

ULDA — Enterprise Architecture Review

Senior Enterprise Data Architect assessment for Fortune-500 production approval

Architecture Review Board — ULDA MDM/CDP Platform

2026-07-16

1. Executive Summary
2. Strengths (what already meets enterprise bar)
3. Architectural Gaps (each: Gap · Why · Solution · Objects · Benefit · Priority)
 - G1 — Disaster Recovery is stale and single-host — **CRITICAL**
 - G2 — Application connects as a superuser role — **CRITICAL**
 - G3 — Order / revenue attribution not integrated — **HIGH**
 - G4 — No hard validation / quarantine gate — **HIGH**
 - G5 — TCPA/consent not enforced at send-time — **HIGH**
 - G6 — No schema migration framework / version control — **HIGH**
 - G7 — Interaction idempotency on webhook retries — **MEDIUM**
 - G8 — Observability beyond a health view — **MEDIUM**
 - G9 — b2c / ad_leads not real-time — **MEDIUM**
 - G10 — Partition automation & master-scale — **MEDIUM/LOW**
 - G11 — pgvector mass-embed & lineage completeness — **LOW**
4. Architecture Maturity Scorecard
5. Production Readiness Assessment
6. Prioritized Implementation Roadmap
7. Additional Enterprise Best Practices to Adopt

1. Executive Summary

The **Unified Lead Dataset Architecture (ULDA)** is a genuinely strong MDM/CDP core: an immutable master identity, deterministic + probabilistic resolution, attribute-level survivorship, an event-sourced activity ledger, a live CDC pipeline, automatic first/last-channel tracking, and a governed 75-field Lead-360 serving layer — all delivered additively with a zero-downtime, fully-reversible track record and clean integrity invariants.

As a reviewer approving for **Fortune-500 production**, my verdict is **Conditional Approval**: the *data architecture* is enterprise-grade, but several **operational-hardening** gaps must close before unconditional sign-off — chiefly **disaster recovery**, **least-privilege for the application**, and **revenue/order integration**.

- **Architecture Maturity Score: 82 / 100**
- **Production Readiness: CONDITIONAL PASS** — approve after the 2 Critical items are remediated.

2. Strengths (what already meets enterprise bar)

AREA	EVIDENCE
Master identity	<code>master_lead_id</code> uuid — generated once, immutable, never reused; merges audited (<code>merged_into</code> , <code>ir_audit</code>)
Integrity	orphan/dup/ids-without-golden invariants = 0 ; 1.07M resolved identities
Event-driven CDC	dual-write trigger → <code>change_outbox</code> → drain worker; 0 backlog
Event ledger	<code>interaction</code> / <code>enrichment</code> partitioned (monthly), append-only
Attribution	automatic first-touch (<code>trg_crosswalk_first_touch</code>) + last-touch (<code>trg_capture_disposition</code>)
Serving	<code>v_lead_master_flat</code> (75 cols) — single-lead read 183 ms
Encryption/auth	TLS on, SCRAM-SHA-256 , backups GPG-encrypted
Governance	PII classification, metric catalog, <code>attribute_lineage</code> , LLM-safe view
Change safety	additive/reversible migrations; whole-stack rollback proven (0 non-ULDA deps)

3. Architectural Gaps (each: Gap · Why · Solution · Objects · Benefit · Priority)

G1 — Disaster Recovery is stale and single-host — CRITICAL

- **Gap.** Newest recovery point is **3 days old** (predates recent changes + production growth). `archive_mode = off` → **no point-in-time recovery**. Entire platform on one Contabo host; no HA replica.
- **Why.** A Fortune-500 will not approve a revenue-critical store with an RPO of “last full dump” and no failover. A host loss or corruption = multi-day data loss.
- **Solution.** (a) Immediate fresh **verified encrypted backup pushed off-host**; (b) nightly automated `pg_dump` + weekly `pg_basebackup`; (c) stand up a **streaming read replica** on a 2nd host to enable continuous WAL/PITR and failover.
- **Objects.** `pg_receivewal` slot (exists: `ulda_replica`), `archive_command` /replica for WAL, cron jobs, backup repo (`InternetSp/db-backups`).
- **Benefit.** RPO → minutes, RTO → hours; failover capability; audit-defensible recovery.
- **Priority.** ● **Critical.**

G2 — Application connects as a superuser role — CRITICAL

- **Gap.** `public.leads` is owned by `root`, a **superuser**; warroom connects as `root`. The app has unlimited privilege.
- **Why.** Least-privilege is a baseline enterprise/SOC-2 control. A compromised app or SQL-injection with a superuser connection = total database compromise.
- **Solution.** Create a dedicated `warroom_app` role with only `INSERT/UPDATE/SELECT` on the tables it needs; migrate the app connection string; revoke superuser from day-to-day use. Keep `postgres` for admin only.
- **Objects.** `CREATE ROLE warroom_app NOSUPERUSER; GRANT ...`; connection-string change + `pm2 restart warroom`.
- **Benefit.** Blast-radius containment; compliance; audit separation.
- **Priority.** ● **Critical.**

G3 — Order / revenue attribution not integrated — HIGH

- **Gap.** `order` / `opportunity` / `order_line` are **empty**; the real revenue source `accounting_db.paysheet_lines` (**65,841 rows**) is not linked to leads. `product_offered` /

`order_details` / `mrr` / `commission` stay blank.

- **Why.** Without closed-loop revenue attribution, the CDP can't measure ROI per source/campaign/agent — the core commercial question.
- **Solution.** Build an **order ingestion connector** from `accounting_db.paysheet_lines` → resolve to `master_lead_id` (email/phone/address) → populate `order` / `order_line` with `mrr`, `install_date`, `sub_dealer`, `commission`; add an attribution match table.
- **Objects.** `ulda_stage.ingest_order()`, FDW or ETL to `accounting_db`, `ulda_gov.order_attribution_match`, cron.
- **Benefit.** True source/campaign/agent ROI; forecasting; commission reconciliation (FiberVolt).
- **Priority.** 🟡 **High.**

G4 — No hard validation / quarantine gate — HIGH

- **Gap.** Bronze landing (`lead_inbox`) exists but isn't the active path; DQ issues are **flagged** (`dq_finding`) but not **rejected** before serving.
- **Why.** “Detect-and-suppress” lets bad data reach golden records (as the 183K state contamination proved). Enterprise pipelines reject/quarantine at ingest.
- **Solution.** Route all sources through `land_lead()` → `validate()` (email RFC, E.164, ZIP, USPS state) → pass to resolution or `quarantine`. Add CHECK constraints where safe.
- **Objects.** `ulda_stage.validate_inbox()`, `ulda_stage.lead_quarantine`, status flow on `lead_inbox`.
- **Benefit.** Golden records stay clean by construction; fewer downstream fixes.
- **Priority.** 🟡 **High.**

G5 — TCPA/consent not enforced at send-time — HIGH

- **Gap.** DNC / litigator / consent / timezone are now in ULDA (42 litigators, 8,072 DNC surfaced), but enforcement in the **dialer/SMS send path** is unverified.
- **Why.** Statutory per-violation fines; litigators bait callers. Data without a fail-closed gate is legal exposure.
- **Solution.** A **consent gate view** (`v_contactable`) the dialer/SMS must query before dialing/texting; fail-closed on DNC/litigator/quiet-hours (timezone).
- **Objects.** `ulda_serve.v_contactable` (excludes DNC/litigator, respects local time window), integration in dialer/SMS.
- **Benefit.** Compliance; litigation risk removed.
- **Priority.** 🟡 **High.**

G6 — No schema migration framework / version control — HIGH

- **Gap.** DDL applied ad-hoc via `s9` – `s15` files; **no version table** (Flyway/Sqitch), no enforced order or checksum.
- **Why.** Enterprise change management requires reproducible, versioned, auditable migrations across environments.
- **Solution.** Adopt **Sqitch** (or Flyway) with a `schema_migrations` table; register existing `s*.sql` as the baseline.
- **Objects.** `sqitch.plan`, `ulda.gov.schema_migrations`, CI hook.
- **Benefit.** Reproducibility; rollback discipline; environment parity.
- **Priority.** 🟡 **High.**

G7 — Interaction idempotency on webhook retries — MEDIUM

- **Gap.** First-touch is idempotent, but disposition/activity capture has **no dedup key** — a retried webhook could double-insert an interaction.
- **Why.** Duplicate events corrupt journey counts and attribution.
- **Solution.** Add a natural `event_key` (source+external_id+occurred_at hash) with a UNIQUE constraint; `ON CONFLICT DO NOTHING` in `ingest_interaction()`.
- **Objects.** `interaction.event_key text`, partial UNIQUE index.
- **Benefit.** Exactly-once activity; trustworthy journey metrics.
- **Priority.** 🟡 **Medium.**

G8 — Observability beyond a health view — MEDIUM

- **Gap.** `v_ulda_health` + `health_alerts()` exist, but no external monitoring, alerting, SLA dashboards, or query-performance tracking.
- **Why.** You can't operate what you can't see; on-call needs alerts, not manual SQL.
- **Solution.** Export metrics to **Prometheus/Grafana** (or a lightweight cron → Telegram alert); enable `pg_stat_statements`; drain-lag + backlog + invariants alerts.
- **Objects.** `pg_stat_statements`, exporter, dashboards, alert rules.
- **Benefit.** Proactive ops; SLA evidence.
- **Priority.** 🟡 **Medium.**

G9 — b2c / ad_leads not real-time — MEDIUM

- **Gap.** Only `leads` is real-time CDC; `b2c` / `ad_leads` are batch-reconcile.
- **Solution.** Extend the dual-write trigger pattern to `b2c_*` / `ad_leads`.
- **Objects.** triggers + drain mapping. **Priority.** 🟡 **Medium.**

G10 — Partition automation & master-scale — MEDIUM/LOW

- **Gap.** Event partitions created statically; master tables unpartitioned (fine ≤ few M).
- **Solution.** `pg_partman` for auto rollout + retention; revisit master partitioning at > 100 M.
- **Priority.** 🟡 Medium (partman) / 🟠 Low (master).

G11 — pgvector mass-embed & lineage completeness — LOW

- **Gap.** 200 sample embeddings; `attribute_lineage` partial.
- **Solution.** Full 1M embed on the replica; complete lineage for all 75 fields.
- **Priority.** 🟠 Low.

4. Architecture Maturity Scorecard

DIMENSION	SCORE
Data model / MDM design	95
Identity resolution & lifecycle	92
Event-driven pipeline & attribution	90
Serving / Lead-360 / MV	92
Data lineage & governance	80
Data quality & validation	78
Security / RBAC / encryption	70
Observability / monitoring	68
Scalability / partitioning / perf	85
HA / DR / backup	55
Integration breadth	80
Revenue/order attribution	55

DIMENSION	SCORE
Overall	82 / 100

5. Production Readiness Assessment

Verdict: CONDITIONAL PASS. The platform is architecturally sound and safe to *operate* today (it already runs live), but for **formal Fortune-500 production sign-off** two Critical controls must close first:

- 1. **G1 — Disaster Recovery** (fresh off-host backup now + PITR/replica path).
- 2. **G2 — Least-privilege application role** (stop running the app as superuser).

With G1 + G2 remediated, the platform clears the production bar; the remaining High/Medium items are fast-follows that do not require redesign.

6. Prioritized Implementation Roadmap

-  **Critical (before sign-off):** 1. Fresh verified off-host backup + nightly/weekly automation (G1). 2. Streaming read replica → PITR + failover (G1). 3. `warroom_app` least-privilege role (G2).
-  **High (next sprint):** 4. Order/revenue integration from `accounting_db.paysheet_lines` (G3). 5. Validation/quarantine gate (G4). 6. Consent gate `v_contactable` for dialer/SMS (G5). 7. Migration framework (Sqitch) (G6).
-  **Medium:** 8. Interaction `event_key` idempotency (G7). 9. Prometheus/Grafana + `pg_stat_statements` (G8). 10. Real-time `b2c / ad_leads` (G9); `pg_partman` (G10).
-  **Low:** full embeddings, lineage completeness, RLS for agent scoping.

7. Additional Enterprise Best Practices to Adopt

- **Environments & CI/CD:** a staging clone + automated migration tests before prod.
- **Data contracts:** per-source schema contracts enforced at Bronze.
- **RLS (row-level security):** agent-scoped visibility for least-privilege reads.
- **Automated restore drills:** monthly PITR restore test (currently blocked by disk → replica).
- **SLOs:** define drain-lag, freshness, and availability SLOs with alerting.
- **Secrets management:** move DB passwords to a vault (currently file-based).
- **Data retention policy:** formal retention/archival for partitioned events + DSAR erasure automation.

Closing note. ULDA's core is enterprise-grade MDM/CDP — the identity spine, event sourcing, and serving layer are done right and will scale without redesign. The gap to unconditional production approval is **operational hardening** (DR, least-privilege, revenue integration, validation) — not architecture. Close the two Critical items and this is a production-ready, scalable, enterprise platform.