures do *not* audit is whether the directory underneath them — the listings of organizations seekers are being routed *toward* — is accurate at the seeker's moment of need. Throughput counts depend on accurate routing. A seeker referred to a service

that does not exist, or that exists at a different address, or that has different intake hours, or that no longer serves their population, is not counted as a placement failure. The failure is invisible to the producer-side throughput layer.

ruby2's discovery-layer accuracy work sits seeker-side — oriented toward what reaches the homeless seeker at the moment of need — and **precedes** the producer-side measurement architecture those throughput counts depend on. The protocol is structurally compatible with the 112 measures (both architectures favor measurable, auditable, provenance-traced records) and operationally upstream of them.

2.5 A different operating system

There is an analytical lens that sharpens the diagnosis: the field is currently organized around a working assumption that **the referral plumbing works** — that the directories exist, that the orgs in them are real, that the schedules are current, that an outreach worker handing a phone number to a seeker is handing them a working line. ruby2 is built from the lived premise that **the plumbing does not work**, that the assumption of working plumbing produces measurement artifacts at the throughput layer that conceal upstream failures, and that the corrective is to make directory accuracy itself a measurable, auditable, methodologically-disciplined artifact.

This is one analytical lens, not the unifying lens. It frames the gap in current discourse; it does not exhaust the methodology.

---

3. The Thesis

3.1 Producer-side and seeker-side

Homeless-services directories have two sides distinguished by their orientation. **Producer-side** instruments — the institutional data systems run by County departments, federal CoC reporting, and referral hubs — are oriented toward measuring institutional throughput: who was served, who entered housing, who reached the next program. **Seeker-side** instruments are oriented toward the moment of discovery — toward what a homeless person, or the frontline advocate sitting with them, can find and use today. Most existing infrastructure is producer-side; ruby2 lives on the seeker side.

ruby2's discovery-layer work measures the listings themselves — are the addresses correct, are the hours current, are the eligibility criteria as stated, is the organization operational today. Producer-side throughput measurements depend on this layer; they cannot count a successful placement until the seeker has actually reached the service.

3.2 Directory accuracy as the precondition

The accountable services discourse — the language of throughput counts, target metrics, performance indicators — presupposes that the directory underneath is accurate. **Discovery-layer accuracy is the precondition for accountable services.** Without it, accountability is calibrated on a moving baseline: improving placement rates against an inaccurate directory tells you the rate is moving, but tells you less than you'd want about whether the system is working for actual seekers. *The seeker won't tell you the systems failed them; they will just lose more trust in them*

3.3 Why the problem is hard

Three structural difficulties make producer-side directory accuracy a research-grade problem, not an engineering exercise:

1. **Source-of-truth fragmentation.** Information about a single org lives across the org's own website (often outdated), its social media presence (often more current but harder to scrape), local news coverage (operational signals embedded in unrelated stories), government referral directories (often the stale ones), 211 records, mutual-aid lists, foundation grant reports, and outreach-worker shared spreadsheets. No single source is authoritative; no two sources agree on the full record.

2. **Standards cross-walk.** The homeless-services data standards landscape spans **HSDS (Human Services Data Specification, the standardized format for human-services records), Open Eligibility (the standardized taxonomy for service categories), and the InformUSA (formerly AIRS) taxonomy -- a longstanding referral-industry classification)**. Each carries different field assumptions, different category granularity, different update mechanisms. Producing records compatible with all three requires explicit cross-walking, which most directory projects skip.

3. **Information decay.** An org's hours change. Its address changes. Its program closes. Its director leaves. Its grant ends. Its language capacity shifts. Producer-side information decays on a roughly monthly timescale for active orgs and faster for orgs under operational stress. A directory built once and updated annually publishes systematically stale data within months.

3.4 Protocol, not directory

A subtle but load-bearing framing distinction: ruby2 is not a single directory. It is a **protocol** — a methodology, and a set of stages that implement it — for *producing* accurate directories. The directory is the artifact; the protocol is the thing. The same protocol can be run against a new geography (a new pilot region, a new county, a new metro area) and produce a directory specific to that region. The directories age; the protocol persists and improves.

---

4. The Methodology / Protocol

4.1 The credential combination

The protocol is grounded in 10+ years of homeless-services navigation as the source corpus — lived expertise mapped to operational patterns about how the discovery problem actually fails — combined with modern AI / LLM-infrastructure capability as the operational layer. The combination is the load-bearing claim: lived expertise alone produces accurate intuitions but not scalable systems; LLM-infrastructure alone produces scalable systems built on assumptions that don't survive contact with the actual seeker experience. The protocol is the formalization of the lived MO — what an experienced navigator does manually (loop query combinations → paginate → collect → repeat → collect links → return → reconcile → store → validate) — into an auditable, reproducible pipeline.

4.2 The pipeline architecture

The protocol's pipeline runs from a geography input through eight stages to a published artifact with full provenance lineage. Three cross-stage services operate across stages as needed. A filter "flywheel" makes the system more efficient and more precise on each subsequent run against the same geography.

The layers

Each stage carries a methodology label and a plain-English translation:

- `geography` *(input)* — the target region (a city, a county, a metro area). Geometry sourced from US Census TIGER/TIGERweb (incorporated places, Census Designated Places, county boundaries) and USGS GNIS (unincorporated communities not in TIGER's CDP inventory); stored on disk as placename rosters (JSON) and boundary polygons (GeoJSON). Future versions are designed to incorporate GTFS (General Transit Feed Specification) transit data for accessibility-aware retrieval.
- **Stage A —** `query_generation` — generates search queries from the cross-product of a vocabulary of service categories (shelter, food, medical, clothing, churches, etc.) and placename terms covering the region's geography
- **Stage B.1 —** `retrieve` — queries an independent search index (Brave Search, currently the only Western developer-scale independent index) for candidate organization URLs
- **Stage B.2 —** `judge` — three independent AI judges (a "trio") vote on whether each returned URL is a legitimate homeless-services-org listing; high agreement signals high confidence, disagreement signals an item for human review or downstream policy. **Disagreement-as-signal is a feature, not a failure mode**: the trio's disagreements have been empirically measured across seven test geographies and consistently land in methodologically interesting cases (news articles about orgs, .gov referral pages, churches with embedded service ministries)

- **Stage C — `fetch_render`** — for URLs that pass judgment, fetches and archives the actual web pages with full provenance: when fetched, what was captured (a frozen archive payload), whether the site's robots.txt was respected, what the upstream judgment was. Every published record is reproducible against this archive; the live web page can change after capture without invalidating the record
- **Stage D — `extract`** — pulls structured fields (addresses, phone numbers, hours, service descriptions, eligibility criteria) from the captured pages, mapping them to HSDS-compatible records
- **Stage E — `resolve_reconcile`** — identity resolution: recognizing that the Salvation Army's eight regional listings represent eight distinct local-chapter records (different EINs, different geographies, different operational states), not one duplicated org; reconciling conflicting field values across sources; surfacing ambiguities for human disposition
- **Stage F — `enrich_promote`** — source-class arbitration: news pages flow into the system as snippet-mining inputs (their operational signals are extracted, but the news pages themselves are never persisted as orgs); `.gov` referral directories are designated as last-resort sourcing (operational reasons, not ideological — government-curated directories empirically produce high-barrier, referral-only listings that org-vocabulary seeds surface anyway); churches are treated as candidates rather than classifications (every church in the catchment is a candidate, scored on signal strength, with miss-rate tracked as a methodology metric)
- **Stage G — `write_with_provenance`** — publishes each record with full provenance lineage back to the captured archive, the trio judgment record, and the extraction span (which sentence on which page sourced which field), so downstream consumers can audit any claim back to its source

The cross-stage services

Three services operate across multiple stages rather than as one stage:

1. **`arbitrator`** — mutative conflict-resolver consumed by Stage B.2 (trio disagreement on relevance / page_type / source_tier), Stage D (extraction conflicts within or across documents), Stage E (identity-resolution edges the heuristic cannot decide), and Stage F (source-class arbitration when sources conflict on a field). When ambiguity surfaces at any stage, the arbitrator can decide and modify records — producing an explicit rationale row in the audit log.
2. **`parser`** — mutative format-converter consumed by Stage C (`fetch_render`) for fetch-time PDFs and Stage F (`enrich_promote`) for enrichment-time PDFs. Converts opaque bytes (most commonly PDF) into structured text plus span anchors, so downstream extraction operates on text and per-field provenance carries source-span pointers back to the original format. The cross-stage placement makes the format-conversion concern reusable across stages and lets the underlying tool (OCR engine, structured-PDF library) be swapped without touching stage code.

3. `review_agent` — annotative; never mutates. The non-destructive counterpart to the arbitrator: surfaces concerns and observations for review — by deep-research agent or human — without altering records. Where the arbitrator decides, the review_agent flags.

The "flywheel"

Brave Search API uses filtering mechanisms called "Goggles" — programmatically-accessible custom re-ranking filters that let you control which websites appear at the top or bottom of your search results. The three available filters are `boost=[1-10]`, `downrank=[1-10]`, and `discard`. When searching a new region, ruby2's Goggles carry discards and downranks for known false-positives and noise (yellowpages.com, psychologytoday.com).

Once a region has been searched, the **flywheel** mechanism programmatically adjusts the Goggles for future runs against the same geography. Each valid org gets converted from a boost candidate to a discard rule on subsequent runs — telling the retrieval layer "we already have this, free the slot for new candidates." Each captured org reduces future retrieval load and opens retrieval slots for finding more orgs. The system gets more efficient and more precise as it operates against the same geography over time.

4.3 Failure detection as methodology core

Most directory projects ship without ever discovering their false-positive rate. ruby2 measures its own — across multiple metrics, at multiple stages — and treats the measurements as the methodology's central artifacts, as well as 'training data' for protocol performance. Gate-fired audits stop publication when data-quality thresholds are missed. Vendor-API evaluations have produced documented mixed and negative verdicts that constrained downstream stack decisions. Per-URL judgment volatility is measured and extensively documented across replicate trio runs and used to size the publishable strict-stable subset.

The pattern is **failure-detection-as-rigor**: a methodology rigorous enough to find its own failures, document them honestly, and design the next iteration around them. This is exceedingly rare in homeless-services work and is part of what positions ruby2 for research-grade collaboration rather than for tech-investor scaling.

4.4 The 100%-or-nothing accuracy bar

ruby2 does not soften its accuracy bar for v1 launch. There is no "good enough to ship" framing internally because there is no shipping path that survives the experiential reality of a wrong field. The methodology builds toward 100% on every published record — with visible freshness flags, provenance lineage, and the operational ability to mark a record `unverified` rather than `verified` when the underlying signal is weak. Records that cannot meet the bar are not published as verified; they are published as unverified-and-flagged-as-such, or they are not published at all.

---

5. Current State (May 2026)

Shipped end-to-end: Stages A through C — geo → query → retrieve → judge → fetch_render — operational across 7 test geographies (Salem-OR, Corvallis-OR, Bend-OR, Sacramento-CA, Fresno-CA, Spokane-WA, Flagstaff-AZ). Vocabulary at v3 (Brave-recommended Option E format, eight service categories including churches as probabilistic candidates), experience-informed 'discovery patterns' at v0.2.6. Stages A through C run against any new geography, with audit trails at each step. End-to-end pipeline replication awaits the planned and design-shaped stages.

Planned / pending: Stage D (`extract`) — the structured-field extraction layer that pulls addresses, hours, services from captured pages. Wired to the integration layer; full implementation pending.

Design-shaped: Stages E (`resolve_reconcile`), F (`enrich_promote`), G (`write_with_provenance`). Architecture committed, specifications documented, implementation pending.

Downstream of the pipeline: seeker-facing surface, three proposed pilot regions (Long Beach, San Gabriel Valley, Antelope Valley), and the methodology-adoption pathway at LA County.

---

6. Path Forward

6.1 Three pilot regions, three distinct demonstrations

Each pilot region is selected to demonstrate a different protocol-strength claim:

- **Long Beach** — standalone-jurisdiction discovery. Long Beach runs its own Continuum of Care under a single Mayor (Rex Richardson, who also chairs LACAHSA), with a consolidated political-reception layer. The pilot demonstrates that ruby2's discovery-layer accuracy work produces a coherent directory artifact across a single integrated jurisdiction with a defined geographic boundary.

- **San Gabriel Valley** — multi-jurisdiction multilingual operation. SGV spans 31 incorporated cities, runs a Council of Governments (SGVCOG) coordination layer, and contains six load-bearing producer-side language ecosystems (Spanish, Korean, Armenian, Chinese, Vietnamese, Tagalog) plus the AAPI-specific service infrastructure (CPAF, NAMI SGV, others). The pilot demonstrates that the protocol produces a coherent directory across a layered, multilingual, multi-jurisdictional surface where a seeker's primary language determines which subset of orgs are legible to them.

- **Antelope Valley** — underserved-region capability. AV sits over the San Gabriel Mountains from the LA basin, has roughly 6,700+ homeless people across 10 communities (Lancaster, Palmdale, and eight unincorporated communities), a producer-side ecosystem of approximately a half-dozen load-bearing organizations, and weather-extreme operating conditions (desert summers, occasional snow). The pilot demonstrates that the protocol works at the hardest end of the service-infrastructure spectrum — isolated, thin-ecosystem, weather-extreme — and that the methodology generalizes to comparable underserved-region surfaces nationwide.

6.2 The methodology-adoption pathway at LA County

ruby2's discovery-layer accuracy work is structurally positioned at the seeker-side discovery layer that LA County's Measure-A-funded producer-side throughput-accountability architecture depends on. The natural adoption flow is:

Data Subcommittee → Leadership Table for Regional Homeless Alignment (LTRHA) → ECRHA → Board of Supervisors.

The Data Subcommittee (under the County's Chief Data Office) provides baseline methodologies and metric specifications. The Leadership Table recommends performance targets. ECRHA — composed of two Supervisors (Barger chair, Horvath member), the LA Mayor seat, the LA City Housing and Homelessness Committee Chair, four sectoral mayoral seats (San Gabriel Valley, North County/SFV, Southeast, Southwest), and a California Governor appointee — sets the final adopted targets. The Board of Supervisors adopts them as County policy (most recently on March 25, 2025, for the FY 2025-26 cycle).

Adjacent to this pathway: the **LACAHSA Board's Goal 5 methodology** workstream (renter protection and producer-side data quality) is a distinct methodology-decision venue. Academic-collaboration entry points exist via researchers seated on the Data Subcommittee (notably at UCLA's California Policy Lab and USC's Homelessness Policy Research Institute).

6.3 The ask

ruby2 is seeking research-grade funding partners — foundations, academic-collaboration funders, impact investors, and methodology-adoption pathways at the County and State levels.

The work fits where:

- **Methodology rigor, lived-expert grounding, and visible-failure-honesty** are recognized as research-grade signals
- **Discovery-layer accuracy** is recognized as the precondition for placement-throughput accountability (the discourse currently dominated by Measure A's 112 performance measures)
- **Pilot-region demonstration** is the appropriate proof-shape (vs market-traction scaling)

A fiscal sponsor is in formation. In parallel, tech-vendor donation conversations are underway for ongoing API and infrastructure capacity (donation / free-credits / discounted-enterprise tier asks via sales channels at active production vendors).

Contact

Jay Dragon · ruby@rubyslist.org

Source documents and underlying canon:

- `FRAMING.md` — *positioning brief*
- `~/ruby2-mvp/docs/METHODOLOGY-DRAFT.md` — *canonical methodology articulation*
- `~/ruby2-mvp/knowledge/DISCOVERY-ENGINE-METHODOLOGY.md` — *seven-stage protocol specification (§4 stage definitions; §10 version history)*
- `~/ruby2-mvp/MVP-PROGRESS.md` — *current implementation state*
- `~/ruby2-outreach/la/` — *LA-ecosystem mapping (entity bundles for HSH, LACAHSA, LACDA, ECRHA, Measure A, HHAP; person profiles for the five-Supervisor BoS, the three pilot-region political-reception layers, the LA mayoral-race triangle; tech-vendor scouts for Jina, Abacus, Brave, TinyFish)*
- `~/ruby2-outreach/funder-onepager.md` — *one-page distillation of this summary*