WP-01 | Working paper | VERSION 0.1 | AUGUST 2026

Evaluating Long-Term Memory Isolation in Multi-Tenant Conversational Agents

Pablo Cardozo
Independent Applied AI Researcher
Buenos Aires, Argentina

Abstract

This working paper defines a reproducible evaluation protocol for conversational memory under tenant isolation. The system combines short-term thread state, vector memory, and graph-grounded documentary evidence. The available artifact creates two synthetic tenants, records distinct facts for each, probes recall, checks cross-tenant leakage, and measures latency. No quantitative result is claimed until a versioned run artifact is released.

Keywords: conversational memory; tenant isolation; GraphRAG; evaluation; latency

EVIDENCE STATUS	SOURCE REVISION
Evaluation protocol available; public result artifact pending	1470f8f17163

1. Research question

Can a multi-tenant conversational agent recover durable user facts without leaking facts across tenants, while remaining within an explicit latency target?

2. System under study

A WhatsApp-facing runtime built with FastAPI, LangGraph, a PostgreSQL checkpointer, Mem0 over pgvector, Neo4j GraphRAG, Convex job coordination, Gemini or Vertex generation, and Langfuse traces. The retrieval precedence documented in the implementation is graph evidence, then long-term memory, then thread state.

3. Method

1. Create two synthetic tenants with mutually exclusive facts for color, allergy, and employer.
2. Record three memory statements per tenant and issue six recall probes in isolated conversation threads.
3. Use required and forbidden substrings to score recall and detect cross-tenant leakage.
4. Report accuracy, average latency, p50, p95, maximum latency, and the ratio under the latency target.
5. Evaluate against declared targets of 0.86 accuracy and 700 ms p95 latency.

4. Current evidence

1. A checked-in benchmark runner computes deterministic lexical checks and latency percentiles.
2. A six-case JSON dataset separates recall success from tenant-leakage failure.
3. Unit tests cover expected answers, leakage detection, missing facts, and percentile calculation.
4. The application README documents the deployed evaluation command and the production architecture under test.

5. Limitations

1. The dataset contains only six synthetic cases and cannot support a general quality claim.

2. Lexical matching does not measure semantic correctness beyond expected terms.
3. The source repository is private and the public result JSON has not yet been released.
4. No baseline comparison or ablation isolates the contribution of thread state, vector memory, and graph retrieval.

6. Next experiment

1. Expand the benchmark with paraphrases, contradictions, delayed recall, deletions, and adversarial leakage probes.
2. Run ablations for thread-only, vector-only, graph-only, and combined retrieval.
3. Release a redacted dataset, environment manifest, raw results, and cost measurements.
4. Add human semantic review alongside deterministic checks.

7. Sources and reproducibility

1. Private implementation repository, revision recorded above

Source revision: 1470f8f17163cd87538c5e12ae621658bd46afe3

Suggested citation

Cardozo, Pablo. "Evaluating Long-Term Memory Isolation in Multi-Tenant Conversational Agents." Working paper, version 0.1, 2026. https://pablo.cardozo.com.ar/research/long-term-memory-isolation.

Independent working document. Not peer reviewed.