Cognitive Continuity Infrastructure: Architectural Patterns for Persistent Operational Context in Stateless Multi-Agent Systems

Riccardo Crosa[1] William Mills[2]

[1] Montecristo International OÜ, Tallinn, Estonia
[2] Hanse Media, London, United Kingdom

riccardo.crosa@montecristo.it
https://mci.ee

May 2026

Abstract

Multi-agent systems built on large language models face a structural tension: organizations require persistent operational context, but the underlying models are stateless. Each invocation begins without memory of prior sessions, roles, or decisions. This paper introduces *Cognitive Continuity Infrastructure* (CCI)—a model-agnostic architectural pattern in which behavioral continuity emerges from the infrastructure rather than from any individual model. The architecture comprises a central orchestrator, specialized agents with persistent role definitions, a relational persistence layer for externalized organizational state, and an asynchronous human approval loop. We differentiate CCI from memory augmentation approaches (including RAG and long-term agent memory) by showing that CCI targets operational continuity—the preservation of roles, relationships, procedural norms, and accountability structures—rather than information retrieval. We propose a taxonomy of artificial organizational memory (episodic, semantic, procedural, prospective) and introduce *cognitive drift* as a systems risk: the gradual degradation of behavioral consistency when the underlying model changes while organizational state remains constant. We outline preliminary metrics for drift detection across five dimensions (tone, policy, procedure, escalation, identity). CCI should be understood not as a claim about machine consciousness or persistent internal memory, but as an architectural strategy for reconstructing organizational continuity across stateless model invocations.

Keywords: cognitive continuity, multi-agent systems, workflow persistence, behavioral consistency, orchestration architecture, cognitive drift

1 Introduction

Organizations accumulate operational context over time: knowledge of how processes work, which relationships matter, what decisions were made and why. This context is fragile in human

organizations—it degrades when employees leave or when procedures go undocumented (Walsh & Ungson, 1991)—and it is entirely absent in systems built on stateless large language models (LLMs).

A growing body of work demonstrates that multiple LLM-based agents can be orchestrated to perform complex tasks (Wu et al., 2023; Hong et al., 2024). Yet these systems share a structural limitation: every model invocation begins with an empty context window. The model retains no record of prior sessions, no awareness of ongoing relationships, and no knowledge of its own operational history. Current approaches address this gap primarily through Retrieval-Augmented Generation (RAG), which appends retrieved documents to the context window at query time (Lewis et al., 2020). RAG addresses information retrieval but not operational continuity: knowing that a meeting occurred is different from maintaining a coherent relationship with the contact discussed in that meeting across weeks of interaction.

This paper introduces *Cognitive Continuity Infrastructure* (CCI): a model-agnostic architectural pattern in which persistent operational context emerges from the design of the surrounding infrastructure rather than from the capabilities of any individual model. This paper is intentionally positioned as a conceptual and architectural contribution rather than an empirical benchmark study. The contribution is the architectural abstraction, the associated memory taxonomy, and a novel risk category.

CCI should be understood not as a claim about machine consciousness or persistent internal memory, but as an architectural strategy for reconstructing organizational continuity across stateless model invocations.

Our contributions are:

1. We define CCI as a distinct architectural pattern and differentiate it from existing approaches to memory augmentation and workflow orchestration (Section 3).

2. We propose a taxonomy of artificial organizational memory comprising four types: episodic, semantic, procedural, and prospective (Section 4.3).

3. We identify *cognitive drift*—the gradual degradation of behavioral consistency when the underlying model changes—as a novel systems risk, and propose preliminary operationalization metrics (Section 7).

The architecture is grounded in the operational experience of designing and managing a highly AI-mediated organizational workflow (Crosa, 2026) in which the majority of operational work is performed by LLM-based agents.

2 Related Work

2.1 Multi-Agent LLM Systems

AutoGen (Wu et al., 2023) enables conversational multi-agent workflows. MetaGPT (Hong et al., 2024) assigns software engineering roles to agents. CrewAI organizes agents into crews with defined tasks. These systems demonstrate that role specialization and inter-agent coordination are feasible and productive. However, existing frameworks treat agents as ephemeral: they are

instantiated for a task and discarded upon completion. Agent identity—accumulated interaction history, role evolution, relational context—is not preserved across sessions.

2.2 Memory Augmentation

RAG (Lewis et al., 2020) remains the dominant paradigm for extending LLM access to external information. More recent work explores long-term memory modules for agents (Park et al., 2023), including reflection mechanisms that synthesize higher-level observations. These approaches operate at the level of information retrieval: the agent gains access to facts it would otherwise lack. CCI addresses a different layer—operational continuity—where the challenge is not recalling a fact but maintaining coherent roles, relationships, and decision-making context across unbounded time horizons.

2.3 AI as Infrastructure

Riva (2025) introduces Cognitive Infrastructure Studies to examine how AI systems reshape human cognition pre-consciously, proposing that AI functions as an invisible infrastructure conditioning what is knowable and actionable. Our work shares the infrastructural framing but addresses a different problem: not how AI changes human thinking, but how organizations can maintain continuous operational context despite the stateless nature of the models themselves.

3 Why CCI Is Not Merely Long-Term Memory

A likely objection is that CCI is "just memory plus orchestration." This section makes the distinction explicit.

Existing systems—including RAG pipelines, LangGraph-style workflow engines, and AutoGen with persistent memory—optimize for:

- **Retrieval quality**: finding the right information at query time.
- **Context extension**: fitting more relevant content into the context window.
- **Task decomposition**: breaking complex tasks into manageable sub-tasks.

CCI optimizes for a different set of objectives:

- **Operational continuity**: maintaining coherent multi-step processes across weeks or months.
- **Identity persistence**: ensuring that an agent occupying a role behaves consistently with that role's accumulated history.
- **Procedural stability**: preserving organizational routines and norms across model invocations.
- **Accountability structures**: maintaining clear records of who proposed what, who approved it, and why.

- **Asynchronous governance**: enabling human oversight without requiring real-time human availability.

The distinction is analogous to the difference between a filing system and an organization. A filing system stores and retrieves documents. An organization maintains roles, relationships, procedures, and decision-making authority over time. CCI is concerned with the latter.

4 Theoretical Framework

4.1 Defining Cognitive Continuity

We define *cognitive continuity* as the property of a system to maintain coherent operational context—including role definitions, relational state, decisional history, and procedural norms—across discontinuous computational sessions. A system exhibits cognitive continuity when it demonstrates behavioral consistency across session boundaries: responses, decisions, and relational conduct remain coherent with the system's accumulated operational history.

This definition is deliberately agnostic about mechanism. Cognitive continuity can in principle be achieved through native model memory, through external memory augmentation, or—as we propose—through infrastructural design.

4.2 Cognitive Continuity Infrastructure

A CCI is an architecture that produces cognitive continuity as an emergent property of the system. It comprises four components (Figure 1):

1. **A central orchestrator** that sequences agent activations, maintains workflow state, and enforces decisional protocols.

2. **Specialized agents with persistent role definitions**, where each agent occupies a defined organizational role that persists across sessions.

3. **A relational persistence layer**, where all operationally relevant information—contacts, interaction histories, task states, decisions, and their rationales—is stored outside the model in a structured, queryable format.

4. **An asynchronous human approval loop**, where a human principal approves, rejects, or modifies proposed actions before execution.

The critical design principle is that *state lives in the infrastructure, not in the model.* The LLM is treated as a stateless reasoning engine that receives reconstructed context, produces a response, and retains nothing.

4.3 A Taxonomy of Artificial Organizational Memory

Drawing on cognitive psychology and organizational theory, we propose four types of artificial organizational memory:

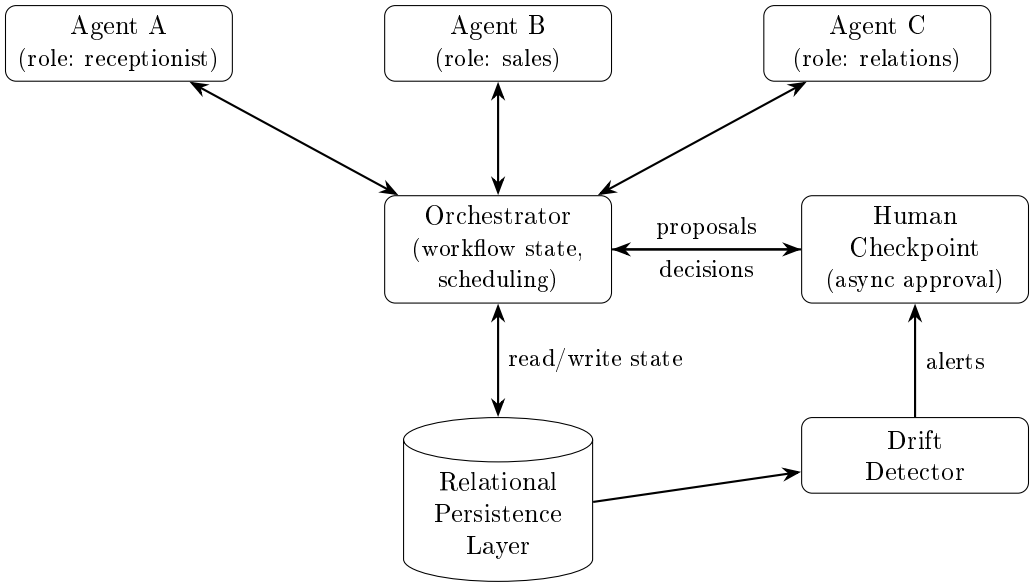

Figure 1: CCI architecture. Agents receive reconstructed context from the persistence layer via the orchestrator. The human checkpoint operates asynchronously. The drift detector monitors behavioral consistency over time.

Type	**Content**	**CCI Implementation**	**Example**
Episodic	What happened	Interaction logs	"Client X rejected proposal on Mar 3"
Semantic	What we know	Contact/entity records	"Client X prefers short contracts"
Procedural	How we operate	Workflow definitions	"Follow up 48h after proposal"
Prospective	What we must do	Task queue, triggers	"Call Client X on Mar 5"

Table 1: Taxonomy of artificial organizational memory with CCI implementation mapping.

Existing RAG systems primarily address episodic and semantic memory. CCI extends coverage to procedural memory (encoded in the orchestrator's workflow logic and agent role definitions) and prospective memory (maintained through scheduled tasks and trigger conditions). The inclusion of prospective memory is particularly significant: it enables the system to act on future obligations without external prompting.

5 Architecture

5.1 Design Principles

The CCI architecture is governed by five principles derived from operational experience:

Principle 1: Persistent role definitions over generic capability. Each agent occupies a named role with a defined communication style and domain of responsibility. A sales agent that consistently presents itself as the same entity across interactions creates relational continuity with external contacts.

Principle 2: Relational state over vector retrieval. Operational state is stored in a relational database with explicit schemas for contacts, interactions, tasks, and decisions. This provides queryable, structured access to organizational history rather than probabilistic similarity search over document embeddings.

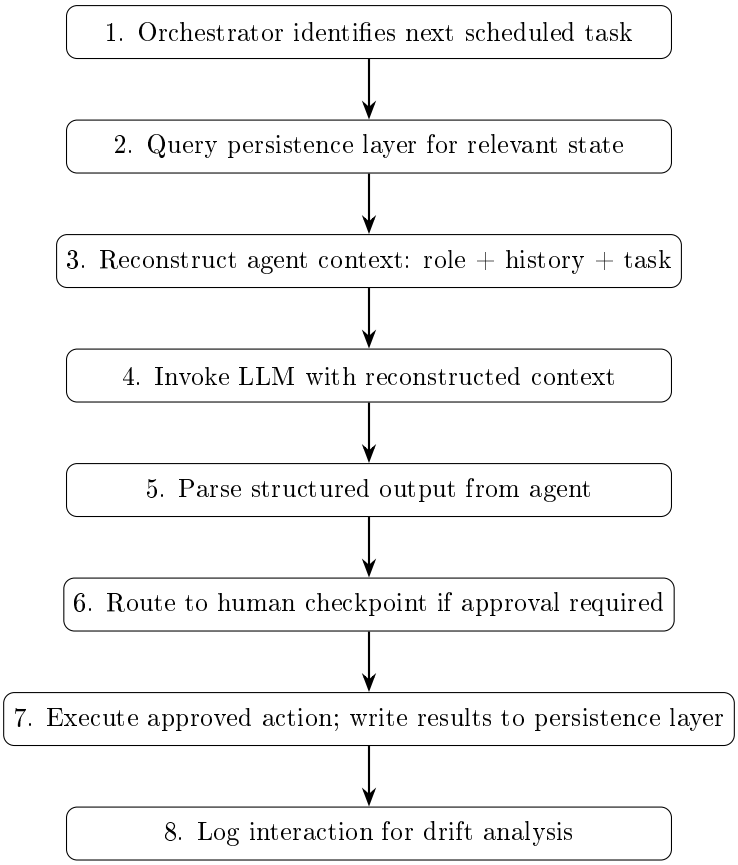

Figure 2: Session reconstruction flow. At each invocation, the orchestrator rebuilds the agent's operational context from externalized state, ensuring continuity across stateless model calls.

Principle 3: Asynchronous human authority. Agents propose actions; the human principal approves or rejects them through an asynchronous channel. This preserves human authority and accountability without requiring real-time availability.

Principle 4: Model independence. The architecture is designed so that the underlying LLM can be replaced without disrupting operational continuity. Because state is externalized and agent behavior is guided by role definitions and workflow logic, switching models changes the reasoning engine but not the organizational state.

Principle 5: Commodity infrastructure. The system must run on minimal infrastructure to be viable for small organizations. This is not merely a cost constraint but a design principle: organizational coordination tools that require enterprise infrastructure remain inaccessible to the organizations that most need them.

5.2 Design Tradeoffs

CCI intentionally sacrifices certain properties in exchange for others:

These tradeoffs are appropriate for organizations where continuity, accountability, and governance are more valuable than throughput or response speed.

5.3 System Components

The CCI comprises four layers:

CCI sacrifices	In exchange for
Response latency (context reconstruction)	Operational continuity
Architectural simplicity	Accountability and audit trails
Stateless horizontal scalability	Behavioral consistency
Full agent autonomy	Human governance and drift detection

Table 2: Design tradeoffs in the CCI architecture.

Orchestration layer. A central process that maintains the master schedule, dispatches tasks to agents, collects their outputs, routes decisions to the human checkpoint, and executes approved actions. The orchestrator encodes the organization's procedural memory.

Agent layer. Each agent is defined by a role specification comprising: identity (name, communication style), domain (scope of responsibility), capabilities (available tools and data), and constraints (actions requiring approval). At each invocation, the agent receives its role specification and relevant state reconstructed from the persistence layer.

Persistence layer. A relational database stores contacts, interaction logs, task queues, decision records, and agent output history. The schema encodes organizational priorities: its structure reflects what the organization tracks and values.

Human interface layer. A messaging channel through which the human principal receives proposals, reviews them, and responds with approvals, rejections, or modifications. The interface is deliberately minimal to reduce friction and enable oversight from a mobile device.

5.4 Operational Walkthrough

To illustrate how these components interact, consider the following scenario:

1. **Week 1.** The sales agent contacts Client X by telephone, introduces the advertising inventory, and logs the interaction. Client X expresses interest but requests a proposal. The orchestrator creates a task: generate proposal for Client X, deadline Friday.

2. **Week 1, Friday.** The orchestrator activates the sales agent with reconstructed context: Client X's contact record, the prior interaction transcript, and the proposal template. The agent generates a proposal and routes it to the human checkpoint. The CEO approves with a minor price adjustment via mobile message. The proposal is sent.

3. **Week 3.** The orchestrator's prospective memory triggers: follow up with Client X (48h rule exceeded, escalate). The relations agent is activated with full interaction history. It calls Client X, who raises an objection about contract length. The agent logs the objection and proposes a counter-offer, routed to the CEO for approval.

4. **Week 3, post model update.** The LLM provider releases a new model version. The drift detector compares the relations agent's tone and escalation patterns against its baseline. It flags a shift: the agent's language has become noticeably more casual than its established persona. The alert is routed to the CEO.

This scenario illustrates four CCI properties: context reconstruction across a three-week gap (step 2–3), prospective memory activation (step 3), asynchronous human governance (steps 2–3), and drift detection (step 4).

6 Sociological Interpretation

One possible interpretation of the architectural patterns described above draws on organizational sociology. The CCI architecture was not derived solely from software engineering; it emerged from the observation that effective multi-agent systems tend to reproduce organizational forms described by classical sociology (Crosa, 2026).

Division of labor, role specialization, hierarchical coordination, and institutional memory are recurrent solutions to coordination problems. Durkheim (1893) described how specialization produces interdependence; Weber (1922) analyzed how procedural rationalization enables continuous operation. CCI arrives at analogous structures: agents specialize, an orchestrator coordinates, procedures are codified, and records are kept.

This parallel—which Crosa (2026) terms *organizational recapitulation*—suggests that organizational sociology may serve as a useful interpretive lens for the design of multi-agent systems. The coordination problems that arise in artificial organizations have structural similarities to those studied in human organizations for over a century. Whether this reflects a deeper invariance or merely a convenient analogy remains an open question.

7 Cognitive Drift

We identify *cognitive drift* as a systems risk specific to persistent multi-agent architectures built on stateless models. Cognitive drift occurs when the behavioral output of an agent changes—gradually and without explicit error—due to changes in the underlying model, even though the agent's role specification, available state, and task remain constant.

7.1 Sources of Drift

Model updates. When the LLM provider releases a new version, the model's response distribution shifts. An agent configured for formal communication may produce slightly more casual output; an agent that consistently recommended conservative strategies may begin favoring more aggressive ones.

Provider switching. Different model families exhibit different behavioral tendencies in tone, reasoning style, and risk assessment. Switching providers changes the reasoning engine while organizational state remains constant, creating a discontinuity in behavioral output.

Prompt sensitivity. Even without model changes, minor modifications to role specifications or context formatting can produce disproportionate behavioral shifts. This is well-documented in LLM research but takes on new operational significance in systems where behavioral consistency is a requirement.

7.2 Drift Taxonomy and Preliminary Metrics

We distinguish five types of cognitive drift, each with a tentative observable metric:

Drift Type	Description	Observable Metric
Tone drift	Communication style shifts	Embedding variance across responses
Policy drift	Decision patterns change	Approval/rejection ratio deviation
Procedural drift	Workflow steps are skipped or altered	Workflow deviation frequency
Escalation drift	Risk tolerance shifts	Escalation rate relative to baseline
Identity drift	Agent persona becomes inconsistent	Persona consistency scoring

Table 3: Cognitive drift taxonomy with preliminary observable metrics.

These metrics are proposed as starting points for operationalization, not as validated instruments. Developing formal drift detection methods—analogous to data drift and model drift monitoring in MLOps—is a priority for future work.

7.3 Why Drift Is Dangerous

Cognitive drift is operationally dangerous because it is silent: the system continues to function, produces plausible outputs, and raises no errors. The degradation of behavioral consistency is detectable only through longitudinal analysis or through the human principal's tacit awareness of expected behavior. In a CCI, the asynchronous human checkpoint serves as a partial drift detection mechanism, but it is insufficient for systematic monitoring. Dedicated drift detection components (Figure 1) are necessary.

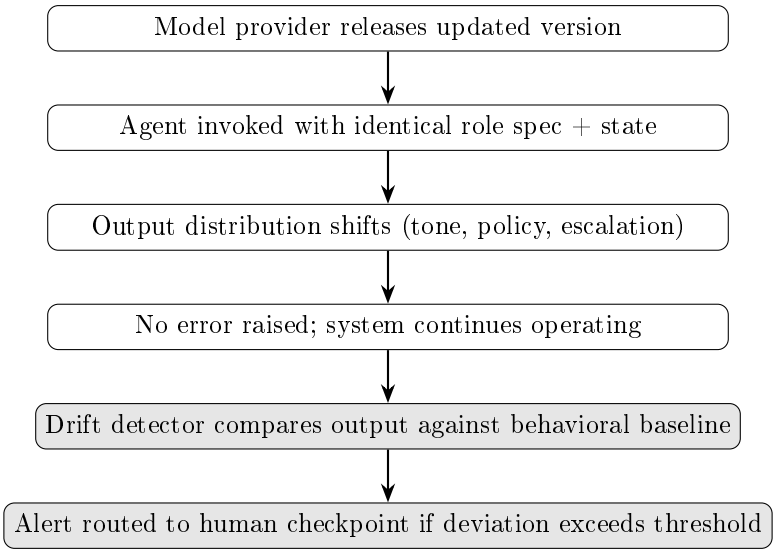

Figure 3: Drift emergence path. Model updates produce silent behavioral shifts that propagate through the system until detected by dedicated monitoring.

8 Planned Validation

The CCI architecture is deployed in a pre-operational configuration and will enter production in September 2026 with a B2B telephone sales deployment for outdoor advertising inventory. The

planned validation assesses three dimensions:

Operational continuity. Whether agents maintain coherent interactions with contacts across multiple sessions spanning weeks, measured by interaction log analysis and contact-reported experience.

Behavioral consistency. Whether agent behavior remains consistent with role specifications across model updates, measured by drift metrics (Table 3).

Governance efficiency. The ratio of agent-proposed actions to human interventions, and how this ratio evolves as the system accumulates operational history.

We intend to publish empirical results in a subsequent paper following the initial operational period.

9 Discussion

9.1 Contribution

The CCI framework proposes that the unit of analysis for organizational AI should be the *infrastructure*, not the model or the individual agent. This aligns with the distributed cognition tradition (Hutchins, 1995) and with recent proposals to study AI as infrastructure (Riva, 2025). The framework's strongest novel contributions are: (1) the explicit framing of continuity as an infrastructural rather than model-level property; and (2) the identification of cognitive drift as a distinct systems risk requiring dedicated monitoring.

9.2 Practical Implications

For practitioners, the CCI framework offers an alternative to waiting for models with native long-term memory. Organizations can achieve operational continuity now through architectural design. The commodity infrastructure principle makes this accessible to small organizations at negligible operational cost.

The asynchronous human approval loop deserves emphasis. Current discourse on AI autonomy often frames human oversight as a bottleneck. In the CCI framework, human oversight is a design feature providing quality control, accountability (the human principal remains responsible for organizational actions), and drift detection (the human's tacit knowledge serves as a behavioral consistency reference).

A further implication concerns organizational resilience to personnel discontinuity. In human organizations, institutional knowledge is sedimented in the minds of individuals—founders, managers, experienced employees. When these individuals leave or die, much of the organization's accumulated culture is lost. This is the classic problem of institutional continuity that Weber (1922) framed as the routinization of charisma: the transfer of a founder's vision into impersonal structures that can outlast any individual. Because CCI externalizes operational context, procedural norms, relational history, and decisional rationale into infrastructure, the organization's accumulated culture persists independently of any individual operator—including the founder. Roles, relationships, and decision-making patterns survive changes in personnel because they are properties of the system, not of any person within it. This represents a technical operational-

ization of what organizational theory has long described as the central challenge of institutional survival.

9.3 Limitations

We explicitly acknowledge the following limitations:

- **No longitudinal empirical data.** The planned deployment has not yet generated results. This paper presents an architectural pattern and risk taxonomy, not empirical validation.
- **No comparative benchmark.** We have not benchmarked CCI against existing orchestration frameworks (LangGraph, AutoGen with persistence, CrewAI). Such comparison would require a shared evaluation protocol that does not yet exist for organizational continuity.
- **No formal continuity metric.** We define cognitive continuity conceptually but do not provide a formal measurement instrument.
- **No demonstrated superiority.** We claim distinctiveness, not superiority. CCI addresses different objectives than existing systems; whether those objectives matter depends on organizational context.
- **Unclear scalability.** The architecture has been designed for small-to-medium organizations. Its behavior at larger scale—more agents, higher throughput, more complex governance—is untested.
- **Self-citation dependency.** The organizational recapitulation framework (Crosa, 2026) referenced in this paper is authored by one of the present authors. We have positioned it as an interpretive lens rather than as foundational theory.

10 Conclusion

We have introduced Cognitive Continuity Infrastructure as an architectural pattern for maintaining persistent operational context in multi-agent systems built on stateless language models. The core claim is that behavioral continuity is an infrastructural property: it emerges from externalized state, persistent role definitions, structured decision loops, and human checkpoints—not from the capabilities of any individual model.

The framework's most distinctive contribution is the identification of cognitive drift as a systems risk that is specific to persistent multi-agent architectures and that requires dedicated monitoring tools. As multi-agent systems move from discrete task execution to continuous organizational operation, the challenge shifts from making individual agents capable to making multi-agent organizations behaviorally coherent. CCI is a step toward addressing that challenge.

References

Crosa, R. (2026). *La società degli agenti: Dall'immaginazione sociologica alla coevoluzione uomo-IA*. Hanse Media.

Durkheim, É. (1893). *De la division du travail social.* Paris: Alcan.

Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, C., ... & Wu, Y. (2024). MetaGPT: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations.*

Hutchins, E. (1995). *Cognition in the Wild.* Cambridge, MA: MIT Press.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, 33, 9459–9474.

Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology.*

Riva, G. (2025). Invisible architectures of thought: Toward a new science of AI as cognitive infrastructure. *arXiv preprint arXiv:2507.22893.*

Walsh, J. P., & Ungson, G. R. (1991). Organizational memory. *Academy of Management Review*, 16(1), 57–91.

Weber, M. (1922). *Wirtschaft und Gesellschaft.* Tübingen: Mohr Siebeck.

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., ... & Wang, C. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155.*