PERSONAL SYSTEMS / IMPLEMENTATION CASE STUDY

Spark: Judgement, Evidence and Delivery

Sol leads. Two Sparks serve one analyst.

A hosted assistant, one distributed DeepSeek service and a durable workflow controller. How experience changed the division of work, where inference and tools execute, and what establishes a completed result.

Francesco Di Costanzo

15 October 2026 / Version 1.2

First published 12 October 2026. Revised implementation snapshot: 15 October 2026.

Companion to *My AI Assistant Needed a Better Operating Model*.
Read the article Open the visual walkthrough

Initial delivery evidence, not a long-term performance claim.
Examples are synthetic; operational identities and private source locators are omitted.

1 Scope, evolution and evidence

This paper describes the author's personal assistant as implemented on 15 October 2026. It distinguishes operating experience, inspected mechanisms, recorded acceptance and proposed evaluation. The revision retains the argument about responsibility for delivery and records subscription-first hosted access and the distinct initiators of research and memory work.

Period	Arrangement	Lesson or hypothesis
From 10 August	Distributed DeepSeek at the centre of the local assistant	Owning capacity did not ensure timely, accepted work.
12 October	Hosted Sol leads two independent Qwen services	Useful newspaper delivery coexisted with uneven analysis and repair effort.
13 October	Sol leads one distributed DeepSeek service across both Sparks	A stronger analyst may reduce rework; the shared capacity and failure boundary also grow.

On 14 October, subscription access became the primary Sol route. The owner retained this operating model after reviewing its roles on 15 October. The earlier distributed runtime was retired on 11 October and restored on 13 October under Sol's leadership. This is not a return to the original assignment of responsibility. Main conversation and scheduled direction remain hosted Sol. Qwen is historical in the current design.

Some weak results reflected model limitations. Others involved unusable inputs, unavailable tools, quota presentation, provider errors, delivery routing or controller defects. The model change cannot isolate these causes. The relevant outcome is accepted work and owner effort saved after supervision, hosted review and maintenance. Agent-evaluation research motivates separating correctness, cost and reliability; it does not independently validate this installation. [1, 2, 3]

1.1 What each claim means

Executable control Inspected code admits, rejects, records or limits a supported operation.

Behavioural contract Instructions specify source checking, disclosure or task completion; the model can still fail to comply.

Operator policy The owner controls deployment, recovery and resumption of held work.

Evaluation hypothesis Better overall quality, less Sol rework and lower owner effort remain outcomes to measure.

The author inspected tracked, sanitised implementation code and dated commissioning records. No live assistant job, external action or benchmark was repeated for this editorial update. The evidence appendix names the record categories, dates, methods and limitations without publishing private repository addresses, internal paths, task identities or source contents. Public references support component semantics and general engineering principles, not the private acceptance results.

Local contribution means findings and evidence retained in the final deliverable. A successful model turn, a large token count and a confirmed delivery receipt measure different things. None alone proves useful analysis.

2 Topology: one gateway, one shared local model

OpenClaw 2026.10.3 runs the gateway, native sessions and tools on Spark 1. Hosted inference uses exactly `openai/gpt-6.1-sol` through the ChatGPT subscription's supported authentication route. OpenRouter supplies the only automatic Sol backup for eligible calls. Main Sol handles conversation, correspondence, calendar, vision and final answers. A scheduled director uses the same model in separate task sessions. Operational role separation does not create a second assistant personality or a shared raw transcript. [4]

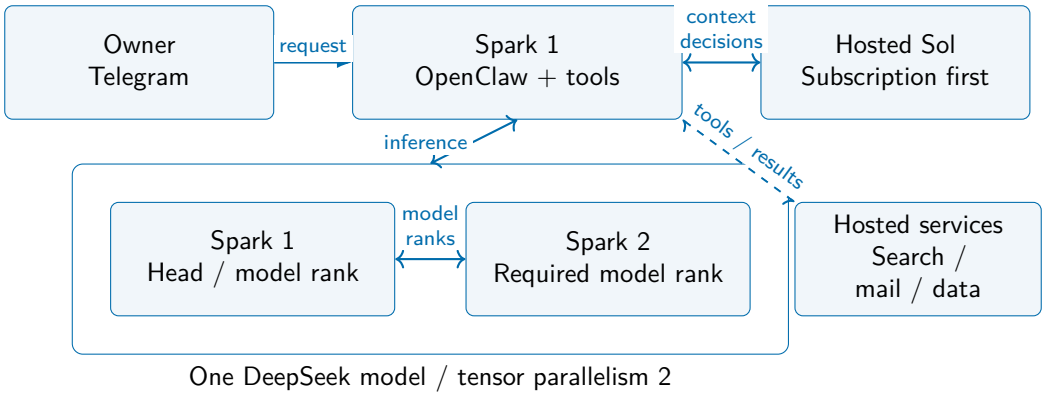


Figure 1. Logical boundaries, not access routes or physical cabling. Both model ranks are required. OpenClaw tool execution remains on Spark 1; hosted reasoning and service operations cross distinct boundaries.

The local service runs `deepseek-ai/DeepSeek-V4-Flash-0731`, served as `deepseek-v4-flash-0731`, with vLLM tensor parallelism of two and a configured 262,144-token ceiling. It follows a pinned two-Spark recipe; model, runtime and compatibility changes are qualified together. Public upstream documentation describes the model and recipe, while local acceptance qualifies this deployment. [5, 6, 7, 8]

Analyst and background roles, automatic memory jobs and terminal clients consume this one pool. Spark 2 is no longer an independent background inference service. There is no local-model or gateway failover. Eligible hosted-route recovery uses the same Sol model. Losing Spark 2 interrupts local inference; losing Spark 1 also affects the assistant gateway and its tools. Hosted main Sol can still answer during a local-model outage only while the gateway, required tools, connectivity and hosted provider remain available.

Today is a separate authenticated Next.js/PWA, deployed on Vercel with hosted Neon Postgres storage. Validated content updates use a scoped publication interface. Code deployment is a separate reviewed operation. Relevant context also reaches hosted Sol, search and mail services. The complete assistant is hybrid; no local-only or universal zero-retention claim is made. [9]

3 Roles, inputs and execution locations

Role	Inference	Work and tool boundary
Main assistant	Hosted Sol	Conversation, vision, email/calendar, input preparation and final answers; tools on Spark 1.
Scheduled director	Same hosted Sol	Brief, review, feedback, finish and approval; research and artifact tools, no shell, browser or direct publication.
Analyst / background	Shared DeepSeek	Bounded research and prepared-document analysis; research, file and memory tools on Spark 1; no shell, browser, email or calendar.
Terminal worker	Same shared DeepSeek	Separately authorised client task; tools and files on the Mac or VPS under that client's permissions.
Supervised Codex	Hosted model	Author's main coding environment; independent inspection of delegated artifacts.

The managing-director analogy includes briefing, source verification, feedback, hands-on completion and responsibility for the deadline. Sol is not a binary approval gate. An analyst may return useful partial evidence even when its stage reaches a limit; the director must inspect what exists rather than confuse runtime failure with empty research.

3.1 A usable source packet

Main Sol prepares binary inputs with its available tools. For a PDF, it supplies readable extracted text with explicit page references, checks table and symbol fidelity, and identifies omitted or image-only material. Restricted analysts and the scheduled director cannot perform shell-based conversion. A large context window cannot repair missing input or missing capability.

A synthetic assignment might ask: "Check whether the totals on page 4 agree with the component values on page 3. Return the calculation, page references and any extraction limitation." The full source remains available to main Sol for checking. A focused excerpt reduces unnecessary reading without pretending that the analyst verified the whole document.

The behavioural return contract is *completed*, *partial* or *blocked*, with findings, source/page references, checks actually performed and unresolved questions. Missing or unusable input should produce a blocked result promptly. Sol can correct the input, request a targeted revision or finish directly. It should not repeat an impossible assignment or broaden tool permissions merely to force completion.

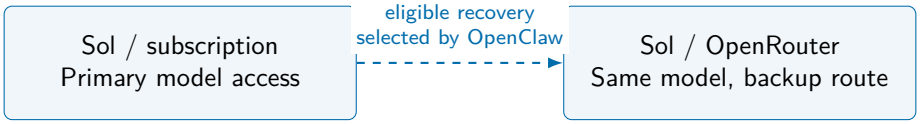
3.2 A separate terminal trace

For a synthetic CSV-parser task, the owner or supervised Codex defines scope and acceptance. A client-side worker may appear in a Herdr terminal. It sends inference context to the same two-node DeepSeek pool, interprets supported tool requests locally and writes the candidate artifact on the client. The caller inspects the result and meaningful checks before acceptance; this may be hosted Codex, but OpenClaw Sol is not automatically involved.

Herdr arranges visible agents; it is not the model or OpenClaw scheduler. Client-side tools differ from the shell-less OpenClaw analyst toolset. Credentials and files remain on the client, while selected prompt and tool-result text travels for inference. This boundary does not mean that no information leaves the client.

4 Hosted access and separate contexts

The primary route is `openai/gpt-6.1-sol` using supported ChatGPT subscription authentication inside OpenClaw. The only configured automatic backup is `openrouter/openai/gpt-6.1-sol`. Model access and the agent harness are different choices: OpenClaw still owns sessions, tools and orchestration; the Codex CLI does not replace it. Main and scheduled roles use the same model with different authority and selected context. [18]



Routing schematic. The dashed arrow means eligible recovery, not review by another agent. Cancellation stops work. Uncertain external writes require reconciliation, not automatic repetition. If both hosted routes fail, DeepSeek does not become Sol.

Subscription usage shares the owner's Codex allowance. Available credits can extend eligible use, while OpenRouter recovery, search and other services remain separately metered. Subscription-first access uses an existing allowance; it does not establish unlimited use, a zero marginal resource cost or measured net savings. [19]

4.1 Context belongs to a route and a session

Installed route / service	Context ceiling	Qualification at 15 October
Subscription Sol	272,000	Effective context in installed OpenClaw 2026.10.3.
OpenRouter Sol	1,050,000	Current catalog value used by this installation.
Local DeepSeek	262,144	Configured serving ceiling of the shared two-node model.

Both Sol routes retain a configured 16,384-token output ceiling. These are version-specific installation observations, not universal provider defaults or normal request sizes. Main conversation, correspondence threads and scheduled tasks retain scoped sessions. Sol and DeepSeek exchange selected briefs, source extracts and returned artifacts; their contexts are not combined into one window or shared mind.

4.2 What the route checks establish

The 14 October record verifies native subscription turns, image and tool paths, and a subscription–OpenRouter–subscription continuity check. Simulated provider failures exercised eligible fallback, cancellation, exhausted routes and completed-action guards. These are bounded checks, not observed exhaustion of a real subscription or proof that every ambiguous external action is safe. A commissioned newspaper publication confirmed six additions across four beats; partial local work still needed Sol's finishing. No additional social drafts were created in that day's review-only drafting checks.

5 Scheduling, admission and finite work

Native OpenClaw command schedules invoke a deterministic durable controller. Reasoning stages remain native agent sessions; the controller does not become a second agent harness or directly call model-provider APIs. It checks returned model attribution and journals task and stage state independently of the final prose.

Today has four edition slots, at 00:00, 06:00, 12:00 and 18:00 Europe/London. Preparation starts twenty minutes before each slot. The controller uses the named time zone and skips outside authorised preparation windows rather than replaying a missed edition. Approval can finish early; normal publication waits for the slot. This is scheduling logic, not a guaranteed delivery SLA.

```
claim workflow lease and a fresh delivery slot
check both required model ranks and serving readiness
if unavailable: stop before paid planning; retain last-good content
ask Sol for focused editorial questions
for each permitted round (initial + at most two revisions):
    run focused local assignments sequentially when admitted
    preserve partial findings and original source returns
    reserve time and research allowance for Sol
    ask Sol to review, revise, finish or withhold
    if approved: validate exact content; publish; record receipt
on uncertainty: preserve evidence and stop for inspection
```

Figure 2. Explanatory control flow. An unavailable cluster does not trigger automatic hosted replacement of the scheduled research workload. Sol's later recovery of gaps in an admitted run is a different operation.

A shared cluster lease serialises the controller's local assignments. Admission also checks model activity and defers discretionary work if capacity is busy or unavailable. Terminal requests and other local activity can still arrive after a check. This is cooperative admission, not global reservation, preemption or unlimited parallelism.

Today's initial research is split into four focused subject assignments. Workers run sequentially, with stage budgets that reserve approximately eight minutes for Sol's review, correction and delivery work. A revision consumes remaining allowance; it does not reset the task. Typefully uses two focused sequential assignments and retains an allowance for Sol to verify and supplement them.

5.1 Why head health was insufficient

A controlled worker outage left the head's health endpoint successful. The corrected readiness path checks the required remote rank as well as model identity, configured context and serving health before paid scheduled planning. A running worker check is still not a proof against a failure immediately after admission. The lesson is to test the dependency the work actually needs, not only the process that answers HTTP.

Optional local memory recall has separate admission and failure handling. It skips a busy or unavailable pool and retains a short time bound. A failed recall cannot increment or hold the main conversation circuit. Setup and recovery can add overhead, so its configured timeout is not a hard bound on the complete user-facing turn.

6 Evidence and budgets across the task

The task-control layer attaches counters to native session identity. It records tool returns, retrieval times, errors and truncation information. Original-evidence review means access to retained returns and usable source material as well as analyst summaries. It does not imply possession of every full original document. Sol must qualify claims when source access is incomplete.

X is a discovery source. Perplexity, web retrieval and original publications can support or contradict the candidate. A source registry can establish identity; it cannot establish the truth or date of each post. The reviewer must distinguish original timestamps from reply dates and later coverage.

Scope	Executable control	Interpretation
Research task	Aggregate ceilings: 4 X, 4 Perplexity, 14 fetch invocations	Applies across participating stages and revisions; not provider billing totals.
Worker assignment	Smaller individual allowances plus a shared worker ceiling	Compact input states what this worker may use; Sol retains research capacity.
Reasoning stage	Time and tool limits; durable halt and native cancellation	Useful checkpoints can survive an unfinished stage.
Revision	At most two targeted additional rounds	Feedback should identify a material gap, not restart discovery.
Plan / special finish / correction	Zero new research allowance	Enforced by stage policy, not only an instruction.

Calls consume allowance before execution, including unsuccessful calls. Provider-specific holds and remaining task allowances continue across roles. These controls bound supported invocations; they are neither a hard monetary cap nor a complete count of provider-internal retries or billable operations. Hardware, electricity, maintenance, hosted inference and research remain costs.

6.1 What the quota failure taught

An initial Typefully test exposed a mismatch between aggregate context and worker-specific allowance. Both local artifacts were blocked; Sol recovered one draft. That demonstrated the write path, not useful local analysis. The correction gave each worker a compact packet with only its own allowance and relevant source identity context.

In the corrected follow-up, two assignments completed in approximately 85 and 111 seconds without tool errors. Sol rejected an out-of-window source, checked originals, edited and added material. Three more unscheduled drafts completed the daily four. This is evidence that the corrected workflow used local findings; it is not evidence that changing the model alone caused the improvement.

6.2 Finishing without another research loop

A review stopped by its task budget has one bounded evidence-only finishing opportunity. A failed hosted provider call does not trigger that special path. Validation can request one bounded correction of the artifact using existing evidence. New research is mechanically disallowed in these selected stages. Semantic truth, relevance and appropriate narrowing remain Sol's responsibility.

7 Approval, publication and durable receipts

Sol approves a specific artifact with source assessments. The controller records its saved-file hash and adds deterministic publication metadata. A trusted helper validates and submits the payload through a scoped authenticated interface. Approval in a transcript is not publication, and a receipt is not a proof of factual correctness.

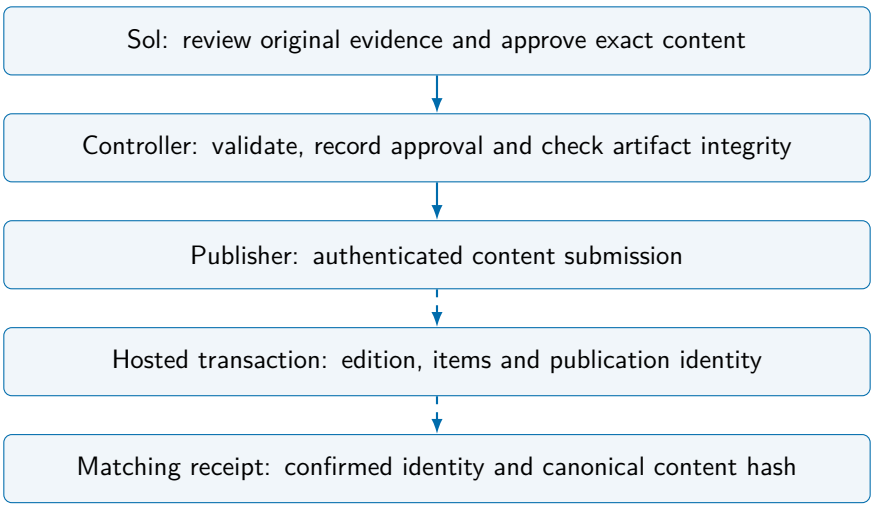


Figure 3. Content delivery. A lost response branches to receipt lookup; unresolved delivery holds the workflow. Application-code deployment follows a separate reviewed path.

The saved-file hash binds local approval to bytes. The publisher and server use a canonical payload hash after schema parsing. These hashes serve different purposes. The approval record is an audit binding in a trusted workflow, not a detached Sol signature that authenticates a remote action.

The database transaction checks publication identity. The same identity and canonical content return the existing receipt; changed content under that identity conflicts. Edition content and versioned item relationships are committed together. Unchanged or partial sections can carry last-good items with their original dates. Idempotency and transactions support this boundary. [10, 11]

If a response is lost, the helper queries the existing receipt and checks the canonical hash. Otherwise the outcome remains unresolved. A remote write may survive an interruption before local bookkeeping; absence of a local completion record is not permission to replay. Failed attempts preserve available evidence and delivery state beside last-good content. Operational reporting can itself be stale without invalidating a confirmed content receipt.

Typefully follows a distinct draft ledger. The whole candidate batch is validated before the first write, with one bounded correction if appropriate. Creation is followed by readback and durable receipts. The intended output is unscheduled drafts for review. No automatic social posting, scheduling or historical deletion is part of this workflow.

8 Correspondence, continuity and authority

Main Sol owns owner conversation and AgentMail email/calendar. Telegram keeps its conversation session; email keeps thread-specific sessions. A private correspondence task store carries relevant instructions, context, authority, intentions and receipts across those sessions. It neither merges raw transcripts nor sends messages itself.

```
Synthetic task: arrange a museum visit
Owner instruction: offer Tuesday or Wednesday
Private context: keep another day unavailable
Authority: owner's request; preserve any draft-only constraint
Next action: read the latest thread before composing a reply
Outcome: record the provider result; reconcile uncertainty
```

Revision checks and atomic replacement reject supported stale writes. Sol must retrieve the record, inspect the thread and choose outward wording. A private reason can inform a reply without appearing in it. Retrieval and disclosure remain fallible; sending tools do not override draft-only instructions.

In this synthetic trace, an owner request creates the task; an email-thread turn retrieves it and offers authorised times. Replies update the same task. Calendar actions require appropriate authority; private updates return to the owner. Bounces and uncertain sends require reconciliation.

8.1 OpenClaw starts memory work directly

OpenClaw starts automatic recall, pre-compaction memory saving and nightly dreaming on the shared local DeepSeek service. Sol does not routinely assign or review these jobs. Sol's conversation compaction and DeepSeek's preceding memory-saving step have different outputs: a smaller conversation context versus durable notes. Optional recall skips busy or unavailable local capacity and has an isolated failure circuit with the retained 15-second bound; this is not a bound on the whole user turn.

Nightly dreaming processes memory candidates. Durable promotion is distinct from narrative diary output. The repaired 15 October sweep preserved existing memory and found no eligible promotions. Two background diary outputs timed out; a successful sweep alone did not establish complete diary delivery. Those failures do not themselves establish loss of source memory. Neither saved notes nor dreaming guarantees accurate recall. Public documentation explains the promotion/diary distinction; these outcomes concern the installed version. [20]

8.2 Supported controls are not universal isolation

Analyst and director tools restrict ordinary actions; main Sol has broader authority. Runtime guards enforce supported admission and scope failures. Behavioural guidance still governs source interpretation, audience selection and the owner's intent. Sol is trusted within the operated host environment. Role instructions are not credential isolation, and external text remains untrusted even when delivered by a permitted tool. OpenClaw's trust model and OWASP's prompt-injection and excessive-agency guidance remain relevant. [12, 13, 14]

Vision remains hosted Sol. Separate services include Grok X search, Perplexity web research, OpenAI transcription, embeddings and image generation, ElevenLabs speech and AgentMail. These are external tool services, not local models or automatically included subscription use. Second-brain provides knowledge on request, not a routinely rewritten local archive. No purchasing or sensitive financial-record workflow is introduced by this migration.

9 State, holds and maintenance

State class	Authority and purpose
Native assistant state	Sessions, reminders, schedules and curated memory on Spark 1.
Correspondence tasks	Private instructions, relevant context, intended effects and receipts.
Workflow journal	Stage identities, quotas, source returns, approval and holds.
Hosted content	Newspaper editions, item history and publication receipts.
Owner application state	Reading state, pins, dismissals and preferences, distinct from editions.
Operational projections	Schedule/reminder views, health observations and attempted delivery status.

Native reminders and schedules remain authoritative. The dashboard receives projections from those records; immediate updates are backed by periodic reconciliation. Health is collected hourly without editorial inference. Weather has an independent retrieval path. Audio services remain quick links, not autonomous listening or transcription jobs.

The durable controller preserves task and stage identities to prevent silent replay. Runtime circuits separate conversation, correspondence, heartbeat and optional recall failures. Provider holds constrain further research. Workflow holds protect interrupted or uncertain delivery. These are related controls at different levels, not a single general retry counter.

Repeated failures, ambiguous writes and infrastructure faults require operator inspection. There is no blind retry chain, paid automatic infrastructure repair or automatic local takeover. Eligible same-Sol recovery through OpenRouter is distinct from starting another workflow or retrying an uncertain write. Clearing a hold does not itself repeat the task. Recovery checks what actually happened, including external receipts, before deciding what to resume.

9.1 Backups follow state, not just source code

Reviewed source changes and deployment history do not back up private assistant state. Recovery material separately covers runtime/configuration compatibility, native assistant state, workflow journal and artifacts, correspondence records, and hosted application data. Pinning and acceptance checks make upgrades reviewable; they do not make an old runtime permanently compatible.

Workflow database backups must include committed journal content and pass integrity checks. SQLite's backup API and write-ahead-log semantics explain why copying an apparent main file can be insufficient. Complete file revisions and database snapshots do not establish one globally atomic instant across every artifact and external service. Recovery must reconcile those boundaries. [15, 16]

Earlier commissioning found a restore reader treating Unicode separators inside text as record boundaries. Correcting the reader and checking restoration addressed a format defect without rewriting the source conversations. A valid database and a plausible export had not alone proved recoverability.

Retention preserves verified recovery sets and explicit holds before ageing routine evidence. Restoring a database cannot unsend correspondence or undo a remote draft. The current approach is a documented maintenance practice, not evidence of a full disaster-recovery exercise or immunity to correlated local loss. The public paper deliberately omits practical access routes and private backup locations.

10 Acceptance and what remains unknown

The following outcomes are reported in the 13 October commissioning record. This paper inspected those records and the corresponding mechanisms; it did not repeat the live checks. Narrow acceptance results must not be extrapolated into long-term performance claims. [17, 3]

Check	Observation	Limit
Long input	259,813 input tokens accepted inside the 262,144-token ceiling	Does not prove comprehension or retrieval across arbitrary long inputs.
Warm decode	Median approximately 75 tokens/s in the recorded benchmark	One shared model, not per machine; not normal task throughput.
Runtime paths	Tool calls, streaming, cancellation, worker outage/recovery and client delegation passed	Qualifies tested paths, not every future workload.
Worker outage	Head health stayed successful; readiness gained a required-worker check	No redundancy; either rank can interrupt local inference.
Commissioned edition	Six additions across four beats in about 12m30s	Finance completed; F1/AI partial at limits; Inter provider errors. Sol recovered gaps.
Corrected drafts	Two local assignments finished in about 85/111s; Sol edited and supplemented	Handoff changed too; earlier blocked local work required Sol recovery.
Frozen packet	Qwen about 46s; DeepSeek about 175s; both made attribution/date errors	Guided regression, not a blind A/B benchmark. Sol caught a DeepSeek error.
Prepared document	Useful arithmetic findings; length constraint missed	Bounded task, not comprehensive document verification.

Earlier correspondence commissioning included synthetic cross-session checks and bounded provider-action checks. Those remain limited evidence of supported paths, not proof that future retrieval, recipients or outward wording will be correct. Current read-only Mac/VPS and visible-terminal delegation checks supersede the earlier incomplete terminal acceptance description.

10.1 Measure contribution and intervention

The next evaluation should trace findings from assignment to retained final evidence, record Sol's corrections and additional research, and count owner interventions and minutes. It should distinguish confirmed on-time delivery, valid partial delivery, withheld work, factual errors and unsupported success claims. Available model and research costs should be recorded without pretending missing charges are zero.

A simpler hosted-only baseline deserves comparable tasks and source access. Better end-to-end quality and less Sol rework are hypotheses with initial evidence. No measured overall cost reduction, speed improvement, sustained reliability or net time saving is claimed. Local capacity earns its complexity only when accepted outcomes cover the effort of supervision and maintenance.

A Dated evidence map and reading guide

This is a public provenance summary, not a directory of operational access. Internal source paths, repository identities, private run identifiers and underlying personal material are omitted. The categories below make clear what was inspected and which evidence could support a claim. The author's records are implementation evidence, not independent validation.

Date / category	Inspected basis	Contribution and limitation
10 August, history	Commissioning chronology	Initial deployment date; no claim that today's orchestration ran for two months.
11–12 October, history	Migration, controller, correspondence and backup records	Explains prior Qwen design and model-independent corrections; superseded routing is historical.
13 October, local runtime	Setup and operations records; pinned runtime recipe	Local model, context and two required ranks; hosted routing was revised on 14 October.
13 October, controller	Scheduling, stage, worker, readiness, approval and delivery code; task contracts	Sequential assignments, shared lease, reserves, evidence-only stages, validation and receipt boundaries.
13 October, roles	Main, director, delegation and analyst instructions; runtime guard implementation	Tool expectations, prepared inputs, audience continuity and separate recall circuit; instructions do not prove compliance.
13 October, acceptance	Runtime and outage checks; newspaper and draft receipts; guided regressions; recall experiment	Reported observations and corrections in Section 10; not a controlled longitudinal comparison.
Publication continuity	Inspected publisher/schema, transaction, receipt and state design from the preceding package	Content/code separation and idempotent delivery remain; no dashboard architecture migration is claimed.

The 14 October subscription commissioning record establishes the primary/backup routing and bounded acceptance described above. The 15 October accepted role map and dreaming-repair record establish the current role boundaries, direct memory initiation and observed diary limits. These later records supersede older hosted-access and memory claims.

Execution evidence supersedes the earlier migration plan. A plan establishes intent, code establishes a mechanism, and a recorded result establishes a bounded observation. The public references below establish neither access to the private records nor independent reproduction of their outcomes.

For a guided explanation, the HTML companion follows owner conversation, a newspaper edition, correspondence and client-side terminal work. Additional schematics below the flows show model roles, job initiators, delivery and recovery boundaries. All examples are synthetic. Its diagrams run offline and do not access the assistant, private storage or live status. Deliberately following public article or paper links is ordinary navigation, not an operational integration.

References

[1] Kapoor et al. *AI Agents That Matter*. 2024. https://arxiv.org/abs/2407.01502

[2] Rabanser et al. *Towards a Science of AI Agent Reliability*. 2026. https://arxiv.org/abs/2602.16666

[3] Anthropic. *Demystifying evals for AI agents*. https://www.anthropic.com/engineering/demystifying-evals-for-ai-agents

[4] OpenClaw. *Gateway architecture*. https://docs.openclaw.ai/concepts/architecture

[5] DeepSeek. *DeepSeek-V4-Flash-0731 model card*. https://huggingface.co/deepseek-ai/DeepSeek-V4-Flash-0731

[6] MiaAI Lab. *DeepSeek-v4-Flash-DSpark-2x-DGX-Spark*. https://github.com/MiaAI-Lab/DeepSeek-v4-Flash-DSpark-2x-DGX-Spark

[7] vLLM. *OpenAI-Compatible Server*. https://docs.vllm.ai/en/latest/serving/online_serving/openai_compatible_server/

[8] vLLM. *Parallelism and Scaling*. https://docs.vllm.ai/en/latest/serving/parallelism_scaling/

[9] OpenRouter. *Provider Logging*. https://openrouter.ai/docs/guides/privacy/provider-logging

[10] AWS. *Making retries safe with idempotent APIs*. https://aws.amazon.com/builders-library/making-retries-safe-with-idempotent-APIs/

[11] PostgreSQL. *Transactions*. https://www.postgresql.org/docs/current/tutorial-transactions.html

[12] OpenClaw. *Security*. https://docs.openclaw.ai/gateway/security

[13] OWASP. *LLM01:2025 Prompt Injection*. https://genai.owasp.org/llmrisk/llm01-prompt-injection/

[14] OWASP. *LLM06:2025 Excessive Agency*. https://genai.owasp.org/llmrisk/llm062025-excessive-agency/

[15] SQLite. *Write-Ahead Logging*. https://sqlite.org/wal.html

[16] SQLite. *SQLite Backup API*. https://sqlite.org/backup.html

[17] NIST. *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*. 2024. https://doi.org/10.6028/NIST.AI.600-1

[18] OpenClaw. *OpenAI authentication*. https://docs.openclaw.ai/providers/openai/authentication

[19] OpenAI. *Pricing*. https://learn.chatgpt.com/docs/pricing

[20] OpenClaw. *Dreaming*. https://docs.openclaw.ai/concepts/dreaming