

Craig's AgentOS Architecture Review

Ten questions | 30–45 minute gut-check | August 10, 2026

We would rather hear “this is wrong” than “looks fine.” Brief answers, a voice memo, or comments on the PDF are all useful. Skip anything outside your expertise. There is no deadline pressure; the 30–45 minutes is a suggested time box, not an expectation to answer everything.

1. Authoritative store: Given the current scale, what would you use as the authoritative store for reviewed knowledge—files, Git, a database, an event log, or a hybrid—and why?
2. Boundary between files and structured data: What belongs in human-readable files versus structured/relational storage, and what measurable signal would tell us to move that boundary?
3. Review authority: One coordinating process currently holds sole agent promotion authority for shared long-term memory. Is that the right control at this scale? What would change your answer?
4. Independent reviewer path: What access and submission model would you use for a more-independent reviewing process—bounded files, Git pull requests, signed bundles, database views, or something else?
5. Worker memory: Should bounded worker processes maintain durable local memory, or should they return evidence and results to one reviewed store? Where would you draw that line?
6. Versioning and recovery: What is the minimum viable versioning, backup, restore, and rollback design you would require before trusting the system with important internal knowledge? Is Git the right tool, overkill, or insufficient?
7. Bad-memory response: If an incorrect or malicious entry reached shared memory, what controls would let us detect it, trace its origin and effects, revoke it, and recover cleanly?
8. Retrieval thresholds: Which observed query failures, data volumes, or latency needs would justify keyword/vector RAG or a graph layer? What tests or failure modes would disqualify them?
9. Operations and failure: What is the largest single point of failure or maintenance burden in this design, and what is the cheapest practical fix?
10. Sequence: If this were yours, what would you build in the next 30, 60, and 90 days—and what would you deliberately not build yet?

Please finish with

- Your recommended architecture in a sketch or short description.
- The top three risks, ranked.
- One assumption teams usually get wrong when building a system like this.
- Any important question we failed to ask.