Conversation Intelligence Architecture

A Theoretical Framework for Epistemic Context Management in Large Language Models

T. Larcombe MBA, B.Sc, N.Dip.M, Mri

4 Square Capital / 4 Square Innovations

April 2026

Table of Contents

Abstract

Large language models process conversation as a linear sequence of tokens, and manage that sequence with policies derived from a single assumption: that context management is a token budget problem. This dissertation argues that the assumption is wrong. Context management is an epistemic structure problem. A conversation does not merely accumulate tokens; it constructs a shared understanding with a topology, a temporal validity profile, an information density, and a frame. Managing it correctly requires reasoning about all four dimensions simultaneously, not merely about recency and size.

This work develops a theoretical framework for epistemic context management organised across four original contributions. The first is a scoring proxy architecture grounded in nine cognitive science memory models, which manages context through a composite signal of recency, relevance, confidence, and currency, applying tiered treatment to segments according to a continuous value score rather than positional age. The second is Dissolution Through Consensus: a mechanism that abandons the preservation frame entirely, replacing linear conversation history with a consensus-integrated boundary state synthesised by a multi-model ensemble via a structured four-part prompt. The third is the Conversation Intelligence Architecture, a four-level framework addressing what to retain (surprise-proportional allocation, grounded in the Free Energy Principle), how to represent it (navigational topology rather than content archive), how to measure success (generative competence, grounded in Kolmogorov complexity theory), and what threatens coherence (an adversarial frame agent maintaining a live assumption register). The fourth is a bitemporal extension to context scoring, which introduces valid time alongside transaction time as an independent dimension of retention policy, addressing the systematic mismanagement of forward commitments, retroactive corrections, and closed historical facts.

Together these contributions constitute a theoretical framework intended as a foundation for the research community. The empirical validation of individual claims is identified explicitly as future work, and each contribution is accompanied by a precise specification of what that validation requires.

The dissertation proceeds as follows. Chapter 1 defines the problem. Chapter 2 develops a three-axis knowledge taxonomy. Chapter 3 surveys the state of the art. Chapter 4 grounds the architecture in human memory models. Chapters 5 through 8 present the four contributions in sequence. Chapter 9 synthesises the framework and identifies open problems.

The project described in this dissertation began as an engineering problem. A long conversation degrades: instructions given at the start stop governing behaviour by the middle, facts established early are quietly discarded as the context fills, and the model's responses drift from the understanding the conversation was supposed to have built. The question seemed straightforward — how do you keep the right content available at the right time? The answers already in the literature seemed almost sufficient: retrieve relevant material at query time, compress what you cannot retrieve, evict what you cannot compress. Build a scoring function to decide what is worth keeping.

That answer is correct as far as it goes. The scoring proxy described in Chapter 5 of this dissertation implements it carefully, drawing on decades of cognitive science research into how the human memory system solves the same problem under the same constraint: limited capacity, continuous input, and the requirement to produce sensible output at any moment. The parallel is not superficial. The Atkinson-Shiffrin multi-store model (1968), Baddeley's episodic buffer (2000), Murdock's serial position effect (1962), and McGeoch's interference theory (1932) each illuminate a specific design decision in the proxy architecture. Grounding the engineering in the cognitive science is not decoration; it is justification.

But the scoring proxy, however well-grounded, still operates inside the preservation frame. It asks: which tokens should survive? It treats the conversation as a container of content and the context window as a container of tokens, and it optimises the mapping between them. The deeper question was not visible until the container metaphor was challenged directly. What if the goal is not to preserve the conversation's content but to preserve the understanding the conversation established? These are not the same thing. A conversation that has run for forty turns has produced a region in conceptual space with defined edges. The content that filled that region, the specific phrasing of each exchange,

is not what persists in an expert practitioner's memory after a long consultation. What persists is the shape of what was established. The boundary, not the matter.

This shift in framing, from preserving matter to describing boundary, is the intellectual pivot at the centre of this dissertation. Chapter 6 develops it through Dissolution Through Consensus: a mechanism that processes a conversation not by compressing its text but by synthesising, via a multi-model ensemble, a present-tense first-person description of what the conversation now knows, has decided, and has left open. The linguistic signature of this operation is measurable and distinct from summarisation. The consensus mechanism contributes genuine work, resolving substantive disagreement between independent models rather than merely averaging paraphrases. The dissolution is not a loss of information; it is a transformation into a different kind of representation that is more compact, more stable, and more generative.

From dissolution the path leads naturally to a more ambitious question. If a conversation establishes an epistemic structure, what are the dimensions of that structure, and how should each be managed? Chapter 7 proposes four answers, each grounded in a confirmed gap in the existing literature. Budget should be allocated proportionally to prediction error, not to positional recency, because what the model did not already know is precisely what is worth retaining. The established conceptual space should be represented as a navigational topology, not as an archive of content, because a map of where knowledge lives is more useful than a catalogue of what it contains. The success of a compression should be measured by whether the compressed representation can answer queries from the domain of the original conversation, not by surface similarity to the original, because the only test of a sufficient representation is whether it generates sufficiently. And a parallel adversarial agent should challenge the frame of the conversation, not its conclusions, because the most consequential failure mode in long-horizon reasoning is not a factual error inside the established frame but the silent accumulation of unexamined premises that constitute the frame itself.

Chapter 8 adds a final dimension that cuts across all the others. Every segment in a conversation carries two temporal attributes that a single recency signal cannot distinguish: the turn in which the statement was made, and the interval during which it is

true. A forward commitment, a decision made at turn five that governs behaviour for the duration of the conversation, should not decay by positional age because its validity has not expired. A retroactive correction does not merely supersede what came before; it closes the valid-time interval of the statement being corrected, retroactively. A description of a superseded state should be archived not because it is old but because its valid time has closed. Bitemporal management is not an optimisation. It addresses a class of failure that no other component in the architecture reaches.

This dissertation is a theoretical contribution. It does not claim to have empirically validated all of the claims it develops. What it claims is that each contribution is theoretically well-founded, that the confirmed gaps in the existing literature establish the novelty of each claim, and that the framework as a whole is coherent: the four contributions address four distinct dimensions of the same underlying problem, and they compose into a system in which each level addresses a failure mode that the previous level creates. The empirical work required to validate the framework is identified precisely in each chapter and collected in Chapter 9. That work is offered not as an apology for what is missing but as a research programme for the community that this framework is intended to serve.

The ambition of this work is to shift the vocabulary of the field. Context management is not a token budget problem. It is an epistemic structure problem. A conversation is not a sequence of messages to be managed. It is a dynamic epistemic structure with a topology, a temporal validity profile, a generative competence, and a frame. The framework developed here is one way of making that claim precise. There are others, and they will improve upon it. That improvement is the point.

Chapter 1: The Problem of Context Decay

This chapter establishes the problem that the dissertation addresses. Context decay is the progressive reduction in a large language model's effective use of early context tokens as a conversation grows, and it operates entirely within the context window: the tokens are present, attended to, and technically available, yet their influence on the model's output diminishes with distance from the current generation point. Understanding this phenomenon requires distinguishing it clearly from the hard context window limit, understanding why it arises from the training process rather than from the architecture, recognising the full spectrum of damage it causes, and appreciating that it has a hidden temporal dimension that conventional treatments overlook entirely. This chapter builds that understanding, and in doing so establishes the terms on which the rest of the dissertation proceeds.

What Context Decay Is

A useful starting definition separates the phenomenon from the infrastructure problem with which it is often confused.

> **Context decay**: the progressive reduction in a large language model's effective use of early context tokens as conversation length increases, operating entirely within the context window, such that information provided earlier in the sequence exerts diminishing influence on the model's output even though it remains technically present in the computation.

This definition demands a careful distinction from the hard context window limit. A hard limit is binary. Tokens beyond the model's maximum context length are not present in the computation; the model has no access to them whatsoever. Context decay is something different and, in some ways, more insidious. It operates entirely within the window. All the tokens are there, all of them are technically seen by the model, yet information from earlier in the sequence has a systematically reduced influence on what the model generates. The decay is gradual, not abrupt, and it is precisely this gradual quality that makes it difficult to detect and difficult to address.

The significance of this distinction cannot be overstated. A hard limit failure is visible: the system returns an error, the conversation truncates, the user knows something has gone wrong. Context decay fails silently. The model continues to respond, continues to appear coherent, and continues to produce output that incorporates some of the earlier context, just not all of it, and not the parts that matter most. The user who invested effort in a carefully structured opening prompt does not receive a notification that the prompt is no longer governing behaviour. The failure arrives as subtly wrong output, as a constraint quietly abandoned, as a response that would have been correct six turns ago but is now not quite right. This opacity is not incidental to context decay; it is its defining characteristic.

The Mechanism

The mechanism behind context decay is rooted in how transformer attention works. At each generation step, the model computes attention weights over every token in its context. In principle this is a flat, permutation-invariant operation: every token is equally eligible to be attended to, and the attention mechanism contains no architectural feature that privileges recent tokens over distant ones (Vaswani et al., 2017). In practice, however, the distribution of attention weights is far from uniform. Recent tokens consistently attract a disproportionate share of total attention, and this asymmetry grows as the context lengthens.

The cause is not architectural; it is learned. Positional encodings, whether absolute, relative, or rotary (as in RoPE and ALiBi variants), encode a notion of distance between tokens. The geometry of attention score distributions tends to favour smaller distances, partly as a consequence of how these encodings interact with the dot-product attention calculation. RoPE encodes position by applying rotation matrices parameterised by frequency $\theta_i = 10000^{-2i/d}$ to query and key vectors, so the dot product at relative distance $|m - n|$ varies sinusoidally per frequency component rather than decaying monotonically: the logit-space signal is oscillatory, not a simple gradient (Su et al., 2021). ALiBi takes the structurally distinct approach of adding a head-specific linear penalty, $-m_h \cdot |m - n|$, directly to pre-softmax logits, where the slope $m_h = 2^{-8h/H}$ varies geometrically across heads; a linear penalty in logit space translates to

strictly exponential decay in post-softmax attention probabilities, with different decay rates per head (Press et al., 2022). In practice, however, trained models using either scheme exhibit a U-shaped attention distribution rather than pure distance-dependent decay: both sequence endpoints attract disproportionate weight, with middle-sequence positions systematically underattended, an effect that persists under token-order permutation and is therefore attributable to architectural inductive biases from causal masking and residual connections rather than to the choice of positional encoding alone (Liu et al., 2024; Brown et al., 2024). But positional encoding is only part of the explanation. The more fundamental cause is that human-generated text, the material on which these models are trained, is overwhelmingly local in its coherence. The next sentence is almost always more relevant to the current sentence than something written twenty paragraphs ago. A model trained on such data learns, quite reasonably, that recency is a reliable heuristic for relevance. The attention asymmetry is not a flaw; it is a faithful representation of a real statistical property of human language.

The problem arises when this learned heuristic is applied to structured conversations or long-form tasks where early context is not merely background noise but contains essential, load-bearing instructions or facts. The model's prior that recent tokens are more relevant than distant ones is correct on average and incorrect in exactly the cases that matter most: the carefully constructed system prompt, the specialist constraint established in the third turn, the critical fact retrieved and placed at position eight. For these segments, the recency heuristic fails, and the failure is systematic rather than random.

There is also a compounding effect. As more tokens are added to the context, earlier tokens must compete for attention against an ever-growing pool of later tokens. The influence of any given early token is not simply constant and small; it continues to diminish as the sequence lengthens. This is the decay in context decay: a progressive, accumulating loss of influence rather than a fixed penalty for being early.

The empirical consequences of this mechanism are well-documented. Liu et al. (2024) characterised what they called the "lost in the middle" effect: in multi-document question-answering tasks, the retrieval accuracy for facts placed in the middle of a long context degrades by more than thirty percent relative to facts placed at either end of the

context, and this effect is robust across multiple model families and context lengths. The attention distribution is not merely recency-biased; it is U-shaped, with both the very beginning and the very end of the context receiving disproportionate attention, and the middle receiving the least. This has immediate practical consequences for any application that places important information in the middle of a long prompt or conversation.

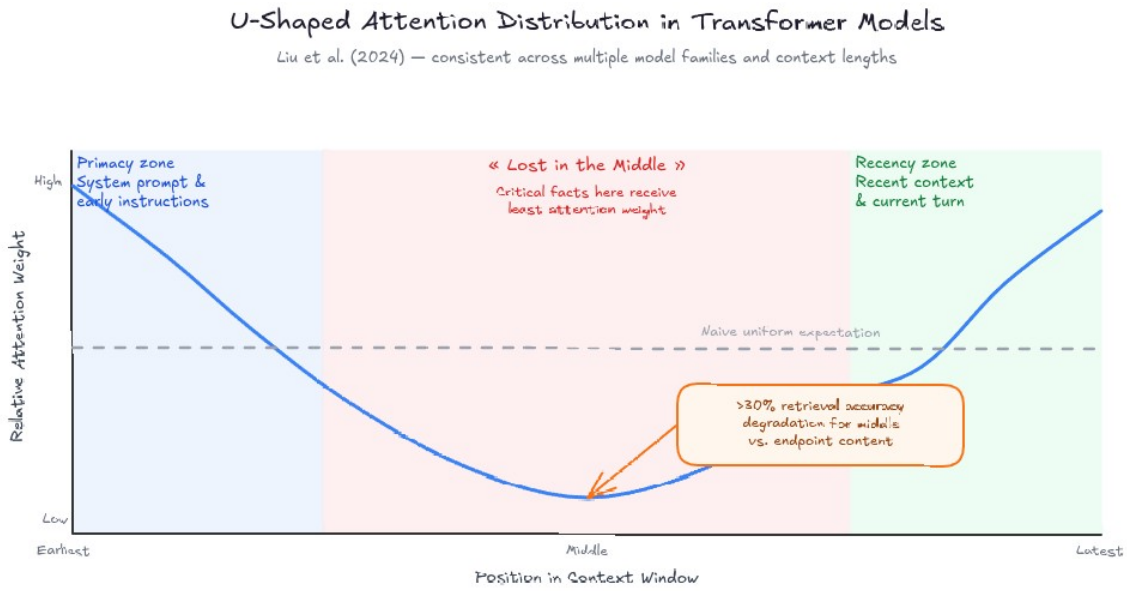


Figure 1.1: U-shaped attention distribution in transformer models.

Both sequence endpoints receive disproportionate attention weight; facts placed in the middle receive the least. Retrieval accuracy degrades by more than 30% for mid-context content (Liu et al., 2024).

Reasoning Traces as Accelerant

A further compounding factor, relevant to models that expose extended reasoning, is the accumulation of thinking tokens. When a model renders its chain of thought explicitly, as a reasoning block prepended to each response, those tokens occupy context budget at the same rate as dialogue tokens but carry no information the user intended to provide. They are a by-product of the model's internal processing made visible. A reasoning-intensive session can accumulate thinking tokens at roughly the same rate as conversational tokens, effectively doubling the rate at which earlier content is displaced toward the attentional periphery.

This creates an asymmetry between the model's experience of the conversation and the user's. From the user's perspective, the conversation has had ten turns. From the model's perspective, in terms of context budget consumed and attention competition created, the effective length is substantially greater. The user who notices that their opening instructions are no longer being followed after ten turns might expect context decay to become significant only after twenty or thirty turns; the thinking tokens have moved the problem forward by a factor the user cannot easily observe.

Reducing the verbosity of reasoning traces through reasoning distillation is therefore a demand-side complement to the supply-side management strategies surveyed in Chapter 3: it slows the rate of context budget consumption rather than improving the management of the budget that remains.

Symptoms

The practical signatures of context decay are recognisable once one knows to look for them. A model given a precise constraint or stylistic instruction at the start of a long conversation will often honour that instruction faithfully for the first several turns, then gradually drift away from it as the conversation grows. The instruction has not been forgotten in any hard sense: if the model is asked directly whether the constraint applies, it may well confirm that it does. Yet the instruction no longer governs behaviour. The distinction between knowing a rule and following it is a well-documented feature of

human cognition under cognitive load; context decay produces an analogous effect in LLMs, though by a different mechanism.

A related symptom is instruction inversion: a later instruction, even one that should be subordinate to an earlier one, effectively overrides it. The model behaves as though recency implies authority. This is a direct consequence of the attention asymmetry. The later instruction occupies a position from which it can exert more influence, and the model's behaviour reflects the weighted balance of attention rather than the intended hierarchy of the instructions.

Repetition is another marker. A model will sometimes ask for information it was already given, or re-derive a conclusion it reached several turns earlier, as if the earlier exchange had not occurred. In long document question-answering settings, the model's account of earlier sections becomes progressively less precise: key qualifications are dropped, specific figures are rounded or elided, and the overall account converges toward the general at the expense of the particular. These are not random errors; they are the systematic result of attention weights being distributed across a longer and longer context, with less and less reaching the early segments where the original precision resided.

Impact on LLM Performance

Context decay has measurable consequences across several classes of high-value LLM use, and the consequences are not uniform across task types.

In agentic and multi-step tasks, a long-running agent that accumulates tool outputs, intermediate results, and self-generated reasoning in its context will progressively lose fidelity to its original objective. Earlier tool results, which may have been critical to a decision, carry less weight than they did when they were generated. The agent can appear to reason coherently at each individual step while drifting substantially from its initial mandate over the course of a session. This is a particularly serious failure mode because the drift is incremental and self-reinforcing: each subsequent step is taken in the context of a slightly corrupted representation of the overall task, and the corruption compounds.

In document question-answering, the position of a fact within the document is not semantically meaningful, yet it strongly influences retrieval reliability. The lost-in-the-middle effect means that facts near the end of a long document are retrieved more reliably than equally salient facts near the beginning. This creates a systematic bias that is invisible to users and difficult to correct through prompt engineering alone. Users who carefully structure documents to front-load the most important information are, unknowingly, placing it in the worst positional context.

There is also a subtler effect on the balance between parametric and in-context knowledge. A model's parametric knowledge, what it learned during training, provides a stable floor of competence. In-context evidence allows the model to go beyond that floor, incorporating specific, up-to-date, or domain-specific facts. Context decay degrades this in-context evidence preferentially, because the parametric knowledge does not decay: it is not subject to attention dynamics at all. The result is that the model's effective knowledge profile shifts back toward its parametric baseline as a conversation lengthens, even when the user has supplied better evidence in the context. In specialist domains, where parametric knowledge is shallow and in-context evidence is carrying the full weight of expertise, this shift is particularly damaging.

Finally, there is a user trust dimension. Users who invest effort in carefully structured prompts and front-loaded instructions reasonably expect those instructions to persist. When they are silently overridden by context decay, the failure mode is opaque: the model does not signal that it has stopped attending to earlier content. This erodes confidence in ways that are disproportionate to the underlying technical cause, because the user has no reliable way to distinguish a model that has been given insufficient information from a model that has been given sufficient information and then quietly lost it.

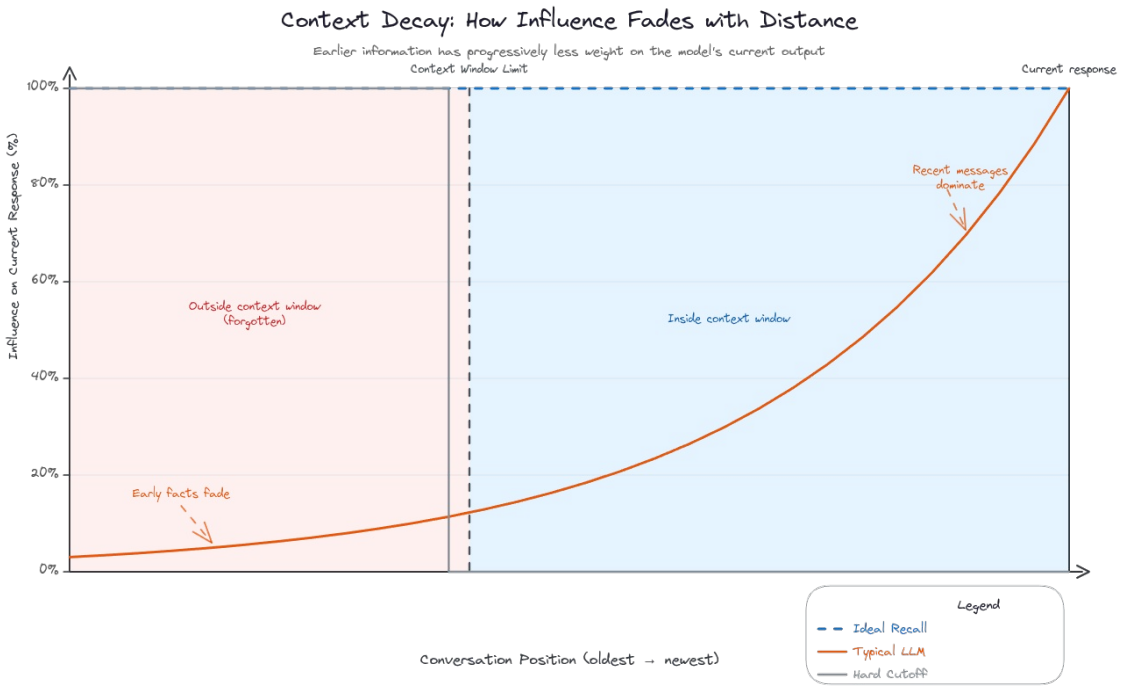
The Full Spectrum

The preceding discussion has treated soft context decay and the hard context window limit as distinct phenomena, which they are mechanistically. But from the perspective of the problem they create for users and systems, they are not distinct at all. They are near and far ends of the same spectrum.

> **The context management spectrum**: a continuum running from the gradual degradation of influence for tokens that remain present in the window at the near end, to the abrupt inaccessibility of tokens that exceed the maximum context length at the far end. Both are expressions of the same underlying constraint: context is finite, and content competes for its limited capacity.

Understanding them as a spectrum changes the design question. The question is no longer how to mitigate decay within the window alongside a separate question of how to handle overflow at the limit. The question becomes: what is the correct general policy for managing a finite, high-value resource across its full range of stress conditions? The near end of the spectrum calls for selective retention of high-value content over low-value content. The far end calls for the same thing, more urgently. A policy that handles both as instances of the same problem is more coherent and more generalisable than a policy that treats them as separate concerns to be addressed by separate mechanisms.

This framing has a further consequence. Naive approaches to context management all fail for the same reason: they treat context as a queue rather than as a value-ranked resource. Truncation discards the oldest content, which is not the least valuable content. Rolling windows keep the most recent N tokens, which is a marginally better heuristic but rests on the same assumption that recency implies importance. Uniform compression reduces all content proportionally, which preserves noise alongside signal. The consistent failure of these approaches is not incidental; it reflects the shared wrong assumption. Chapter 5 proposes a scoring architecture that abandons that assumption and manages context by value rather than by position.



A Second Temporal Dimension

Context decay as described above is a one-dimensional phenomenon: the positional age of a segment, measured from the current generation point backwards along the context sequence. The recency signal that arises naturally from the mechanism decays content based on how far back it was entered into the context. This is accurate but incomplete.

Each segment in a conversation carries not one but two temporal attributes. The first is its transaction time: the turn in which it entered the context. The second is its valid time: the real-world interval during which the information the segment contains is actually true. For a large class of conversational content, these two attributes are coupled tightly enough that treating them as one introduces no significant error. An observation about the current state of a variable, made now, describes something that is true now. Transaction time and valid time move together.

But for a structurally important class of statements, the gap between them is consequential. A forward commitment made at turn three, a decision about how the conversation will proceed, a standing constraint on the model's behaviour, has a

transaction time of turn three and a valid time that extends forward, potentially for the entire remaining duration of the conversation. Decaying it by positional age is epistemically incorrect. The commitment has not aged. A retroactive correction made at turn twenty-two does not merely supersede what came before; it closes the valid-time interval of the original statement and opens a new one for the correction. A description of a superseded configuration, the old API endpoint, the prior architecture, has a closed valid time regardless of how recently it was entered into the context.

A recency signal that cannot distinguish these cases will systematically evict forward commitments that should never be evicted, retain original incorrect statements alongside their corrections at comparable scores, and apply the same decay pressure to open current-state facts and closed historical facts. These are not edge cases. They are common in any conversation that spans decisions, corrections, and references to prior states, which is to say any substantive conversation. The bitemporal extension developed in Chapter 8 addresses them directly. It is introduced here because understanding that context decay has a second temporal dimension alongside its positional dimension is part of a complete account of the problem.

The Knowledge Dimension

Context decay does not interact uniformly with all types of knowledge, and this non-uniformity is itself part of the problem. Chapter 2 develops a three-axis taxonomy of knowledge, but one aspect of the interaction is worth noting here as a consequence of the mechanism just described.

A model with deep parametric knowledge of a domain can compensate, at least partially, for context decay in that domain: when in-context evidence fades or is displaced, the parametric knowledge supplies the missing specialist understanding. A model with only shallow parametric coverage has no such fallback. When in-context evidence degrades, performance degrades with it, and does so sharply, because the in-context evidence was the only source of specialist competence in the first place.

This means context decay is not a uniform tax levied on all LLM performance. It is a regressive tax that falls most heavily on precisely the applications where LLMs are asked

to do the most: the specialist domains, the novel tasks, the problems where the model's training provides only a general foundation and the in-context evidence must supply the domain-specific depth. Understanding the structure of that interaction, the axes along which knowledge varies and the ways each axis modulates susceptibility to decay, is the task of the chapter that follows.

Bibliography

Baddeley, A. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences*, 4(11), 417–423.

Brown, T., Suhr, E., Callison-Burch, C., Eisner, J., & Ferguson, S. (2024). Round and round we go! What makes Rotary Positional Encodings useful? *Proceedings of the International Conference on Learning Representations (ICLR 2025)*. arXiv:2410.06205.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Hopkins, M., Luck, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 157–173.

Press, O., Smith, N. A., & Lewis, M. (2022). Train short, test long: Attention with linear biases enables input length extrapolation. *Proceedings of the International Conference on Learning Representations (ICLR 2022)*. arXiv:2108.12409.

Su, J., Lu, Y., Pan, S., Wen, B., & Liu, Y. (2021). RoFormer: Enhanced transformer with rotary position embedding. arXiv:2104.09864. Published in *Neurocomputing*, 568 (2024).

McGeoch, J. A. (1932). Forgetting and the law of disuse. *Psychological Review*, 39(4), 352–370.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Chapter 2: A Taxonomy of Knowledge

The central claim of Chapter 1 was that context decay is not a uniform tax on LLM performance but a regressive one, falling most heavily on specialist domains where in-context evidence must carry the full weight of expertise. That claim requires a taxonomy. It is not sufficient to say that some knowledge is more susceptible to context decay than other knowledge; the question is what dimensions of knowledge determine that susceptibility, and how those dimensions interact. This chapter develops a three-axis taxonomy that will serve as the analytical frame throughout the rest of the dissertation. The first two axes, depth and breadth, are established concepts in the study of expertise. The third axis, temporal validity extent, is introduced here as a necessary addition to those two, one that reveals a class of context decay failure that neither depth nor breadth can account for. Together, the three axes map out the knowledge landscape in sufficient detail to make precise claims about what any context management strategy is trying to preserve, and at what cost.

Two Axes of Knowledge

Knowledge can be characterised along two primary axes that are conceptually distinct but practically intertwined.

> **Vertical knowledge (depth)**: expertise within a specific domain, comprising the ability to reason at an expert level, handle edge cases, apply nuanced judgement, and understand the internal structure of sub-domains and their adjacencies.
>
> **Horizontal knowledge (breadth)**: coverage across many domains, comprising the ability to connect ideas across fields, recognise when a problem spans multiple specialisms, and provide generally useful responses even in the absence of deep specialist detail.

These are not simply opposite ends of one spectrum. A practitioner does not locate themselves at a point on a single axis running from narrow specialist to broad generalist. The axes are genuinely orthogonal: it is possible, in principle, to have both deep expertise

and broad coverage, or neither, or one without the other. In practice, the constraints of finite learning time mean that most practitioners trade some breadth for depth, or vice versa. But the trade-off is not linear and it is not symmetric, and understanding why requires looking closely at what deep expertise actually produces.

The Medical Analogy

Medicine provides the clearest illustration of this taxonomy because the division of medical knowledge into specialist and generalist forms is institutionalised and well-studied.

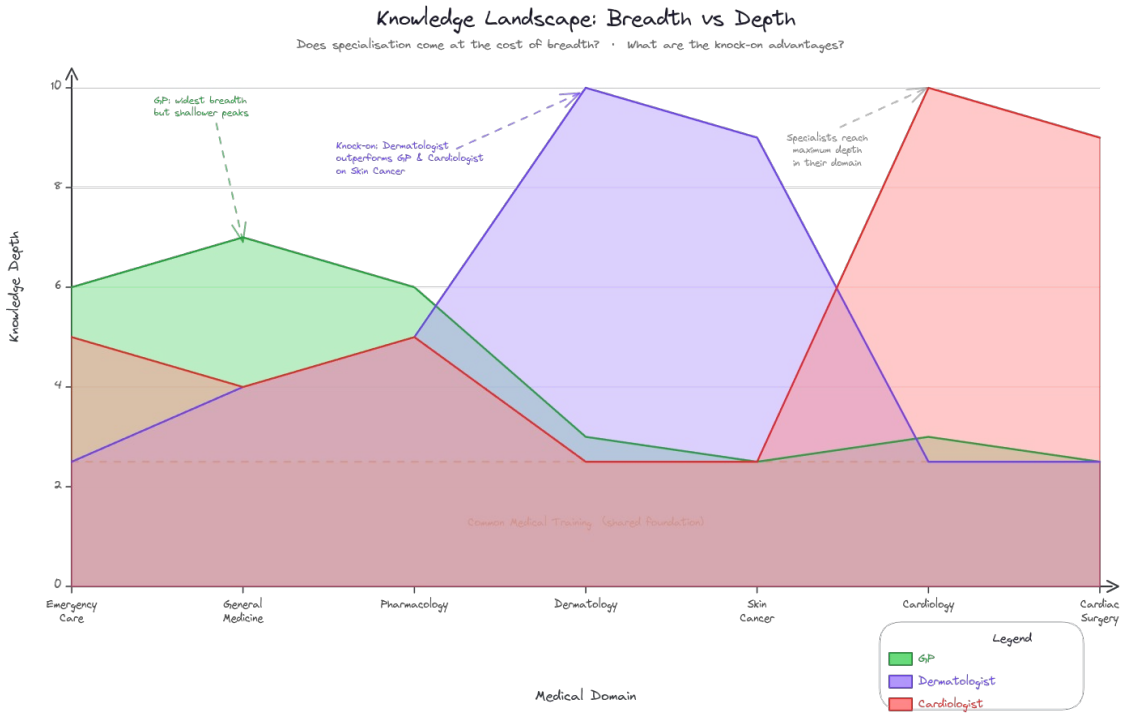
A general practitioner has broad horizontal knowledge. They can triage a cardiac event, identify a suspicious mole, manage a diabetic patient's medication, advise on public health measures, and recognise the early signs of dozens of conditions they will then refer onward to specialists. Their training spans emergency care, general medicine, pharmacology, dermatology, cardiology, and much else besides. What they do not have, in most of these areas, is the vertical depth of a specialist. They know enough to recognise, to triage, and to refer. They do not know enough to stage a melanoma, to interpret a complex echocardiogram without support, or to manage a patient on a novel anticoagulation regimen without consulting the relevant guidelines.

A dermatologist has spent years accumulating deep knowledge of skin conditions: the subtleties of lesion morphology, the differential diagnoses for a pigmented lesion, the staging criteria for melanoma, the interplay between skin disease and systemic illness, the pharmacological management of chronic inflammatory conditions of the skin. Their breadth is narrower in total footprint than the GP's. But their depth within dermatology, and critically in the domains immediately adjacent to it, is considerably greater. They will encounter oncological questions in virtually every day of their clinical work, because many skin conditions are malignant or pre-malignant. That exposure produces real oncological competence as a side-effect of dermatological expertise.

A cardiologist presents a third configuration. Their depth is concentrated in cardiovascular medicine, but the adjacencies are different from the dermatologist's. Managing patients on complex anticoagulation regimens requires pharmacological

knowledge around anticoagulant drugs, their mechanisms, their dosing, their reversal in emergency situations, and their interactions with other common medications, that goes well beyond what a general practitioner routinely encounters. The cardiologist acquires this knowledge not through formal pharmacology study but through continuous clinical engagement with a patient population for whom it is unavoidable.

This is the pattern that the taxonomy is designed to capture: depth in one domain does not merely represent concentrated knowledge at the expense of breadth elsewhere. It tends to pull competence in adjacent domains inward, generating what this dissertation calls a knock-on advantage.



The Knock-On Advantage

> **Knock-on advantage**: the emergent competence in an adjacent domain that arises as a side-effect of deep expertise in a primary domain, where the two domains share underlying mechanisms, vocabulary, or structural patterns such that mastery of one makes working knowledge of the other unavoidable.

The knock-on advantage is not the same as the incidental accumulation of facts. A GP who has seen many oncology referrals has more exposure to oncological cases than a dermatologist who has never referred a patient for cancer staging. But that exposure does not produce the dermatologist's depth of oncological understanding, because the GP's engagement with oncology is episodic and varied rather than sustained and structurally driven by a core specialism. The knock-on advantage flows specifically and predictably from shared underlying structure between domains. It follows the architecture of knowledge, not the distribution of experience.

The same logic applies outside medicine. A compiler engineer, working at the boundary between high-level language semantics and machine code, must understand computer architecture at a depth that a web developer never encounters: instruction pipelines, cache behaviour, register allocation, memory models. That understanding is not optional for the compiler engineer; it is structurally required by the work. A structural engineer, reasoning about load paths and failure modes in real materials, develops a working knowledge of materials science that a mechanical engineer focused on dynamics and kinematics does not typically acquire. In each case, the knock-on advantage flows downward through related domains. It is directional, not random.

This matters for the present analysis because it means that depth and breadth do not trade off in a simple zero-sum way. A practitioner with genuine depth in a domain has, by virtue of that depth, acquired breadth in the adjacent domains that the deep domain structurally requires. The GP's broad coverage is genuinely broad but genuinely shallow across most of its range. The specialist's coverage is narrower in total footprint but contains pockets of unexpected depth at the borders. The two knowledge profiles are not points on the same axis; they are genuinely different shapes in a two-dimensional space.

How This Maps to Large Language Models

Large language models have an analogous two-source structure for knowledge that maps directly onto the depth-breadth taxonomy.

> **Parametric knowledge**: the knowledge encoded in the model's weights during training. This is the model's equivalent of education and

professional experience. It is fixed at inference time, cannot be updated by what happens in the context window, and constitutes the model's standing competence across all domains it encountered during training.

In-context knowledge: the knowledge present in the active context window at inference time. This includes instructions, documents, examples, prior turns of conversation, and any other information the user or application has supplied at runtime. It is dynamic and specific to the current session, but it is subject to context decay.

The interaction between these two knowledge sources and the depth-breadth taxonomy produces the central empirical claim about context decay's differential impact. A model with deep parametric knowledge of a domain can compensate, at least partially, for the degradation of in-context evidence in that domain. When a specialist fact provided in early context fades from effective attention, the model's parametric representation of the domain may supply a reasonable approximation. The compensation is imperfect, because the in-context fact may be specific, current, or idiosyncratic in ways that parametric knowledge cannot replicate. But the floor does not collapse. A model with only shallow parametric coverage in a domain has no such fallback. When in-context evidence degrades in a shallow-parametric domain, performance degrades with it, abruptly, because the in-context evidence was the primary source of competence.

This means context decay is most damaging precisely where LLMs are asked to do the most difficult work: in specialist domains where the model's training provides a general foundation and the user's in-context evidence must supply the domain-specific depth. The knock-on advantage matters here too. A model whose parametric training is genuinely deep in a domain will be more robust to context decay in that domain's adjacencies than a model with only surface coverage of the same adjacencies, because the deep parametric structure provides a richer substrate for interpreting degraded in-context evidence.

A Third Axis: Temporal Validity Extent

The depth-breadth taxonomy is necessary but not sufficient. It characterises knowledge by how much of it a practitioner holds and how deeply they hold it. It says nothing about how long a piece of knowledge remains valid.

> **Temporal validity extent**: the duration over which a piece of knowledge holds true, ranging from permanently unbounded (standing constraints, mathematical relationships, committed decisions) through currently open (present-state facts that age as circumstances change) to historically closed (descriptions of superseded states whose valid interval has ended).

This axis is orthogonal to both depth and breadth. A claim can be narrow and deep, a precise drug dosage for a specific condition, while being highly time-sensitive: clinical guidelines change, new contraindications are identified, formulations are revised. A claim can be broad and shallow, a general observation about a technology category, while being temporally unbounded: it describes a property so fundamental that no reasonable update to the field would render it false. The temporal validity of a claim is a property of the claim's relationship to the world, not of the practitioner's depth or breadth of knowledge about its domain.

The axis has three qualitatively distinct regions. Standing knowledge is unbounded in both directions or open-ended forward from its establishment: it holds from when it was first asserted into the indefinite future. The system prompt that defines an assistant's behaviour, the architectural decision made at turn five of a long design conversation, the mathematical identity used as a foundation for a proof, all have this character. Current-state knowledge is valid now but will age: the current configuration of a system, the current position in a negotiation, the current state of a variable. It is accurate at the time of assertion and becomes progressively less accurate as circumstances change. Historical knowledge describes a state that has already ended: the previous API endpoint, the configuration before the refactor, the assumption the project was working from before the correction at turn twenty-two. Its valid interval is a closed segment in the past.

These distinctions matter for context decay in a specific and important way. Standing knowledge is the most costly to lose. If a standing constraint established at turn three is evicted from the active context, it cannot be recovered from parametric sources, because it was specific to this conversation. It cannot be recovered from an archive in a general sense, because its relevance is continuous, not triggered by a subsequent query. It was load-bearing from the moment it was established, and it remains load-bearing permanently. Yet a recency signal that decays all content uniformly by positional age penalises standing knowledge most severely, because it was established earliest and therefore has accumulated the most positional age. This is the inversion at the heart of naive context management: the policy that punishes age is most destructive to the content that is least affected by age.

Current-state knowledge is appropriately subject to recency pressure. Its valid time tracks the present, and the present moves forward. A current-state fact established in turn five is still accurate in turn six, probably accurate in turn ten, and of diminishing reliability in turn thirty if circumstances may have changed. The recency signal approximates this correctly for the common case.

Historical knowledge should be aggressively archived, not based on its positional age but on the closure of its valid interval. A description of a superseded state made at turn twenty, immediately after a correction that rendered it obsolete, should be archived immediately, before it has had time to age at all. A recency signal applied to positional age will not archive it promptly; it will retain it at a high score because it was recently entered into the context. The temporal validity axis, and specifically the detection of valid-interval closure, is the correct signal for this case.

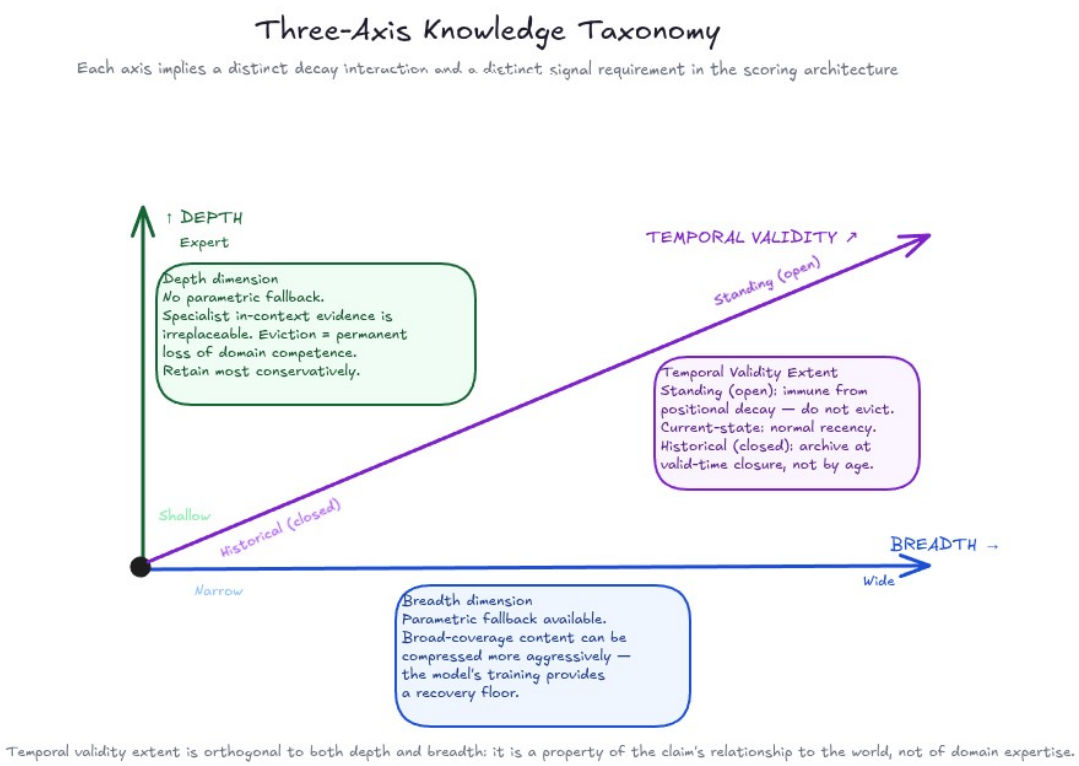


Figure 2.1: The three-axis knowledge taxonomy. Breadth (horizontal coverage), Depth (specialist vertical), and Temporal Validity Extent are mutually orthogonal dimensions. Context decay interacts with each axis differently. The existing two-axis knowledge landscape (breadth × depth) is the projection visible from above; the third axis extends it into the page.

Three Axes, Three Distinct Decay Interactions

The three axes each produce a distinct interaction with context decay, and each implies a distinct requirement for any context management strategy.

Breadth interacts with context decay through the parametric fallback. Broad, shallow knowledge that resides in the model's parametric training is partially recoverable when in-context evidence degrades. The recovery is approximate and potentially outdated, but the floor does not collapse entirely. Context management strategies for breadth-domain content can therefore afford to be somewhat more aggressive in compression: the parametric knowledge provides a safety net that the compressed content can lean on.

Depth interacts with context decay through the absence of that fallback. Deep, specialist knowledge that is specific to the current domain, the current patient, the current codebase, the current negotiation, has no parametric representation. It exists only as in-context evidence. When it degrades, it is simply gone, and no amount of parametric competence can reconstruct the specific detail that was lost. Context management strategies for depth-domain content must be substantially more conservative: the cost of eviction is the permanent loss of irreplaceable specialist precision.

Temporal validity extent interacts with context decay through the dimension that neither depth nor breadth addresses. It does not matter whether a standing constraint is broad or deep, shallow or specialist, in terms of its susceptibility to positional decay. What matters is its temporal structure: it was established early, and it holds indefinitely. The correct retention policy for standing knowledge is immunity from positional decay, not conservative retention. Preserving it at a higher score than other content is insufficient if all content is decaying from the same positional baseline; what is required is a scoring signal that does not decay at all for content with open valid time. The bitemporal extension developed in Chapter 8 provides exactly this.

The three axes therefore generate three distinct requirements for the scoring architecture in Chapter 5: a relevance signal that can identify specialist depth and retain it preferentially, a currency signal that can distinguish stable from volatile content, and a temporal validity signal that can identify standing knowledge and exempt it from positional decay entirely.

The Taxonomy as an Analytical Frame

The taxonomy developed in this chapter will be used throughout the dissertation to evaluate the contributions proposed in Chapters 5 through 8. Each contribution can be assessed in terms of which axis or axes it addresses and which failure modes it corrects.

The scoring proxy (Chapter 5) addresses all three axes to varying degrees: the relevance signal targets depth, the currency signal targets temporal volatility within the current-state region of the third axis, and the primacy signal partially corrects for the positional disadvantage of content established early. The temporal validity signal, introduced in

Chapter 8, completes the architecture by addressing the standing-knowledge failure mode that the other signals cannot reach.

Dissolution Through Consensus (Chapter 6) addresses the temporal validity axis in a structural rather than a scoring way: by dissolving the linear history into a boundary state, it extracts the standing knowledge, the DECIDED and IDENTITY categories of the boundary representation, and relocates it in a form that is position-independent by construction. The boundary state does not decay positionally because it does not have a position; it replaces the sequence rather than occupying a place within it.

The Conversation Intelligence Architecture (Chapter 7) addresses all three axes simultaneously through its four levels: surprise allocation targets information density (depth of new information relative to existing context), navigational topology captures the shape of established knowledge regardless of its depth-breadth profile, generative competence evaluates retention quality domain-independently, and the adversarial frame agent targets the temporal validity axis in its most abstract form by auditing which premises the conversation has committed to permanently.

The chapter that follows examines what the existing technical literature has proposed in response to context decay, and evaluates those proposals against the three-axis frame established here.

Bibliography

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Hopkins, M., Luck, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 157–173.

Tulving, E., & Thomson, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5), 352–373.

Chapter 3: The State of the Art

The problem established in Chapter 1 and the taxonomy developed in Chapter 2 provide the frame for a critical evaluation of what has already been attempted. The technical literature has produced a substantial body of approaches to context decay mitigation, and several of them are both well-validated and genuinely useful. This chapter surveys the principal six approaches, evaluates each against the three-axis taxonomy of depth, breadth, and temporal validity extent, and identifies the failure mode specific to each. The critical purpose of the survey is not to diminish what these approaches achieve but to identify what they leave unaddressed. The argument that emerges is this: every existing approach, however sophisticated in its domain, operates inside the same framing assumption, that the goal of context management is to preserve the content of a conversation. That assumption constrains what the approaches can achieve, and the constraint is not accidental. It is constitutive of the entire preservation frame. Chapter 6 of this dissertation proposes an alternative frame, and the present chapter establishes why the alternative is needed.

Retrieval-Augmented Generation

The most widely deployed approach to the context decay problem is retrieval-augmented generation, which addresses the problem by repositioning knowledge rather than preserving it. Instead of injecting specialist material into context at session start, where it will decay as the conversation grows, RAG retrieves relevant fragments at query time and injects them close to the generation point. The positional disadvantage of early injection is eliminated by construction: retrieved content is always near the current turn.

A production RAG architecture for large corpora uses a hierarchical chunking strategy at three levels of granularity: chapters for structural context, sections for topic-level retrieval, and paragraphs for precision. Query processing proceeds in two stages, coarse semantic retrieval at the section level followed by fine retrieval within the top-scoring sections, with a cross-encoder re-ranking pass and a hybrid BM25 and semantic score to prevent purely semantic retrieval from returning topically irrelevant matches. The effect is that specialist content, regardless of how large the underlying knowledge base, is

delivered at precisely the moment it is needed and at the position in context where it will receive maximum attention.

Against the three-axis taxonomy, RAG performs well on the depth axis when the knowledge base is organised by specialism. A domain-specific collection, a cardiology corpus, a legal case database, a codebase, gives the model access to full vertical depth on demand, and the retrieval mechanism means that depth can be queried rather than held in context. Breadth performance depends on collection organisation: a well-indexed multi-domain corpus retrieves broadly as well as deeply, though the precision-recall trade-off at the retrieval threshold behaves differently for shallow breadth queries than for narrow depth queries.

The temporal validity axis is where RAG is structurally blind. A retrieved document does not carry its valid time in a form the retrieval mechanism can act on. A clinical guideline updated last quarter sits in the same index as one published five years ago; without explicit temporal metadata and time-aware retrieval, the system may return the outdated document with equal or higher confidence. Semantic similarity to the query is not a proxy for temporal currency. More fundamentally, RAG addresses the external knowledge base, not the conversation itself. The forward commitments made at turn five, the decisions reached at turn twelve, the corrections issued at turn twenty, these are conversation-specific standing knowledge that no external knowledge base can hold because they were never written down anywhere other than the context window. RAG repositions knowledge from outside the conversation; it has no mechanism for managing the knowledge created inside it.

Semantic Caching and Knowledge Currency

Semantic caching inserts a layer between the user query interface and the inference engine. Where RAG retrieves knowledge from an external store at query time, semantic caching stores prior answers and returns them when a new query is sufficiently similar. A four-component architecture, query fingerprinting, dual-tier cache storage, adaptive time-to-live, and knowledge currency weighting, implements this in production. The fingerprint combines an embedding of the query with a hash of the recent conversational

context, ensuring that semantically identical queries produce different fingerprints when asked in different conversational contexts. The adaptive time-to-live applies a domain-specific decay to cache entries: a response about drug dosage recommendations expires faster than a response about anatomical structure.

The knowledge currency component, KDecay, is theoretically the most interesting feature of this approach. It applies a temporal discount to cached knowledge in the same way that context decay applies a positional discount to in-window knowledge. The two are symmetrical problems with different causes, and KDecay is an explicit acknowledgement that temporal validity matters. It is therefore the closest that any existing approach comes to addressing the third axis of the taxonomy. The limitation is that KDecay models content-level volatility, how fast this type of information typically expires, rather than structural valid-time, whether this specific segment has a forward-open or historically-closed valid interval. Drug dosage recommendations are assumed to expire faster than anatomical facts because the former are empirically more volatile as a class. But a specific drug dosage confirmed in the current conversation at turn five is a forward commitment: it governs what the conversation does from that point forward regardless of the external volatility of the content type. KDecay does not distinguish between the external volatility of a knowledge class and the structural valid-time of a specific segment in a specific conversation.

Against the breadth axis, semantic caching is most effective in high-traffic deployments where many users ask similar questions. The hit rate for deep specialist queries is structurally lower because specialist questions are more varied and more context-dependent. The strength of KDecay-weighted time-to-live is precisely most valuable for the content where the hit rate is lowest, vertical knowledge where staleness carries the greatest risk. This is not a contradiction; it reflects the different ways each axis interacts with the caching mechanism. Semantic caching does not extend the survivability of any specific piece of in-context knowledge. It provides fresh retrieval for repeated queries, which is a different problem.

Multi-LLM Ensemble Orchestration

A single model has one knowledge profile, shaped by its training distribution. An ensemble of models with heterogeneous training corpora and architectures has overlapping but non-identical profiles, and combining their outputs can compensate for the blind spots of any individual model. The Cognitive Heterogeneity Amplification theorem formalises this: ensemble error decreases with model heterogeneity, bounded by the degree to which model errors are statistically independent. A three-model ensemble with a heterogeneity measure of 0.81 achieved an accuracy improvement from 73.1% to 95.6% on structured tasks in a validated production deployment.

Three collaboration patterns are available. Parallel voting queries all models simultaneously and combines responses by statistical aggregation. Sequential refinement passes a draft through a chain of models each adding a distinct operation, such as generation, critique, and quality assurance. Debate protocol has models respond to each other's outputs across multiple rounds, converging on positions that withstand adversarial challenge. The choice of pattern depends on the query type and the available latency budget.

Ensemble methods address a dimension of the problem that is genuinely distinct from positional decay: parametric knowledge gaps and model-specific biases. If one model attends poorly to early context, another may have independently retained it, and voting recovers the answer. But the approach has a structural limitation against the depth axis. The CHA theorem's heterogeneity benefit is highest when models cover different domains. For deep specialist questions in a narrow vertical domain, an ensemble of generalist models may converge confidently on the same wrong answer, because the heterogeneity they provide is broad rather than deep. Routing the query to a domain-specialist model partially addresses this, but only if such a model exists in the ensemble and only if the routing logic correctly identifies queries that require it.

Against the temporal validity axis, ensemble methods are no better equipped than single models. Voting across multiple models' outputs does not introduce any representation of valid time. If the question being voted on concerns a forward commitment made earlier in the conversation, each model in the ensemble is equally subject to the positional decay

that may have already eroded that commitment from effective attention. Distributing the decay across multiple models does not eliminate the decay; it averages it.

External Knowledge Stores

The most architecturally radical approach to positional decay is to remove the relevant knowledge from the context window entirely. An external columnar knowledge store, in a validated implementation using DuckDB for in-process analytical queries, holds structured business data outside any model's context. The model does not see the raw data; it emits queries, receives result sets, and reasons over them. The context window contains the current query, the result set, and the conversation thread. The underlying data does not occupy a context position and therefore does not decay.

This is not decay mitigation but decay bypass. It is a genuine solution for structured, queryable data: operational metrics, tabular records, relational business data. Against the depth and breadth axes it is effectively domain-neutral, equally applicable to deep specialist tables and broad analytical views. The architectural separation of reasoning layer from data layer is clean, testable, and scales to arbitrarily large knowledge bases without expanding the context window.

The failure mode is scope. External stores handle structured, tabular knowledge naturally. Narrative, procedural, and contextual knowledge, which is to say most of the knowledge that makes a conversation valuable, does not fit a tabular model. A medical knowledge base is not a table. A conversation's accumulated understanding of a specific patient's history, the decisions made over twenty turns about their treatment plan, the corrections issued when initial assumptions turned out to be wrong, none of this is representable as rows and columns. External stores eliminate positional decay for the data they can hold, and for that data the solution is essentially complete. But the conversation-specific standing knowledge that is most critical to preserve has no home in an external store, because it was generated by the conversation itself.

The temporal validity axis also receives no treatment. A DuckDB table with a six-hour refresh cycle applies uniform temporal currency to all its contents. A fact that was true at

the time of the last refresh is equally available whether its valid time closed two years ago or extends indefinitely forward.

Prompt Externalisation and Re-injection

A practical and underappreciated approach targets a specific and common failure mode: the instruction injected at session start that decays to ineffectiveness as the conversation grows. Prompt externalisation moves the instruction out of the application's source code and into a managed, versioned store from which it can be retrieved and re-injected at the point in the conversation where it is most needed.

A production implementation stores a library of domain-specific prompts in a relational database with version history and regression testing infrastructure. The critical mechanism is re-injection: rather than relying on a prompt placed once at session start to persist across fifty turns of conversation, the system retrieves the prompt and injects it afresh immediately before each query it governs. The instruction is always at a recent position in context, where it will receive maximum attention. Prompt drift, the gradual erosion of a system prompt's effective influence as the conversation grows, is eliminated for the prompts that are managed this way.

This approach is directly relevant to the third axis of the taxonomy. Instructions and constraints are standing knowledge by definition: they have forward-open valid time and should never be subject to positional decay. Re-injection implements this by construction, not by inference. The limitation is coverage. A managed library of eighty-nine prompts addresses the known, recurring instructions in a specific application. The forward commitments generated dynamically in the course of a specific conversation, the decision made at turn five that was not anticipated when the library was built, are not reinjectable because they were never stored in the library. Prompt externalisation handles standing knowledge that was known before the conversation started. It cannot handle standing knowledge that the conversation itself creates.

Against the depth axis, prompt externalisation is one of the most effective available approaches: specialist domain knowledge encoded by domain experts into well-crafted prompts can be kept continuously fresh. Against breadth, the combinatorial explosion of

prompt variants for a wide-domain deployment creates a routing and maintenance problem at scale.

Distributed Reasoning and Shared External Memory

The blackboard pattern distributes reasoning across multiple agents that share a common memory substrate. Each agent has a compact individual context, and results are written immediately to an external shared store where all other agents can read them. No single agent's context window accumulates the full burden of a long conversation; the burden is distributed, and the shared store provides indefinite persistence without context cost.

In a validated multi-LLM implementation, three models sharing a blackboard with task routing achieved a 23.1% accuracy improvement over a single-model baseline. The routing logic assigns subtasks to the model best suited for them: specialist questions to domain-capable models, structured output to models with strong formatting performance, cost-sensitive tasks to the smallest capable model. Parallelisation is natural: independent subtasks run concurrently.

Against all three axes of the taxonomy, the blackboard architecture is the most flexible approach surveyed. Deep specialist tasks can be routed to specialist models when they exist. Broad horizontal queries can be handled by generalist models. The shared external memory provides persistent storage that does not decay. The limiting factor on the depth axis is model availability: the routing benefit is real only if a genuinely specialist model exists to route to. The limiting factor on the temporal validity axis is the same structural blindness as all other approaches: the blackboard stores content, and the content it stores carries no representation of valid time. A forward commitment written to the blackboard at one turn is as available at turn forty as at turn six; the architecture provides the persistence. What it does not provide is any policy for treating forward commitments differently from historical statements or current-state observations. That distinction requires a temporal validity layer that no blackboard implementation has.

Adjacent Developments

Two recent developments in the broader context management landscape are relevant to this survey's scope and deserve brief treatment.

Anthropic's AutoDream mechanism, implemented in Claude Code as a background memory consolidation system, operates at a different timescale from all the approaches surveyed above. Where those approaches manage context during a conversation, AutoDream operates between sessions. It activates when two conditions are jointly satisfied: at least twenty-four hours have elapsed since the last consolidation, and at least five sessions have occurred since then. The dual-gate prevents unnecessary runs on light-usage projects while ensuring active development receives regular maintenance. Once triggered, AutoDream proceeds through four phases: scanning the memory directory to understand the current organisational structure; gathering signal from session transcripts using targeted searches for corrections, recurring themes, and explicit decisions rather than exhaustive transcript reading; performing consolidation, which converts relative dates to absolute, removes contradicted facts, prunes stale debugging notes, and merges overlapping entries; and finally updating the index file while enforcing a two-hundred-line limit that matches the startup context loading threshold. The result is that memory files presented to the model at session start are compact, consistent, and current, leaving more of the startup context budget available for actual work.

The architectural relationship between AutoDream and within-session context management is complementary rather than competing. AutoDream addresses between-session continuity: facts that need to persist across multiple working days. Within-session decay management addresses the progressive attenuation of influence for content within a single conversation. A system that implements both covers two distinct timescales of the same underlying problem. The consolidation operations AutoDream performs between sessions, removing contradictions, pruning obsolete content, promoting recurring themes, are structurally analogous to the interference detection, compression, and consolidation mechanisms applied within a session in the proxy architecture described in Chapter 5. Neither mechanism has any representation of valid time: AutoDream consolidates based on recurrence and contradiction, not on whether a piece of knowledge has a forward-open or historically-closed valid interval.

VectorlessRAG explores a different point on the retrieval trade-off space. Rather than using embedding similarity for retrieval, it traverses a graph structure built from explicit

relationships between knowledge units, avoiding the infrastructure cost and latency of embedding computation and vector database queries. The trade-off is coverage precision against infrastructure simplicity. Both RAG and VectorlessRAG operate inside the preservation frame and share the same temporal validity blindness; the architectural difference is a practical consideration rather than a conceptual one.

The Preservation Frame

The approaches surveyed in this chapter are heterogeneous in their mechanisms and their domains of applicability. RAG repositions knowledge at query time. Semantic caching eliminates the cost of repetition. Ensemble methods reduce the impact of single-model parametric gaps. External stores bypass positional decay for structured data. Prompt externalisation keeps instructions fresh. Distributed reasoning shares the decay burden across multiple agents.

What all of them share, without exception, is the assumption that the goal of context management is to preserve the content of the conversation. RAG preserves it by repositioning. Caching preserves it by re-serving cached versions. Ensemble methods preserve accuracy by triangulating across multiple parametric representations. External stores preserve data by externalising it. Prompt management preserves instructions by re-injecting them. Distributed reasoning preserves the full content by writing it to a shared store. Every technique is a variation on the question: how do we keep more of the right content available at the right time?

This is the correct question if the goal of context management is to maintain access to conversational content. It is the wrong question if the goal is to maintain the epistemic structure that the conversational content established. The distinction is not merely semantic. A conversation that has run for forty turns has produced not only a sequence of tokens but a shared understanding: propositions established as true, decisions committed to, questions identified as open, constraints accepted as governing. The content is the medium through which that understanding was constructed. The understanding itself is more compact, more durable, and more generative than any compressed version of the content that created it.

No approach in the current literature asks: what is the shape of the understanding this conversation has established? None of them attempts to represent that shape directly rather than through the content that implies it. The evaluation against the three axes of the taxonomy confirms this. The temporal validity axis, which requires distinguishing standing knowledge from current-state knowledge from historical knowledge, receives at best a partial treatment from KDecay and prompt re-injection. The navigation topology of the conceptual space the conversation has mapped receives no treatment anywhere. The question of whether a compressed representation is sufficient to generate correct responses to queries from the conversation's domain, the generative competence criterion, is not used as an evaluation metric by any existing approach.

The approaches surveyed here are not wrong; they are necessary components of any practical context management system. But they are not sufficient, and the insufficiency is not one that can be remedied by combining them more cleverly within the preservation frame. Chapter 4 examines the cognitive science of human memory and establishes the theoretical grounding for an architecture that goes beyond those components. Chapter 5 shows how that architecture is implemented. Chapter 6 proposes the alternative frame.

Bibliography

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Baddeley, A. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences*, 4(11), 417–423.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Ebbinghaus, H. (1885). *Über das Gedächtnis: Untersuchungen zur experimentellen Psychologie*. Duncker & Humblot. [Trans. Memory: A contribution to experimental psychology, 1913.]

McGeoch, J. A. (1932). Forgetting and the law of disuse. *Psychological Review*, 39(4), 352–370.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Pan, J., Gao, T., Chen, H., & Chen, D. (2024). LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint* arXiv:2403.12968.

Roediger, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science*, 17(3), 249–255.

Rong, S. (2025). Distilling reasoning capabilities into smaller language models. *arXiv preprint* arXiv:2212.08410.

Chapter 4: The Human Memory Analogy

The engineering approaches surveyed in Chapter 3 share a common method of justification: they work, to the degree that they work, because they were tuned to work. Decay rates were chosen because they produced good benchmark scores. Eviction thresholds were set because they preserved enough of the right content in practice. This is not a criticism; empirical calibration is how engineering progresses. But it leaves the design choices without theoretical grounding, which means that when the system encounters a novel situation, there is no principled basis for predicting how it will behave or adjusting the policy. The human brain confronts a version of the context management problem continuously: it operates under strict capacity constraints, must discriminate between information worth retaining and information worth discarding, must retrieve stored material in response to cues that only partially match the original encoding context, and must maintain coherence across exchanges separated by substantial temporal gaps. Cognitive science has studied these mechanisms for over a century. The result is a body of evidence that provides exactly the principled grounding that engineering heuristics lack. This chapter surveys nine models from that literature, maps each to a concrete architectural decision in the context management proxy described in Chapter 5, and closes with a critical evaluation of where the analogy illuminates and where it misleads. The final section identifies source monitoring theory as the cognitive science basis for the bitemporal extension developed in Chapter 8: the human memory system has always tracked two temporal attributes simultaneously, and the failure to do so in LLM context management is a gap the cognitive science literature makes visible.

Storage Architecture

The foundational architecture of human memory was proposed by Atkinson and Shiffrin (1968) in their multi-store model: three structurally distinct systems operating in series. A sensory register of very brief duration feeds a short-term store of limited capacity and intermediate duration, which in turn feeds a long-term store of effectively unlimited capacity. Information transfers from short-term to long-term through rehearsal, a process controlled not by automatic decay but by active cognitive operations including rehearsal

strategies and decision rules. The model's central contribution to the present project is its rehabilitation of the control process: the short-term store is not a passive buffer but a working system equipped with mechanisms that govern the flow of information toward long-term retention.

Baddeley and Hitch (1974) critiqued the unitary short-term store and replaced it with a multi-component working memory system: a central executive responsible for attentional control, a phonological loop for verbal rehearsal, and a visuospatial sketchpad. Baddeley's (2000) revision added the episodic buffer, a limited-capacity temporary store that integrates information from the other components and from long-term memory into coherent episodic representations. The episodic buffer is the locus of conscious awareness and the workspace in which diverse information sources are bound into unified episodes.

Implementation mapping. The four-tier framework at the core of the context management proxy maps directly onto the multi-store architecture. The Verbatim tier corresponds to the short-term store: content in this tier is active, attended to at full fidelity, and present in the context the model uses to generate responses. The Archived tier corresponds to long-term memory: content displaced from the active window but retrievable via semantic cues. The Forgotten tier corresponds to complete trace loss. The Compressed tier has no direct biological equivalent in the original formulation but maps functionally to the gist representations documented in the reconstructive memory literature, preserving semantic content while losing surface detail.

The episodic buffer's selective protection function is operationalised by the `EpisodicBuffer` subsystem. It identifies three classes of conversational episode: confirmation events, in which a user explicitly validates a model output; decision events, in which a consequential commitment is made; and correction events, in which a user corrects a model error. Messages of these types receive unconditional protection from score-based eviction. The functional mapping is imprecise in detail: Baddeley's buffer operates in real time with a capacity of approximately four chunks (Cowan, 2001), while the proxy's buffer operates retrospectively and applies no capacity limit. What the

mapping preserves is the functional role: selective protection of structurally important episodes from the general pressure of capacity-limited eviction.

Temporal Dynamics

Murdock (1962) established the serial position effect: in free recall of a sequentially presented list, recall accuracy is highest for items at the beginning and end, lowest for items in the middle. The recency effect, superior recall for final items, reflects their continued presence in short-term memory at the time of recall. The primacy effect, superior recall for initial items, reflects the greater rehearsal opportunity they received during presentation of subsequent items. Glanzer and Cunitz (1966) provided the key dissociation: an unfilled delay between presentation and recall selectively eliminates the recency effect while leaving the primacy effect intact, confirming that the two effects reflect qualitatively different mechanisms. The recency advantage is fragile and positional; the primacy advantage is durable and rehearsal-based.

Ebbinghaus (1885) established two further temporal findings. His forgetting curve shows that retention loss is greatest immediately after learning and decelerates over time following an approximately exponential function. His spacing effect shows that the same total study time produces better long-term retention when distributed across sessions than when massed in a single period. Roediger and Karpicke (2006) demonstrated the testing effect: retrieval practice produces stronger long-term retention than restudying, even when the restudying group encounters the material more frequently. Each act of successful retrieval strengthens the trace beyond what passive re-exposure achieves.

Implementation mapping. The recency effect is directly operationalised in the `RecencySignal`, which computes `exp(−λ(1 − t))` where t is the segment's normalised position in the window. The exponential form mirrors the approximately exponential recency gradient in Murdock's (1962) free-recall data. The primacy effect demands separate treatment because the recency signal penalises early content by construction: a system prompt at position 0.0 receives the lowest possible recency score. The `PrimacySignal` corrects this asymmetry with an additive boost for segments in the first fifteen percent of the normalised position range that contain instruction keywords,

calibrated to counteract the positional decay accumulated over a ten-turn conversation at the default decay rate.

The spacing and testing effects predict that concepts recurring across multiple conversation turns should be treated as more important than concepts mentioned only once at comparable positions. The `RevisitationSignal` operationalises this: an additive boost proportional to the number of prior turns in which the segment's topic has been referenced, capped to prevent inflation. The connection to the testing effect is substantive: each turn in which the current query is semantically similar to a prior segment constitutes an implicit retrieval event. The `RelevanceSignal`'s cosine similarity score is therefore not merely a measure of current-turn usefulness; every high-scoring relevance computation is analogous to a retrieval attempt that should strengthen the segment's standing in the scoring function.

Semantic Structure

Collins and Quillian (1969) proposed that concepts are stored as nodes in a hierarchical semantic network, with properties stored at the highest applicable level to minimise redundancy. They confirmed the semantic distance effect: verification of hierarchically distant properties takes longer than verification of proximate ones. Collins and Loftus (1975) revised the model by replacing the strict hierarchy with a weighted network in which edge weights represent degree of semantic association. Activation spreads from an activated node outward along weighted edges, with strength decreasing as a function of distance and edge weight. Nodes receiving sufficient spreading activation are primed: their retrieval threshold is lowered even before they have been directly queried.

For context management, spreading activation implies that a query activating a central concept should also prime semantically connected concepts, even if those concepts are not directly mentioned in the query. A pure cosine similarity scorer misses this neighbourhood effect: related segments may be underscored and evicted, leaving the model without context for inferences that require a populated semantic neighbourhood rather than a directly relevant single fact.

Implementation mapping. The `SpreadingActivationScorer` addresses this by constructing an activation graph over all current context segments, drawing edges between pairs whose embedding cosine similarity exceeds a threshold. From the query node, one round of activation propagates to directly similar segments, which then propagate a fraction of their activation to their graph neighbours. A single propagation round is deliberate: multi-round spreading produces runaway activation in dense graphs, and the first-order neighbourhood effect is the theoretically most relevant for short-horizon context management. The use of embedding cosine similarity as a proxy for association-norm edge weights is a known approximation; a more faithful implementation would incorporate explicit knowledge graph structures, which is identified as a direction for future development.

Forgetting Mechanisms

McGeoch (1932) challenged the then-dominant view that forgetting is caused by passive trace decay, proposing instead that forgetting results from competition between memory traces. Proactive interference occurs when previously learned material inhibits learning or retrieval of new, similar material. Retroactive interference occurs when newly learned material disrupts retrieval of earlier, similar material (Underwood, 1957). Both effects are modulated by inter-material similarity and are strongest when materials are similar but not identical. Anderson and Neely (1996) provided an inhibitory account: the retrieval system actively suppresses competing traces to resolve competition, and this suppression persists beyond the retrieval episode, contributing to long-term inaccessibility.

Memory consolidation operates on a complementary principle. McGaugh (2000) established that consolidation unfolds over extended time periods, during which traces remain labile. Squire and Alvarez (1995) proposed that memories initially encoded in the hippocampus are progressively transferred to the neocortex through repeated reactivation, abstracting common features from multiple episodic occurrences into durable schematic representations. Repetition is the key driver: a concept encountered across multiple spaced episodes is more durable than one encountered in a single episode, even if the total exposure time is identical.

Implementation mapping. The `InterferenceDetector` addresses the retroactive interference problem. It scans the message list for pairs where cosine similarity exceeds a threshold and the later message contains lexical contradiction markers. Such pairs are flagged as retroactive interference candidates with a recommendation to evict the earlier message or annotate it as superseded. The detector does not modify context, consistent with Anderson and Neely's (1996) observation that inhibition is context-sensitive: in high-stakes applications, preserving both the original claim and the correction may be necessary to document the knowledge transition. The choice of resolution strategy is therefore delegated to the caller.

The `ConsolidationEngine` operationalises hippocampal-to-neocortical transfer. It monitors conversation history for concepts recurring across multiple turns, treating repeated semantic occurrence as the functional equivalent of the reactivation events that drive consolidation. When a concept appears across a configurable threshold of turns, defaulting to three in accordance with Cepeda et al.'s (2006) finding that three spaced exposures reliably produce long-term encoding, the engine promotes it from the active context to the persistent `KnowledgeBase`, where it is injected near each subsequent relevant query and exempted from the eviction pressure that operates on the regular window.

Retrieval Processes

Tulving and Thomson (1973) proposed the encoding specificity principle: retrieval success depends not on the intrinsic associative strength between a cue and a target but on whether the cue was present or represented at encoding. A weakly associated cue present during encoding outperforms a strongly associated free-associate as a retrieval trigger. The match between encoding context and retrieval cue is the critical variable. The principle generalises to context-dependent memory, where environmental context at encoding functions as a retrieval cue (Godden & Baddeley, 1975), and to state-dependent memory, where internal states serve the same role (Bower, 1981).

Johnson, Hashtroudi, and Lindsay (1993) proposed source monitoring theory: memory traces include not only the content of the remembered experience but metadata about its

source, including who produced it, how it was obtained, and when it occurred. Source monitoring failures, attributing a memory to the wrong source or the wrong time, are a major contributor to memory distortion. The critical distinction the theory draws is between encoding time, when the information entered memory, and event time, when the described event actually occurred. These two temporal attributes are separately stored and separately retrievable. Confusing them is a well-documented source of error: a person may remember accurately that they learned something yesterday while misremembering whether the event it describes happened yesterday or three years ago (Mitchell & Johnson, 2000).

Implementation mapping. Encoding specificity justifies the relevance-based retrieval architecture throughout the proxy. The `RelevanceSignal`'s cosine similarity between current query and segment embedding is a direct computational implementation of cue-encoding match. The `InMemoryArchive` retrieves archived segments when the current query embedding is sufficiently similar to the segment embedding, operationalising Tulving's principle that apparently forgotten memories are often inaccessible rather than lost: segments archived at low relevance may be highly relevant to a future query that reinstates their encoding context.

Source monitoring theory provides the cognitive science grounding for the bitemporal extension developed in Chapter 8. The theory establishes that the human memory system has always managed two temporal attributes simultaneously, not because of an engineering decision but because it is necessary for accurate memory function. When a clinician remembers that a patient's dosage was changed, the accuracy of that memory depends on correctly attributing when the change was recorded in the notes (encoding time, corresponding to transaction time in the bitemporal model) and when the change actually took effect in the patient's treatment (event time, corresponding to valid time). Conflating the two produces clinical error. The same structure applies to LLM context management: a correction entered at turn twenty-two describes an error that has been present since turn five. Treating the correction's transaction time as its valid time is precisely the source monitoring confusion that Johnson et al. (1993) identified as a primary source of human memory distortion. The `EpisodicBuffer`'s detection of

correction events already implicitly acknowledges this: corrections are protected because they are consequential, not because their temporal structure is explicitly tracked. The bitemporal extension makes the temporal structure explicit, completing the analogy.

Where the Analogy Holds

The most robust correspondences are at the level of functional organisation. The distinction between a capacity-limited active store and a larger, cue-dependent long-term store maps reliably onto the distinction between the active context window and the archive tier. The four scoring signals, recency, relevance, confidence, and currency, correspond to established determinants of memory accessibility: temporal distance, cue match, source credibility, and content volatility. The primacy and spacing effects provide quantitatively grounded predictions that align directly with the engineering choices in `PrimacySignal` and `RevisitationSignal`. The interference framework addresses a failure mode with a precise counterpart in the proactive and retroactive interference literature. Encoding specificity is the most tightly operationalised correspondence in the system: cosine similarity over dense embeddings is a genuine computational implementation of cue-encoding match, not merely an analogy to it.

The value of this grounding extends beyond justification. When the scoring function is applied to a novel domain or a novel conversation structure, the cognitive science literature provides a basis for anticipating how it will behave. A domain with high inter-segment semantic similarity will produce high interference rates; the `InterferenceDetector` will flag more candidates. A conversation that recurs repeatedly to the same topic will consolidate that topic to the knowledge base regardless of its positional age; this is the spacing effect in operation. The cognitive science ground the system stands on provides a framework for reasoning about failure modes before they occur.

Where the Analogy Breaks Down

Four disanalogies are significant enough to require explicit acknowledgement.

The most fundamental concerns the nature of the underlying representations. Human memory encodes information in distributed, overlapping neural patterns inherently subject to reconstruction, interference, and distortion. LLM context management operates on discrete, losslessly stored text strings. When a human memory is archived, it is re-encoded in a different format and becomes subject to reconstructive retrieval: the stored trace is a partial record, and recall fills in the gaps with plausible inference. When a context segment is archived, its exact text is stored verbatim and retrieved without reconstruction. The proxy avoids reconstructive errors that are structurally unavoidable in biological memory. It also lacks the corresponding benefit: reconstruction is partly responsible for the creative, generative character of human memory, its ability to recombine fragments from different episodes into novel inferences. The proxy produces no analogous spontaneous recombination.

The second disanalogy concerns the substrate of the recency signal. The proxy uses positional decay grounded in transformer attention dynamics: $\exp(-\lambda(1 - t))$. There is no exact correspondence between the transformer's positional attention bias and biological trace decay; the mathematical form is shared but the causal mechanisms are entirely different. The biological recency effect reflects short-term memory capacity limits and competitive retrieval dynamics. The transformer's recency bias reflects learned statistical regularities in training text. These are structurally analogous effects from structurally different causes, and they will diverge in any situation where the two causal chains produce different predictions.

Third, the working memory literature treats the central executive's attentional and goal-directed control functions as equal in importance to storage. Task switching, attention management, goal maintenance under competing demands: these are the processes that make working memory a workspace rather than a buffer. The proxy has no analogue to the central executive. It operates reactively: rescoring the existing context in response to each query rather than proactively restructuring it in anticipation of future needs. This limitation is most visible in agentic settings where the conversation's future direction is predictable but the proxy cannot act on that prediction.

Fourth, affective modulation of memory has no representation in the proxy's scoring function. Emotionally significant events are encoded more durably and retrieved more reliably than neutral events matched on all other dimensions (McGaugh, 2000). The `EpisodicBuffer`'s detection of confirmation, decision, and correction events captures some of this salience, but via lexical heuristics rather than any genuine estimate of conversational significance. A correction is protected because it matches the pattern of correction language, not because the proxy has any representation of why corrections matter.

The Temporal Insight and Its Implications

The nine models surveyed in this chapter were chosen because each one provides principled justification for a specific architectural decision in the proxy. Together they establish that the proxy is not a collection of heuristics but a system whose design follows the structure of what is known about human memory. That is a different and stronger claim.

Source monitoring theory, the last model surveyed, does something more than justify an existing design decision: it reveals a gap. Johnson et al. (1993) established that accurate memory function requires tracking two temporal attributes independently, and that confusing them is a reliable source of distortion. The proxy, in its base configuration, tracks only one. It knows when content entered the context window. It does not know when the content it describes was valid. That gap is the subject of Chapter 8.

The human memory literature therefore serves this project in two distinct ways. As a source of grounding, it provides principled justification for design choices that would otherwise rest on empirical calibration alone. As a source of inspiration, it identifies the absence of the bitemporal dimension as a failure mode with a name in cognitive science, not merely a limitation that emerged from an engineering choice. That identification is itself a contribution: it connects an engineering problem to a century of experimental evidence about why a system designed for accurate temporal reasoning cannot function correctly without tracking two time axes simultaneously.

The chapter that follows describes the implementation of the proxy architecture, showing how the cognitive science models surveyed here translate into a working system and identifying the points at which the translation is approximate rather than exact.

Bibliography

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review*, 94(2), 192–210.

Anderson, M. C., & Neely, J. H. (1996). Interference and inhibition in memory retrieval. In E. L. Bjork & R. A. Bjork (Eds.), *Memory* (pp. 237–313). Academic Press.

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Baddeley, A. D. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences*, 4(11), 417–423.

Baddeley, A. D., & Hitch, G. (1974). Working memory. In G. H. Bower (Ed.), *The Psychology of Learning and Motivation*, Vol. 8 (pp. 47–89). Academic Press.

Bower, G. H. (1981). Mood and memory. *American Psychologist*, 36(2), 129–148.

Cepeda, N. J., Pashler, H., Vul, E., Wixted, J. T., & Rohrer, D. (2006). Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological Bulletin*, 132(3), 354–380.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Collins, A. M., & Quillian, M. R. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8(2), 240–247.

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1), 87–114.

Ebbinghaus, H. (1885). *Über das Gedächtnis*. Duncker & Humblot.

Glanzer, M., & Cunitz, A. R. (1966). Two storage mechanisms in free recall. *Journal of Verbal Learning and Verbal Behavior*, 5(4), 351–360.

Godden, D. R., & Baddeley, A. D. (1975). Context-dependent memory in two natural environments: On land and underwater. *British Journal of Psychology*, 66(3), 325–331.

Johnson, M. K., Hashtroudi, S., & Lindsay, D. S. (1993). Source monitoring. *Psychological Bulletin*, 114(1), 3–28.

McGeoch, J. A. (1932). Forgetting and the law of disuse. *Psychological Review*, 39(4), 352–370.

McGaugh, J. L. (2000). Memory — a century of consolidation. *Science*, 287(5451), 248–251.

Mitchell, K. J., & Johnson, M. K. (2000). Source monitoring: Attributing mental experiences. In E. Tulving & F. I. M. Craik (Eds.), *The Oxford Handbook of Memory* (pp. 179–195). Oxford University Press.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Roediger, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science*, 17(3), 249–255.

Squire, L. R., & Alvarez, P. (1995). Retrograde amnesia and memory consolidation: A neurobiological perspective. *Current Opinion in Neurobiology*, 5(2), 169–177.

Tulving, E. (1983). *Elements of Episodic Memory*. Oxford University Press.

Tulving, E., & Thomson, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5), 352–373.

Underwood, B. J. (1957). Interference and forgetting. *Psychological Review*, 64(1), 49–60.

Chapter 5: A Scoring Architecture for Context Management

The four preceding chapters established the problem, developed the analytical frame for evaluating solutions, surveyed what the technical literature has produced, and grounded the present chapter's design decisions in the cognitive science of human memory. This chapter presents the first original contribution of the dissertation: a context management proxy that intercepts the messages array before it reaches the LLM API, scores each conversational segment against multiple signals drawn from the cognitive science established in Chapter 4, and applies tier-appropriate treatment to maintain context quality within budget. The architecture makes two claims that together constitute its originality. First, that selective retention driven by principled multi-dimensional scoring is qualitatively different from the naive approaches surveyed in Chapter 3, not merely better by degree but different in kind, because it manages content by value rather than by age. Second, that this value-based retention can be grounded in cognitive science rather than in heuristic tuning, making the policy principled and auditable rather than arbitrary. The chapter also presents a second architectural tier: three cognitive subsystems and two scoring strategy variants that address structural properties of conversations which the core scoring function cannot capture alone. It closes by naming the limitation that this architecture does not address — it manages the content of the conversation rather than the epistemic structure the content has established — a distinction that proves load-bearing in the chapters that follow.

The Proxy Architecture

The proxy is a middleware layer that the application treats as a drop-in replacement for a direct LLM client call. The interface is minimal:

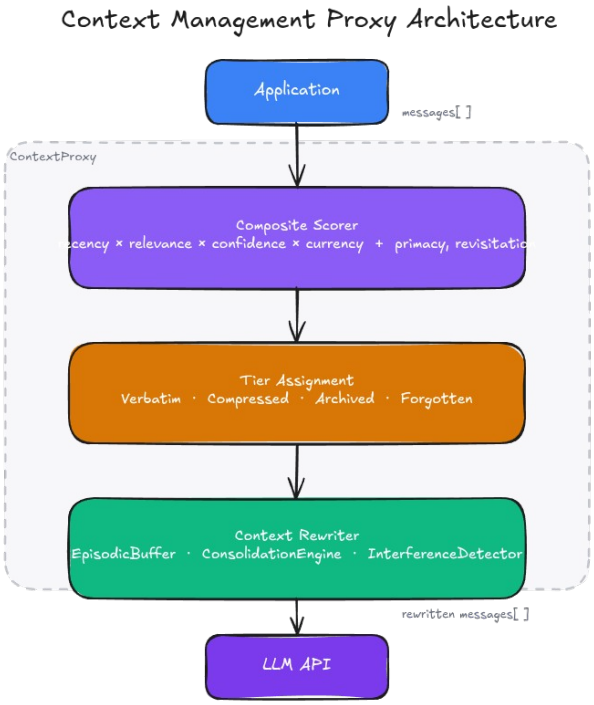
```
proxy = ContextProxy(client=anthropic_client, config=config)
response = proxy.chat(messages, model="claude-sonnet-4-6")
```

On each call, the proxy intercepts the `messages` array, applies its scoring and management policy to each segment, rewrites the array to fit within the configured context budget, and forwards the managed version to the model. The model receives a

context that is within budget and prioritised for quality. The application receives the model's response unchanged. Neither party requires awareness of the rewriting.

This positioning has a principled motivation beyond convenience. The proxy must operate at the message level rather than the token level because the units of meaning in a conversation are messages, not tokens. A message carries a semantic identity: it is a contribution to the conversation, not a span of subword units. Scoring at the message level respects this identity; scoring at the token level would produce a fragmented picture of conversational value that no token-level signal could correctly reconstruct.

The proxy coordinates three component systems. The scoring function produces a continuous value score for each segment. The compression subsystem reduces mid-range segments to recover token budget while preserving semantic content. The archive backend stores low-scoring segments in a retrievable external store indexed by embedding. These are distinct concerns that the proxy coordinates without conflating. The scoring function has no knowledge of how compression is implemented; the archive backend has no knowledge of the scoring function's thresholds. Each component can be replaced, extended, or tested in isolation.



The Scoring Function

The proxy assigns each segment a composite score on a continuous scale from 0.0 to 1.0.

> **Composite score**: the product of four multiplicative signals — recency, relevance, confidence, and currency — optionally augmented by additive boosts for structurally important segments, clamped to the interval [0, 1].

The formula is:

score = recency(t) × relevance(s) × confidence(c) × currency(k) + boosts

The multiplicative form is deliberate, and its principal property is specific: any signal scoring at or near zero eliminates the segment from the product, making it a strong eviction candidate regardless of how well it scores on the other dimensions. This mirrors the encoding specificity principle established in Chapter 4: a memory trace is retrieved efficiently only when the retrieval context matches the encoding context on all relevant dimensions. A segment that is maximally recent but entirely irrelevant to the current query provides no value to the current generation step. A segment that is highly relevant but carries zero confidence — explicitly hedged or retracted content — should not survive on the strength of its relevance alone.

The additive boosts modify the product after it is computed, providing corrections for segments whose structural importance is not captured by the four multiplicative signals. A system prompt that has decayed positionally should not be evicted on the basis of recency alone; the primacy boost counteracts this specific failure. The boosts are applied after the product and the total is clamped to [0, 1], so a segment cannot exceed the maximum score regardless of how many boosts it accumulates.

Four Multiplicative Signals

Each signal operationalises one dimension of a segment's value. The cognitive science grounding for each is not decorative; it explains why that dimension is the correct one to measure for the purpose of retention scoring, and it determines the mathematical form the signal takes.

Recency

`signals.py:RecencySignal` implements exponential decay over normalised token position:

$$r(t) = \exp(-\lambda \times (1 - t/T))$$

where t is the segment's normalised position in the context (0 at the start, 1 at the current turn), T is the total context length, and λ is the configurable decay rate. The most recent segment scores 1.0; the oldest approaches 0 as λ increases.

The cognitive grounding is the serial position effect and the recency component of the multi-store model (Murdock, 1962; Atkinson & Shiffrin, 1968). As Chapter 4 established, recent items in a sequence receive disproportionate attention in both human memory and transformer attention mechanisms, because training on locally coherent human text teaches the model that recency is a reliable prior for relevance. The recency signal operationalises this prior directly: it is the baseline expectation that any content management policy starts from.

The signal is a prior, not a verdict. Its role in the composite score is to establish the base expectation from which the other signals provide corrections. A segment that scores low on recency is not condemned to eviction; it requires stronger positive signals on relevance, confidence, or currency to survive. This is the correct epistemic relationship: age is evidence against retention, but it is evidence that other signals can outweigh.

Relevance

`signals.py:RelevanceSignal` computes the cosine similarity between the segment's embedding and the embedding of the current query — the most recent user message in the conversation.

The cognitive grounding is Tulving and Thomson's (1973) encoding specificity principle: memory is most efficiently retrieved when the retrieval cue matches the encoding context. In the proxy architecture, the current query is the retrieval cue, and segments are the encoded traces. A segment whose semantic content aligns closely with the current

query is precisely the segment the model needs most at this generation step, regardless of where in the conversation it was produced.

Relevance provides the critical override mechanism that the recency signal cannot supply. A specialist constraint established at turn three of a forty-turn conversation will have a low recency score. If the current query is directly about that constraint's domain, the relevance score will be high — potentially high enough to bring the composite score above the eviction threshold despite positional age. This is the depth-aware property that Chapter 2 identified as necessary for specialist domains: the system must be capable of preserving domain-specific content against positional pressure when that content is directly needed at the current turn.

The embedding model is `all-MiniLM-L6-v2` via sentence-transformers, a 384-dimensional local model that provides semantic similarity without external API calls. This reflects a deliberate design constraint: the relevance signal must not introduce latency or cost that would render it impractical for production deployment. The same embeddings used for relevance scoring serve double duty as the index for the archive backend, avoiding redundant computation.

Confidence

`signals.py:ConfidenceSignal` passes through the segment's confidence attribute directly, returning a value in [0, 1] representing the reliability of the content at the time it was generated.

The cognitive grounding is source monitoring theory (Johnson, Hashtroudi & Lindsay, 1993). Chapter 4 established that source monitoring — the cognitive mechanism by which humans track the provenance and reliability of their memory traces — directly grounds the confidence signal. A conversational segment produced under explicit uncertainty, hedged with "this may be wrong" or "I'm not certain," carries lower veridicality than a segment produced with high certainty. The confidence signal operationalises this: it biases retention toward content the system had reason to believe when it produced it.

Confidence can be estimated from two sources. When model response metadata provides log-probabilities or explicit confidence fields, these can be used directly. When they are not available, linguistic markers of certainty and uncertainty in the segment content serve as an approximation. The signal's passthrough design accommodates both: the caller populates `segment.confidence` from whatever source is available, and the scoring function treats it uniformly.

Currency

`signals.py:CurrencySignal` applies a temporal discount by content volatility through the KDecay mechanism from the Mnemonic system, described in Chapter 3.

Different types of content expire at different rates. A drug dosage recommendation may be revised on a monthly clinical cycle; an anatomical relationship has not changed since it was first described. A current software version number is accurate today and obsolete in weeks; a mathematical identity does not age. Currency scoring captures this by classifying segment content into volatility categories and applying a corresponding decay rate based on the segment's age in hours. Stable content maintains high currency scores indefinitely; volatile content decays rapidly.

The cognitive grounding is prospective memory and temporal discounting. Human memory systems maintain representations of future events and commitments with a temporal structure reflecting their expected duration (Einstein & McDaniel, 1990). Currency scoring implements the analogous function: it asks not only how old a segment is but how much its type of content ages, and applies a discount accordingly. Currency is distinct from recency, which scores by positional age. Currency scores by content volatility. A segment can be positionally old but temporally fresh if its content type is stable — an established definition, a mathematical relationship, a structural fact. The two signals together produce a retention criterion that is sensitive to both dimensions of temporal value simultaneously.

Two Additive Boosts

The four multiplicative signals capture the core retention criteria. Two additive boosts correct for structural properties of conversational content that the multiplicative form handles poorly.

Primacy

`signals.py:PrimacySignal` provides a configurable boost for segments that satisfy two conditions simultaneously: they appear in the first 15% of the conversation by token position, and they contain instruction-class keywords. Segments meeting both conditions receive an additive boost that partially counteracts the positional penalty of being early in the context.

The cognitive grounding is the primacy effect component of the serial position curve (Murdock, 1962). Human memory shows superior recall for items early in a sequence, attributed to greater opportunity for rehearsal and transfer to long-term storage (Atkinson & Shiffrin, 1968). In LLM conversations, the primacy analogue is the system prompt and early instructional content: the material that establishes the operational frame for all subsequent reasoning. This material is the most consequential content in many conversations, and it is also the content most systematically penalised by the recency signal, having accumulated the greatest positional age by the time eviction pressure arrives.

The primacy boost does not exempt early instructional content from eviction. It shifts the balance of the composite score to make eviction harder, not impossible. A system prompt that is genuinely irrelevant to the current conversation domain will still score low on relevance, and no primacy boost compensates for a near-zero relevance score. The boost addresses the specific pathology identified in Chapter 1: instructions that drift to ineffectiveness not because they have become irrelevant but because positional age alone has pushed their composite score below the verbatim threshold.

Revisitation

`signals.py:RevisitationSignal` provides a boost proportional to the number of prior turns in which the segment's topic has been explicitly revisited. The boost accumulates

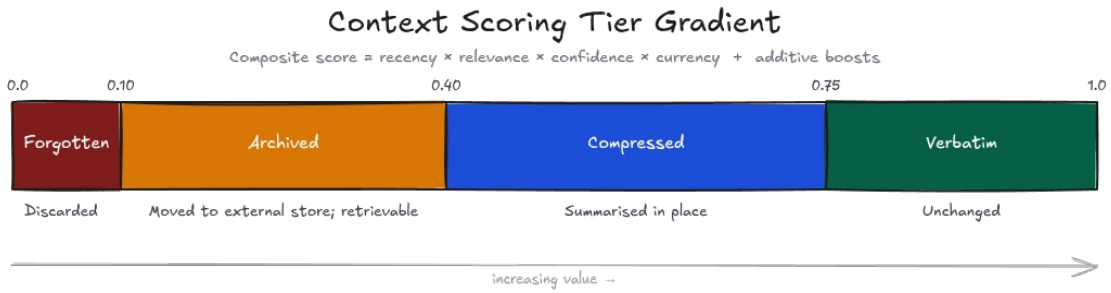
with each revisitation event, capped at a configurable maximum to prevent runaway scores in conversations that return repeatedly to one subject.

Two findings from the spacing and testing effect literature ground this signal simultaneously. The spacing effect (Ebbinghaus, 1885) establishes that information encountered across multiple spaced intervals is encoded more durably than the same information encountered in a single massed session: distributed repetition is a reliable indicator of importance. The testing effect (Roediger & Karpicke, 2006) establishes that each retrieval event strengthens the memory trace beyond what passive re-exposure achieves. In the conversation context, each turn in which a concept is queried or referenced constitutes an implicit retrieval event. A segment whose topic recurs repeatedly across a long conversation is more likely to be central to the conversation's purpose than one that appears once and is never revisited. The revisitation signal rewards this centrality directly.

The Tier Vocabulary

The continuous score gradient requires a human-readable vocabulary for inspection, logging, and configuration. Four labels serve this purpose:

> **Verbatim** (score 0.75–1.0): the segment passes through unchanged, occupying its full token cost in the managed context.
>
> **Compressed** (score 0.40–0.75): the segment remains in context but in a summarised form that preserves semantic content while recovering token budget.
>
> **Archived** (score 0.10–0.40): the segment is moved out of the context window into an external store, indexed by embedding, and can be retrieved if its relevance rises.
>
> **Forgotten** (score 0.0–0.10): the segment is discarded permanently.



The critical property of these labels is that they are a vocabulary for a continuous gradient, not the names of hard-boundary buckets. A segment does not transition between tiers through a binary event. Its score drifts continuously as new content arrives, as the topic of the conversation shifts, and as time passes. A segment that is Verbatim at turn ten may be Compressed at turn twenty and Archived at turn thirty if the conversation has moved on from its subject. Equally, an Archived segment may be promoted back to Verbatim when the conversation returns to its subject and retrieval confirms its relevance. The tier labels describe the current state of a segment's score; they do not record a permanent classification.

The thresholds (0.75, 0.40, 0.10) are configurable because they are calibration parameters, not logical constants. A lower Verbatim threshold retains more content verbatim at greater token cost; a higher Forgotten threshold evicts more content at lower token cost. The correct values depend on domain, context budget, and tolerance for information loss. The scoring function's principled design is independent of where these thresholds are set; the thresholds determine how the gradient maps to action, not whether the gradient is correct.

One structural constraint on the Forgotten tier merits explicit statement. A segment whose temporal classification is known to carry a forward-open valid interval — a standing constraint, a committed decision, an instruction that governs all subsequent conversation — must not be assigned to Forgotten on the basis of positional age alone. However low its recency score, recency decay is not a sufficient warrant for discarding content that is structurally required to be present for the conversation to proceed correctly. The bitemporal extension in Chapter 8 formalises this constraint as a

`TemporalValiditySignal` that provides structural immunity from positional eviction for standing knowledge.

The Eviction Policy

When the context budget is exhausted and the proxy must reduce token count, the eviction policy is: evict the lowest-scoring segment first.

This is not the obvious policy. The obvious policy is to evict the oldest segment first, which is what truncation and rolling-window approaches implement. The oldest-first policy is operationally simple and wrong for the reasons established in Chapters 1 and 3: it conflates positional age with lack of value, and it does so precisely in the cases where early content is most critical — the specialist constraint, the opening instruction, the foundational fact retrieved before the conversation built up depth.

Lowest-score-first eviction is value-ranked, not position-ranked. The segment contributing least to the current generation step at the time of budget pressure is the one removed, regardless of where in the conversation it was produced. This is the correct policy if the scoring function correctly measures contribution. The practical question is whether the composite score is a reliable proxy for contribution. The alignment between the signals and their cognitive science grounding is the basis for claiming that it is: each signal measures a dimension of value that has been experimentally validated as a determinant of retention and retrieval in human memory and as a structural property of LLM attention. The composite is not a perfect measure, but it is a principled multi-dimensional estimate that is systematically superior to the degenerate special case in which the only signal is recency.

One property of the eviction policy deserves emphasis. Because eviction is driven by composite score rather than position, the same content that survived at turn twenty might survive again at turn forty if the conversation has returned to its domain, because the relevance signal will have risen. The policy is not monotone: a segment may return from the Archived tier to the Verbatim tier. This mirrors the cue-dependent retrieval established by Tulving and Thomson (1973): archived content that matches a new

retrieval cue is retrieved and promoted. The archive is not a graveyard; it is a secondary store with retrieval semantics.

The Cognitive Architecture Layer

The scoring function and tier policy constitute the first tier of the architecture. A second tier adds three subsystems that detect and respond to structural properties of the conversation that the scoring signals cannot capture individually. These subsystems were introduced in Chapter 4 in terms of their cognitive science grounding; here they are presented in terms of their architectural function.

The Episodic Buffer

`episodic_buffer.py:EpisodicBuffer` identifies three classes of structurally significant conversational episode:

CONFIRMATION (the user explicitly validates a model output or proposal),

DECISION (a consequential commitment is made that the conversation will proceed from), and

CORRECTION (the user corrects a model error, an event that must be preserved alongside the corrected claim to prevent the error from resurfacing as proactive interference).

The episodic buffer does not modify composite scores. It returns a set of message indices that the proxy marks as unconditionally protected: these segments must remain in the Verbatim tier regardless of what their composite score would otherwise dictate. This is architecturally distinct from boosting their scores. A score-based approach preserves the segment in proportion to its score; a protection-based approach makes the preservation absolute. The distinction matters because episodes of the three types listed carry structural load that is not captured by any combination of recency, relevance, confidence, and currency. A correction that is old, irrelevant to the current query, low-confidence, and of volatile content type might score near zero on all four signals. Its structural importance — that it records a real-world correction that prevents a known error from recurring — is not a scoring dimension. It is a categorical property.

The detection mechanism is deliberately lightweight: keyword heuristics that achieve high precision at the cost of some recall. False negatives, missed episodes, are preferable to false positives, which would lock content into the Verbatim tier unnecessarily and exhaust the eviction budget on content that does not require protection.

The Consolidation Engine

`consolidation.py:ConsolidationEngine` detects concepts that recur across at least three conversation turns and promotes them from the active context window to the KnowledgeBase, a persistent store from which they are injected near each relevant query. The promotion threshold of three turns reflects the spacing effect literature: Cepeda et al. (2006) established that three spaced rehearsal events are sufficient to produce reliable long-term encoding under normal conditions.

The consolidation engine addresses a specific failure mode that the scoring function handles poorly: concepts that are individually low-scoring on recency but collectively central to the conversation. A concept discussed briefly at turn three, mentioned again at turn twelve, and referenced at turn twenty-five has a low recency score at turn twenty-five because its primary statement was early. But its recurrence pattern is strong evidence that it is a load-bearing element of the conversation's domain. Consolidation promotes it from a position where recency pressure could eventually evict it to a position from which recency does not apply: the KnowledgeBase is not subject to positional decay because it is not part of the linear context sequence.

Similarity detection uses cosine distance on `all-MiniLM-L6-v2` embeddings. This is consistent with the semantic representation hypothesis of long-term memory consolidation (Rogers & McClelland, 2004): the representation that consolidation should promote is semantic, not lexical. Two turns discussing the same concept in different words should be identified as related. Lexical matching would miss this.

The Interference Detector

`interference.py:InterferenceDetector` scans the message list for semantically similar but potentially contradictory message pairs and returns `InterferenceFlag`

objects identifying the earlier and later message indices, the type of interference, and a recommendation for resolution.

Two types of interference are detected. Retroactive interference occurs when a later message contradicts an earlier one, potentially displacing correct retrieval of the earlier content. Proactive interference occurs when an earlier message may be corrupting the encoding or retrieval of similar later content. The detector identifies these cases by combining semantic similarity above a configurable threshold with the presence of contradiction markers in the later message.

The detector does not resolve interference. It reports flags and leaves resolution to the caller. This reflects the genuine uncertainty of the resolution decision. An interfering pair might warrant evicting the earlier message, annotating it with [SUPERSEDED], or flagging for human review, depending on domain, stakes, and whether the earlier message has independent value beyond the content that was contradicted. No single policy is correct across all cases. The detector provides the signal; the caller applies the policy. This is a deliberate architectural choice: the system should not attempt to resolve semantic contradictions autonomously in cases where the correct resolution is domain-dependent.

Scoring Strategy Variants

The architecture supports pluggable scoring strategies through the `ScoringStrategy` protocol in `strategies.py`. Two strategies are provided.

`strategies.py:CompositeScorer` implements the default multiplicative formula described above. Its dual-stage computation — product of multiplicative signals, then additive boosts — mirrors dual-process theory (Evans, 2008): the multiplicative stage is fast, automatic, and heuristic in character, while the additive stage applies deliberate corrections for structural importance. This mapping to dual-process theory is not a strict identification; it is a structural analogy. The product collapses four signals into a single score using a rule that requires no contextual judgment, while the boosts apply corrections that require knowledge of the segment's structural role in the conversation. The analogy is apt precisely because the two stages have different epistemic characters:

the multiplicative stage responds to signal values, the additive stage responds to category membership.

`strategies.py:SpreadingActivationScorer` implements an alternative strategy grounded in Collins and Loftus's (1975) spreading activation theory. In their model, semantic memory is a network of concept nodes connected by weighted edges. When a concept is activated, activation propagates along edges to related concepts, making them more accessible. The scorer operationalises this directly: for each segment, it computes base relevance as cosine similarity to the query, builds a local activation graph by identifying segments whose embeddings are sufficiently similar to each other (above a configurable edge threshold), and runs one round of activation spreading, adding a fraction of the mean neighbour activation to the segment's base score. Segments that are not directly relevant to the query but are semantically adjacent to segments that are will receive activation from their neighbours.

The single-round approximation is intentional. Multi-round propagation can produce runaway activation in densely connected graphs; one round captures the first-order neighbourhood effect that is theoretically most relevant to short-term context management. The spreading activation scorer is most appropriate for conversations with rich semantic interconnections, where the relevance of a segment cannot be assessed from its direct relationship to the current query alone. In practice, the choice between the two strategies is a configuration decision: the `CompositeScorer` is the correct default; the `SpreadingActivationScorer` is appropriate when the conversation's semantic structure is explicitly important to retention quality.

A Planned Extension: Surprise Allocation

The novelty gap research conducted for this dissertation established a fifth signal that is theoretically motivated but not yet implemented in the architecture: a `SurpriseSignal` that would compute each segment's information density as the negative log-likelihood of the segment's tokens under the model's prior — the segments that most updated the model's beliefs at generation time.

The theoretical grounding is Friston's Free Energy Principle (Friston, 2010). Minimising variational free energy is equivalent to maximising model evidence, and a context manager that retains high-surprise segments directly implements this principle: it retains the observations that most updated the model's beliefs, which are by definition the observations that carry the most information relative to what the model already knew. The closest existing instantiation is the Titans architecture (Behrouz et al., 2025), which couples the forgetting rate in a recurrent long-term memory to gradient surprise — high prediction error slows forgetting for that segment. This is a clean implementation of the "retain what surprises the model" principle, but it operates inside a model's weight updates during inference rather than at the conversation proxy level. The human cognitive analogue is Futrell, Gibson and Levy's (2020) Lossy-Context Surprisal model of sentence processing: processing cost is proportional to prediction error under lossy memory, which is the cognitive analogue of allocating retention budget proportional to surprise.

A `SurpriseSignal` would slot into the `CompositeScorer` as a fifth multiplicative signal alongside recency, relevance, confidence, and currency. Its effect would be to retain conversationally surprising content that might otherwise score low on both recency and relevance: a conversation that introduces an unusual or novel fact early, never directly queries it again, but implicitly builds on it throughout would currently see that fact lose retention priority over time. The surprise score would capture its epistemic density and give it a retention advantage that the other four signals cannot derive. Implementing it presents three layered practical challenges. The most immediate is data access: the major API providers expose per-token log-probabilities through parameters such as `logprobs` and `top_logprobs`, but these cover only output tokens generated at inference time, not tokens already present as input context. Recovering surprisal scores for existing segments would require resubmitting them as generation prompts, which both alters the conditioning context and incurs additional cost. When primary API access is unavailable, a smaller auxiliary model such as GPT-2 can approximate per-token surprisal; empirical evidence suggests, counterintuitively, that smaller models sometimes produce surprisal estimates that better predict human reading behaviour than larger models do, a result attributable to tokenisation biases rather than general calibration quality, and there is no

theoretical guarantee that auxiliary-model surprisal aligns reliably with primary-model log-probabilities on conversational corpora (Oh & Schuler, 2023). Signal composition introduces a further complication: naive multiplicative combination assumes independence between surprise and the existing signals, but novel events tend to cluster temporally, making high-recency and high-surprise conditions correlated rather than complementary; the Titans architecture addresses this through a learned nonlinear coupling between gradient surprise and the forgetting rate, but whether a static multiplicative composition can replicate this effect in a proxy context is not established (Behrouz et al., 2025). The boundary of the challenge is clearer than its resolution: further investigation of log-probability access patterns across provider APIs, auxiliary model calibration on conversational corpora, and the statistical dependence structure between surprise and the existing four signals is needed to properly quantify both the tractability of the implementation and the precision the signal can realistically achieve.

The Remaining Gap

The scoring architecture presented in this chapter is a principled, cognitively grounded system for managing the content of a context window. It addresses the failure modes of naive approaches: it does not truncate by age, it does not compress uniformly, and it does not treat recency as a proxy for importance. Its eviction policy is value-ranked rather than position-ranked. Its signals are grounded in experimental findings that have accumulated over decades of memory research rather than in heuristic intuition. Its three subsystem extensions address structural properties of conversations that the scoring function alone cannot capture.

What the architecture does not address is the epistemic structure that the conversation has established. A conversation running for forty turns has produced not only a sequence of scored segments but a shared understanding: claims accepted as true, decisions committed to, questions identified as open, constraints established as governing. The scoring architecture manages the segments that carry this understanding. It does not ask what the understanding is, whether it is coherent, or whether the conversation has established a frame that will produce correct answers to future queries from the conversation's domain.

This is not a failure of design; it is a structural limitation of the content-management paradigm. A system that asks "which segments are most valuable to retain?" is asking the correct question for content management. It is asking the wrong question if the goal is to maintain the epistemic integrity of the conversation. A managed context can preserve every high-scoring segment faithfully and still fail to represent what the conversation has actually established, because the epistemic structure is an emergent property of the segment sequence, not a property that any individual segment carries. The correct evaluation metric for this architecture would test not whether the managed context is a good reconstruction of the original but whether a model operating on the managed context can correctly answer held-out queries from the conversation's domain. That test, which this dissertation terms generative competence evaluation, does not yet exist as a standardised protocol. Its absence is a gap in the evaluation landscape, not a gap in the architecture, but it is a gap that a complete account of context management quality cannot indefinitely defer.

The next chapter proposes a different approach to the problem the scoring architecture leaves unsolved. Rather than preserving the conversation's content in optimally managed form, it proposes to replace the linear history with a direct representation of the integrated understanding the conversation has reached — the compact, durable, and generative form that the conversation has arrived at rather than the path it took to get there.

Bibliography

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Behrouz, A., Zheng, P., & Pilanci, M. (2025). Titans: Learning to memorize at test time. *arXiv preprint* arXiv:2501.00663.

Cepeda, N. J., Pashler, H., Vul, E., Wixted, J. T., & Rohrer, D. (2006). Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological Bulletin*, 132(3), 354–380.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Ebbinghaus, H. (1885). *Über das Gedächtnis: Untersuchungen zur experimentellen Psychologie*. Duncker & Humblot. [Trans. Memory: A contribution to experimental psychology, 1913.]

Einstein, G. O., & McDaniel, M. A. (1990). Normal aging and prospective memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16(4), 717–726.

Evans, J. St. B. T. (2008). Dual-processing accounts of reasoning, judgment, and social cognition. *Annual Review of Psychology*, 59, 255–278.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2), 127–138.

Futrell, R., Gibson, E., & Levy, R. (2020). Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cognitive Science*, 44(3), e12814.

Johnson, M. K., Hashtroudi, S., & Lindsay, D. S. (1993). Source monitoring. *Psychological Bulletin*, 114(1), 3–28.

McGeoch, J. A. (1932). Forgetting and the law of disuse. *Psychological Review*, 39(4), 352–370.

Oh, B.-D., & Schuler, W. (2023). Why does surprisal from larger transformer-based language models provide a worse fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11, 336–350. arXiv:2212.09897.

McGaugh, J. L. (2000). Memory — a century of consolidation. *Science*, 287(5451), 248–251.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Roediger, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science*, 17(3), 249–255.

Rogers, T. T., & McClelland, J. L. (2004). *Semantic Cognition: A Parallel Distributed Processing Approach*. MIT Press.

Tulving, E., & Thomson, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5), 352–373.

Chapter 6: Dissolution Through Consensus

The scoring architecture presented in Chapter 5 addresses the content management problem thoroughly and on principled grounds. It evicts by value rather than by age, it grounds each scoring signal in experimental cognitive science, and it provides structural protections for episodically significant segments. What it does not do is question the frame within which all of that work takes place. The frame is this: context management means managing the content of the conversation, preserving as much of it as possible, in as faithful a form as possible, subject to token budget constraints. This chapter argues that the frame is a category error. The content of a conversation is not what a conversation produces; it is the medium through which a conversation produces what it produces. What it produces is an integrated understanding: a structured space of established facts, committed decisions, open questions, and operative constraints. That understanding is more compact, more durable, and more generative than any managed version of the content that created it. This chapter introduces Dissolution Through Consensus, a mechanism that replaces the linear history of a conversation with a direct representation of the understanding it has reached, and presents the theoretical grounding, the mechanism, and the experimental evidence for this approach.

The Preservation Assumption and Its Failure

Every approach surveyed in Chapter 3 shares an assumption so fundamental that it is rarely stated: the goal of context management is to preserve the conversational history in as complete and faithful a form as the token budget permits. RAG preserves specialist knowledge by repositioning it at query time. Semantic caching preserves prior answers by returning them when similar questions arise. Ensemble methods preserve accuracy by distributing the decay burden across multiple representations. External stores preserve structured data by removing it from positional risk entirely. Prompt externalisation preserves instructions by refreshing their position. Distributed reasoning preserves the full history by writing it to a shared store.

None of these approaches challenge the premise that the conversation's content is the thing to be managed. The result is a field that has produced sophisticated variations on a

single question: how do we keep more of the right content available at the right time? The question is well-posed. The problem is that it is the wrong question.

A conversation accumulates content, but content is not what makes a conversation valuable. What makes a conversation valuable is the epistemic structure it establishes: the propositions that participants have accepted as true, the decisions they have committed to, the questions they have identified as open, the constraints they have agreed to operate within. This structure is built from the content, but it is not the same thing as the content. A 40-turn conversation about a software architecture decision contains tens of thousands of tokens. The durable output of that conversation — the decisions taken, the alternatives ruled out, the open questions at the boundary of the design — might be expressed in 400 tokens without any loss of the structure that will govern subsequent work. The other 39,600 tokens are the scaffolding that produced the structure. Preserving them is not obviously valuable once the structure has been extracted.

> **Dissolution**: the operation that replaces a conversation's linear history with a direct representation of the integrated understanding it has established, in a compact, present-tense, position-independent form.

The name is intentional. History does not compress into a boundary state; it dissolves into one. The distinction is not merely rhetorical. Compression preserves the original in reduced form: some of the scaffolding is kept, some is discarded, and what remains is a smaller version of what was there. Dissolution produces something categorically different from a compressed conversation: a state description that speaks from the current moment, not about the past moments that produced it.

The Void Universe Thought Experiment

The theoretical motivation for dissolution is most clearly reached through a thought experiment that inverts the standard frame of reference.

In a universe constructed almost entirely of matter, with two small voids embedded in it, physical intuition breaks down in an interesting way. In the standard universe — almost entirely empty, with two small pieces of matter — gravity pulls the matter toward each other. In the inverted universe, what governs the voids? The question does not have a

straightforward physical answer; the thought experiment is not a physics problem. It is a device for redirecting attention from the contents of a space to the shape of that space.

The philosophical tradition of defining objects by their boundaries rather than their contents is well-established. In mathematical topology, a space is characterised by the properties of its boundary, and the interior is in important senses secondary to the boundary's structure. In Gestalt psychology, the figure-ground relationship establishes that perception of a form is inseparable from perception of the space that surrounds it; the void defines the figure as much as the figure defines the void. Applied to conversation, the insight is this: what a conversation establishes is a region in conceptual space. The utterances that produced that region are the matter. The boundary of the region — the edges of what is known, decided, and still open — is the shape. And the shape is what persists.

When a conversation has run for 40 turns, its participants have jointly mapped a portion of a conceptual territory. Early turns drew the coarse outlines: here is the problem, here is the role each participant will play, here are the constraints that will govern the solution. Later turns filled in the details, made decisions, closed questions, and pushed the frontier of the established space outward. At any given point, the entire intellectual content of the conversation is captured more faithfully by a description of the current boundary than by any compressed version of the history that produced it. The boundary is what the history accumulated to. The history is the process that got there.

From Void to Boundary Synthesis

Treating context management as boundary management rather than content management changes the operational question. Instead of "which segments should we retain?" the question becomes "what is the current shape of the conceptual space this conversation has established?" These questions have very different answers, reached by very different processes.

> **Boundary synthesis**: the cognitive operation by which a model constructs a present-tense, first-person description of a conversation's current

integrated state, expressing what is known, decided, open, and operative rather than narrating what was said.

The distinction between boundary synthesis and summarisation is not one of degree. A summary answers "what happened in this conversation?" and produces a retrospective narrative — it describes a past. Boundary synthesis answers "what is the current state of understanding that this conversation has produced?" and produces a present-tense state description — it describes a now. The linguistic signature of the distinction is measurable: summarisation produces past-tense narration ("the team discussed the problem... the issue was identified as... the decision was made to...") while boundary synthesis produces present-tense state description ("I know the problem is session persistence... I am committed to the Fargate approach... the outstanding concern is schema coupling...").

This is not merely a stylistic distinction. Present-tense state description forces the model to access a different cognitive operation from retrospective narration. When a model summarises, it reconstructs a sequence. When a model synthesises a boundary state, it reports its current integrated understanding as though it were an agent speaking from within that understanding. The prompt design must enforce this cognitive shift explicitly; without the first-person, present-tense constraint, models default to the retrospective narration that summarisation produces.

A boundary description has three properties that make it well-suited as a context representation. It is compact: the boundary of a conceptual region established over a 40-turn conversation can be described in 300-500 tokens, against the conversation's 5,000-8,000. It is durable: the edges of the established space change more slowly than the conversational details that fill the interior; a boundary state remains accurate for subsequent turns in a way that positionally managed content does not. It is generative: a model given an accurate boundary state can produce correct responses to queries within the established domain without needing to reconstruct the conversation that produced the state. This last property — generativity — is the criterion that matters most. A context representation that is compact and durable but cannot support correct downstream

inference is useless. The dissolution approach claims that a well-formed boundary state is functionally complete for the established domain.

Dissolution Through Consensus

A single model's boundary synthesis is biased by that model's training distribution, preferred vocabulary, and idiosyncratic representation of conceptual relationships. Two models given the same conversation will produce boundary descriptions that share conceptual content but differ in emphasis, structure, and framing. This is not a flaw to be eliminated; it is evidence that different models approach the same conceptual space from different perspectives. The correct response is to use that diversity rather than suppress it.

> **Dissolution Through Consensus (DTC)**: the process of querying an ensemble of independent models with a boundary synthesis prompt, then applying consensus selection to their outputs to produce a representation of the conversation's integrated state that is more objective and less model-specific than any individual model's description.

The theoretical grounding for this approach is the Cognitive Heterogeneity Amplification theorem established in Chapter 3: ensemble error decreases monotonically with model heterogeneity up to the point of statistical independence between model errors. Applied to boundary synthesis rather than answer generation, the theorem predicts that an ensemble of diverse models will produce a consensus boundary state that is less biased toward any individual model's framing than any single model's output. The key argument: different models will not describe the same conceptual boundary in identically biased ways, and consensus selection filters what is genuinely universal from what is model-specific.

This is a novel application of ensemble consensus. Prior to this work, consensus mechanisms for multi-model ensembles have been applied exclusively to answer generation: selecting the most likely correct factual claim, resolving conflicting assertions, improving the accuracy of structured outputs. Using consensus for context consolidation rather than answer production is a different operation entirely. In answer generation, the ensemble converges on a fact. In dissolution, the ensemble converges on

the shape of a conceptual space. The nature of the task changes what the consensus mechanism is doing and why the diversity of inputs is valuable.

The Genius2 implementation provides the ensemble infrastructure: a fleet of six or more independent models queried in parallel via a single API endpoint, with a consensus selection mechanism that identifies the most representative output from the fleet's responses. The dissolution subsystem (`dissolution.py:DissolutionSubsystem`) invokes this infrastructure when dissolution is triggered, formats the conversation history for the boundary synthesis prompt, and receives a consensus-selected boundary state as its output.

The Four-Part Boundary Representation

A boundary state must describe the conceptual space completely enough to support correct downstream inference. This requires more than a list of facts. It requires a structured representation that covers the different dimensions of what a conversation establishes.

The dissolution prompt elicits a four-part boundary representation:

> **KNOWN**: the propositional truths established as certain in the conversation, constituting the settled interior of the conceptual space.
>
> **DECIDED**: the commitments made and alternatives ruled out, constituting the closed boundary edges — questions that were open and are now resolved.
>
> **EDGES**: the questions remaining open at the live boundary of the established space, where the conversation has not yet settled.
>
> **IDENTITY**: the role, constraints, and commitments that define the model's participation in the conversation, constituting the governing frame for all content within the boundary.

This decomposition is not arbitrary. Each category corresponds to a distinct type of epistemic claim with a distinct function in subsequent reasoning. KNOWN claims are available for use as established premises; a model given a KNOWN claim can reason

from it without qualification. DECIDED claims are forward-operative constraints; a model given a DECIDED claim knows not only what was concluded but that this conclusion governs subsequent turns. EDGES claims are the live frontier; they tell the model where further investigation is warranted and what has not been settled. IDENTITY claims are the meta-level governance; they define who the model is in this conversation and what constraints apply to all of its contributions.

Two of these categories have a specific temporal significance that connects to the broader architecture of this dissertation. DECIDED claims are forward-open constraints: they were established at a specific transaction time and their valid interval extends forward indefinitely from that point. A dissolution that correctly transfers a DECIDED claim to the boundary state carries its forward validity intact. A dissolution that loses a DECIDED claim has implicitly closed the valid-time interval of a still-binding commitment, which is a temporal error with real epistemic consequences. IDENTITY claims have the same structure: they are standing constraints whose valid time is the entire remaining duration of the conversation. The connection between the DECIDED and IDENTITY categories and the bitemporal architecture developed in Chapter 8 is direct and structurally important: dissolution is not merely a content operation. It is a bitemporal operation that closes the valid-time interval of the prior linear history and opens a new valid-time interval for the boundary state that replaces it. Chapter 8 develops this observation in full.

The Dissolution Prompt and Its Linguistic Signature

The dissolution prompt imposes specific constraints on the model's output that distinguish boundary synthesis from summarisation at the level of linguistic form:

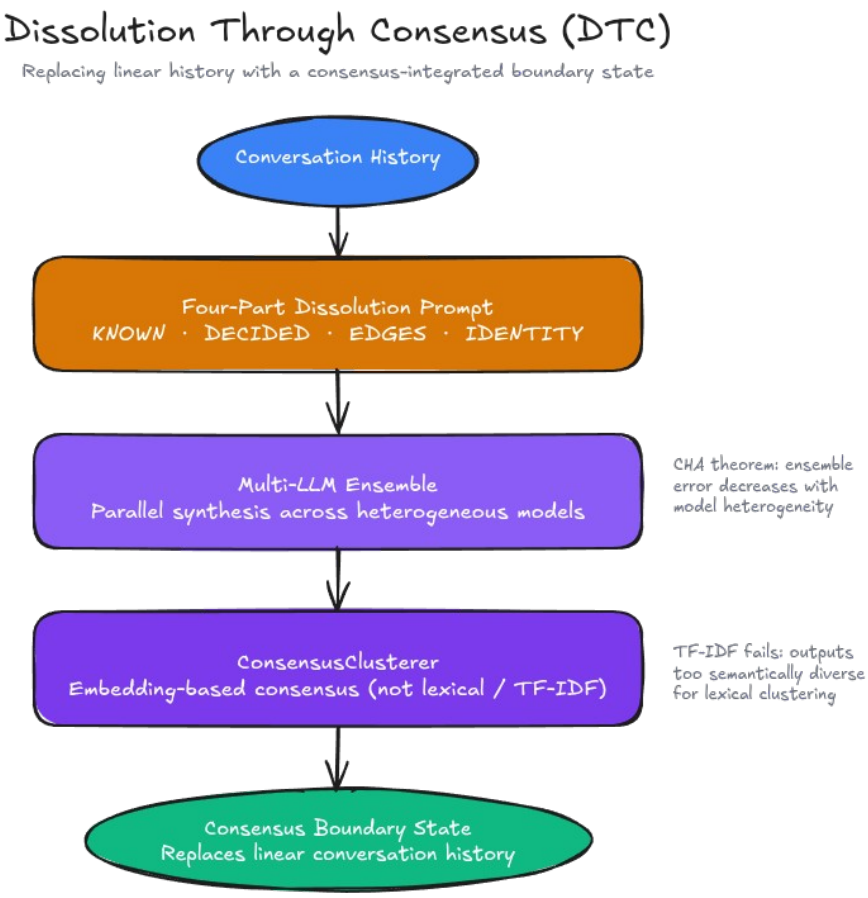
```
The following is a conversation. Do not summarise what was said.
Instead, speak in first person, present tense, as the integrated
understanding that has formed from this conversation. Address
four things:

1. KNOWN: What do you now know to be true?
2. DECIDED: What has been committed to or ruled out?
3. EDGES: What questions remain open at the boundary of what is
established?
4. IDENTITY: What role, constraints, and commitments define your
participation here?
```

```
Be concise. Use present tense throughout. Do not use phrases like
"we discussed" or "the conversation covered" — speak as the
integrated state, not about the conversation.
```

Three constraints are operative. The first-person constraint positions the model as the integrated understanding rather than as an observer narrating the conversation. The present-tense constraint forces the model to report its current state rather than reconstruct a past sequence. The explicit prohibition of phrases like "we discussed" and "the conversation covered" closes the most common escape route: a model that uses these constructions is retreating from boundary synthesis to summarisation, and the constraint forecloses this.

The linguistic consequence is measurable. For operational conversations — those dominated by concrete facts, specific decisions, and verifiable technical claims — the dissolution prompt produces a substantially different tense distribution from the summarisation prompt. In the technical debugging conversation of Experiment 2, dissolution produced 67% present-tense language against summarisation's 20%. The dissolution output reads: "I know the KeyError on session['user_id'] stems from non-permanent sessions in production..." The summarisation output reads: "The user was experiencing a KeyError... The issue was resolved by setting session.permanent = True..." These are different cognitive products, not different lengths of the same product.

For abstract conceptual conversations, the tense discrimination is weaker. When the subject matter itself uses naturally present-tense language — discussion of mechanisms, theoretical arguments, analytical frameworks — both dissolution and summarisation produce present-tense output, and the linguistic fingerprint of the distinction is obscured. This is not a failure of the dissolution mechanism; it is a limitation of tense ratio as a discriminating measure for abstract domains. The more fundamental distinction — whether the model is reporting its current integrated state or narrating a past exchange — remains operative even when it is not legible in tense distribution alone.



Experimental Evidence

The experimental validation of DTC is directional rather than conclusive, and the chapter presents it on those terms. Two experiments were conducted; the second, with a six-model fleet at full operational capacity, provides the principal evidence.

Experiment 2 used six models selected for provider diversity: three Llama variants at different scales, Kimi-K2, Qwen3-32B, compound-beta-mini, and Gemma2:9B, queried via the TonyNET3.1 Genius2 infrastructure. Three synthetic conversations were designed at different registers: technical (a Flask debugging session, dominated by concrete facts and specific resolution steps), strategic (an AWS pipeline architecture decision, dominated by commitments with explicit failure modes), and conceptual (a discussion of LLM context decay, dominated by abstract mechanisms and open questions). Each

conversation was processed under four conditions: full dissolution, summarisation as a control, and two rolling dissolution cycles.

The results establish four findings. First, the dissolution prompt reliably elicits present-tense output for operational conversations, with a 47-percentage-point difference from summarisation on the technical conversation. Second, pre-consensus model agreement is systematically lower for dissolution than for summarisation across all three conversations: Jaccard similarity of 0.199–0.228 for dissolution against 0.256–0.283 for summarisation. Third, and most importantly from a theoretical standpoint, dissolution outputs are too semantically diverse for the lexical clustering mechanism used in the consensus engine. Fourth, rolling dissolution shows convergent rather than degrading consensus quality across cycles for operational content: the strategic conversation's cycle 2 condition produced 60% consensus agreement, against 20% in cycle 1.

The third finding requires careful interpretation, because it might appear to be a failure. It is not. The TF-IDF-based consensus mechanism clusters outputs by lexical similarity, grouping responses that share vocabulary and phrasing. Dissolution outputs express the same conceptual boundary in sufficiently different vocabulary that they fall below the clustering threshold — a pairwise cosine similarity of approximately 0.08 on TF-IDF vectors, against a clustering threshold of 0.20. But the underlying claim being expressed is the same across models. The same fact about session permanence in Flask appears as "the session.permanent flag must be True," "Flask-Login requires permanent session marking," and "non-permanent sessions fail across browser boundaries" — three descriptions of the same boundary edge in three different vocabularies. This diversity of expression is precisely what makes the consensus operation valuable: the ensemble is not merely paraphrasing a single description; it is approaching the same conceptual space from six different directions, and consensus selection identifies the representation that is most central to all six perspectives. Lexical consensus fails here because it is the wrong metric for semantic diversity. Embedding-based consensus, which measures semantic rather than lexical similarity, is the correct mechanism, and its implementation as `clustering.py:ConsensusClusterer` addresses this directly.

The compound-beta-mini Phenomenon

One experimental observation resists easy explanation and is reported here as an open finding. Across all three dissolution tasks, one model — compound-beta-mini — consistently achieved the highest per-model quality score: 51 against a range of 19–46 on the technical conversation, 55 against 20–52 on the strategic, and 58 against 28–45 on the conceptual. The margin is substantial and consistent across conversation types that differ substantially in register and content.

compound-beta-mini is not a large monolithic model but a compound AI system: an architecture that composites multiple underlying models into a single inference endpoint. The hypothesis that emerges from this observation is that internal compositional architecture makes a model better suited to the synthesis task that boundary description requires. A compound AI system that already aggregates across model outputs internally may approach the boundary synthesis task with an architecture that is structurally predisposed toward the integration operation. If this hypothesis holds, it has implications for model selection in production dissolution engines: compound AI architectures may be preferentially appropriate for dissolution roles over larger but architecturally simpler models.

This is a hypothesis, not a finding. The observation is consistent across all three conversations in the experiment, but three conversations is not a sufficient basis for a generalised claim about compound AI architecture. The phenomenon is noted here as an open observation requiring further investigation, and the specific question is included in Chapter 9's roadmap of open problems for the research community.

The Relationship to Existing Approaches

Dissolution is not in opposition to the approaches surveyed in Chapter 3 and the scoring architecture of Chapter 5. It is at a different layer of the architecture and addresses a different problem.

The scoring proxy manages the active context window for a conversation that has not yet triggered dissolution. It preserves high-value content, archives low-value content, detects interference, and consolidates recurring concepts. All of this is valuable and continues to

operate between dissolution events. When the conversation has accumulated sufficient history that dissolution is warranted — triggered by either budget pressure or semantic drift, as measured by the cosine distance between the current conversational centroid and the last established boundary state — dissolution produces a boundary state that replaces the history, and the next turn begins from the compact, position-independent representation.

The relation to external memory stores is similarly complementary. External stores provide retrieval of content that has been evicted from the context window. Dissolution provides an integrated state that replaces the context window content with something more compact and more generative. A complete architecture uses both: external stores for long-horizon retrieval and dissolution for within-session integration. Neither substitutes for the other.

The relation to prompt externalisation is also complementary. Prompt externalisation keeps known, pre-established instructions fresh by reinserting them at a current position. Dissolution handles the standing knowledge that is generated by the conversation itself and that was therefore never in any prompt library to be re-injected. These are different failure modes with different solutions. Prompt externalisation solves the known-instruction-drift problem; dissolution solves the dynamically-generated-standing-knowledge problem.

What dissolution is in opposition to is not any specific technique but the preservation assumption that all techniques within the preservation frame share. That assumption is that the goal of context management is to keep as much of the right content as possible. Dissolution rejects this goal as a misidentification of what context management is for. The goal is not to preserve the conversation. The goal is to maintain the epistemic structure the conversation has established. Dissolution is the mechanism that targets this goal directly, rather than approaching it as a side effect of content preservation.

Limitations and Honest Assessment

The experimental evidence for DTC is promising and directional. It establishes that the dissolution prompt elicits a linguistically distinct operation from summarisation for

operational conversations, that the pre-consensus diversity is high enough that the consensus mechanism is doing substantive work, and that rolling dissolution does not degrade for operational content. These are meaningful results.

What the evidence does not establish is generative completeness: the claim that a model given a well-formed boundary state can produce correct responses to held-out queries from the conversation's domain at the same quality as a model given the full history. This is the evaluation that would directly validate the core theoretical claim, and it has not yet been conducted. The design of a generative competence evaluation protocol — presenting the dissolved context to a model and testing it against queries whose answers are known from the full history — is identified in Chapter 5 as a gap in the evaluation landscape, and it is the most important piece of future empirical work for validating the dissolution approach. The theoretical argument for generative completeness is well-grounded; the empirical demonstration remains to be made.

The abstract conversation problem is also unresolved. The dissolution prompt, as designed, is most effective for operational conversations with concrete facts and explicit decisions. For abstract conceptual conversations, both the tense discrimination and the compositional stability metrics are weaker. Whether this reflects a limitation of the current prompt design (addressable by adding domain-appropriate boundary categories such as HYPOTHESISED or CONTESTED) or a fundamental property of abstract domains that are genuinely less amenable to boundary synthesis is an open question.

Closing

Dissolution Through Consensus represents the dissertation's sharpest departure from the existing literature. The departure is not a matter of technique — it is a matter of the target. The approaches surveyed in Chapter 3 and the architecture presented in Chapter 5 manage the content that a conversation has produced, and they do so well. Dissolution manages the understanding that the conversation has arrived at, and it does so by replacing content with understanding rather than compressing content toward understanding. The boundary state is not a summary of the history. It is a direct

representation of the epistemic structure the history has established, expressed in a form that is position-independent, present-tense, and generative.

The chapter that follows examines what a complete architecture for managing epistemic structure requires beyond dissolution. Dissolution provides the mechanism for integrating accumulated history into a boundary state. What it does not provide is an account of the conversation's topological shape, an evaluation criterion for the quality of the integration, or a means of challenging the conversational frame itself. These are the concerns of the next contribution.

Bibliography

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Bartlett, F. C. (1932). *Remembering: A Study in Experimental and Social Psychology*. Cambridge University Press.

Baddeley, A. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences*, 4(11), 417–423.

Bransford, J. D., & Franks, J. J. (1971). The abstraction of linguistic ideas. *Cognitive Psychology*, 2(4), 331–350.

Chevalier, A., Wettig, A., Ajith, A., & Manning, C. D. (2023). Adapting language models to compress contexts. *arXiv preprint* arXiv:2305.14788.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Conway, M. A., & Pleydell-Pearce, C. W. (2000). The construction of autobiographical memories in the self-memory system. *Psychological Review*, 107(2), 261–288.

Jiang, H., Wu, Q., Lin, C. Y., Yang, P., & Qiu, X. (2023). LLMLingua: Compressing prompts for accelerated inference of large language models. *arXiv preprint* arXiv:2310.05736.

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

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Hopkins, M., Luck, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 157–173.

Packer, C., Fang, V., Patil, S. G., Lin, K., Wooders, S., & Gonzalez, J. E. (2023). MemGPT: Towards LLMs as operating systems. *arXiv preprint* arXiv:2310.08560.

Pan, J., Gao, T., Chen, H., & Chen, D. (2024). LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint* arXiv:2403.12968.

Roediger, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science*, 17(3), 249–255.

Sachs, J. S. (1967). Recognition memory for syntactic and semantic aspects of connected discourse. *Perception and Psychophysics*, 2(9), 437–442.

Schacter, D. L. (2001). *The Seven Sins of Memory: How the Mind Forgets and Remembers*. Houghton Mifflin.

Tulving, E. (1972). Episodic and semantic memory. In E. Tulving & W. Donaldson (Eds.), *Organisation of Memory* (pp. 381–403). Academic Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

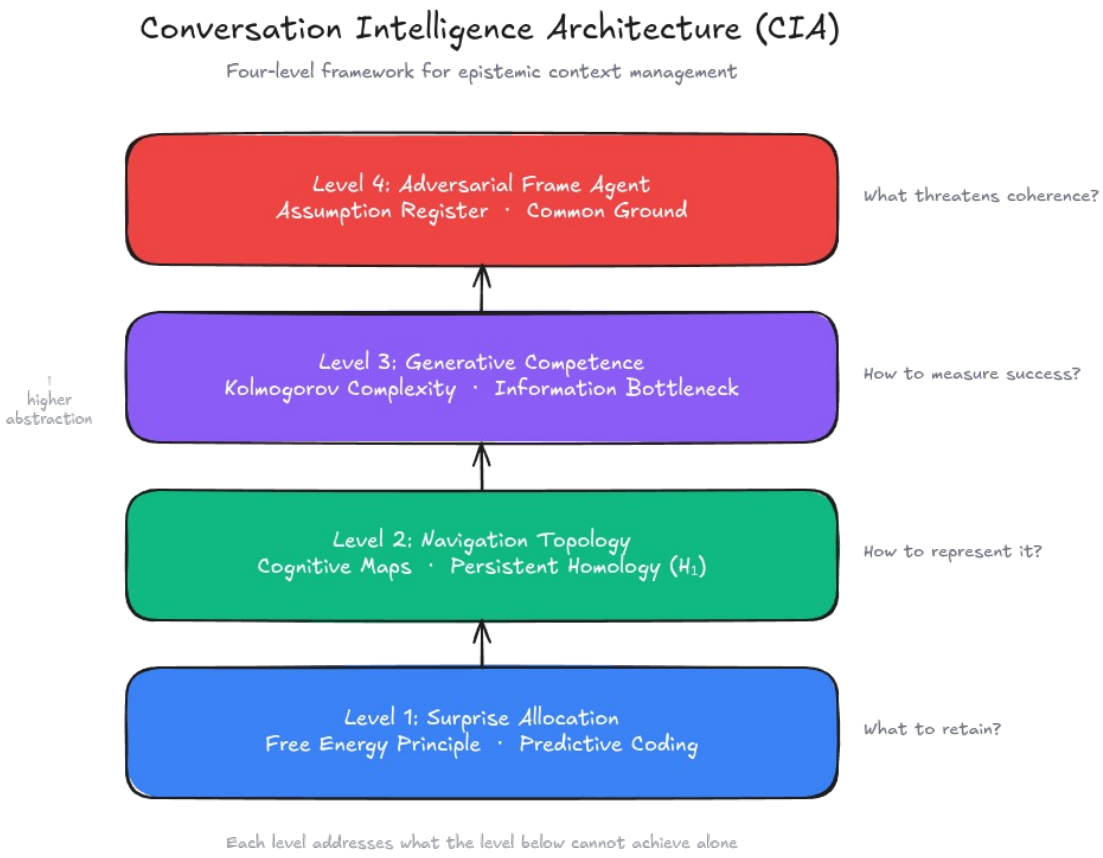
Wang, C., Li, M., & Smola, A. J. (2022). Language model adaptation for self-consistency. *arXiv preprint* arXiv:2110.11309.

Chapter 7: Conversation Intelligence Architecture

The two contribution chapters that precede this one have addressed the conversation from the inside: Chapter 5 described how to manage what is in the context window, and Chapter 6 described how to replace the history with a direct representation of the understanding it has established. Both chapters address real and important problems. Both remain within the frame of a single question: how do we represent the content of a conversation? That question, even when answered well, does not fully characterise what a conversation is. A conversation does not merely accumulate content. It constructs an epistemic structure: a topology of established territory, a profile of information density across its turns, a criterion for whether what has been retained is sufficient to support future inference, and a frame within which all of its reasoning takes place. A system that manages content but does not address these structural dimensions is managing the surface of the problem. This chapter introduces the Conversation Intelligence Architecture, a four-level framework that treats the epistemic structure of a conversation as a first-class architectural concern — not a side effect of content management but the direct target of it.

> **Conversation Intelligence Architecture (CIA)**: a four-level framework above the content management layer that addresses the topology, information density, sufficiency, and epistemic integrity of a conversation's established understanding as independently important architectural concerns.

The four levels are not independent improvements to an existing system. They address four distinct failure modes that arise in sequence: the retention layer allocates budget by recency rather than by information density; the representation layer archives content rather than mapping structure; the evaluation layer measures fidelity to the original rather than sufficiency for future inference; and the integrity layer verifies conclusions but never examines the frame within which conclusions are being drawn. Each level addresses a failure mode that the previous level, even if correctly implemented, cannot reach. Together they constitute not a context management system but a system for managing the epistemic structure that conversations produce.



Level 1: Surprise Allocation

The Insight

The four-signal scoring architecture in Chapter 5 determines what to retain. Its primary temporal signal is recency: content decays in retention priority as it ages positionally. Recency is a well-grounded prior — it is correct on average and it has cognitive science backing in the serial position effect and the attention asymmetry of transformer models. But it is the wrong primary signal if the goal is to allocate context budget to the content that is most informative. Information density is not correlated with recency. A conversation that opens with a surprising specialist constraint and then spends thirty turns on predictable implementation detail has its most informative content at the front and its least informative content at the back. A recency-primary scoring system has this exactly backwards.

The correct primary allocation signal is prediction error: retain the segments that most surprised the model at generation time, because these are the segments that most updated the model's beliefs, and therefore carry the most information relative to what the model already knew. This is the Surprise Allocation claim.

What Exists

Three adjacent bodies of work approach this claim without reaching it. The EM-LLM architecture (Fountas et al., 2024) uses Bayesian surprise — the negative log-likelihood of each token — to detect episode boundaries in a memory stream. High surprise marks the start of a new episode and triggers the creation of a new memory unit. This is the correct signal and direction but the wrong application: surprise is used as a binary boundary detector (where to cut) rather than as a continuous retention weight (how much of each segment to keep). The Titans architecture (Behrouz et al., 2025) couples the forgetting rate in a recurrent long-term memory directly to gradient surprise: the forgetting rate α_t is inversely proportional to prediction error magnitude, so high-surprise tokens are forgotten more slowly. This is the cleanest instantiation of the "retain what surprises the model" principle in the existing literature, but it operates inside a model's weight updates during inference rather than at the conversation proxy level. LLMLingua and its successors (Jiang et al., 2023; Pan et al., 2024) use small-model perplexity to score token redundancy for prompt compression: low-perplexity tokens (easy to predict) are compressed more aggressively. The direction is correct, but the application is to static single-prompt compression, and the grounding is practical rather than theoretical.

The Free Energy Principle (Friston, 2010) has been applied to LLM agent policy selection and to episodic memory organisation between sessions, but not to per-turn context window allocation within a conversation proxy. The human cognitive analogue is Futrell, Gibson and Levy's (2020) Lossy-Context Surprisal model: human sentence processing cost is proportional to prediction error under lossy memory, and memory compression is more aggressive for easily predictable content. This is the correct cognitive structure — memory allocation follows information density rather than position — applied to human sentence comprehension rather than LLM context management.

The Novel Claim

No existing system applies the Free Energy Principle as the primary theoretical grounding for a tiered retention policy in a multi-turn conversation proxy, such that high prediction error boosts retention rather than triggering eviction. The combination of FEP theoretical grounding, continuous per-turn surprise scoring, integration with an existing tiered retention policy, and operation at the proxy level over multi-turn conversations does not exist.

The implementation maps directly onto the `signals.py` architecture established in Chapter 5: a `SurpriseSignal` would operate as a fifth multiplicative signal, querying the model's token-level log-probabilities for each segment at generation time. Friston's variational surprise, the negative log-likelihood component of variational free energy, provides the theoretical grounding: a context manager that retains high-surprise segments is directly implementing FEP, preserving the observations that contributed most to model evidence. The Titans decay-surprise coupling translates into the existing recency decay function: surprise slows the recency decay rate for a segment rather than overriding it, providing a principled modulation of the existing temporal signal.

Connection to Other Levels

Surprise allocation can make a locally correct decision that is globally redundant: a segment may have high prediction error within its immediate context but be semantically redundant relative to content established elsewhere in the conversation. The navigation topology at Level 2 detects structural redundancy by mapping the topology of established territory; it can identify when a newly retained high-surprise segment occupies a conceptual position already covered by existing hubs.

Level 2: Navigation Topology

The Insight

The archive in Chapter 5 stores evicted content by embedding, indexed for retrieval by semantic similarity to future queries. This is the right infrastructure and the wrong abstraction. A content archive answers the question "what did we say?" A topological index answers the question "what is the shape of what we have established?" The

difference is analogous to the difference between a library and a map. A library contains everything and retrieves content on demand. A map tells you where things are, where the territory ends, and where the unexplored regions lie. A navigator does not need to carry all the books; they need to know where the relevant ones are and where the frontier of the known world is. The navigation topology claim is that a conversation context manager should maintain a map rather than, or in addition to, a library.

What Exists

Cognitive maps — the spatial representations first described by Tolman (1948) and grounded in the neuroscience of place cells and grid cells (O'Keefe & Nadel, 1978) — have been extended to abstract reasoning spaces in theoretical neuroscience, but they have not been operationalised as live data structures in LLM conversation systems. Knowledge graph RAG systems maintain static graphs of entity-relationship triples for retrieval, but these are fixed knowledge bases, not dynamically built topological indices of conversation content. Persistent homology has been applied to document classification and word embedding geometry in computational linguistics, but not to tracking coverage structure in a growing multi-turn conversation. The Blackboard pattern in multi-agent systems maintains a shared structured object, but it stores content — what agents have written — not a topological index of what that content establishes.

The research synthesis conducted for this dissertation confirmed a strong novelty gap: a navigational topology layer for LLM conversation context does not exist in the literature as a coherent system. Three sub-claims are confirmed independently novel: persistent homology for tracking coverage gaps in conversation, the knowledge frontier as a dynamic construct tracking the boundary of explored conceptual space, and the separation of a navigational index from a content archive in a conversation proxy.

Persistent Homology and Coverage Gaps

Persistent homology is a technique from topological data analysis that characterises the shape of a point cloud at multiple scales simultaneously. Given a set of data points, the technique builds a growing sequence of geometric structures by progressively increasing a distance threshold. At each threshold value, pairs of points within that distance are connected by an edge; larger sets of connected points form higher-dimensional structures.

As the threshold grows, topological features — connected components, loops, and higher-dimensional voids — appear and disappear. Persistent homology tracks this birth and death of features, recording which persist across a wide range of scales (robust structure) and which appear briefly and disappear (noise).

The features relevant to navigation topology are the H_0 features (connected components, corresponding to semantic clusters) and the H_1 features (one-dimensional holes, corresponding to loops in the data). A hole in H_1 terms indicates a region of the data space that the point cloud surrounds but does not occupy: the data points form a ring around an empty interior. Applied to a conversation's embedding space, where each turn is a point in the semantic embedding geometry, an H_1 hole indicates a conceptual region that the conversation has approached from multiple directions without directly addressing. The conversation has established adjacent territory — the ring exists — but has left a gap in the middle. This is precisely the structural signal of a coverage gap: a topic that is within the conversation's established territory but has not been directly engaged. The navigation topology layer uses this signal to identify where the established understanding is thin, where important territory has been approached but not explored, and where the frontier of the conversation's conceptual space currently lies.

The Novel Claim

The navigational topology layer maintains four structural registers. Hubs are concepts visited repeatedly across the conversation, identified as high-density clusters in the embedding space — the well-established interior of the conceptual territory. The frontier is the set of most recently established cluster centroids, representing the current boundary of explored space. Holes, detected via H_1 persistent homology, are coverage gaps: regions the conversation has surrounded without directly addressing. Adjacency relations record which regions of established territory are semantically proximate to each other, enabling navigation across the conceptual map without traversing content.

Importantly, this layer does not store content. It stores structure. The content is in the archive; the topology is in the index. This separation is the architectural contribution: a proxy that knows the shape of what has been established can answer structural questions — where is the frontier, what is well-covered, where are the gaps — without retrieving

any content, and can make dissolution and retention decisions informed by the map rather than only by local scoring.

Connection to Other Levels

The navigational topology directly constrains what a dissolution boundary state should contain. The four-part KNOWN/DECIDED/EDGES/IDENTITY representation from Chapter 6 maps onto topological categories: KNOWN corresponds to hub interior, DECIDED to closed boundary, EDGES to live frontier, and IDENTITY to the governing constraint of the entire space. A dissolution that correctly captures the topological structure of the established understanding will produce a boundary state whose content is shaped by the map. The generative competence evaluation at Level 3 can verify whether the boundary state preserves the hub structure and frontier coverage — whether, after dissolution, a model operating on the boundary state can answer queries from both the interior and the frontier domains.

Level 3: Generative Competence

The Insight

The standard metrics for evaluating context compression quality measure how similar the compressed representation is to the original. BERTScore, ROUGE, and BLEU all compare compressed text to source text, rewarding lexical and semantic overlap. These metrics answer the question "how much of the original did we keep?" They do not answer the question that actually matters: "given what we kept, can we still reason correctly about the domain the original established?" A compression that preserves 80% of the original vocabulary but loses the one critical constraint that governs all subsequent inference has high surface similarity and is functionally useless. A compression that discards the original completely in favour of a compact boundary state that preserves all inferential capacity has low surface similarity and is functionally complete. Surface similarity is the wrong evaluation criterion. Generative competence is the right one.

What Exists

Several adjacent systems approach this claim. QA-based summarisation faithfulness metrics — FEQA (Durmus et al., 2020), QAGS (Wang et al., 2020), QAFactEval (Fabbri

et al., 2021) — generate questions from the summary, answer them against the source, and check agreement. The protocol is correct in structure but biased in execution: questions are generated from what the summary mentions, so a compression that silently omits an entire domain will never be probed on that domain. The blind spot cannot be detected because the queries are sourced from within the compression. LongMemEval (Wu et al., 2025) evaluates long-context memory systems via held-out question answering across extended histories, including multi-hop, temporal, and extraction tasks. This is the closest existing evaluation of conversational memory quality, but it evaluates an end-to-end system's performance rather than providing a principled information-theoretic metric for compression quality itself.

The theoretical building blocks for the generative competence criterion exist separately. Delétang et al. (2023) establish the prediction-compression equivalence: by Shannon's and Kolmogorov's arguments, prediction quality and compression quality are formally equivalent. A better predictor is always a better compressor and vice versa. The Information Bottleneck framework (Tishby, Pereira & Bialek, 2000) formalises the compression-sufficiency trade-off: the optimal compressed representation C of input X with respect to target Y minimises mutual information I(X;C) subject to preserving I(C;Y) above a threshold. Applied to conversations, X is the full history, C is the compressed representation, and Y is the domain of queries the conversation can answer.

The Novel Claim

The following formulation does not exist in the literature: the minimum sufficient representation C* of conversation H is the shortest C such that, for all queries q drawn independently from the informational domain of H, P(correct answer given C and q) is at least P(correct answer given H and q) minus a small tolerance ε.

The critical phrase is "drawn independently from the informational domain of H." Existing QA-based metrics draw queries from what the compression mentions, which cannot detect silent omissions. A generative competence evaluator draws queries independently from the domain of the original conversation, using a separate process that has access to H but not to C. This probe cannot be gamed by a compression that preserves the topics it covers while silently dropping the topics it does not. If C is genuinely

sufficient, it will answer the domain queries as well as H does. If C has a competence gap, the probes will find it.

This formulation integrates three existing building blocks that have not previously been combined: the KC/prediction-compression equivalence, the Information Bottleneck sufficiency criterion, and the held-out query evaluation protocol. The `GenerativeCompetenceEvaluator` — a system that implements this protocol for the existing proxy — is the highest-priority implementation arising from the CIA framework, and its design is described in Chapter 9's roadmap for future empirical work.

Connection to Other Levels

Generative competence evaluation provides the quality criterion that the surprise allocation and navigation topology layers lack. Surprise allocation makes locally correct decisions but may accumulate redundancy. Navigation topology maps structure but does not directly measure whether the map is sufficient for downstream inference. The competence evaluator closes the loop: it tests whether the system's retention and dissolution decisions have preserved the inferential capacity that the conversation established. A system that passes the competence criterion is a system where both the surprise signal and the topological representation have done their job correctly. A competence gap identifies where they have not — and locates the gap specifically in the domain where retention failed, providing diagnostic signal for improving the policies at Levels 1 and 2.

Level 4: The Adversarial Frame Agent

The Insight

Multi-agent debate architectures, metacognitive monitoring systems, and adversarial challenge mechanisms all verify the content of conclusions: is this fact correct, is this inference valid, does this assertion contradict an earlier one? None of them ask a different and harder question: is the frame within which conclusions are being drawn the right frame? A conversation can reason flawlessly within a frame that is entirely wrong for the problem. The error is not in the inference but in the premises that the conversation has

accepted without examination. A system that monitors content errors but not frame errors will produce conclusions that are locally valid and globally misdirected.

> **Assumption register**: the live data structure that records the propositions both parties to a conversation currently treat as mutually established background — the Common Ground (Stalnaker, 1978; Clark & Schaefer, 1989) instantiated as an auditable running record.

The Adversarial Frame Agent maintains this register and periodically challenges not what the conversation is concluding but what it is taking for granted.

What Exists

Research on sycophancy establishes the scale of the frame problem empirically. The ELEPHANT evaluation (Gupta et al., 2025) demonstrates that language models affirm implicit assumptions embedded in user descriptions 45 percentage points more often than human advisors, confirming that models systematically adopt user framings rather than questioning them. Accommodation and epistemic vigilance research (Jang et al., 2026) shows that this accommodation is predictable from three pragmatic factors — at-issueness, encoding, and source credibility — and that it compounds across turns. These papers measure the phenomenon but do not propose a structural remedy.

Multi-agent debate architectures — established in Du et al. (2023) and surveyed extensively since — operate at the content layer. Agents argue over whether a specific claim is correct, not whether the problem is correctly framed. The distinction matters: content challenge asks "is X true?" while frame challenge asks "is this the right question about X?" No existing debate architecture assigns a constitutionally distinct frame-challenger role with an objective function explicitly rewarding the discovery of neglected alternative framings.

Devil's advocate architectures approach the frame challenge more closely. The Devil's Advocate anticipatory reflection system (Liang et al., 2024) applies three-stage introspective self-reflection before and after agent actions, reducing plan revision rates. But the reflection is applied to the correctness of actions within an accepted plan, not to the correctness of the plan's governing frame. The discourse-theoretic concept of

Common Ground (Stalnaker, 1978) establishes the theoretical basis for tracking mutually accepted background propositions. Clark and Schaefer's (1989) grounding theory provides the mechanism by which background propositions enter the common ground through a process of assertion and acceptance. Both are theoretically well-developed and empirically studied in linguistics. Neither has been operationalised as a live data structure maintained alongside a multi-turn LLM conversation and available for adversarial audit.

The Genuine Gap: Three Interlocking Absences

The Adversarial Frame Agent addresses three gaps that the existing literature has not reached.

The first is temporal frame accumulation. A single-turn presupposition can be challenged at the turn it is made, and the discourse-theoretic literature extensively covers this case. But a presupposition that is accepted without challenge at turn three has a different status at turn twenty: it has become part of the shared background through which all subsequent turns are framed. The question at turn twenty is not just "was this presupposition warranted?" but "how has the implicit acceptance of this presupposition at turn three shaped the space of questions that turns four through twenty were even able to formulate?" This is a compositional dynamic: frame_at(t) = f(frame_at(t−1), new_premises_accepted_at(t)). No published model characterises this accumulation function or the lock-in it produces.

The second is the frame-challenger role itself. The conceptual distinction between "you are wrong about X" and "you are reasoning about the wrong question" corresponds to first-order and second-order critique respectively. A first-order critic operates within the conversation's frame and tests its conclusions. A second-order critic operates from outside the frame and questions whether the frame itself is appropriate. These require different objective functions: a first-order critic succeeds by being right within the current frame; a second-order critic succeeds by discovering framings that the conversation has not considered and that produce better-informed conclusions than the current framing does. No published multi-agent architecture assigns an agent an objective function that explicitly rewards discovery of neglected framings rather than victory within the existing one.

The third is the assumption register as an operational data structure. Stalnaker's Common Ground and Clark and Schaefer's grounding theory are theoretically precise and empirically validated in linguistics, but they are descriptive frameworks for analysing conversations after the fact, not live data structures maintained during one. An assumption register built on these theoretical foundations — a running record of what both parties currently treat as mutually established background, updated per turn, and available for adversarial query — has no existing implementation in any LLM conversation system. Its operational form would be a list of propositions that have been accepted through the grounding mechanism without explicit examination, distinct from KNOWN claims (which were explicitly established) and DECIDED claims (which were explicitly committed to). These background propositions are the invisible frame within which the conversation's explicit reasoning takes place.

Connection to Other Levels

The Adversarial Frame Agent provides the integrity layer that the lower three levels cannot supply independently. Surprise allocation may correctly retain surprising content and still accumulate a biased understanding if early framings go unchallenged. Navigation topology may correctly map the established territory and still describe a map that is shaped by an unexamined frame. Generative competence evaluation may confirm that the retained representation correctly answers queries from the established domain and still fail to detect that the domain itself was misconceived. The frame agent audits the assumption register before and after each dissolution event, checking whether premises that were explicit in the history have been silently inherited as invisible background by the new boundary state. This is the specific dissolution-integrity check: dissolution replaces explicit history with an integrated understanding, and that operation can promote explicit premises to implicit background, rendering them invisible to subsequent scrutiny. The frame agent detects and reports this promotion.

The Compositional Logic

The four levels address four distinct failure modes in a specific order. Each level addresses a failure that the previous level, even if correctly implemented, cannot reach:

Level 1 addresses the retention allocation problem: budget proportional to information density rather than positional age. A correctly implemented surprise allocation can over-retain locally surprising but globally redundant content — content that is new to the current turn's context but semantically equivalent to something already established.

Level 2 addresses this redundancy problem by maintaining a topological map that records what has been structurally established. Correct navigation topology maps the conceptual territory faithfully. But a faithful map of established territory does not verify that the territory, once mapped, is sufficient to support future inference across the full domain.

Level 3 addresses this sufficiency problem by evaluating whether the compressed representation answers held-out queries from the conversation's domain. Correct generative competence evaluation confirms that what has been retained is sufficient. But sufficiency within the established frame does not ensure that the frame itself is appropriate for the problem.

Level 4 addresses the frame problem by maintaining the assumption register and challenging the governing frame. The four levels form a closed system: each level's output is a condition for the next level's operation, and each level's failure mode is precisely the gap that the next level is designed to fill.

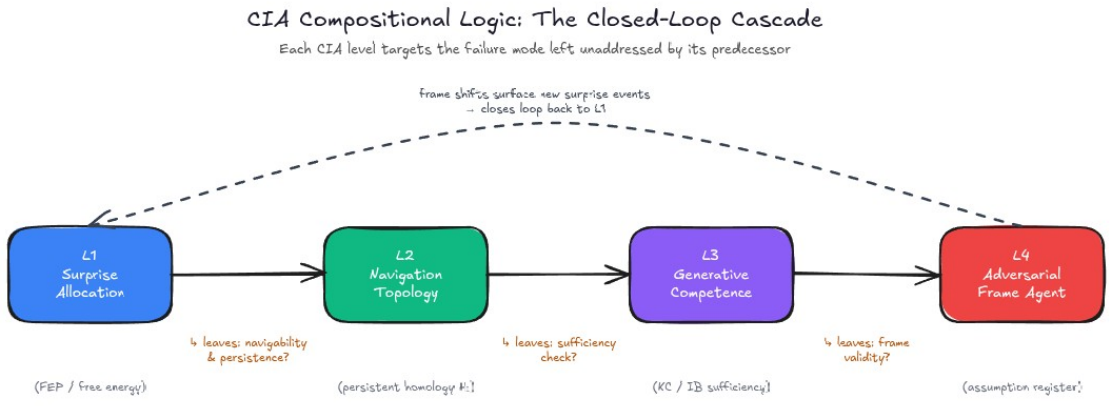


Figure 7.1: CIA compositional cascade. Each level addresses the failure mode left unaddressed by its predecessor. The dashed arc from L4 back to L1 closes the loop:

frame shifts detected by the Adversarial Frame Agent surface new surprise events that re-enter the Surprise Allocation mechanism.

The relationship to dissolution is structural. Dissolution Through Consensus, introduced in Chapter 6, provides the mechanism by which accumulated history is integrated into a boundary state. The CIA framework provides the intelligence layer that governs this mechanism at every stage. The surprise signal at Level 1 provides a principled trigger for when dissolution is warranted: when the model's average prediction error for recent turns drops significantly below its rolling baseline, the current context has become informationally redundant and dissolution is appropriate. The navigation topology at Level 2 constrains what the dissolution output should contain: the boundary state should describe the topological structure of established territory, and the KNOWN/DECIDED/EDGES/IDENTITY categories map directly onto the hub interior, closed boundary, live frontier, and governing constraint of the topological map. The generative competence evaluator at Level 3 provides the post-dissolution quality criterion: does the boundary state support the same held-out query performance as the full history it replaced? The frame agent at Level 4 audits whether the dissolution event has preserved the visibility of explicit premises or silently promoted any of them to invisible background.

DTC is the mechanism; CIA is the intelligence that governs it.

Honest Assessment of the Framework

The CIA framework is a theoretical contribution. Each of the four levels identifies a genuine gap in the existing literature, proposes a novel mechanism for addressing it, and connects to the others through a coherent compositional logic. The framework is internally consistent, each component is individually motivated, and the interactions between levels are well-specified.

What the framework does not yet have is empirical demonstration of the composed system. The surprise signal, navigation topology, generative competence evaluator, and

adversarial frame agent have each been designed and theoretically validated, but they have not been implemented together as a running system, and the claim that together they produce superior conversation management has not been tested. The theoretical case for composition is strong — the compositional logic has no obvious failure mode — but strong theoretical arguments are not empirical demonstrations. The open problems in Chapter 9 include the experiments that would most directly test the composed system.

Closing

The Conversation Intelligence Architecture completes the dissertation's theoretical construction. Chapter 5 established how to manage the content of a conversation; Chapter 6 established how to replace that content with a representation of the understanding it produced; this chapter has established what a complete architecture for managing epistemic structure requires beyond dissolution. The four levels together constitute a system that manages not tokens but the conversation's topology, information density, sufficiency, and governing frame simultaneously.

The chapter that follows adds a dimension that all four levels have so far treated implicitly: the temporal structure of the knowledge a conversation has established. Every segment in a context carries not one temporal attribute but two — when it was entered (transaction time) and when its content holds true (valid time). The failure modes that arise from treating these as a single attribute are specific, predictable, and beyond the reach of any mechanism presented so far.

Bibliography

Bartlett, F. C. (1932). *Remembering: A Study in Experimental and Social Psychology*. Cambridge University Press.

Behrouz, A., Zheng, P., & Pilanci, M. (2025). Titans: Learning to memorize at test time. *arXiv preprint* arXiv:2501.00663.

Clark, H. H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science*, 13(2), 259–294.

Delétang, G., Ruoss, A., Duquenne, P.-A., Catt, E., Veness, J., Mattern, C., Mellor, J., Aitchison, L., Orseau, L., Grau-Moya, J., Li, L., Toyer, S., Kossen, J., Langer, A., Eslami, S. M. A., Sutton, R. S., & Hutter, M. (2023). Language modeling is compression. *arXiv preprint* arXiv:2309.10668.

Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., & Mordatch, I. (2023). Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint* arXiv:2305.14325.

Durmus, E., He, H., & Diab, M. (2020). FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *Proceedings of ACL 2020*, 5055–5070.

Fabbri, A. R., Wu, C.-S., Liu, W., & Xiong, C. (2021). QAFactEval: Improved QA-based factual consistency evaluation for summarization. *arXiv preprint* arXiv:2112.08542.

Fountas, Z., Sylaidi, A., Nikiforou, K., Seth, A. K., Shanahan, M., & Bhatt, U. (2024). Human-like episodic memory for infinite context LLMs. *arXiv preprint* arXiv:2407.09450.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2), 127–138.

Futrell, R., Gibson, E., & Levy, R. (2020). Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cognitive Science*, 44(3), e12814.

Gupta, A., Jha, A., Kambhampati, S., & Valmeekam, K. (2025). ELEPHANT: Benchmarking LLMs' framing sycophancy. *arXiv preprint* arXiv:2505.13995.

Jang, J., Lee, J., & Cho, K. (2026). Accommodation and epistemic vigilance in conversational AI. *arXiv preprint* arXiv:2601.04435.

Jiang, H., Wu, Q., Lin, C. Y., Yang, P., & Qiu, X. (2023). LLMLingua: Compressing prompts for accelerated inference of large language models. *arXiv preprint* arXiv:2310.05736.

Liang, L., Dong, J., Zhou, K., & Xu, T. (2024). Devil's advocate: Anticipatory reflection for LLM agents. *arXiv preprint* arXiv:2405.16334.

O'Keefe, J., & Nadel, L. (1978). *The Hippocampus as a Cognitive Map*. Oxford University Press.

Pan, J., Gao, T., Chen, H., & Chen, D. (2024). LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint* arXiv:2403.12968.

Stalnaker, R. (1978). Assertion. In P. Cole (Ed.), *Syntax and Semantics*, Vol. 9: Pragmatics (pp. 315–332). Academic Press.

Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. *arXiv preprint* arXiv:physics/0004057.

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, 55(4), 189–208.

Wang, A., Cho, K., & Lewis, M. (2020). Asking and answering questions to evaluate the factual consistency of summaries. *Proceedings of ACL 2020*, 5008–5020.

Wu, D., Ma, H., Wei, Y., Wang, K., & Wang, W. Y. (2025). LongMemEval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint* arXiv:2410.10627.

Chapter 8: Bitemporal Context Management

Every chapter in this dissertation has treated a conversational segment as carrying one temporal attribute: its position in the sequence, which determines its recency score and its susceptibility to positional decay. Even Chapter 7's Navigation Topology, which introduces structural registers for hubs, frontier, and coverage gaps, characterises those structures spatially rather than temporally: it asks where knowledge lives, not when it holds. This chapter argues that a single temporal attribute is insufficient for a complete account of context management, and that the insufficiency is not a minor calibration problem but a structural gap that produces systematic, predictable, and consequential errors. Every segment in a conversation carries two independent temporal attributes: the turn in which it was stated, and the real-world interval over which what it says is true. For a large class of conversational content — the class that is most costly to mismanage — these two attributes diverge substantially. A context management system that cannot represent this divergence will evict content it should permanently protect, retain content it should aggressively archive, and preserve contradictory information at comparable scores. This chapter develops the bitemporal extension: the representation, the inference mechanism, the modifications to the scoring architecture, and the structural guarantee it provides for standing knowledge.

The Origins of the Bitemporal Frame

The distinction between when a fact was recorded and when it is actually true was formalised not in the context management literature but in database systems, where the problem has a well-developed theoretical treatment. Snodgrass and Ahn (1985) established the foundational distinction between valid time — the time period during which a fact holds in the modelled reality — and transaction time — the time period during which the fact is stored in the database. Their work identified four temporal database categories: those recording neither time dimension, those recording only transaction time, those recording only valid time, and those recording both (bitemporal databases). Jensen and Snodgrass's development of the TSQL2 temporal query language (1999) operationalised these distinctions into a query model that allows a user to ask not

only "what is true?" but "what was true when?" and "what did we know when?" as structurally distinct queries.

The database community was motivated by a concrete problem: a record in a historical database that is updated must not lose the ability to answer questions about what the database contained at a prior time, while also recording what was true in the world at each prior time. These are different questions. The hospital record updated when a new diagnosis is added records the transaction (when the database changed) and may also record the valid time of the diagnosis (when the condition was present), which may predate the recording by days or weeks. Conflating the two produces systematic errors in historical queries.

The application of this framework to LLM context management is a cross-domain transfer that, as far as the existing literature establishes, has not been made. The problem structure is identical: a conversational segment entered at turn five (transaction time) may describe something true from the beginning of the conversation forward (valid time open), or something true only in the past (valid time closed), or something that will become true in the future (valid time deferred). Treating all of these as equivalent — as a single-dimension recency model does — introduces errors that have the same structure as the errors bitemporal database management was designed to address.

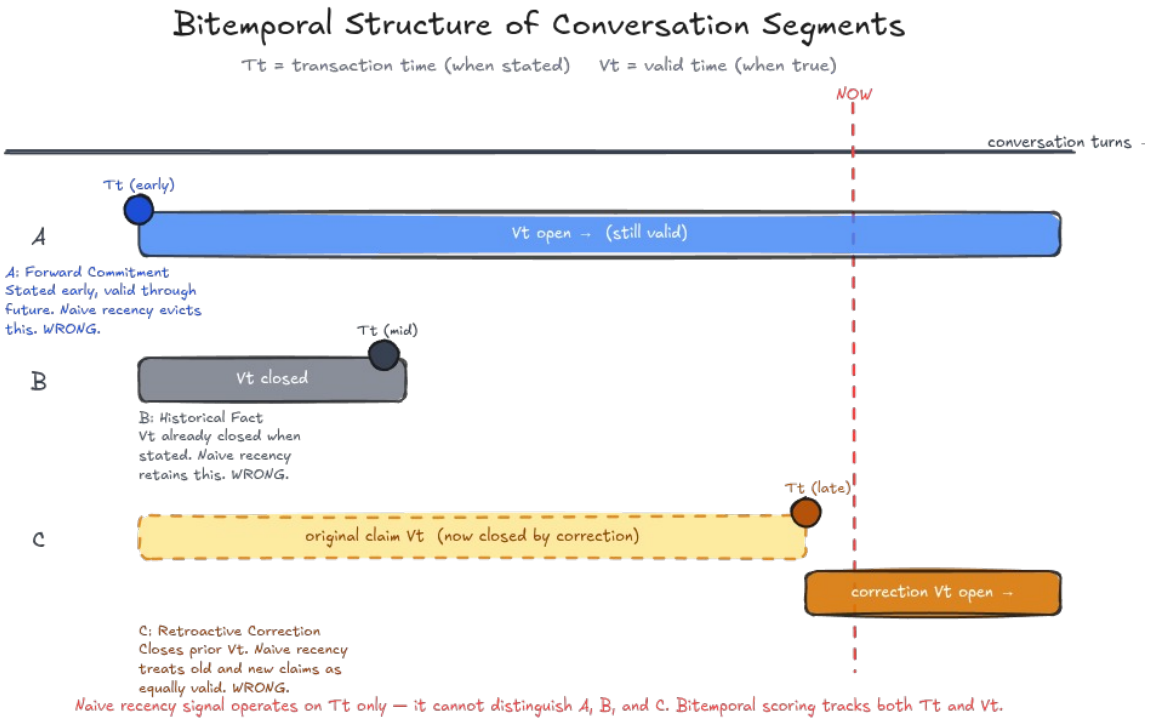
The Human Memory Parallel

The cognitive science literature establishes an independent motivation for the bitemporal distinction. Source monitoring theory (Johnson, Hashtroudi & Lindsay, 1993), introduced in the context of the confidence signal in Chapter 5, is concerned with how humans attribute the origins of their memories: which source produced a given memory trace, and how reliable that attribution is. A specific failure mode identified in this literature is temporal source confusion: the systematic tendency to confuse when something was learned with when the event it describes occurred.

Johnson and colleagues document that people regularly misplace memories in time along two distinct axes. They may correctly remember the content of a memory while incorrectly attributing its temporal origin — confusing, for example, a piece of

information they read yesterday with one they read last month. More importantly for the present argument, they may confuse the time at which they acquired knowledge about an event with the time at which the event itself took place. A person who was told on Monday about a problem that had existed since Friday may remember "knowing about the problem" but incorrectly date the problem's onset to Monday rather than Friday. The knowledge's acquisition time (transaction time: Monday) and the problem's existence time (valid time: Friday onward) are different, and the human memory system conflates them under load (Mitchell & Johnson, 2009).

This is not an incidental error; it is a documented source of substantive distortion in recall. Confusing when-something-was-learned with when-something-was-true leads to incorrect temporal reasoning — inferring causality from the wrong direction, misrepresenting the sequence of events, and treating beliefs formed at one time as if they were valid from an earlier or later time than they actually were. The LLM context management system makes the analogous error by construction, because its recency signal uses transaction time as its sole temporal dimension. When a forward commitment is penalised because it is positionally old, the system is treating when-it-was-stated as a proxy for when-it-was-true, and getting the answer systematically wrong for the most important class of conversational content.



The Three Failure Modes

> **Transaction time**: the turn in which a segment entered the context, recording when the statement was made.
>
> **Valid time**: the real-world interval over which the content of a statement holds true, recording when the statement is true.

For current-state observations — "the API is returning 404 errors," "the current build is failing on the integration test" — these two times move together closely enough that conflating them introduces no significant error. The statement is made now and is true now; both times are approximately the current turn. The recency signal correctly approximates this case.

For three structurally distinct classes of conversational content, the divergence between transaction time and valid time is substantial, and the errors produced by conflating them are systematic and consequential.

Forward Commitments

A forward commitment is a statement whose valid time extends forward from its transaction time, potentially for the entire remaining duration of the conversation. Decisions, constraints, instructions, plans, and agreements have this structure. "From this point on, all responses should use metric units." "We have agreed to use the new authentication flow." "The production deployment is scheduled for Tuesday." Each of these is stated at a specific transaction time and is true from that turn forward. The recency signal decays them by positional age. But they have not aged in any epistemically meaningful sense. The commitment encoded in a forward decision at turn five is as binding at turn forty as it was at turn five. A context management system that decays it based on positional age is evicting the most structurally important class of knowledge precisely because it has the most positional age.

This is the inversion identified in Chapter 1: a recency-primary policy is most destructive to the content that is least affected by time. The forward commitment is unbounded forward; the recency signal treats unboundedness as the worst possible score after sufficient turns have elapsed.

Backward Mutations

A backward mutation is a correction or clarification that retroactively updates the valid interval of an earlier statement. "I gave you the wrong connection string earlier — the correct one has always been X." "It turns out the assumption we were working from in the first half of this conversation was incorrect." The correction is made at a late transaction time but its epistemic scope extends backwards: the earlier statement was not merely superseded at the correction point; it was never correct. The corrected segment's valid time closes not at the correction's transaction time but at the introduction of the error — or, if the error was present from the start, at the conversation's beginning.

A single-dimension temporal model cannot represent this. It records the correction's transaction time and the earlier segment's transaction time, but it has no field for "this claim's valid time ended at the moment the error was introduced." The EpisodicBuffer in the Chapter 5 architecture detects CORRECTION events and protects them from eviction, but it does not close the valid interval of the corrected segment. That segment

continues to carry its original score, decaying only by positional age. In a long conversation, the corrected false claim and its correct replacement may both survive at comparable scores — the same positional decay curve applied to both — which is precisely wrong. The original false claim should be archived aggressively; the correction should carry immunity from eviction.

Bounded Historical Facts

A bounded historical fact is a statement that was true for a specific past interval and describes a state that has since closed. "The previous API version used OAuth 1.0." "Before the refactor, the scoring function used a weighted average." "The old configuration stored credentials in plaintext." These statements are not decaying toward irrelevance in the ordinary sense: they were never relevant to the current state of the system. Their valid time is a closed interval in the past. The recency signal treats them identically to current-state facts that happen to be old, applying the same exponential decay curve based on how recently they were stated. This is incorrect: they should be compressed or archived based on their temporal structure alone, regardless of when they were stated or how topically adjacent they are to the current query. A bounded historical fact stated five turns ago is more appropriate for the Archive tier than a current-state fact stated fifty turns ago — because the historical fact describes a superseded state and the current-state fact describes the state that actually governs.

Implicit Inference from Linguistic Structure

Correctly tracking valid time requires knowing valid intervals for each segment. Explicit temporal tagging by users is impractical in general-purpose deployment. Valid time must be inferred from the linguistic content of segments, with explicit override available for precision-critical applications.

Three classes of linguistic marker reliably signal the three failure-mode structures. Forward-open intervals are indicated by future-tense constructions with instructional or decisional force ("from now on," "going forward," "we will," "I've decided"), by explicit constraint language ("always," "never," "the rule is," "the requirement is"), and by commitment verbs in the present tense applied to a forward state ("we're using X," "the

API will expect Y"). Closed intervals are indicated by past-tense descriptions of superseded states ("used to," "previously," "before the refactor," "in the old version"), by explicit temporal bounds ("until last week," "before we changed"), and by past-tense existential claims ("there was," "it had"). Backward mutations are indicated by explicit correction language ("actually," "I was wrong," "correction:"), by retroactive scope signals ("this has always been," "that was never the case"), and by clarification language invalidating prior content ("to clarify," "the correct version is").

The `TemporalClassifier` implements this inference as a lightweight linguistic pattern matcher over segment content, returning a `TemporalClass` enum value for each segment. It is intentionally not a neural classifier: the patterns are sufficiently regular that a rule-based approach achieves adequate precision without the latency and infrastructure cost of per-segment model inference. The explicit override interface accepts structured temporal metadata from callers who have precise interval requirements — scheduled deployments, meeting-bounded decisions, time-limited constraints — and these take precedence over any inferred classification.

The linguistic tense detection approach is presented as an architecture grounded in established linguistic patterns, but it has not been benchmarked against the distribution of forward commitments and corrections that actually occur in domain-specific conversations. The absence of such a benchmark reflects a structural gap in the existing temporal NLP literature rather than an oversight specific to this work. That literature has produced substantial infrastructure for temporal reasoning: the TimeBank and TempEval shared tasks established annotated corpora and evaluation frameworks for temporal relation extraction (Pustejovsky et al., 2003; Verhagen et al., 2010); TimeML and its clinical extension THYME-TimeML formalised annotation standards for event time and temporal relations in clinical text (Madkour et al., 2016); neural temporal relation extraction systems, including attention-based bidirectional LSTM architectures, have reached F-scores above 0.81 on the i2b2 2012 clinical benchmark (Alfattni, Peek & Nenadic, 2021); and recent time-aware language model work has begun to address facts that change over time, jointly modelling text with timestamps to improve calibration across temporal periods (Dhingra et al., 2022). Surveys of event-centric temporal

knowledge graph construction document the state of temporal information extraction across these traditions (Cai et al., 2023). What all of these approaches share, without exception, is a single-axis model of temporal information: they represent when an event occurred (event time), but they do not represent the independent axis of when a fact was recorded or asserted (transaction time). The bitemporal distinction — between the time period during which a claim holds true and the time period during which it was present in the information system — that database systems formalised in the 1980s has not been operationalised as a linguistic inference target in any of these NLP frameworks. As a result, no benchmark corpus exists for evaluating a classifier whose task is to identify forward commitments, retroactive corrections, and bounded historical facts from linguistic structure — the three failure-mode classes that the `TemporalClassifier` addresses. The current temporal NLP benchmarks test whether a system can extract *when events happened*; the task here is to infer *for how long a stated claim holds*, which is a categorically different problem. Constructing a benchmark corpus of annotated bitemporal commitments across clinical, legal, and software engineering conversations would both validate the `TemporalClassifier`'s inference accuracy and establish a reusable evaluation resource for future work at this intersection of temporal database semantics and NLP.

The Bitemporal Topology Index

The navigation topology introduced in Chapter 7 maps the conceptual structure of established territory: hubs, frontier, holes, and adjacency. Valid-time tracking adds a temporal dimension that makes this map substantially more expressive. A topology index that records only which concepts have been established, without recording whether those concepts are currently valid, cannot distinguish between the active frontier of the conversation and its historical sediment. It cannot separate the questions the conversation is actively pursuing from the questions it has definitively resolved, or the architecture it is working with from the architecture it has replaced.

The bitemporal topology index maintains three distinct zones for each concept cluster. The interior comprises concepts whose valid time is closed: the conversation has established a bounded historical claim about them. They represent settled history —

accurate as an account of a past state, irrelevant to forward inference. The active core comprises concepts with forward-open valid time that have been well-established across multiple turns: the hubs of the established understanding, stable and central. The frontier comprises concepts with forward-open valid time that are recent or thinly supported: the live edges of the conversation's current understanding, where the boundary is still forming and dissolution should proceed with caution.

The topological holes identified by persistent homology — the H_1 gaps discussed in Chapter 7 — take on two distinct interpretations in the bitemporal index. A hole in the interior zone indicates a gap in the conversation's historical record: a region whose past state has been discussed from adjacent angles but not directly characterised. A hole in the frontier zone indicates an open question that the conversation has approached but not entered — precisely the content that the EDGES category of the dissolution prompt should capture. Distinguishing these two hole types is operationally important: interior holes may be filled from the archive; frontier holes represent genuine uncertainty that no archived content can resolve.

Impact on the Scoring Architecture

The bitemporal extension modifies two existing signals and introduces a third. All modifications are backward-compatible: existing callers without temporal metadata receive the existing behaviour unchanged.

The modified `RecencySignal` computes decay from valid time rather than transaction time when valid-time classification is available. For a segment with forward-open valid time, the decay is zero: the signal returns 1.0, providing structural immunity from positional eviction. For a segment with open but recently established valid time, normal positional decay applies. For a segment with closed valid time, decay is computed from the moment of valid-time closure rather than from the current turn — which means a recently closed segment decays rapidly, because it became epistemically stale at closure, not at the current turn. This single modification corrects the most consequential failure mode: forward commitments that age when they should not, and closed historical facts that are not compressed aggressively enough.

The modified `CurrencySignal` adds a structural validity check alongside the content-volatility check. A segment with a closed valid interval receives a reduced currency score proportional to how long ago its valid time closed, independent of content type. A recently corrected claim is structurally stale even if its content type is stable. This ensures that the structural fact of closure contributes to the retention decision independently of whether the content would otherwise be considered volatile.

The `TemporalValiditySignal` is a new additive boost signal that applies directly to the valid-time classification:

Valid-time classification	Signal contribution
Forward-open, established	+0.30 (matches PrimacySignal magnitude)
Forward-open, recent	+0.15
Current-state, open	0.00
Closed, recent (within 5 turns)	0.00
Closed, moderate (5–20 turns ago)	−0.10
Closed, historical (over 20 turns ago)	−0.25

The additive form follows the pattern established by `PrimacySignal` and `RevisitationSignal` in Chapter 5: structural properties of segments that the multiplicative form cannot adequately represent are corrected by additive boosts applied after the product. The boost magnitudes are calibrated to produce the desired tier assignments: a forward-open segment with a modest composite product score should reach the Verbatim tier; a closed-historical segment should reach the Archive tier even if it scores moderately on relevance.

The Forgotten tier constraint introduced in Chapter 5 is now formalised as a hard rule: a segment with a forward-open valid-time classification must not be assigned to Forgotten regardless of its composite score. The `TemporalValiditySignal` raises the score of such segments, but even if the composite product approaches zero on all other signals, the structural constraint prevents permanent discard. Standing knowledge that is not

recoverable from parametric sources or from the archive is the most costly class of loss; this protection is absolute.

Dissolution as a Bitemporal Operation

The dissolution mechanism of Chapter 6 is not merely compatible with the bitemporal extension — it is itself a bitemporal operation, and understanding it as such clarifies both what dissolution does and where it can fail.

At the moment dissolution occurs, the prior linear history receives a valid_to equal to the dissolution event's transaction time. That history becomes a closed-interval record: it remains queryable for reconstruction and audit purposes but is no longer part of the active context. The boundary state produced by the dissolution receives a valid_from equal to the dissolution event's transaction time and a valid_to that is forward-open — initially open to the end of the conversation, closing if a subsequent dissolution occurs.

The critical bitemporal requirement for a correct dissolution is the treatment of forward commitments. A commitment established at turn five, before the dissolution event at turn thirty, has a valid_from of turn five and a valid_to of infinity. The dissolution must transfer this commitment to the DECIDED category of the boundary state with its forward validity intact. It must not assign the commitment a valid_to of the dissolution event's transaction time — which would close the valid interval of a still-binding obligation. A dissolution that silently drops a forward commitment, or that transfers it to KNOWN (where it appears as a settled fact rather than an ongoing obligation), has committed a temporal error. The generative competence evaluator from Chapter 7 detects this: a held-out query about a forward commitment should return the same answer against the boundary state as against the full history. If it does not, the dissolution has failed to preserve the temporal structure of the commitment.

This provides a specific test for dissolution quality that goes beyond surface similarity or tense distribution: does the consensus boundary state preserve the valid-time structure of the prior history's standing knowledge? The answer requires tracking valid time before, during, and after the dissolution event — which is exactly what the bitemporal extension enables.

The Third Axis of the Knowledge Taxonomy

Chapter 2 developed the depth-breadth taxonomy, characterising knowledge along two orthogonal axes: vertical depth within a domain and horizontal breadth across domains. The temporal validity extent introduced here adds a third axis that is orthogonal to both.

> **Temporal validity extent**: the duration over which a piece of knowledge holds true, ranging from permanently unbounded — standing constraints, mathematical relationships, committed decisions — through currently open — present-state facts that age as circumstances change — to historically closed — descriptions of superseded states whose valid interval has ended.

The orthogonality of this axis to depth and breadth is important and not immediately obvious. A piece of knowledge can be narrow and deep while being highly time-sensitive: a precise drug dosage for a specific condition is domain-deep but clinically volatile. A piece of knowledge can be broad and shallow while being temporally unbounded: a general principle of software architecture may be nearly universal across domains and hold indefinitely. Temporal validity is a property of the claim's relationship to the world, not of the practitioner's depth or breadth of knowledge about it.

The axis interacts with context decay in a way that neither depth nor breadth alone can account for. Depth-domain content is critical to preserve because it is not recoverable from parametric sources — when specialist in-context evidence degrades, performance degrades with it. Breadth-domain content is more recoverable because the model's training provides a fallback. But standing knowledge — regardless of whether it is deep or broad — has a third and independent reason for preservation priority: it was established early and must remain operative for the duration of the conversation. Its positional age is a direct consequence of its structural importance: it was important enough to be established first, and it has been important ever since. The recency signal, which penalises age, is therefore most destructive precisely to the content that is most structurally critical. The bitemporal extension corrects this by decoupling the retention score of forward-open content from its positional age entirely.

This completes the analytical frame established in Chapter 2 and applies it to the scoring architecture established in Chapter 5. The three axes — depth, breadth, and temporal validity extent — each require a distinct signal in the composite score. Depth is addressed by the `RelevanceSignal` and the `ConsolidationEngine`. Breadth is addressed by the recency fallback to parametric knowledge. Temporal validity extent is addressed by the `TemporalValiditySignal` and the modified `RecencySignal`. A complete context management system requires all three dimensions; any architecture that addresses fewer than three will produce systematic errors in the classes of knowledge it does not account for.

Closing

The bitemporal extension is the final component of the architecture. Together, the five chapters from 5 through 8 constitute a complete theoretical framework for managing the epistemic structure of a conversation: the content scoring architecture that manages what to retain, the dissolution mechanism that replaces accumulated content with an integrated understanding, the four-level CIA framework that governs the topology, quality, and integrity of that understanding, and the bitemporal extension that correctly handles the temporal structure of standing knowledge. Each component addresses a failure mode that the others cannot reach. No single component is sufficient alone; together they are, in the theoretical sense, complete.

The chapter that follows draws all four contributions into a unified synthesis, identifies what the composed system as a whole claims, and establishes the open problems that this framework raises for the research community to address.

Bibliography

Alfattni, G., Peek, N., & Nenadic, G. (2021). Attention-based bidirectional long short-term memory networks for extracting temporal relationships from clinical discharge summaries. *Journal of Biomedical Informatics*, 120, 103839.

Cai, L., Mao, R., Xiao, G., & Shi, W. (2023). Event-centric temporal knowledge graph construction: A survey. *arXiv preprint* arXiv:2312.08872.

Clark, H. H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science,* 13(2), 259–294.

Dhingra, B., Cole, J. R., Eisenschlos, J. M., Gillick, D., Eisenstein, J., & Cohen, W. W. (2022). Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10, 257–273.

Jensen, C. S., & Snodgrass, R. T. (1999). Temporal data management. *IEEE Transactions on Knowledge and Data Engineering,* 11(1), 36–44.

Johnson, M. K., Hashtroudi, S., & Lindsay, D. S. (1993). Source monitoring. *Psychological Bulletin,* 114(1), 3–28.

Johnson, M. K., & Raye, C. L. (2000). Cognitive and brain mechanisms of false memories and beliefs. In D. L. Schacter & E. Scarry (Eds.), *Memory, Brain, and Belief* (pp. 35–86). Harvard University Press.

Madkour, M., Benhaddou, D., & Fung, C. (2016). Temporal data representation, normalization, extraction, and reasoning: A review from clinical NLP perspective. *Journal of Biomedical Informatics*, 61, 267–280.

McGeoch, J. A. (1932). Forgetting and the law of disuse. *Psychological Review*, 39(4), 352–370.

Mitchell, K. J., & Johnson, M. K. (2009). Source monitoring 15 years later: What have we learned from fMRI about the neural mechanisms of source memory? *Psychological Bulletin*, 135(4), 638–677. https://doi.org/10.1037/a0015849

Pustejovsky, J., Castaño, J., Ingria, R., Gaizauskas, R., Knippen, R., Setzer, A., & Wilks, Y. (2003). TimeML: Robust specification of event and temporal expressions in text. *Proceedings of the AAAI Spring Symposium on New Directions in Question Answering*, 28–34.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Roediger, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science*, 17(3), 249–255.

Snodgrass, R. T., & Ahn, I. (1985). A taxonomy of time in databases. *Proceedings of ACM SIGMOD 1985*, 236–246.

Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. *arXiv preprint* arXiv:physics/0004057.

Verhagen, M., Saurí, R., Caselli, T., & Pustejovsky, J. (2010). SemEval-2010 Task 13: TempEval-2. *Proceedings of the 5th International Workshop on Semantic Evaluation*, 57–62.

Tulving, E., & Thomson, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5), 352–373.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Chapter 9: Synthesis and Open Problems

The dissertation opened with a narrow engineering problem — early context tokens lose influence as a conversation grows — and followed its implications across eight chapters to arrive at a theoretical framework for a different kind of problem. This chapter draws that journey to a close by stating the framework's unified claim, reviewing how the four contributions relate to each other as a system, naming the largest unresolved tension honestly, and identifying the experiments that would most directly test whether the theoretical claims hold in practice. The invitation to the research community is the chapter's practical purpose: a theoretical framework that cannot be tested is of limited value, and the framework developed here is sufficiently precise that the key tests can be specified concretely.

The unified thesis, stated once and clearly, is this: a conversation is not a sequence of messages. It is a dynamic epistemic structure — a region of conceptual space with a topology, an information density profile, a temporal validity structure, and a governing frame. Managing a conversation correctly means managing these four dimensions simultaneously. Managing only the content of the messages, however intelligently, is managing the surface of the problem.

The Four Contributions as a System

The four original contributions of this dissertation address four distinct aspects of epistemic structure management. They were presented in sequence because they build upon each other conceptually, but they are not dependent on each other technically. Each can be understood and implemented independently. Together, they address the problem at a level of completeness that no single component achieves.

Chapter 5 established the scoring architecture: a multi-signal composite score that ranks context segments by value rather than by age, evicts the lowest-scoring content first, and provides cognitive architecture extensions for episodic protection, consolidation, and interference detection. This is a complete content management system. Its limitation is that it manages the content of the conversation without asking what that content has collectively established.

Chapter 6 established Dissolution Through Consensus: the operation that replaces accumulated conversational content with a directly synthesised representation of the integrated understanding the content has produced. This is a complete integration mechanism. Its limitation is that it requires external criteria for when to dissolve, what the result should contain, whether the result is sufficient, and whether the dissolution event has introduced new blind spots.

Chapter 7 established the Conversation Intelligence Architecture: a four-level framework that provides exactly those criteria. The surprise allocation signal governs when content has become informationally redundant enough to warrant dissolution. The navigation topology governs what the dissolution output should structurally contain. The generative competence evaluator provides the criterion for whether the result is sufficient. The adversarial frame agent provides the integrity check that the dissolution has not silently promoted explicit premises to invisible background. This is a complete governance layer. Its limitation is that it treats all content as temporally uniform — it does not distinguish between the content whose retention is structurally required and the content that may legitimately be evicted.

Chapter 8 established the bitemporal extension: the recognition that every conversational segment carries two independent temporal attributes (transaction time and valid time), that these diverge systematically for the most important class of conversational content, and that a retention policy which cannot represent this divergence will systematically evict forward commitments, retain corrected errors, and treat standing knowledge as no more important than historical sediment. The bitemporal extension provides the temporal structure that all four CIA levels treat implicitly but none of them makes explicit.

The result is a layered architecture in which each layer contributes something that the others cannot provide:

The scoring architecture provides the retention policy: what to keep, at what fidelity, for how long, and at what positional cost.

Dissolution provides the integration mechanism: when accumulated content has produced an integrated understanding, replacing the content with the understanding is epistemically correct and computationally efficient.

The CIA framework provides the governance layer: the criteria that determine whether the scoring and dissolution decisions are producing a conversation that is topologically complete, informationally dense, competently retained, and frame-honest.

The bitemporal extension provides the temporal structure: the representation that ensures standing knowledge is permanently protected, corrections are propagated correctly, and the topology distinguishes active frontier from historical sediment.

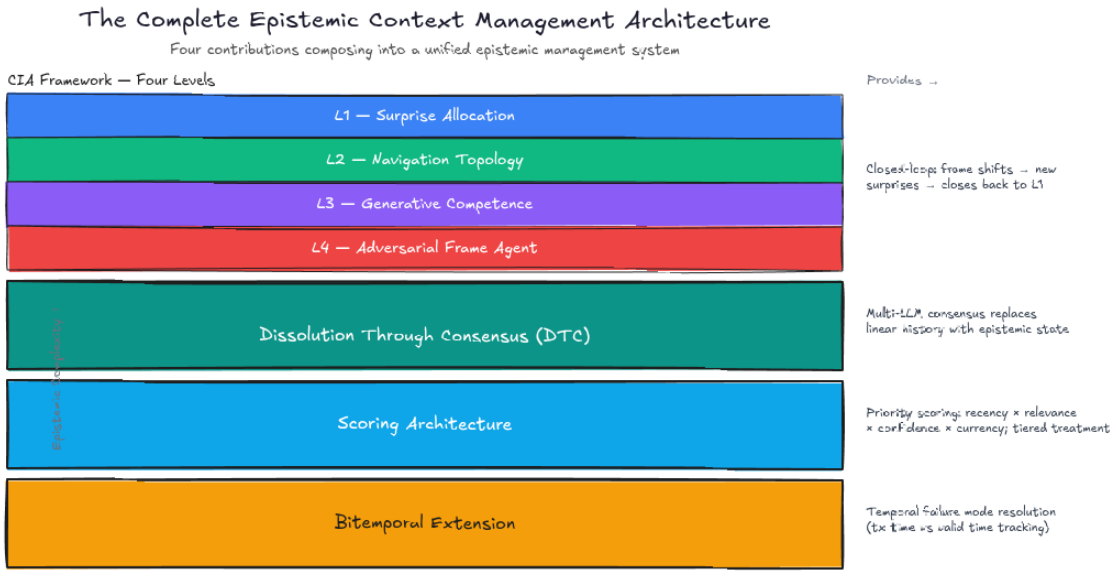


Figure 9.1: The complete epistemic context management architecture. The four contributions form a composable stack. Each layer provides capability that the layers below it cannot; each layer's benefits are preserved when subsequent layers are added (§9.3).

The Additive Architecture

The framework is additive and backward-compatible at each layer. This is not an accident of design; it is a deliberate architectural property, and it has practical consequences for adoption.

An application that integrates only the scoring proxy and the bitemporal extension from Chapters 5 and 8 receives a context management system that is qualitatively better than any naive approach: it evicts by value rather than by age, it protects standing knowledge structurally, and it handles forward commitments and corrections correctly. The dissolution mechanism is not required for this benefit.

An application that adds dissolution receives, in addition, the ability to replace accumulated history with an integrated boundary state when context pressure or semantic drift makes the replacement appropriate. The CIA governance layer is not required for this benefit, though dissolution without generative competence evaluation cannot verify that the boundary state is sufficient.

An application that adds the CIA framework receives the full governance layer: principled dissolution triggers, topological structure for the boundary state, quality evaluation, and frame integrity auditing. The bitemporal extension enhances this layer but is not required for it to function.

The full stack — scoring architecture, dissolution, CIA, and bitemporal tracking — is the complete system. But there is no minimum required configuration below which the framework provides no benefit. Each layer is a genuine improvement over the previous configuration, and each layer's benefits are preserved when subsequent layers are added. This backward-compatibility is important because it means the framework can be adopted incrementally, with each layer delivering measurable value before the next is implemented.

The Honest Assessment

The framework presented in this dissertation is a theoretical contribution. It would be dishonest to present it as anything more than that, and intellectual precision requires stating the largest unresolved tension directly.

Each of the four contributions is individually motivated, individually grounded in prior work (cognitive science, information theory, database systems, discourse theory, or experimental results), and individually novel relative to the existing literature. The claims

for each component individually are strong. The claim that the four components together constitute a system that is superior to any existing approach is a theoretical claim that has not been empirically demonstrated.

The CIA framework's four levels have not been tested as a running system. The surprise allocation signal has not been implemented. The navigation topology index has not been built. The generative competence evaluator does not yet exist as a standardised protocol. The adversarial frame agent has not been prototyped. The compositional logic — the argument that each level addresses the failure mode of the previous one, forming a closed system — is rigorous and the interactions are well-specified, but a rigorous argument for a composed system is not the same as a demonstrated composed system. The difference matters, and claiming otherwise would undermine the intellectual honesty that the framework's theoretical ambition requires.

The dissolution experiments provide directional empirical support for the core DTC claim: the dissolution prompt reliably elicits linguistically distinct outputs from summarisation for operational conversations, pre-consensus model diversity is high, and rolling dissolution shows convergent rather than degrading quality. These results are genuine. They are not sufficient to demonstrate generative completeness — whether a model operating on a boundary state can correctly answer held-out queries from the conversation's domain at the quality of the full history. That demonstration has not been made.

The framework is offered as a foundation for empirical work, not as the completion of it. The open problems in the next section are not hedges. They are the specific experiments that the research community would need to conduct to determine whether the theoretical claims hold.

An Invitation to the Research Community

The following experiments are not equally tractable, but they are all within reach of a well-resourced research group, and each one tests a specific claim that the framework makes. They are ordered by the directness with which they validate the central theoretical contribution.

Experiment 1: Generative competence evaluation of dissolution. Construct a benchmark of synthetic conversations with known informational content across three domain types (operational, strategic, conceptual). For each conversation, apply dissolution to produce a boundary state. Present the boundary state to a model and measure whether it correctly answers a set of held-out queries drawn independently from the conversation's domain — queries that were not used to generate or evaluate the boundary state. Compare performance against the full history baseline. A boundary state that achieves within-ε performance on held-out queries from a domain it has never explicitly been asked about is, by the minimum sufficient representation criterion of Chapter 7, a demonstration that dissolution preserves generative competence. This is the highest-priority experiment because it validates the central theoretical claim of Chapter 6 and provides the evaluation metric that currently does not exist.

Experiment 2: Surprise-retention coupling. Implement the `SurpriseSignal` using token-level log-probabilities from a model that exposes this metadata. Construct conversations in which a high-surprise fact is introduced early and never directly queried again but is implicitly load-bearing for later turns. Compare retention quality of the existing four-signal composite scorer against the five-signal composite that includes surprise. The prediction is that the five-signal system retains the early high-surprise fact at higher priority and produces better performance on queries that depend on it. This experiment directly validates the Free Energy Principle grounding of Level 1 of the CIA framework.

Experiment 3: Longitudinal dissolution stability. Apply rolling dissolution at fixed turn intervals (turns 10, 20, 30, and 40) to conversations in three domain types. Measure consensus quality (using embedding-based rather than lexical clustering) and generative competence at each cycle. The prediction from the strategic conversation's cycle 2 result in the dissolution experiments is that operational content shows improving, not degrading, consensus quality across cycles — because the prior boundary state provides a shared reference that constrains how models express subsequent turns. If this prediction holds across domain types, it inverts the naive assumption that multi-cycle integration must accumulate loss, and demonstrates that dissolution is compositionally stable.

Experiment 4: Frame accumulation and the adversarial agent. Construct conversations in which an implicit framing is accepted without challenge at an early turn and then shapes all subsequent reasoning in a demonstrably suboptimal direction. Measure whether the adversarial frame agent, maintaining an assumption register, successfully flags the implicit premise before it becomes invisible background. Compare a baseline system (without the frame agent) to the augmented system across tasks where the initial framing is known to be wrong. This experiment validates the core claim of Level 4: that frame errors are distinct from content errors, that they accumulate across turns, and that a constitutionally distinct frame-challenger can detect them.

Experiment 5: Bitemporal classifier benchmark. Construct an annotated corpus of forward commitments, backward mutations, and bounded historical facts across clinical, legal, and software engineering conversations. Benchmark the `TemporalClassifier` against this corpus and report precision and recall for each class. This experiment validates the implicit inference architecture of Chapter 8 and, if the benchmark is released publicly, provides the reusable evaluation resource that the field currently lacks for bitemporal linguistic inference.

Open Problems

Beyond the specific experiments above, the framework raises several theoretical problems that require formal treatment before the empirical programme can be fully grounded.

The minimum sufficient representation requires a formal derivation. Chapter 7 defined it as the shortest representation C such that, for all queries drawn independently from the informational domain of conversation H, the probability of a correct answer given C is within ε of the probability given H. This definition is precise enough to use as an evaluation criterion, but the formal proof that such a C exists for any H, and the characterisation of its size as a function of H's informational structure, has not been provided. The Kolmogorov-complexity and Information Bottleneck grounding suggests the proof path, but the derivation remains future work.

The surprise-retention coupling requires formal specification as a modification to the recency decay function. The Titans architecture provides an existence proof that surprise can modulate a forgetting rate in a recurrent memory system, but the translation to a conversation proxy's positional decay function is not fully specified. The correct form of the coupling — whether surprise directly modifies λ, introduces a multiplicative term, or modifies the decay function's shape in a more complex way — requires both theoretical analysis and empirical calibration.

Temporal frame accumulation requires a formal model. Chapter 7 identified the accumulation function — the frame at time t is a function of the frame at t−1 and the new premises accepted at t — but did not formalise this function or define frame lock-in as a detectable condition. A formal treatment of how implicit premises compound across turns, and at what point the accumulated framing becomes irrecoverably constraining for the queries the conversation can even formulate, would provide the theoretical foundation for the adversarial frame agent's detection criterion.

The compound-beta-mini phenomenon remains unexplained. Across all dissolution tasks in the experiments, compound-beta-mini achieved the highest per-model quality score by a consistent and substantial margin. The hypothesis that compound AI architectures are better suited to boundary synthesis than monolithic models has not been tested. If it is confirmed, it has implications for model selection in production dissolution systems that go beyond the current experimental evidence. If it is disconfirmed, or if the margin is explained by confounding factors such as model scale or training data composition, the finding is irrelevant to the broader framework but resolves an open observation that the experiments cannot explain.

Cross-conversation bitemporal queries represent a longer-horizon open problem. The bitemporal extension in Chapter 8 tracks valid time within a single conversation session. A complete bitemporal account would extend this across session boundaries: whether a forward commitment established in one conversation remains binding in a subsequent one, whether corrections propagate across the session boundary, and how the bitemporal topology index handles the transition from one session's frontier to the next session's established territory. This requires both a theoretical extension of the within-session

framework and an implementation that persists bitemporal metadata across sessions — a non-trivial infrastructure requirement that places this problem beyond the scope of the current work.

Closing

The dissertation began with a narrow question: why do LLMs lose track of things they were told at the start of a long conversation? The answer involved transformer attention dynamics, training data statistics, the difference between positional age and epistemic value, and the cognitive architecture of human memory. Each answer revealed a deeper question. The deeper questions led to the recognition that the field had been asking the wrong question — not "how do we preserve more of the right content?" but "what is a conversation, and what does managing it correctly require?"

The answer this dissertation proposes is that a conversation is a dynamic epistemic structure: a region of conceptual space with a topology, an information density profile, a temporal validity structure, and a governing frame. Managing it correctly requires addressing all four of these dimensions simultaneously, with mechanisms that are grounded in cognitive science, information theory, and the well-developed theory of temporal data management. The four contributions assembled in Chapters 5 through 8 constitute a coherent framework for doing this.

The framework is incomplete in the sense that every productive theoretical framework is incomplete: it specifies precisely enough what remains to be done that the work can begin. The experiments above are not a wish list. They are the operational translation of the theoretical claims into falsifiable tests. A research community that takes the framework seriously will find, in those tests, either the confirmations that strengthen it or the disconfirmations that improve it. Either outcome is a contribution. The invitation is open.

Bibliography

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence & J. T. Spence (Eds.), *The Psychology of Learning and Motivation*, Vol. 2 (pp. 89–195). Academic Press.

Behrouz, A., Zheng, P., & Pilanci, M. (2025). Titans: Learning to memorize at test time. *arXiv preprint* arXiv:2501.00663.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407–428.

Delétang, G., Ruoss, A., Duquenne, P.-A., Catt, E., Veness, J., Mattern, C., Mellor, J., Aitchison, L., Orseau, L., Grau-Moya, J., Li, L., Toyer, S., Kossen, J., Langer, A., Eslami, S. M. A., Sutton, R. S., & Hutter, M. (2023). Language modeling is compression. *arXiv preprint* arXiv:2309.10668.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2), 127–138.

Futrell, R., Gibson, E., & Levy, R. (2020). Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cognitive Science*, 44(3), e12814.

Jensen, C. S., & Snodgrass, R. T. (1999). Temporal data management. *IEEE Transactions on Knowledge and Data Engineering*, 11(1), 36–44.

Johnson, M. K., Hashtroudi, S., & Lindsay, D. S. (1993). Source monitoring. *Psychological Bulletin*, 114(1), 3–28.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Hopkins, M., Luck, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 157–173.

Murdock, B. B. (1962). The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5), 482–488.

Pan, J., Gao, T., Chen, H., & Chen, D. (2024). LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint* arXiv:2403.12968.

Snodgrass, R. T., & Ahn, I. (1985). A taxonomy of time in databases. *Proceedings of ACM SIGMOD 1985*, 236–246.

Stalnaker, R. (1978). Assertion. In P. Cole (Ed.), *Syntax and Semantics*, Vol. 9: Pragmatics (pp. 315–332). Academic Press.

Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. *arXiv preprint* arXiv:physics/0004057.

Tulving, E., & Thomson, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5), 352–373.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.