

# The Continuity Layer After Memory

*The Mayorga Mnemosyne AI Continuity Framework for AI Systems That Preserve  
Meaning Over Time*

**By Francisco J. Mayorga, Jr.**

v1.0 | June 2026

Artificial intelligence is learning not to forget.

That may become one of the defining shifts in the next era of AI.

For years, most people experienced AI as a brilliant but forgetful assistant. You asked a question. It answered. You returned later. The world around the question had vanished. The project had to be rebuilt. The goal had to be restated. The vocabulary had to be explained again. The constraints, the prior decisions, the rejected options, the evidence, the assumptions, and the purpose behind the work all had to be carried back into the room by the user.

The machine could speak with astonishing fluency, but it could not reliably continue.

That contradiction has shaped the first public era of AI. Models became impressive inside the boundary of the prompt, but weak across the horizon of time. They could write, reason, summarize, translate, code, tutor, plan, and imitate expertise. Yet the moment the conversation reset, much of the accumulated meaning dissolved.

I have called this condition **brilliant amnesia**.

Brilliant amnesia is not the absence of intelligence. It is the absence of continuity. It is what happens when a system is locally capable but longitudinally fragile. The answer may be impressive today, but the reasoning behind it may not survive tomorrow. The system may know what to say in the moment, but not what must be preserved so that the next moment still makes sense.

This is the problem space that the **Mayorga Mnemosyne AI Continuity Framework** has been built to address.

The framework comes from a broader body of work I have been developing through books, essays, public framework materials, private research, implementation labs, and AI-native product experiments. Its central concern has remained consistent: artificial intelligence does not merely need better memory, larger context windows, or more sophisticated retrieval. It needs a governed continuity architecture capable of preserving meaning, evidence, decisions, assumptions, definitions, and justified change across time.

The current movement toward memory systems, AI-maintained knowledge bases, and agentic workflows does not originate this problem.

It confirms that the problem is becoming impossible for the industry to ignore.

OpenAI's Dreaming memory system points toward more dynamic, time-aware memory. Karpathy-style LLM Wiki patterns point toward persistent, interlinked, compounding knowledge bases. Recent research and preprints on agent memory, wiki-memory compilation, governed memory, and self-evolving retrieval suggest that the field is moving beyond one-shot prompting and static retrieval toward systems that carry context forward.

This is a meaningful development.

It is also only the beginning.

Memory is not continuity. A wiki is not continuity. Retrieval is not continuity. Even a larger context window, useful as it may be, does not by itself preserve why something mattered.

The missing layer is governed continuity: the architectural ability to preserve meaning, evidence, decisions, assumptions, definitions, contradiction handling, canon status, human review, and justified change across time.

The movement toward memory and AI-maintained knowledge bases does not make the Mnemosyne AI Continuity Framework less relevant. It makes it more urgent. It shows that the flag was planted in the right place.

## The Problem Was Never Just Forgetting

At first, the problem looked simple: the model forgot.

It forgot the project, the style, the previous answer, the assumptions, the reasons behind a decision, what had already been ruled out, and what mattered.

So the obvious answer seemed to be memory.

Let the system remember more. Let it store facts about the user. Let it retrieve past conversations. Let it hold more tokens. Let it search documents. Let it build a knowledge base.

All of those things help.

But the deeper problem was never just forgetting.

The deeper problem was the breaking of meaning.

Forgetting is when information disappears. Continuity failure is when the thread between information, judgment, purpose, and time breaks apart.

A system may remember a sentence and still forget what the sentence meant. It may retrieve a document and misunderstand its status. It may preserve an old decision but lose the reason that made the decision wise. It may remember a personal preference and confuse it with an institutional rule. It may summarize a source and discard the one qualification that kept the claim honest.

Memory brings the past forward.

Continuity makes the past intelligible to the future.

That difference matters. A box of photographs is not the same as a family history. A filing cabinet is not the same as institutional memory. A library without librarians, catalogs, marginalia, provenance, and scholarly discipline is not yet a civilization of knowledge. It is only a room full of books.

AI has begun building the room.

The next task is to preserve the civilization inside it.

## The Old Pattern: Ask, Answer, Forget

The old pattern of AI interaction was simple:

Ask.

Answer.

Forget.

The model responded to the immediate prompt. If documents were uploaded, it used them. If prior context existed in the conversation, it used that too. But the interaction remained trapped inside the moment.

The user carried the continuity burden.

The user remembered what had already been decided. The user restated the project's history. The user explained which terms meant what. The user corrected the model when it drifted. The user maintained the

line between draft and final, verified and speculative, old and current, preference and policy, source and inference.

For casual use, this is inconvenient.

When AI becomes infrastructure, it becomes dangerous.

A student should not have to explain the same learning goal every session. A founder should not have to restate an entire product strategy each time an AI agent helps with a build. A researcher should not have to rebuild a literature map every week. A company cannot depend on a system that forgets why one vendor was rejected and another was approved. A university cannot trust a curriculum assistant that remembers a syllabus but forgets the philosophy behind the course.

No serious institution can rely on intelligence that answers well today but loses the reasoning tomorrow.

The problem is not that the machine lacks words.

The problem is that the machine lacks a durable thread.

## The Industry Is Moving Toward Memory

The industry now appears to recognize that stateless AI is not enough.

OpenAI's Dreaming memory system is one clear signal. In its public explanation, OpenAI describes memory as a way for ChatGPT to apply preferences and constraints from past conversations and stay current as time passes. The example is simple but important: a system should know when a trip that was once future has become past, so it does not keep making recommendations as if the user were still traveling. [1]

That is more than a convenience feature.

It reveals an architectural truth: time changes meaning.

"I am going to Singapore" means one thing before the trip, another during the trip, and another after returning home.

"I am considering this vendor" changes meaning after evaluation, after rejection, after selection of a competitor, and after a later market shift.

"This is our draft policy" cannot be treated as "this is our approved policy."

"This source supports the claim" is not the same as "this source was later contradicted."

Memory that does not age becomes a fossil. Memory that updates without explanation becomes a rumor. Continuity must do something harder: it must preserve change without erasing the reasoning that came before.

OpenAI explicitly recognizes that traditional memory can become stale. Dreaming is presented as a way to revise memories over time, making them more useful and less frozen. [1]

That is progress.

But it is progress at the memory layer.

Memory helps an assistant remember what may matter.

Continuity asks what must remain meaningful.

Those are different questions.

## The Rise of the AI Wiki

At the same time, another movement is emerging outside the model itself: AI-maintained knowledge bases.

The Karpathy-style LLM Wiki pattern captures this shift well. Instead of treating documents as raw material to be retrieved again and again at query time, the LLM incrementally builds and maintains a persistent, interlinked wiki. It reads sources, extracts information, updates topic pages, identifies contradictions, and keeps knowledge current rather than rediscovering it on every question. [3]

This is a genuine step forward.

It moves from "search my files" to "compile my knowledge."

It turns a pile of documents into something closer to a living map.

In the old model, knowledge sits scattered across PDFs, transcripts, notes, articles, emails, screenshots, spreadsheets, and chat histories. The user knows that something important is "in there somewhere," but the structure is missing. Every serious question requires another hunt. Subtle connections remain invisible. Contradictions sleep undisturbed in forgotten folders.

An AI wiki changes the posture.

It says: do not merely retrieve fragments. Build a persistent artifact. Connect the concepts. Preserve summaries. Make relationships visible. Let knowledge compound.

That is a major improvement over the disposable prompt.

It is also deeply aligned with the continuity problem I have been naming.

But precision matters.

A wiki is not continuity.

It is a powerful continuity surface.

It is not the full architecture.

## Why a Wiki Is Not Continuity

A wiki can connect ideas, but connection is not judgment.

A graph can show relationships, but a graph is not governance.

A summary can compress a document, but compression is not evidence.

A contradiction can be flagged, but a flag is not a reviewed decision.

An interlinked markdown folder can help an AI reason through a project, but a folder is not institutional memory.

The beauty of an AI-generated graph can create the illusion of continuity. The user sees nodes, links, pages, references, and paths. It feels like the system understands. Sometimes it may. Sometimes it may only be arranging shadows on the wall.

A wiki may contain weak claims, hallucinated summaries, outdated assumptions, flattened distinctions, false equivalences, lost source details, or links that imply relationships without explaining their nature.

A basic wiki link says:

"This is connected to that."

Continuity asks a deeper set of questions.

How is it connected? Does it support, contradict, depend on, define, revise, supersede, exemplify, or merely resemble? Who decided the connection matters? What source supports it? What confidence does it deserve? Is the claim verified, speculative, analogical, contradicted, superseded, or approved for public use? If the answer changes, what changed and why?

Without those distinctions, a wiki may organize information while failing to preserve meaning.

A well-organized archive can still lose the thread.

## The Compilation Problem

Recent research and preprints on wiki-memory systems reinforce this concern. One paper on Wiki-memory Compile, Evaluate, Refine systems describes the promise of compiling domain knowledge into persistent artifacts, while warning that raw documents can lose critical facts when they are distilled into a wiki. [5]

That risk matters.

It names one of the core dangers of AI memory systems: compression can become deletion.

A system may summarize a source and lose the qualification that made the claim safe. It may consolidate several notes and erase a disagreement. It may merge competing interpretations and accidentally present a false consensus. It may preserve the obvious facts and discard the subtle exceptions. In trying to build a clean map, it may remove the difficult terrain.

This is why continuity cannot be reduced to knowledge compilation.

A continuity architecture must preserve not only what the system thinks is important now, but also the reasons, tensions, contradictions, uncertainties, and review history that make later interpretation possible.

Civilization does not continue merely because it stores records.

It continues because someone knows how to read them.

## Memory Is Personal. Continuity Is Structural.

Memory often begins with the user.

Continuity begins with the work.

Memory asks:

What should this assistant remember about me?

Continuity asks:

What must this system preserve so that future action remains meaningful?

Memory can remember that a user prefers concise responses. Continuity preserves why a strategy was chosen.

Memory can remember that a user is vegetarian. Continuity preserves which assumptions governed a recommendation.

Memory can remember that a user is working on a book. Continuity preserves the book's canon, evidence, terminology, unresolved claims, rejected directions, approval status, and publishing history.

Memory can personalize.

Continuity can govern.

This is why the Mnemosyne AI Continuity Framework is not merely a "better memory" proposal. It is an argument for a higher-order architectural layer.

The question is not only whether an AI system can remember.

The question is whether it can continue with meaning.

## The Continuity Layer After Memory

The emerging AI stack can be understood in layers.

Prompting gives the model an immediate task. It is powerful, flexible, and fast. But unless something preserves the result, much of the value evaporates.

Retrieval allows the model to search external documents. It reduces amnesia by giving the system access to stored information. But retrieval does not automatically preserve judgment, status, lineage, or meaning.

Memory carries user preferences, facts, and recurring context across conversations. It reduces repetition and makes the assistant feel less like a stranger.

AI wikis and compiled knowledge bases turn scattered materials into persistent, interlinked artifacts. They help knowledge compound. They make structure visible.

Governed continuity is the next layer.

This is the layer the Mnemosyne AI Continuity Framework defines.

Governed continuity asks not only what is stored, retrieved, remembered, or linked. It asks what must remain meaningful across time, who governs that meaning, how change is justified, and how future intelligence can inherit the reasoning of the past without becoming trapped by it.

That is the continuity layer after memory.

## The Mnemosyne AI Continuity Framework

The Mnemosyne AI Continuity Framework defines governed continuity as the missing architectural layer for AI-native systems.

Its premise is simple:

AI systems do not merely need more memory.

They need continuity architecture.

A continuity architecture preserves the living logic of work over time.

It does not try to remember everything. That would create noise.

It does not trust every summary. That would create fragility.

It does not treat every saved fact as equal. That would flatten judgment.

It does not assume that a link equals understanding. That would confuse structure with meaning.

Instead, it preserves the categories that allow future reasoning to remain coherent.

At minimum, an AI continuity architecture must preserve several kinds of continuity: decision continuity, evidence continuity, assumption continuity, definition continuity, contradiction continuity, workflow continuity, revision continuity, and human-governed continuity.

These are not decorative categories. They are the load-bearing structures. Remove them, and the system may still look impressive from a distance, but it will not carry the weight of serious work.

## Decision Continuity

Decision continuity preserves not only what was decided, but why.

Most organizations are full of decisions whose rationale has vanished. A team remembers that a product feature was delayed, but not the tradeoff. A company remembers that a vendor was rejected, but not the risk analysis. A school remembers that a course was redesigned, but not the educational philosophy. A founder remembers that a feature was deferred, but not the security, cost, or strategy constraint behind the decision.

When the reason disappears, the decision becomes brittle.

Future teams either repeat old mistakes or reverse decisions without understanding what they are undoing.

Decision continuity preserves alternatives considered, reasons accepted, reasons rejected, human approvals, unresolved concerns, and the conditions under which the decision should be revisited.

It is the ship's log of institutional judgment.

Without it, future captains inherit only the route, not the storm.

## Evidence Continuity

Evidence continuity preserves the relationship between claims and sources.

AI systems are very good at producing claims. The harder task is preserving the evidence trail behind them.

A claim may be verified. It may be plausible but unverified. It may be an analogy, a hypothesis, or a claim supported by a weak source. It may have been true last year but outdated now. It may be useful in one context and dangerous in another.

Without evidence continuity, AI systems produce confident prose that slowly detaches from its foundations.

Evidence continuity keeps the claim tied to the source, the source tied to its reliability, and the conclusion tied to its limits.

It is the footnote not as academic decoration, but as moral plumbing. It keeps the water of argument connected to its source.

## Assumption Continuity

Every serious decision rests on assumptions.

The danger is that assumptions often disappear into the floorboards.

"We should build this product first" may assume a certain customer, budget, timeline, regulatory context, technical stack, or market opening.

"We should publish this essay now" may assume a certain strategic goal, audience, risk tolerance, and desired association.

"We should use this model" may assume cost, quality, latency, privacy, and availability.

When assumptions are not preserved, later reasoning becomes ungrounded. People continue acting on conclusions after the premises have changed.

Assumption continuity asks:

What had to be true for this recommendation to make sense?

Is it still true?

If not, what must change?

This is where continuity becomes intellectual honesty. It reminds the system that every conclusion is a house built on foundations, and foundations must be inspected.

## Definition Continuity

Many projects do not fail because people disagree.

They fail because people use the same words to mean different things.

In AI work, this problem is especially dangerous.

Terms like memory, retrieval, context, agent, reasoning, alignment, knowledge, autonomy, workflow, governance, and continuity are often used loosely. A team can appear aligned while silently drifting.

Definition continuity preserves key terms and their intended meaning.

For the Mnemosyne AI Continuity Framework, this is essential. A model's response is not a decision. A draft is not canon. A source is not evidence until it has been evaluated. A recommendation is not approval. A memory is not automatically a truth, and a connection is not automatically a judgment.

Definition continuity protects a project from semantic drift. Imagine two teams using the same word "approved," but one means "the AI generated it" and the other means "a human owner signed off." They may believe they agree while building incompatible systems. Continuity begins by making the language stable enough for responsibility to attach.

## Contradiction Continuity

Contradictions are not failures.

Ignored contradictions are failures.

A mature AI continuity architecture should not hide contradictions to make the system look clean. It should preserve them, classify them, and create a path for review.

A new source may contradict an old claim. A new decision may conflict with a prior assumption. A model may produce a recommendation that violates a constraint. A project may contain two definitions of the same term. A public claim may exceed the evidence available.

Contradiction continuity ensures that tensions remain visible until they are resolved, deprecated, or consciously accepted.

This is especially important as AI systems become more agentic. An agent that cannot remember contradictions may continue acting on broken premises.

Contradiction is not the enemy of intelligence.

It is the smoke alarm of continuity.

## Workflow Continuity

Work does not happen in isolated prompts.

It happens through sequences: source intake, analysis, drafting, review, verification, revision, approval, publication, distribution, monitoring, and reassessment.

Workflow continuity preserves where work stopped, what remains pending, what depends on what, and what should happen next.

This is one of the most practical forms of continuity. It is also one of the easiest to underestimate.

A project does not only need knowledge.

It needs a memory of motion.

Without workflow continuity, even good ideas become stranded. A draft waits for verification. A verified claim waits for publication. A published essay waits for distribution. A useful insight waits for translation into product design. The work may be excellent, but without continuity, it remains on the wrong side of the river.

## Revision Continuity

Change is unavoidable.

The question is whether change is justified.

Revision continuity preserves why something changed, what changed, when it changed, who approved it, and what prior version it replaced.

Without revision continuity, updates become erasures.

A system may silently rewrite a conclusion. A team may lose the historical reason a phrase was protected. A project may forget why a feature was removed. A book may drift away from its original argument. A policy may change without retaining the path of reasoning.

A continuity system should be able to say: this term was changed because the old wording created confusion; this claim was softened because the evidence was weaker than expected; this feature was delayed because the security model was not ready; this public statement replaced an earlier draft after human review.

Revision continuity allows a system to evolve without losing itself.

It is the difference between growth and drift.

A tree grows by adding rings. It does not survive by pretending last season never happened.

## Human-Governed Continuity

The final layer is human governance.

AI can propose, organize, retrieve, summarize, detect patterns, flag contradictions, and suggest revisions.

But in serious work, AI should not silently decide what becomes canon.

The Mnemosyne AI Continuity Framework is built around a simple principle:

**AI proposes. Humans govern. Mnemosyne preserves.**

That means a continuity architecture must distinguish between AI-generated suggestions and human-approved knowledge.

It must know what is draft, reviewed, verified, rejected, private, public, deprecated, and canonical.

Without human-governed continuity, AI memory can become a fog of plausible summaries.

With human-governed continuity, AI becomes a partner in preserving meaning.

This is not a rejection of AI capability. It is a recognition of responsibility. A compass can guide a traveler, but it should not decide the purpose of the journey.

## Why This Matters More as AI Agents Mature

The continuity problem becomes more urgent as AI systems become more agentic.

A passive chatbot that forgets is annoying.

An agent that forgets can be dangerous.

If an AI agent books appointments, sends messages, writes code, generates proposals, updates records, analyzes contracts, supports training, drafts policies, or manages workflows, then continuity becomes part of safety.

What did the agent believe it was doing? Which instruction governed its action? What memory shaped the recommendation? What constraint did it follow? What evidence did it use? What prior decision did it inherit? What assumption was active? What changed between the first plan and the final action? What human approval was required? What should be remembered, forgotten, archived, or escalated?

These are not philosophical luxuries.

They are operational necessities.

The more capable AI becomes, the more expensive forgetting becomes.

Intelligence raises the stakes of amnesia.

## Convergence and Naming the Layer

As the industry wakes up to memory, wikis, agents, and knowledge-compounding systems, ideas will begin to converge. Large AI labs, independent researchers, open-source builders, and product companies will all move toward similar territory.

That is how technological waves work.

But convergence can erase origin.

A concept can be absorbed into product language. A framework can be diluted into feature names. A vocabulary can spread without attribution. An early flag can be buried under later implementation.

This essay continues my ongoing work on the **Mayorga Mnemosyne AI Continuity Framework**, a framework developed across public essays, framework materials, books, implementation labs, and private research. The public record now includes the Mnemosyne framework repository, published essays on retrieval and continuity, public-facing framework pages, and related books and laboratories. [9-12]

The point is not that no one else has worked on memory, retrieval, knowledge graphs, provenance, governance, or agent context.

Many have.

The point is more specific.

The Mayorga Mnemosyne AI Continuity Framework names and structures a specific missing layer after memory: governed continuity architecture.

It defines continuity as the preservation of meaning, evidence, decisions, assumptions, definitions, contradiction handling, canon status, human review, and justified change across time.

That is the flag I am planting through the Mayorga Mnemosyne AI Continuity Framework.

The present industry movement does not originate the argument. It makes the argument more visible.

## What the Industry Is Actually Discovering

The industry is discovering that intelligence alone does not compound.

Memory lets intelligence accumulate.

Wikis let knowledge become navigable.

Retrieval lets information become accessible.

Agents let systems act.

Yet without continuity architecture, all of these layers remain fragile.

A memory can become stale. A wiki can drop critical facts. A retrieval system can surface the wrong source. An agent can act on outdated assumptions. A summary can erase uncertainty. A graph can imply false coherence. A recommendation can hide its evidence. A project can drift without noticing.

Continuity is the layer that makes these systems trustworthy over time.

It does not replace memory. It governs memory.

It does not replace wikis. It gives wikis status, lineage, review, and meaning.

It does not replace retrieval. It tells retrieval what kind of thing it is retrieving and how it should be interpreted.

It does not replace agents. It gives agents a memory of purpose, constraint, approval, and consequence.

This is the next architecture.

## A Public Boundary and a Private Architecture

This essay intentionally defines the public doctrine, not the full implementation blueprint.

The public doctrine matters because the category needs language.

The industry needs better words for the difference between memory and continuity, personalization and governance, retrieval and evidence, links and meaning, summaries and institutional knowledge.

Implementation matters too. A full continuity system requires design choices around provenance, review processes, versioning, publication readiness, and human approval.

Those implementation details belong in technical notes, private research, product specifications, and controlled systems.

The public point is this:

AI-native organizations will need continuity architecture.

The Mayorga Mnemosyne AI Continuity Framework defines one such architecture.

## The Continuity Test

Here is a simple test for any AI memory system, wiki system, retrieval system, or agent platform.

Do not ask only:

Can it remember?

Ask:

Can it preserve why something mattered?

Can it distinguish draft from approved?

Can it tell whether a claim is verified, speculative, contradicted, or superseded?

Can it preserve the assumptions behind a recommendation?

Can it show which source supported which claim?

Can it explain what changed between versions?

Can it keep human approval separate from AI suggestion?

Can it tell future users not only what happened, but why?

If not, then it may have memory.

It may have retrieval.

It may have a wiki.

But it does not yet have continuity.

## Why the Name Mnemosyne Matters

Mnemosyne, in Greek mythology, is the goddess of memory and the mother of the Muses.

The name matters not because this framework is mythological, but because memory has always been tied to culture, creativity, education, and civilization.

Without memory, there is no learning.

Without continuity, memory becomes inert.

AI systems are now entering many of the spaces where human beings preserve meaning: schools, companies, archives, laboratories, creative work, and public institutions.

If AI systems become tools through which people learn, decide, write, govern, build, and remember, then the architecture of AI memory will shape the architecture of human continuity.

That is why this work matters.

## The Next Frontier

The frontier is no longer only model intelligence.

It is not only bigger context windows, better retrieval, personal memory, or wikis.

The next frontier is governed continuity.

AI systems must learn not only how to answer, but how to preserve the living logic of work over time.

They must know what changed, why it changed, what evidence supports a claim, what assumptions are active, what definitions govern a project, what contradictions remain unresolved, what humans approved, what should be carried forward, and what should be left behind.

They must know how to continue.

## Conclusion: The Systems That Can Continue

OpenAI Dreaming is a signal.

LLM Wikis are a signal.

Agent memory research is a signal.

Wiki-memory compilation research is a signal.

Self-evolving retrieval is a signal.

All of them point toward the same larger truth: AI is escaping the disposable prompt.

But escaping the prompt is not enough.

AI must also escape brilliant amnesia.

It must move from answering to remembering, from remembering to organizing, and from organizing to continuing.

That final movement requires governed continuity.

The Mayorga Mnemosyne AI Continuity Framework names this missing layer and gives it an architectural purpose: to preserve meaning, evidence, decisions, assumptions, definitions, contradiction handling, canon status, human review, and justified change across time.

The future will not belong merely to the AI systems that remember more.

It will belong to the systems that can continue with meaning.

## Source Notes

- [1] OpenAI. "Dreaming: Better memory for a more helpful ChatGPT." OpenAI, 2026. <https://openai.com/index/chatgpt-memory-dreaming/>
- [2] OpenAI Help Center. "Memory FAQ." OpenAI, updated 2026. <https://help.openai.com/en/articles/8590148-memory-faq>
- [3] Karpathy, Andrej. "LLM Wiki." GitHub Gist, created April 4, 2026. <https://gist.github.com/karpathy/442a6bf555914893e9891c11519de94f>
- [4] Du, Pengfei. "Memory for Autonomous LLM Agents: Mechanisms, Evaluation, and Emerging Frontiers." arXiv, 2026. <https://arxiv.org/abs/2603.07670>
- [5] Huerta, Juan M. "WiCER: Wiki-memory Compile, Evaluate, Refine Iterative Knowledge Compilation for LLM Wiki Systems." arXiv, 2026. <https://arxiv.org/abs/2605.07068>
- [6] Ming, Haoliang, Feifei Li, Xiaoqing Wu, and Wenhui Que. "Retrieval as Reasoning: Self-Evolving Agent-Native Retrieval via LLM-Wiki." arXiv, 2026. <https://arxiv.org/abs/2605.25480>
- [7] Miteski, Stefan. "Memory as Metabolism: A Design for Companion Knowledge Systems." arXiv, 2026. <https://arxiv.org/abs/2604.12034>
- [8] Lam, Chingkwun, Jiaxin Li, Lingfei Zhang, and Kuo Zhao. "Governing Evolving Memory in LLM Agents: Risks, Mechanisms, and the Stability and Safety Governed Memory Framework." arXiv, 2026. <https://arxiv.org/abs/2603.11768>
- [9] Mayorga, Francisco J., Jr. Mnemosyne Framework v0.1: Initial Public Conceptual Release. GitHub repository: cidvalue/mnemosyne-framework, April 27, 2026. <https://github.com/cidvalue/mnemosyne-framework>
- [10] Mayorga, Francisco J., Jr. "Retrieval Is Not Continuity." Franciscomayorga.com; related Zenodo record: Mnemosyne Framework v0.2: Continuity vs Retrieval, DOI 10.5281/zenodo.19816291. <https://franciscomayorga.com/essays/retrieval-is-not-continuity>
- [11] Mayorga, Francisco J., Jr. "When AI Builds Itself, Who Remembers Why?" Franciscomayorga.com, June 2026; Zenodo DOI 10.5281/zenodo.20599060. <https://franciscomayorga.com/essays/when-ai-builds-itself-who-remembers-why>
- [12] Mayorga, Francisco J., Jr. Mnemosyne AI Continuity Framework. Franciscomayorga.com, public framework hub. <https://franciscomayorga.com/>