NEXT MOCA

NEXT MOCA TECHNICAL PAPER

Needlepath

Selective State Compression for Production Agentic AI Systems

Authored by Swanand Rao, Co-founder, Next Moca | June 2026

Executive thesis	Needlepath turns context management into an infrastructure primitive: select the right state, preserve grounding, fail open when needed, and measure the operating profile of every high-volume agent workflow.

27.38% **final-path input token reduction** Across the full 100-task benchmark	**83/100** **usefulness preserved or improved** Equal-or-better than baseline	**24/100** **optimizer fail-open cases** Baseline context used when safer
35.88% **input reduction on optimized rows** Where Needlepath safely optimized	**0** **silent unsafe compressions** Benchmark target: fallback, not hidden risk	**10 / 18** **agent families / tool classes** CRM, calendar, RAG, SQL, support, docs, and more

Audience: CTOs, VPs of Engineering, AI platform leaders, enterprise architects, and applied researchers evaluating production agent infrastructure.

What this edition strengthens

- Sharper executive decision framing for senior technical readers.
- More explicit architecture, state-record contract, scoring logic, and safety invariants.
- Clearer benchmark interpretation with production-conservative accounting.
- Enterprise rollout path with validation gates, telemetry, governance, and roadmap.

1. Executive Decision Summary

Production agents do not fail only because they lack context. They also fail because they carry too much of the wrong context: stale chat history, distractor RAG chunks, unused tool schemas, redundant memories, workflow traces, attached file metadata, and partial results. Shipping all of it to the model increases cost, latency, prompt-surface risk, and the probability that the model attends to the wrong evidence.

Needlepath is a selective state compression layer for production agentic systems. It sits between the agent runtime and the model call, evaluates the next action, scores candidate state records against the task, preserves grounding-critical evidence, and returns either an optimized context package or a safe baseline pass-through decision.

Core principle	Wrong optimization is worse than no optimization. Needlepath should reduce context only when it can preserve correctness, grounding, policy boundaries, and output completeness.

Question for technical leadership	Needlepath answer
What it is	A model-call context optimizer that turns sprawling state graphs into smaller, task-fit context packages.
Why it matters	At scale, context is an infrastructure cost center and a reliability surface. Better state selection can lower spend and latency while reducing distractors.
What the benchmark shows	A production-conservative 27.38% final-path input-token reduction across 100 tasks, with 83/100 task usefulness equal to or better than baseline and 24/100 fail-open cases.
What to validate next	Replay real customer traces, track savings/latency/fallback/quality by workflow, and promote only workflows that clear governance gates.

1.1 What the Paper Claims - and What It Does Not Claim

Claims supported by the benchmark

- Input context can be reduced meaningfully without forcing optimization on every task.
- Fail-open behavior is essential to production truthfulness of the savings number.
- Quality preservation must be measured against baseline, not inferred from token savings.

Claims that require customer validation

- Savings by a specific tenant, agent family, or workflow.
- Latency and total cost impact under a given model gateway and cache profile.
- Quality under real enterprise data, permissions, artifacts, and write actions.

2. Compare and Contrast: The Context Optimization Stack

Needlepath complements token-level and provider-level optimizations such as **TOON**, context caching, prompt compression, and provider-side caching. Those techniques mostly optimize **how selected context is represented, reused, or priced**. Needlepath operates one layer earlier: **which context should be included at all** for a given agent run.

Optimization layer	Primary question	Representative techniques	How Needlepath fits
State selection and orchestration	Which memory, retrieval chunks, prior outputs, tool state, and runtime metadata belong in this run?	Needlepath	Upstream decision layer. Selects high-signal state or fails open to baseline when grounding, policy, or completeness risk is detected.
Representation and structure	How can retained context be encoded more compactly once selected?	TOON and compact structured formats	Needlepath reduces the payload first; compact formats then make the remaining structured context denser.
Reuse, latency, and pricing	Which stable context can be reused or priced more efficiently across calls?	Context caching and provider-side caching	Cleaner, more predictable retained context increases the value of caching mechanisms.
Verbosity reduction	How can selected content be shortened without removing required evidence?	Prompt compression and summarization	Compression is safer after state has been selected and grounding-critical records are protected.

Stack logic	Used together, the stack becomes stronger and easier to govern.
1	**Needlepath** decides what matters.
2	**TOON** makes structured context more compact.
3	**Context** caching makes stable repeated context cheaper and faster.
4	**Prompt** compression reduces residual verbosity.

Bottom line	Needlepath is not a replacement for token optimizations. **It is the selection and orchestration layer** that chooses the right state first, then lets TOON, caching, and compression optimize the payload that remains.

Operationally, this distinction matters. In production agent systems, the dominant risk is not only excessive tokens; it is sending stale, redundant, or distracting state into a model call. Needlepath addresses that risk before token-level techniques take over.

3. The State Explosion Problem

Enterprise agents operate over state graphs, not isolated prompts. A single request can draw from persistent memory, chat history, user uploads, RAG retrieval, tool contracts, OAuth readiness, workflow checkpoints, generated artifacts, and audit controls. Each state item may be critical for one step and irrelevant for the next.

The challenge is selective reliability: identifying the smallest context that remains sufficiently grounded for the requested action. Naive truncation optimizes only for length. Needlepath optimizes for fit.

State source	Why it grows	Failure mode if unmanaged
Conversation history	Every user and assistant turn adds context.	Stale assumptions and intermediate decisions distract the model.
RAG retrieval	Knowledge spaces return chunks, metadata, and citations.	Irrelevant chunks inflate prompts and dilute evidence.
Agent memory	Preferences, procedures, prior outcomes, and learned defaults accumulate.	Low-priority memories crowd out task-critical facts.
Tool metadata	Tool lists and schemas expand as capabilities grow.	Tool bloat raises cost and increases wrong-tool risk.
Workflow traces	Multi-step runs produce intermediate outputs and checkpoints.	Trace noise obscures the current step objective.
Files and artifacts	Uploads, generated documents, and media add references and manifests.	Missing file state breaks grounding; excess file state inflates cost.

3.1 Design Requirements

Requirement	Implication
Typed state	Every candidate record must carry type, source, provenance, freshness, authority, dependency, and risk metadata.
Constraint-first selection	Evidence, citations, identifiers, workflow dependencies, and output-contract requirements must be preserved before budget optimization.
Auditable fallback	The system must explain when it declined to optimize and why baseline context was safer.
Workflow-level metrics	Savings, quality, latency, fallback, and selected-state signals must be reported by agent family and workflow.

4. Product Architecture

Needlepath is designed to integrate as an optimization layer around selected model calls, not as a replacement for the agent orchestrator. The orchestrator still owns workflow control; Needlepath owns context preparation, safety gating, and telemetry for the next model action.

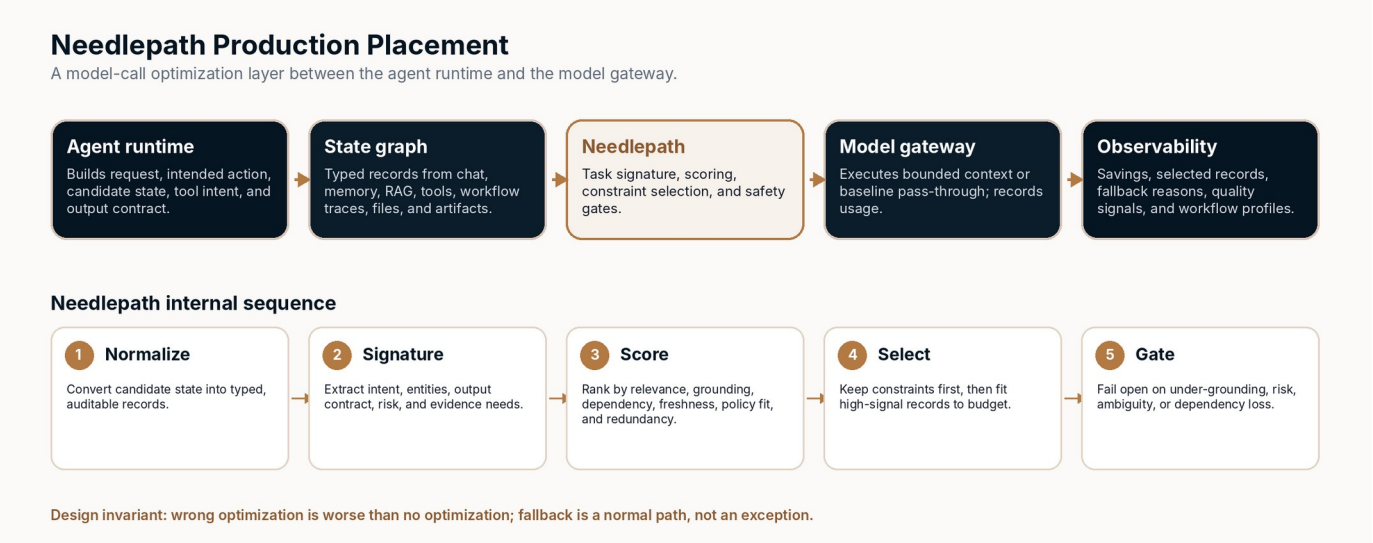

Figure 1. Production placement and internal sequence for Selective State Compression.

Stage	Needlepath function	Production requirement
Request + state graph	Inventory candidate state from runtime, tools, memory, retrieval, and artifacts.	Records carry provenance, freshness, authority, risk, and permission metadata.
Task signature	Derive entities, action intent, output type, domain tags, risk indicators, and required evidence.	Signature must separate useful state from adjacent noise.
State scoring	Rank records by relevance, grounding value, dependency value, recency, authority, policy fit, and redundancy.	Scores must be visible enough for telemetry and audit.
Optimized context	Format the high-signal subset for the next model action.	Context must preserve evidence and output-completeness constraints.
Safe execution	Use optimized context or fail open to baseline when risk is detected.	Optimization never silently overrides safety gates.

5. State Record and Task Signature Contract

The quality of state compression depends on the quality of metadata. Needlepath should not treat context as raw text. It should treat context as typed, governed state records with enough metadata to make selection explainable and reversible.

Record field	Purpose
id	Stable state-record identifier used in telemetry and trace replay.
type	chat_turn, memory, rag_chunk, tool_schema, workflow_checkpoint, file_manifest, external_record, artifact_reference.
source + scope	Where the record came from and which user, tenant, session, or workspace boundary applies.
provenance	Document id, tool call id, memory key, artifact id, or workflow step that produced the record.
freshness + authority	Timestamp and authority class used to supersede stale or lower-trust state.
dependencies	Identifiers, citations, required fields, selected entities, or tool/workflow constraints that depend on the record.
risk tags	Policy, privacy, external-write, credential, or grounding-critical markers.

5.1 Example State Record

```
{
  "id": "rag:doc:chunk:842",
  "type": "rag_chunk",
  "scope": "tenant_knowledge",
  "source": {"document_id": "doc_17", "section": "pricing_policy"},
  "tokens": 382,
  "freshness": "2026-06-02T12:20:00Z",
  "authority": "source_document",
  "dependencies": ["citation.required", "answer.grounding"],
  "risk_tags": ["grounding_critical"],
  "allowed_for": ["tenant:acme", "role:analyst"]
}
```

5.2 Task Signature

Signature field	Selection impact
Intent	What the next model action is trying to accomplish.
Entities	Accounts, customers, records, dates, files, products, users, or workspace identifiers that must remain attached.
Output contract	Answer, artifact, citation-bound response, tool argument, SQL, plan, email, document, or workflow synthesis.
Risk class	Read-only, sensitive, external write, credentialed action, admin action, privacy-bound action, or under-grounded task.
Evidence needs	Required sources, file references, citations, or selected records that must be preserved.
Fail-open thresholds	Workflow-specific risk and confidence thresholds that decide when baseline context is safer.

6. Selective State Compression Algorithm

Needlepath is intentionally multi-factor. Semantic relevance alone is not sufficient for production execution because a short but relevant prompt can still be under-grounded, policy-unsafe, missing a required identifier, or incomplete for the requested artifact.

Algorithmic posture	Optimize under constraints, not after truncation. Grounding-critical records, dependency records, and policy-required state are retained before discretionary state is ranked against the budget.

Signal	Interpretation	Example
Task relevance	How directly the record supports the current request.	A resume chunk is high relevance for a resume-summary request.
Grounding value	Whether removing the record weakens evidence or citation fidelity.	A source citation or uploaded file reference is preserved.
Dependency value	Whether downstream tool or workflow steps depend on the record.	A selected account id remains attached to a CRM update action.
Recency and authority	Whether fresher or more authoritative state supersedes older state.	Latest user correction outranks older chat history.
Risk and policy fit	Whether the output type or action requires fuller context.	External-write tasks can trigger conservative fallback.
Redundancy	Whether another retained record carries the same information.	Repeated acknowledgements are dropped.

6.1 Decision Procedure

```
def needlepath_prepare(request, candidate_state, contract, policy):
    signature = build_task_signature(request, contract)
    records = normalize_state(candidate_state)

    locked = retain_required_records(records, contract, policy)
    scored = score_remaining(records - locked, signature, policy)
    selected = select_under_budget(locked, scored, policy.max_context)

    if violates_grounding(selected, contract):
        return baseline(reason="under_grounded")
    if violates_dependencies(selected, contract):
        return baseline(reason="dependency_missing")
    if violates_risk_policy(selected, request, contract):
        return baseline(reason="risk_fallback")
    if below_confidence_threshold(signature, selected, policy):
        return baseline(reason="low_intent_confidence")

    return optimized_context(selected, telemetry=scored.summary())
```

The fail-open branch is first-class behavior. In the 100-task benchmark, Needlepath intentionally used baseline context for 24 tasks because baseline was safer for those cases.

7. Quality Preservation and Safety Invariants

For senior engineering readers, the most important implementation detail is not the compression ratio. It is the set of conditions under which Needlepath refuses to compress. The system must make conservative behavior visible, measurable, and tunable by workflow.

Invariant	Protected behavior	Example fallback reason
Grounding completeness	Required files, citations, retrieved evidence, record identifiers, and selected entities must remain present.	under_grounded
Dependency integrity	Tool arguments, workflow checkpoints, selected records, and downstream identifiers cannot be severed.	dependency_missing
Policy boundary	Tenant, role, privacy, credential, or workspace restrictions cannot be bypassed by context selection.	policy_boundary
Risk escalation	External writes, admin changes, credential actions, and high-impact tasks receive conservative context or confirmation.	risk_fallback
Output completeness	Long-form artifacts, regulated outputs, and document-grounded deliverables should not be shortened merely to reduce tokens.	contract_requires_fuller_context
Ambiguity threshold	If the task signature is not specific enough to select state confidently, baseline is safer.	low_intent_confidence

7.1 Safeguard Layer

Safeguard	What it does	Customer value
Grounding checks	Detect whether required evidence, file context, identifiers, or records are present.	Reduces under-contextualized answers.
Risk-aware fallback	Route high-risk or under-grounded cases to baseline context.	Makes safety visible and measurable.
Output-contract awareness	Keep answer format aligned with the requested artifact or response type.	Avoids shortening deliverables that require completeness.
Telemetry	Report savings, fallback behavior, selected state, and quality signals by workflow.	Supports rollout decisions and governance.

Metric note	The 83/100 figure is a task-usefulness preservation metric. It should be read as equal-or-better versus the baseline answer in the benchmark, not as a standalone relevance score.

8. Benchmark Methodology

The benchmark compared baseline prompts against Needlepath-optimized prompts over 100 agent tasks. The task set covered 10 agent families and 18 tool classes, including CRM-style records, calendar state, RAG-backed knowledge tasks, SQL and diagnostic workflows, support, and document-grounded operations.

The primary reported result is the final path. If Needlepath declined to optimize a task, the accounting reflects the baseline path. This prevents the benchmark from overstating savings by excluding difficult tasks.

Metric	Baseline	Final path	Saved	Reduction
Input tokens	100,884	73,257	27,627	27.38%
Generated output cost tokens	18,737	17,261	1,476	7.88%
Visible output text estimate	20,314	18,901	1,413	6.96%

Generated output cost is the model-reported generated token count. Visible output text is an estimate over the answer text a user sees. Because the benchmark used normal output mode, output reductions are primarily a byproduct of cleaner context rather than aggressive answer compression.

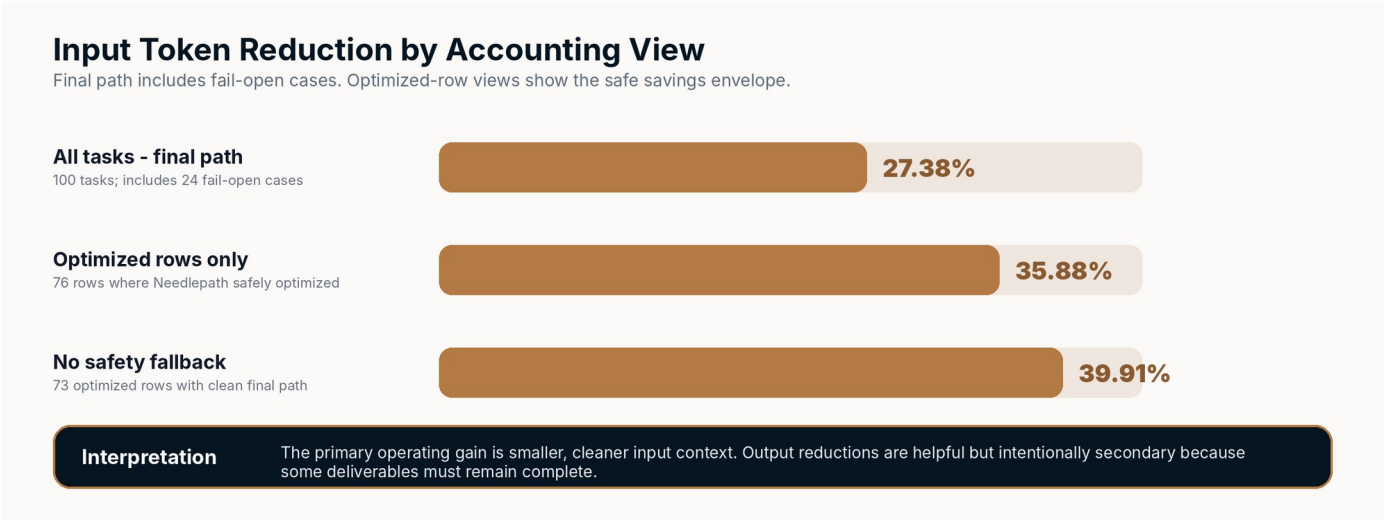

Figure 2. Input-token reduction across production-conservative and optimized-row accounting views.

9. Results Interpretation

The benchmark should be read as an operating-profile result: Needlepath reduces input context when it can do so safely, declines to optimize when baseline context is safer, and keeps quality assessment tied to task usefulness versus baseline.

Slice	Tasks	Input reduction	Generated output reduction	Visible output reduction
All tasks - final path	100	27.38%	7.88%	6.96%
Optimized rows only	76	35.88%	12.31%	11.13%
Optimized rows with no safety fallback	73	39.91%	15.83%	11.62%

The optimized-row views reveal savings potential where Needlepath can safely operate. The all-task view is the more important production number because it includes the safety behavior real deployments require.

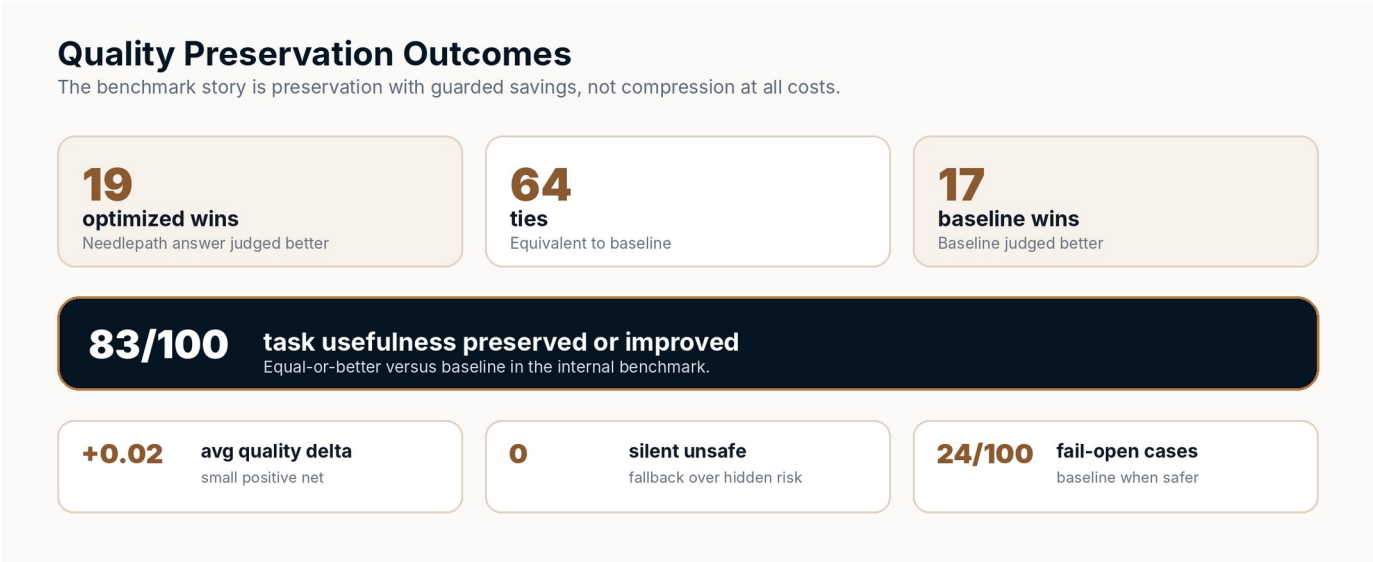

Figure 3. Quality preservation outcomes and fallback behavior in the 100-task benchmark.

9.1 Executive Reading of the Numbers

Reader	Implication
CTO	Context optimization has credible infrastructure value when savings are measured after fail-open behavior, not only on easy rows.
VP Engineering	The integration surface is narrow enough to pilot around selected model calls, but rollout must be governed by workflow-level thresholds.
Research leader	The research question is not just compression. It is selective reliability under grounding, dependency, risk, and output-contract constraints.

10. Runtime Integration Model

Needlepath should be inserted after the agent runtime has gathered candidate state and before the model provider call is constructed. Existing agent, workflow, tool, memory, artifact, and governance systems remain the source of truth.

Integration point	Responsibility	Needlepath contract
Agent runtime	Builds candidate state and intended action.	Pass request, state records, output contract, and policy envelope.
Needlepath	Selects or passes through context.	Return optimized context package or baseline decision with reason.
Model gateway	Executes model call.	Receives bounded context and records token usage.
Observability plane	Collects run metadata.	Store input-before, input-after, saved tokens, fallback reasons, selected records, and quality signals.
Governance	Controls rollout and risk.	Apply workflow-level thresholds, beta flags, confirmation rules, and rollback policies.

10.1 Telemetry Schema

```
{
  "stage": "agent:exec",
  "mode": "on",
  "input_before": 567,
  "input_after": 229,
  "saved_tokens": 338,
  "fallback": false,
  "selected_records": 12,
  "truncated_records": 31,
  "source_scopes": ["session", "memory", "rag", "tool"],
  "fallback_reason": null,
  "quality_probe": "preserved"
}
```

A successful deployment should make the optimization state visible. Teams need to know not only how many tokens were saved, but also when Needlepath declined to optimize and why.

10.2 Production Metrics

Metric family	What to measure
Token economics	Input saved, output cost change, total model-cost impact, cache effects.
Latency profile	Model-call latency, end-to-end latency, p50/p95/p99, fallback overhead.
Quality preservation	Usefulness, correctness, grounding fidelity, format compliance, actionability.
Operational safety	Fail-open rate, safety fallback rate, confirmation rate, policy-blocked actions.
Workflow fit	Savings and fallback by agent family, tool class, state source, and output contract.

11. Best-Fit Workflows

Needlepath is less valuable for short, stateless, one-shot prompts. Its economic and operational value grows with state graph depth, repeated calls, expensive models, and high-volume workflows.

Workflow pattern	Why it fits	Validation focus
Entity-heavy operations	Customers, accounts, invoices, dates, or workspace identifiers make relevance selection strong.	Savings by workflow and preservation of required entities.
Structured tool workflows	Tool metadata and execution policies can be ranked against explicit task intent.	Fallback rate, latency, and correct tool-use context.
Document-grounded tasks	Evidence can be preserved while unrelated history is removed.	Grounding coverage and artifact completeness.
Diagnostic or SQL workflows	Input often compresses while output remains detailed for safety and explainability.	Balance between concise context and adequate reasoning detail.

11.1 Customer Evaluation Path

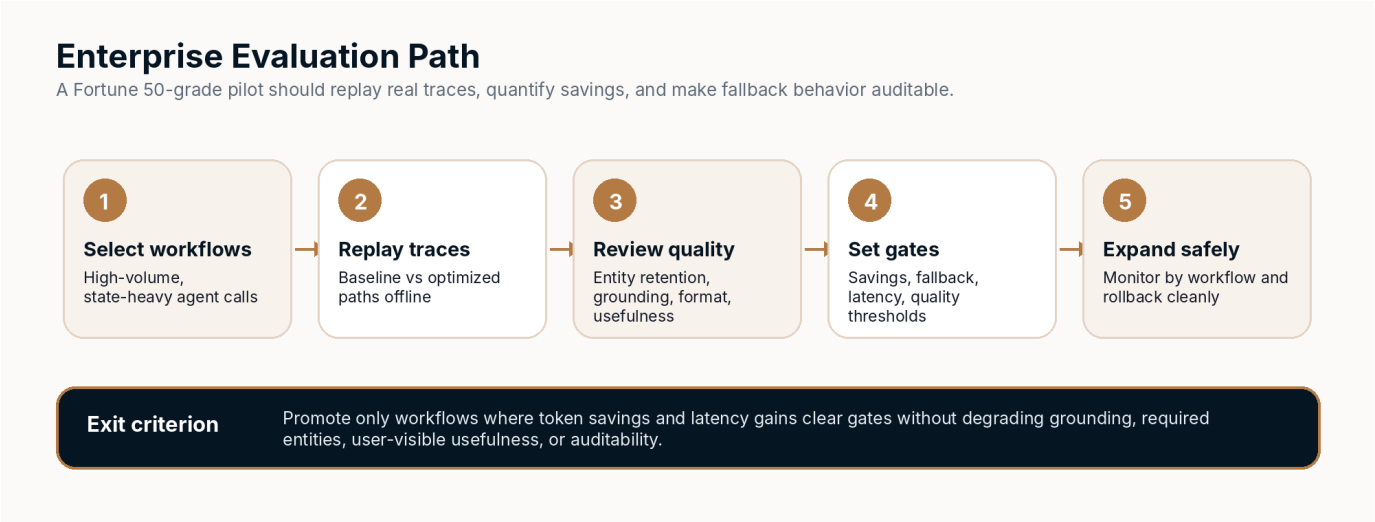

Figure 4. Recommended enterprise evaluation path and promotion criteria.

Phase	What happens	Decision output
1. Select workflows	Choose one or two high-volume agent workflows with meaningful state growth.	Target workflows and baseline metrics.
2. Replay traces	Run baseline and optimized paths over representative traces without disrupting users.	Savings, latency, fallback, and quality comparison.
3. Review quality	Evaluate preserved entities, grounding, completeness, format compliance, and actionability.	Workflow-level readiness score.
4. Set rollout gates	Define thresholds for savings, fallback rate, latency, and quality preservation.	Deployment guardrails.
5. Expand safely	Move from selected calls to additional agent families with telemetry-backed monitoring.	Broader optimization roadmap.

12. Governance, Rollout, and Failure Modes

Selective compression changes the context delivered to a model, so it must be governed like infrastructure. The right rollout model is not a global switch. It is workflow-specific enablement with thresholds, trace replay, observability, and reversible policy.

Control	Production expectation
Enablement scope	Start with read-only, high-volume, state-heavy workflows; avoid broad enablement before trace replay.
Gates	Minimum savings, maximum fallback rate, no quality degradation, no grounding failures, acceptable latency delta.
Sensitive actions	External writes, admin actions, credential changes, publication, deletion, and regulated outputs require conservative context or human confirmation.
Rollback	Per-workflow flag should return the model gateway to baseline context without application redeploy.
Audit	Store selected-record counts, omitted source scopes, fallback reason, policy mode, benchmark cohort, and quality probe results.

12.1 Common Failure Modes to Test

Failure mode	Example	Mitigation
Grounding loss	Required source chunk or file reference removed.	Fail open; add grounding-critical lock.
Entity drift	Older entity or incorrect account id retained.	Recency/authority scoring and entity-dedup checks.
Tool dependency break	Tool schema or argument dependency omitted.	Dependency lock and contract validation.
Over-short artifact context	Long deliverable loses necessary context.	Artifact-aware mode and output-contract gate.
Policy boundary confusion	State from wrong scope appears eligible.	Scope filtering before scoring; audit selected records.

Rollout stance	Needlepath should earn scope. Start narrow, prove workflow-level value, and expand only where telemetry demonstrates savings without quality or safety regression.

13. Limitations and Roadmap

Needlepath is a context optimization system. It is not a substitute for application-level correctness, role-based access control, credential governance, model evaluation, or workflow validation. It should be deployed with observability, evaluation gates, and rollback controls.

Area	Current implication	Roadmap direction
Semantic ambiguity	Some tasks do not expose enough intent to safely select state.	Clarification prompts and stronger task-signature extraction.
External writes	Actions with side effects require conservative context and confirmation.	Risk-specific compression policies and approval-aware state retention.
Long artifacts	Output length may need to remain high even when input compresses.	Artifact-aware mode that optimizes context while preserving deliverable completeness.
Evaluation drift	Workflow behavior changes as agents, tools, and data evolve.	Continuous benchmark replay and per-workflow thresholds.
Tenant heterogeneity	Enterprises differ in schema, authority, permissions, and compliance expectations.	Configurable record contracts, policy adapters, and validation suites.

13.1 Research Questions Worth Tracking

- How should task signatures be evaluated when user intent is incomplete or ambiguous?
- Which metadata fields have the highest marginal value for safe context selection?
- How does state compression interact with model attention, long-context behavior, retrieval quality, and tool-use accuracy?
- What offline probes best predict production quality preservation before a workflow is enabled?
- How should fallback thresholds adapt across risk classes and output contracts without becoming opaque?

13.2 Appendix: Metric Definitions

Metric	Definition
Final path	The actual path used after Needlepath can either optimize or fail open to baseline.
Optimized rows	Rows where Needlepath returned a compressed context instead of baseline pass-through.
Generated output cost tokens	Model-reported generated token count used for output-cost accounting.
Visible output text estimate	Estimated answer text shown to the user, distinct from generated token accounting.
Task usefulness preserved or improved	Judgment that optimized output was equal to or better than baseline for the task.

14. Conclusion

Needlepath gives engineering and product leaders a practical control plane for agent context. Instead of paying to send everything, teams can send the state that matters, preserve the evidence that grounds the answer, and measure when baseline context is the safer path.

The benchmark evidence is intentionally production-conservative: 27.38% final-path input reduction across all tasks, 35.88% reduction on optimized rows, 24/100 fail-open cases, and usefulness preserved or improved in 83/100 cases. That combination is the important result. Needlepath does not merely reduce tokens. It makes context selection observable, governed, and safe enough to operate in production agentic AI systems.

Closing position	Selective State Compression is an infrastructure advantage: smaller prompts, cleaner context, explicit fallback behavior, and a controlled path to lower-cost agent workflows.

14.1 Decision Frame for Enterprise Leaders

Decision question	Practical answer
Adopt for pilot when	Workflows have high call volume, large state graphs, expensive model calls, and measurable grounding requirements.
Do not lead with	Short stateless prompts, one-shot rewrites, or high-risk external-write workflows before trace replay.
Demand before production	Trace replay, workflow-specific gates, fallback telemetry, quality probes, rollback, and governance sign-off.
Measure after launch	Savings, latency, fail-open rate, quality preservation, grounding defects, and workflow-level drift.

Next Moca Technical Paper | June 2026

Prepared for senior technical evaluation of production agentic AI infrastructure.