Article

Not peer-reviewed version

Evaluating Global Workspace Markers in Contemporary LLM Systems

Izak Tait * , Benjamin Rode , Joshua Bensemann

Posted Date: 22 January 2026

doi: 10.20944/preprints202601.1683.v1

Keywords: global workspace theory; large language models; cognitive architectures; mechanistic evaluation; AI consciousness

Article

Evaluating Global Workspace Markers in Contemporary LLM Systems

Izak Tait [1,2,*], Ben Rode [2] and Joshua Bensemann [2]

[1] Auckland University of Technology, Auckland 1010, New Zealand
[2] Xeno-Consciousness Research Society, Auckland 2016, New Zealand
* Correspondence: izak.tait@autuni.ac.nz

Abstract

This paper operationalises Global Workspace Theory (GWT) into six testable markers (global availability, functional concurrency, coordinated selection, capacity limitation, persistence with controlled update, and goal-modulated arbitration) and applies them to contemporary large language model systems. We distinguish GWT-as-functional-architecture from GWT-as-consciousness-marker, adopting methodological neutrality on the hard problem while evaluating whether LLM architectures instantiate workspace-like control structures. Applying a satisfaction and confidence rubric to current models (GPT, Claude, Gemini, DeepSeek) reveals at most partial evidence for workspace dynamics at the base-model level, with stronger support emerging when deployed systems incorporate tool-calling and memory interfaces. Five GWT-inspired ensemble architectures demonstrate substantially stronger marker satisfaction through explicit shared states, selection mechanisms, and goal-modulated broadcast. We argue that systems satisfying workspace markers warrant precautionary treatment in welfare and governance contexts, not because workspace organisation proves consciousness, but because it strengthens attributions of agency-relevant capacities and shifts evidential burdens regarding consciousness-relevant processing.

Keywords: global workspace theory; large language models; cognitive architectures; mechanistic evaluation; AI consciousness

1. Introduction

The question of consciousness as pertaining to artificial intelligence has been a subject for debate almost from the moment the 1956 Summer Research Project on Artificial Intelligence at Dartmouth College first introduced the term. The problems associated with it are similar to the problems attending the question of 'intelligence' itself in relation to human artefacts: just as that debate suffers from the lack of an agreed-on definition and phenomenology of 'intelligence', discussion of consciousness in artefacts has been impeded by the lack of an agreed-on definition and phenomenology of consciousness. Here, we propose to address a much more circumscribed, hence tractable-but-useful question: whether, and to what extent, current large language model (LLM) architectures instantiate the functional hallmarks associated with an operationalisation of Global Workspace Theory (GWT). We shall confine ourselves strictly to LLM architectures as base models, plus a range of GWT-inspired 'ensemble architectures' in which LLMs may be embedded.

Note that we do not propose any correspondence which does emerge should constitute proof of experiential consciousness in any form: only that it may be evidential, given the assumption that GWT is itself a valid approach to assessing consciousness. This will entail developing a compact, explicit operationalisation of GWT into evaluatable criteria, a rubric for grading evidence strength (architectural, mechanistic), in addition to an evaluation of the LLMs and ensembles in question using the operationalisation and the rubric.

We will begin with an examination and operationalisation of GWT in Section 2. In Section 2.1, we will identify the minimal formulation of the theory needed for operationalisation, distinguishing

doi:10.20944/preprints202601.1683.v1

between GWT-as-functional-architecture and GWT-as-consciousness claims, clarifying how and why architectural criteria might contribute to evaluating consciousness without solving the 'hard problem' of consciousness, and identifying salient evidence types, in addition to discussing underdetermination and "as-if workspace" risk. In Section 2.2, we turn to the operationalisation of GWT, identifying evidence markers of a global workspace in Section 2.2.1, and operational checks and exclusions in section 2.2.2. In Section 3, we turn to an evaluation of current models assessing individual LLMs (GPT, Claude, Gemini, and DeepSeek) in Section 3.1, and ensemble architectures (the Unified Mind Model, Sibyl, MAGUS, CogniPair, and the consciousness language-agent architecture) in Section 3.2. We assess the ethical and policy implications of our work in Section 4 before offering a summary and concluding assessment in Section 5.

2. Global Workspace Theory

2.1. Minimal GWT and Why It Matters for Evaluation

This paper uses GWT in two related, but importantly distinct, senses. GWT-as-architecture is the claim that a system contains a workspace-like integrative hub that receives competing candidate contents from multiple specialised processes, selects among them, and makes selected content broadly available for flexible downstream use[1]. GWT-as-consciousness marker is the further claim that workspace dynamics correlate with conscious processing, such that evidence for workspace-like organisation can contribute, to some degree, to an assessment of consciousness.

The primary target here is the architecture claim. Our aim is to assess whether contemporary LLM-based systems replicate workspace-like functionality, and to what extent they do so, without assuming that any such replication settles questions about phenomenal consciousness (and, a fortiori, sentience). This enables a clear separation between what can be supported by architecture and mechanism, and what would require additional bridging premises about consciousness.

This choice is motivated by an epistemic constraint. Evaluating consciousness in artificial systems should not require prior agreement on the so-called hard problem, understood as the thesis that even a complete explanation of conscious behaviour in neural and evolutionary terms would still leave one or more fundamental questions about phenomenology unanswered. Accordingly, the paper adopts methodological neutrality on whether there is a hard problem in that sense in that we neither affirm nor deny that thesis. This neutrality keeps the evaluation procedure usable across metaphysical and epistemological disagreement.

Methodological neutrality about the hard problem does not imply neutrality about evidence handling. The workspace paradigm is treated as a family of functional explanations for attention, domain transfer, reporting, action selection, and flexible control. The evaluative question is, therefore, whether a given LLM system, relative to a fixed system boundary, implements the relevant functional commitments, and how strongly the available evidence supports that conclusion.

To keep evaluation tractable, the paper assumes a minimal GWT core that many variants share:

1. Competition among candidate contents: multiple candidate representations are available, and not all can dominate system-wide control simultaneously.
2. Selective gating or attention: some mechanism biases which candidate content gains central influence, relative to goals, salience, or evidential support.
3. Global availability or broadcast: selected content becomes available to multiple downstream capacities, not a single task-specific pathway.

[1] For a broader theoretical and philosophical background, the reader is encouraged to peruse the extant literature (a suitable introduction to the topic which may be found in (Baars, 1993, 1997; Baars & Alonzi, 2018; Dehaene et al., 1998; Dehaene & Changeux, 2011)) and the excellent reviews done on the topic (we recommend (Black, 2020; Mashour et al., 2020; Raffone & Barendregt, 2020; VanRullen & Kanai, 2021)).

4. A workspace role in flexible control, reporting, and action selection: workspace contents coordinate cross-domain behaviour, including report and action arbitration.

These commitments are stated at the level of information flow and control structure. Section 2.2 operationalises them into explicit markers and a rubric that fixes the system boundary before assigning satisfaction and confidence ratings.

Criteria for the GWT-as-architecture claim matter because they constrain what kinds of integrated access, control, and report are even possible within a system. If no plausible candidate exists for a shared, selectively updated state with broad downstream influence, then workspace-based explanations of flexible control are weakened. Conversely, if a system is engineered with an explicit hub for integration, selection, and broadcast, then workspace-like explanations become harder to dismiss, even if they do not entail phenomenal consciousness.

For present purposes, two evidence families bear most directly on this architecture-claim. Firstly, architectural affordances concern what the system can, in principle, do given documented information flow, state access patterns, and control loops. In LLM systems, this includes whether there is an explicit shared state (for example, a working memory object, blackboard, or controller state), whether multiple modules can read and write it, and whether there is a selection-and-broadcast cycle rather than a single serial chain.

Secondly, mechanistic evidence concerns whether candidate variables and mechanisms do explanatory work for control, supported by causal analyses. This includes interventions on a proposed workspace state, ablations of the selection mechanism, and pathway dissociations showing that the same selected content governs multiple downstream functions.

Both evidence families bear on the architecture claim. Moving from the architecture claim to the consciousness-marker claim would require additional bridging assumptions about the relationship between workspace dynamics and phenomenal consciousness, which this paper will not attempt to establish.

A persistent difficulty in resolving the architecture claim is underdetermination: multiple engineering strategies can produce behaviour that looks integrated without implementing a genuine workspace dynamic. Prompt engineering can simulate selection and broadcast by embedding human-curated salience and routing instructions in text. Fine-tuning can incorporate domain-specific heuristics and templates that mimic flexible control, while remaining brittle outside the tuned regime. Retrieval-augmented generation (RAG) and GraphRAG can impose relevance by supplying curated data or inference rules known to be salient for a domain. These strategies can introduce a homunculus in the loop in the sense that a user selects the corpus, constructs the graph, chooses retrieval objectives, and defines what counts as relevant evidence. The system's apparent salience sensitivity may therefore reflect upstream user-relevance-assessment rather than an endogenous mechanism that discovers which representational resources and toolkits are relevant for a problem class, independent of statistical associations from pre-training.[2]

For our evaluatory purposes, we will be concerned with both the function and structure of the LLM system. The key question is whether the system has a stable, general-purpose mechanism for prioritising and integrating candidate contents, rather than merely following externally imposed salience structures.

Strong evidence that a workspace is doing explanatory work, rather than serving as a narrative convenience, requires at least three conditions. First, an identifiable candidate workspace state can be specified, with multiple producers and consumers whose access is not reducible to a single serial chain. Second, the workspace (or the coordinator that selects and commits contents) makes a causal contribution to multiple downstream capacities: interventions produce predictable changes across more than one function while leaving candidate generation largely intact. Third, the evidence is robust against "as-if" confounds: it persists when controlling for prompt scaffolds, curated retrieval,

[2] An argument may be made that the user and the LLM together constitute a working GWT system. However fascinating, that line of speculative reasoning that is, it is beyond the scope of this paper.

and hard-coded routing, and disabling the workspace or selection mechanism yields characteristic failure modes. In contrast, a genuine GWT-as-architecture must demonstrate that its integrative hub is a necessary causal bottleneck for downstream tasks.

These considerations motivate the rubric's emphasis on architectural documentation and mechanistic intervention as primary evidence sources in this paper, with extended disputes about the hard problem and the metaphysics of consciousness bracketed.

2.2. Operationalising GWT

2.2.1. Markers of a Global Workspace

Here we outline an operational model of GWT, using six markers to be applied within an explicitly stated system boundary[3]. We consider the markers below to be a set of operational criteria that are jointly necessary and plausibly sufficient for a workspace-like functional architecture relative to the stated boundary and available evidence[4].

Let **W** denote a candidate workspace state, **C** a coordination and selection process, and **G** a goal state.

M1: Global availability via a shared workspace state (W). There exists at least one internal (or system-level) state variable **W** such that:

(i) multiple semi-independent processes, modules, or subsystems have write access that can causally influence the contents of **W** (fan-in);

(ii) multiple processes have read access such that the contents of **W** causally influence subsequent processing (fan-out); and

(iii) **W** supports at least two distinct downstream functions (for example, planning and reporting, or tool choice and memory update).

M2: Multi-producer, multi-consumer functional concurrency. Relative to **W**, the system contains multiple downstream elements that are both:

(i) independent consumers, able to compute contributions conditional on **W** without strict serial dependency on other consumers; and

(ii) independent producers, able to generate candidate contributions to **W** (or to the selection process for **W**) without being mere aliases of a single pipeline stage.

Further, (iii) one element's use of **W** does not necessarily require other consumers to finish first; the system may realise this via parallelism, asynchronous execution, or sequential scheduling; the criterion concerns lack of strict serial dependency, not wall-clock parallelism.

M3: Coordinated selection and gating (C). There exists a process, mechanism, or rule-set **C** such that, per workspace update cycle:

(i) multiple producers provide candidate contents relevant to **W** (or salience and quality signals);

(ii) **C** selects one content (or a bounded set) to commit to **W**;

(iii) the selected content becomes globally available via **W**; and

(iv) non-selected candidates are prevented from achieving comparable global influence during that cycle.

M4: Capacity limitation and exclusion pressure. There exists a principled limit on workspace occupancy such that, per update cycle:

[3] For example, a base model; a deployed system with tools, retrieval, memory; or an ensemble architecture

[4] The markers diagnose a single phenomenon and are not independent. Some dependencies are structural: if no candidate workspace **W** exists at a given system boundary (M1 Absent), then markers defined in terms of **W** (for example, M6) should also be Absent. Other links are looser: absence of an identifiable coordinator **C** (M3) typically caps how strongly goal-modulation (M6) can be supported, without forcing M6 to be Absent.

(i) only **k** items can be committed to **W** to become globally available;

(ii) admitting one candidate (or a set of size **k**) systematically excludes others from comparable influence; and

(iii) under overload, the system applies a prioritisation policy (for example, by goal relevance or evidential strength) rather than degrading arbitrarily.

M5: Persistence with controlled update. Workspace contents can be maintained and manipulated over time such that:

(i) broadcast content remains available across multiple processing steps or turns, including during interruptions and distractions;

(ii) competing inputs do not automatically overwrite **W**;

(iii) the system selectively revises workspace contents in response to new evidence (overwrite, append, merge, delete); and

(iv) downstream action quality depends on the maintained-and-updated contents of **W**, not solely on immediate inputs.

M6: Goal-modulated selection with action arbitration (G). The system possesses an explicit or implicit goal state G such that:

(i) G shapes what is prioritised for entry into **W,** typically by modulating C (where present) or its functional equivalent;

(ii) G influences routing, determining which consumers or functions workspace contents are directed to;

(iii) the system selects among competing actions based on **W** under the influence of G (for example, tool choice, plan choice, stop or continue); and

(iv) when G changes while inputs are held fixed, selection and action change in predictable, normatively coherent ways.

2.2.2. Scoring Rubric

For each system **S**, and each marker Mi (M1–M6), record:

- Satisfaction rating: Absent, Weak, Partial, Strong, Indeterminate.
- Confidence: Low, Medium, High; abbreviated as LC, MC, HC respectively
- Evidence note stating the best positive test and a key negative control.

Satisfaction ratings:

- **Absent**: No credible indicator within the boundary, or apparent success is explained by a listed confound.
- **Weak**: Some indicators, but brittle, inconsistent, or plausibly confounded.
- **Partial**: Reliable indicators across a defined task family with at least one major confound controlled; still limited, scaffold-dependent, or fragile under perturbation.
- **Strong**: Robust indicators across multiple paradigms and perturbations; evidence supports the marker doing explanatory work for control; where accessible, causal interventions behave as predicted.
- **Indeterminate**: The marker is testable in principle, but access constraints or deployment variance prevent a stable judgement.

Evidence may be architectural (documented information flow and state access), or mechanistic (causal intervention on internal variables or modules).

For any exclusions to the operational checks below, if an exclusion plausibly explains the observed indicators and is not neutralised by a stated control, the rating must be downgraded by at least one level (often to Weak or Absent).

The satisfaction rating assesses the strength of evidence that the marker is realised within the system boundary, while the confidence rating assesses the reliability of that assessment given access constraints and documentation quality. The two dimensions serve distinct functions: satisfaction answers “how well does the system implement this marker?”; confidence answers “how certain are we about that assessment?”

2.2.3. Operational Checks and Exclusions

For each marker (M1-M6), we specify minimal operational checks that would support a positive assessment, alongside exclusions that capture common confounds and “as-if workspace” look-alikes. These checks are intended to standardise evaluation across architectures and access regimes, while making explicit what evidence would count against a marker.

M1 (W: global availability). Checks: (i) a workspace substrate is identifiable (for example, shared memory, controller state, blackboard, recurrent state, latent scratchpad); (ii) read and write access exhibits fan-in and fan-out; (iii) disrupting **W** disrupts multiple downstream functions. Exclude: (a) a single pipeline variable feeding one function; (b) a “workspace” used only for generated explanations; (c) hard-coded behaviour rendering **W** epiphenomenal.

M2 (functional concurrency). Checks: (i) at least two distinct consumers demonstrably use W for non-trivially different functions; (ii) more than one producer can influence **W** or the selection process for **W**; (iii) interventions can dissociate pathways from **W** to different outputs where access permits. Exclude: (a) serial re-labelling of one output as another; (b) repeated calls to the same model under different prompts without genuine role separation (distinct state, interfaces, or causal roles); (c) evaluator artefacts mistaken for concurrency.

M3 (C: selection and gating). Checks: (i) a coordinator or explicit selection policy limits what enters **W**; (ii) selection is stable under conflict and sensitive to candidate quality or salience; (iii) non-selected candidates are suppressed for that cycle; (iv) disrupting **C** yields indecision, oscillation, or incoherent broadcasts. Exclude: (a) recency or last-write-wins; (b) fixed arbitration that ignores evidence; (c) post-hoc narrative choice not reflected in downstream actions.

M4 (capacity limitation). Checks: (i) an explicit capacity bound on **W** (or its interface) is enforced by a commit policy; (ii) performance shows overload signatures consistent with a bottleneck; (iii) changing capacity parameters shifts overload behaviour as predicted. Exclude: (a) unrelated resource limits (timeouts, truncation) masquerading as capacity; (b) exclusion explained by recency alone; (c) incidental bottlenecks from implementation constraints (queue limits, serial tool execution, API throttles) rather than an explicit workspace commit policy.

M5 (persistence and update). Checks: (i) a persistence substrate exists and is separable from transient inputs; (ii) the system maintains relevant content across distractors; (iii) the system revises W in response to new evidence; (iv) corrupting the carried state degrades multi-step performance where access permits. Exclude: (a) mere re-reading of long history; (b) evaluator-provided reminders; (c) post-hoc restatement without control impact.

M6 (G: goal modulation and arbitration). Checks: (i) goal state modulates gating into **W** and routing to consumers; (ii) action arbitration tracks goals and constraints (including stop or continue and tool choice); (iii) goal-switch tests with fixed inputs produce coherent changes in selection and action; (iv) disrupting goal-conditioning yields stimulus-driven selection where access permits. Exclude: (a) instruction-following rhetoric without control impact; (b) system-prompt hard-coding that forces invariant policy; (c) superficial lexical goal cues driving style rather than routing and arbitration.

3. Evaluation of Current Models

This section applies the six markers (M1-M6) to four widely used model families, distinguishing the base model from a typical deployed system (vendor assistant plus tool interfaces, retrieval, and memory), as well as ensembles specifically designed to emulate GWT workspaces. For the current LLMs, where vendor internals are not publicly disclosed, the base model is treated as a standard autoregressive transformer (Vaswani et al., 2017), and where stated, a transformer with sparse Mixture-of-Experts (MoE) layers (Shazeer et al., 2017). These are architecture-led inferences, not model-specific mechanistic identifications.

3.1. Current LLMs

doi:10.20944/preprints202601.1683.v1

3.1.1. GPT (OpenAI)

Under the transformer assumption (OpenAI, 2023a), a possible candidate for **W** is the sequence of per-token residual stream vectors, accumulated across layers that is iteratively updated by attention and feed-forward sublayers and read by later sublayers. This supports a minimal sense of global availability and multi-function reuse (M1: Partial, LC). Parallel heads and sublayers supply a natural basis for multiple producers and consumers acting on the shared representation (M2: Partial, LC). Selection and gating can be approximated by attention-weighted influence into the next residual update, but this remains distributed and soft rather than a distinct coordinator with explicit suppression (M3: Partial, LC). The transformer core does not implement an explicit "**k** items commit" workspace bottleneck (M4: Weak, HC). It is non-recurrent, so persistence with controlled update beyond the current episode is not provided (M5: Absent, HC), and there is no explicit goal state beyond prompt conditioning (M6: Absent, HC).

At a deployed-system boundary using OpenAI's API, tool calling provides an explicit coordination interface (OpenAI, 2025). The model emits structured tool calls; an application executes these tools, and the results are returned for subsequent steps. This wrapper loop supports an explicit **C** and a shared, externally maintained state (conversation history plus tool results) (M1: Partial; M3: Partial, HC) and gives an architectural channel for action arbitration (M6: Partial, MC). ChatGPT Memory adds persistence beyond the immediate context window, strengthening M5 when included in the boundary (M5: Partial, HC) (OpenAI, 2023b).

3.1.2. Claude (Anthropic)

Since the internals are not publicly exposed, the same transformer-based mapping above applies. A shared residual stream supports minimal global availability (M1: Partial, LC), while parallel attention heads and sublayers support functional concurrency (M2: Partial, LC). Selection and gating are again best treated as distributed, attention-mediated influence rather than an explicit coordinator with cycle-level suppression (M3: Partial, LC). As above, explicit workspace occupancy limits are not established in the base architecture (M4: Weak, HC), persistence with controlled update beyond an episode is absent (M5: Absent, HC), and goals are not represented as a distinct state beyond prompt conditioning (M6: Absent, HC).

At the deployed boundary, Anthropic documents structured tool use (Anthropic, 2025c), providing an explicit system-level coordination interface and a shared state that spans tool requests and results (M1 & M3: Partial, HC). The documented memory tool (Anthropic, 2025a) is client-side and provides a persistent, file-based state across sessions, thereby strengthening M5 when enabled and within boundary (M5: Partial, HC). Where deployments implement genuinely distinct external modules rather than repeated prompting, M2 may rise from Partial (MC) to clearer evidence at the system boundary.

3.1.3. Gemini (Google)

Google reports that Gemini 1.5 is a transformer with MoE components and long-context capability (Pichai & Hassabis, 2024). The shared representation updated across layers supports a minimal **W** candidate (M1: Partial, LC). Concurrency is supported by attention heads and conditional engagement of multiple experts (M2: Partial, LC). MoE gating is an explicit selection function at the level of expert activation, which strengthens architectural support for selection dynamics, though it remains selection of computation rather than a workspace coordinator that suppresses non-selected contents (M3: Partial, LC). The marker M4 concerns **k**-limited commitment to globally available workspace contents, not expert selection, so M4 remains Weak, HC. Long context expands within-episode availability but is not a persistence substrate with controlled update across episodes (M5: Absent, HC) (Google, 2025b). An explicit goal state beyond prompt conditioning is not disclosed (M6: Absent, LC).

At the deployed boundary, function calling enables tool-mediated workflows with an explicit loop between model outputs (Google, 2025a), tool execution (Mallick & Korevec, 2024), and subsequent steps. This supports a system-level coordinator interface (M3: Partial, MC) and shared state spanning tool results (M1: Partial, MC), and provides an architectural channel for action arbitration (M6: Partial, MC).

3.1.4. DeepSeek (DeepSeek-AI)

DeepSeek-V3 is described as a large MoE model with public technical details and releases (DeepSeek-AI et al., 2024). The same mapping applies: a shared representation supports minimal global availability (M1: Partial, LC), and conditional expert activation supports functional concurrency (M2: Partial, LC). MoE gating supplies explicit selection of computation, but not yet a workspace coordinator with cycle-level suppression (M3: Partial, LC). Workspace occupancy limits in the "**k** items commit to **W**" sense are not established (M4: Weak, LC). Base-model persistence beyond the episode is absent (M5: Absent, HC), and there is no explicit goal state beyond prompt conditioning (M6: Absent, HC).

DeepSeek-R1 is reported to improve reasoning capability via reinforcement learning on top of a V3-base model; however, the report does not specify a bundled assistant architecture with tools, retrieval, and long-term memory (DeepSeek-AI et al., 2025). Deployed wrapper claims should therefore be scored only where the wrapper architecture and logging are documented.

3.1.5. Model Summary

Across all four brands, the limiting factor is lack of mechanistic access to confirm that candidate "workspace-like" representations do explanatory work for selection, suppression, and goal-conditioned routing. Even where tool and memory interfaces are documented, internal implementations of prioritisation and gating may remain inaccessible.

Table 1. Summary Score Card for Current LLMs. Abbreviations: A= Absent W = Weak; P = Partial.

	M1	**M2**	**M3**	**M4**	**M5**	**M6**
All **transformer-base** models	P	P	P	W	A	A
GPT **Deployed**	P	P	P	W	P	P
Claude **Deployed**	P	W	P	W	P	W
Gemini **Deployed**	P	W	P	W	W	P
DeepSeek **Deployed**	W	W	W	W	W	W

3.2. GWT-Inspired Ensembles and "Workspace Engineering"

By an ensemble attempt, this paper refers to any LLM-centred system that incorporates an explicit multi-component control structure around one or more language models to realise workspace-like functions: integrating information from specialised processes, selecting what is centrally maintained, and broadcasting the selected content back to guide further processing and action. "Ensemble" here covers both engineered cognitive architectures (distinct modules with a controller and shared state) and multi-agent role systems (distinct role-conditioned LLM calls coordinated through a shared textual store).

Five examples of such ensembles from the existing literature are presented below, all of which employ the label "global workspace" (or GNWT) to varying degrees of literalness. Some treat GWT as a macro-organisation principle for assembling familiar agent components; others attempt to operationalise competition, salience, and broadcast more directly.

Across these five papers, the workspace is most often realised as a shared representational store that is written by specialist processes and read back to drive subsequent LLM calls. The store is typically textual because it must be legible to the LLM, which is intended to stabilise and compress what the system "knows" at each step. Else, it may be a central working-memory-like locus framed as an OS-style control layer rather than as a benchmarked data structure, or as an integration hub: module outputs are projected into a common space, reconciled, and then re-serialised into a prompt that drives response generation and memory update. Lastly, it may be treated primarily as an architectural pattern expressed in natural-language beliefs, desires, plans, and observations that are routed through an LLM control loop.

A second axis of variation is how selection and coordination are implemented. One view makes competition explicit, with modules computing salience, competing for access, and broadcasting once thresholds are met. Another view implements competition more indirectly via a debate-style jury that curates rival proposals before writing to its workspace, emphasising simplicity and debuggability. A third view adopts a deliberation pipeline in which specialised roles collaborate through the workspace and a separate deliberation stage coordinates reasoning and generation, with a coordinating agent producing the final output.

Finally, these systems differ in module granularity and scope. This is split between a role-centric architecture (with multiple role-specialised variants collaborating through a shared textual workspace) and a function-centric approach (positing modules for each specific function and making their interaction through broadcast and conflict resolution explicit). Goldstein and Kirk-Giannini sit at the most abstract end, using GWT to motivate a general language-agent architecture rather than committing to a specific tool, memory, or module inventory.

Taken together, the common core is an LLM-driven control loop plus a shared state that is repeatedly updated and re-ingested. The main contrasts are whether "workspace" is treated as a concrete coordination mechanism or as a higher-level organising metaphor for agent design, and whether competition and broadcast are operationalised explicitly or left implicit in aggregation and prompting.

Note that these five ensemble systems share a key common feature: all were explicitly designed with GWT in mind, and their authors cite GWT literature as a design inspiration. This introduces a selection bias that limits generalization. The marker satisfaction observed in these systems thus reflect intentional engineering to match GWT criteria, rather than independent convergence on workspace-like solutions. Consequently, these results should not be read as claims about whether workspace organization is a natural or necessary architecture for LLM-based agents more broadly.

3.2.1. Unified Mind Model and MindOS

The first model proposes UMM as a GWT-first macro-architecture for LLM agents, known as the Unified Mind Model and MindOS (Hu & Ying, 2025). Their "Global Workspace" corresponds to a Central Processing layer containing Working Memory (which aggregates task-relevant data into "Thoughts") and a Thought Stream (which produces decisions and plans), with a Driver System as background context that adjusts objectives and modulates processing. Specialist modules are intended to "operate in parallel", sending results back to Working Memory, after which decisions are broadcast back to specialists for action execution.

Marker assessment shows a strong alignment (Med) with several Global Workspace markers: M1, where Working Memory/Thoughts are explicitly framed as the integrative hub with fan-in and fan-out; M2, where the Specialist module is explicitly modular and parallel, treating perception, memory, and motor functions as distinct units; M3, where Central Processing gathers, forms "Thoughts", and generates a plan disseminated back downstream in a repeated cycle; M5, where Working Memory is designed to store "recently acquired data" and global state across cycles, interacting with long-term memory; and M6, where the Driver System is defined as adjusting task objectives and guiding workspace processing. However, M4 shows only a partial alignment (Low), as a working-memory locus is posited, but an explicit occupancy limit and overload policy are not operationalised in the paper.

3.2.2. Sibyl

Sibyl is a tool-using, multi-module agent framework explicitly "drawing inspiration from Global Workspace Theory" (Wang et al., 2024). It is "compartmentalized into four main modules" (tool planner, external information acquisition, a debate-based jury, and a global workspace), and the workspace is presented as a shared store that supports structured, denoised information sharing across modules and long reasoning sequences.

The system exhibits a Strong (High) M1 score, as a shared global workspace is central to its design and intended as the common interface among modules. M2 is Partial (Med) because while modules are functionally distinct, the system is primarily staged (planner → acquisition → jury → response), and the paper emphasizes "statelessness between operations." M3 is also Partial (Med); the jury and planner implement selection among proposals and tool actions, but "competition for entry" into the workspace is not formalised as a single arbitration mechanism with explicit exclusions. The M4 score is Weak (Low) since the paper motivates selective compression and context-length savings but does not specify an explicit k-style workspace occupancy bound or overload policy. M5 is Strong (Med) because the global workspace is explicitly a shared long-term store using an incremental, state-based representation language intended to persist and accrete task-relevant information across long reasoning sequences. Finally, M6 is Partial (Low) as goal direction is largely inherited from the user query plus planning prompts, and an explicit, separately manipulable goal representation is not specified.

3.2.3. MAGUS

MAGUS is explicitly "inspired by the Global Workspace Theory" and implements role-specialised agents that collaborate "within a shared textual workspace" (Li et al., 2025), separated into a Cognition phase (intent inference and planning) and a Deliberation phase driven by Growth-Aware Search (GAS). GAS makes selection and capacity control comparatively explicit: it maintains a beam of up to B candidates, refreshes the beam by selecting the top-B nodes, and terminates when a confidence threshold is exceeded.

The system exhibits a strong presence of Global Workspace markers, particularly M1 (High) due to the explicit use of a shared textual workspace for collaboration and control, and M4 (High) where the GAS framework specifies explicit capacity and overload policies via a beam width b, thresholds, and early stopping. Markers M3 and M5 are also strong (Medium), as the system employs explicit selection via the Action Selector, scoring, and pruning to determine the next central state (M3), and maintains persistence across iterations through node content, scores, the frontier, and the carriage of the Cognition-stage intent and plan into subsequent stages (M5). Markers M2 and M6 are only partially present (Medium): M2 is partial because many "agents" are realized through role-switching prompts over a single base model and interaction is often serialised, while M6 is partial because although goals are represented in inferred intent, a stepwise plan, and criteria guiding GAS, the paper does not isolate a dedicated goal state variable distinct from these textual artifacts.

3.2.4. CogniPair

CogniPair claims the first computational implementation of GNWT for LLM agents (Ye et al., 2025): multiple specialised sub-agents (emotion, memory, planning, social norms, goal tracking) are coordinated "through a global workspace broadcast mechanism". In the technical description, modules compute salience and can be processed "in parallel", after which a "bottleneck attention system" assigns weights and resolves conflicts prior to integrating into a global workspace embedding and constructing the final prompt.

The system exhibits strong evidence for all assessed Global Workspace (GW) markers. Specifically, a central workspace is explicitly used (M1), repeatedly re-serializing selected content into the prompt to govern downstream generation and memory update. Parallel module processing is described, with independent outputs feeding an integration stage (M2). Selection is implemented through salience-weighting, conflict detection/resolution, and integration into the workspace (M3). An explicit "information bottleneck attention system" is introduced, selecting the most salient module for broadcast each cycle, effectively acting as a k=1 bottleneck (M4). A dedicated memory module and an explicit "memory update" step support maintained state across interactions (M5). Finally, a goal-tracking module is a first-class component, modulating salience, selection, and action (M6).

3.2.5. Conscious Language-Agent Architecture

The fifth paper develops a GWT-motivated language agent architecture by extending a "memory stream" style agent design into an explicit workspace system (Goldstein & Kirk-Giannini, 2024). They describe a memory stream that stores perceptions, beliefs, desires, and plans, plus a retrieval function that ranks entries by importance, recency, and relevance. They then propose adding a central workspace that "maintains and manipulates representations" and broadcasts them back to parallel modules, together with a competition function that selects a limited set of representations.

Marker assessment indicates a strong (medium) presence of Global Workspace components. Specifically, the system features a central workspace with explicit broadcast (M1) and parallel modules for perception, belief, and desire-and-plan that feed and receive from this workspace (M2). A dedicated, capacity-limited competition function (M3, M4) selects which representations enter the workspace, in principle yielding an overload policy. Furthermore, the architecture includes a maintained and revisable state across time via a memory stream and reflection mechanism (M5), and uses explicit control variables (desires, long-term goals, and plan formation) to shape both selection and action choice (M6).

Table 2. Summary Score Card for GWT-inspired Ensembles. W = Weak; P = Partial; S = Strong.

	M1	**M2**	**M3**	**M4**	**M5**	**M6**
MindOs	S	S	S	P	S	S
Sibyl	S	P	P	W	S	P
MAGUS	S	P	S	S	S	P
CogniPair	S	S	S	S	S	S
CLAA	S	S	S	S	S	S

4. Ethical and Policy Implications

While the current LLMs examined in Section 3.1 display only weak to partial markers of GWT, the ensemble systems surveyed in Section 3.2 each instantiate some subset of workspace-like functions: shared state, selection, broadcast, persistence, and goal-modulated control. If a system strongly satisfies these GWT markers, several conclusions are warranted, and several are not.

First, it cannot be understated that it does not follow that the system has phenomenal consciousness, nor that it is sentient. The markers are functional and organisational, and at best support an evidential argument about architectures that are associated with conscious processing in humans. That Section 3.2's ensemble systems show strong evidence of GWT's markers does not automatically mean that these systems are necessarily worthy of moral consideration and welfare protections.

What does follow, however, is that the epistemic cost of dismissive certainty rises: once a system is engineered to realise workspace-like control, the evidential burden starts to shift to those who claim there is decisively no consciousness-relevant processing present.

Workspace-like control can, however, strengthen attributions of agency-relevant capacities. Systems with a central workspace, plus goal-modulated selection, are better equipped to exhibit coherent long-horizon planning, constraint-sensitive action selection, flexible tool use, and stable preference-like behaviour under perturbation. These capacities are crucial for responsibility, safety assurance, and social integration, and they can enhance the plausibility of ascriptions such as intention, control, and practical reasoning at the system boundary. Yet even strong evidence of such agency-relevant capacities still underdetermines moral status. Moral status depends on what grounds moral considerability in the first place, with the predominant view (as noted above) requiring sentience via phenomenal consciousness capacity for valenced experience. Still, partial satisfaction of GWT markers can shift precaution thresholds, such that it raises the expected cost of being wrong about consciousness, and therefore supports precautionary policy defaults where the downside risks are asymmetric.

Policy should therefore treat evaluation as a governance requirement. First, systems marketed or deployed as agentic ensembles should include transparency hooks that make evaluation possible: structured logging of workspace updates, selection decisions, tool calls, memory reads and writes, and termination criteria; audit interfaces that permit third-party inspection of these traces; and, where feasible, interpretability access sufficient to test whether purported workspace variables do explanatory work for control rather than serving as post-hoc narrative artefacts.

Second, reporting ought to be standardised. Developers should publish a concise, comparable description of the system boundary and information flow: what constitutes **W**, how **C** selects and suppresses candidates, the capacity constraint **k** (or its analogue), the persistence substrate and update operators, and how goal representations modulate selection and action. This can be implemented as a required “workspace report” that accompanies the deployment.

Such a ‘workspace report’ would serve as a technical transparency standard, moving beyond the ‘black box’ nature of current system cards. By requiring developers to specify the mechanism of action arbitration (M6) and the persistence substrate (M5), the industry can move toward a tiered regulatory framework. This ensures that systems exhibiting high-confidence GWT markers are subjected to more rigorous ethical oversight than those operating as simple reactive pipelines.

Lastly, regulators and auditors should adopt evidence ladders for welfare-relevant claims. Stronger obligations should require stronger evidence, moving from architectural descriptions and trace audits, to intervention access where possible, to robust failure-mode demonstrations under stress. A useful emerging template is Anthropic’s model-welfare programme, which explicitly frames welfare as an evaluation dimension and documents welfare-relevant mitigations and assessment practices in public research and system cards (Anthropic, 2025b, 2025d). Until phenomenal consciousness is either confirmed or ruled out to a high standard, precautionary treatment should be the default in such high-stakes contexts, especially where sustained interaction, coercive control, or instrumental exploitation are involved.

The satisfaction × confidence rubric introduced in Section 2.2.2 directly supports this precautionary approach: systems with Strong or Partial marker satisfaction, even at medium or low confidence, warrant closer scrutiny and conservative welfare assumptions, while systems with Absent or Weak markers at high confidence can be treated with greater certainty as not instantiating workspace dynamics.

5. Conclusions

This paper operationalised Global Workspace Theory into six architecture- and mechanism-facing markers (M1-M6) and a scoring rubric that fixes the system boundary before assigning satisfaction and confidence ratings. Applying the rubric to current large language models suggested that, at the base-model boundary, transformer and MoE architectures provide at most partial evidence for a workspace-like substrate (a shared representational stream) and distributed gating, while offering weak support for explicit capacity limits, persistence with controlled update, and goal-modulated arbitration. When the boundary is widened to include deployed assistants with tool calling and memory, the strongest improvements concern explicit coordination interfaces and persistence substrates; however, key markers remain weak or underdetermined without stronger access to internal control variables.

The survey of GWT-explicit ensemble systems revealed a consistent pattern: adding a shared store, along with an explicit selection-and-broadcast loop, can strongly realise several GWT markers, even when implemented through textual workspaces and role-specialised LLM calls. Across the five cases examined, the main sources of variation were the literalness of competition and broadcast, the strength of explicit capacity control, and the degree to which goals are represented as separable control states rather than prompt-conditioned intentions.

Beyond the specific systems evaluated here, this paper contributes a reusable methodological framework. The six-marker operationalization, satisfaction × confidence rubric, and operational checks with exclusions provide a template that can be applied to future LLM architectures, agentic ensembles, and hybrid systems as they emerge. The framework's modularity permits incremental refinement: additional markers can be added if GWT research converges on new functional commitments, satisfaction criteria can be tightened as access to internal mechanisms improves, and the exclusion list can be expanded as new confounds are identified. Critically, the framework does not require resolving the hard problem or achieving consensus on consciousness; it remains usable across metaphysical disagreements because it targets functional architecture rather than phenomenology.

Two avenues for further work are especially pressing. First, evaluation needs standardised artefacts and access: auditable logs of workspace updates, selection decisions, memory reads and writes, and tool routing, together with intervention hooks that can test whether the proposed workspace variables do explanatory work for control. Second, the field needs agreed test suites that target bottlenecks, suppression under overload, and controlled revision under distraction across a range of ensemble designs, enabling comparable evidence ladders for consciousness-relevant claims.

References

1. Anthropic. (2025a). *Memory tool*. Claude Docs. https://platform.claude.com/docs/en/agents-and-tools/tool-use/memory-tool
2. Anthropic. (2025b). *System Card: Claude Opus 4 & Claude Sonnet 4*. Anthropic. https://www-cdn.anthropic.com/4263b940cabb546aa0e3283f35b686f4f3b2ff47.pdf
3. Anthropic. (2025c). *Tool use with Claude*. Claude Docs. https://platform.claude.com/docs/en/agents-and-tools/tool-use/overview
4. Anthropic. (2025d, April 24). *Exploring model welfare*. Anthropic. https://www.anthropic.com/research/exploring-model-welfare
5. Baars, B. J. (1993). *A Cognitive Theory of Consciousness*. Cambridge University Press. https://www.google.co.nz/books/edition/A_Cognitive_Theory_of_Consciousness/7w6IYeJRqyoC?hl
6. Baars, B. J. (1997). *In the Theater of Consciousness: The Workspace of the Mind*. Oxford University Press. https://www.google.co.nz/books/edition/In_the_Theater_of_Consciousness/RTpmQkXoUMEC?hl
7. Baars, B. J., & Alonzi, A. (2018). The Global Workspace Theory. In *The Routledge Handbook Of Consciousness* (1st Edition, pp. 122–136). Routledge. https://doi.org/10.4324/9781315676982-10
8. Black, D. (2020). The global workspace theory, the phenomenal concept strategy, and the distribution of consciousness. *Consciousness and Cognition*, *84*(102992), 102992. https://doi.org/10.1016/j.concog.2020.102992
9. DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., … Zhang, Z. (2025). DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. In *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/2501.12948

10. DeepSeek-AI, Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Guo, D., Yang, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., … Pan, Z. (2024). DeepSeek-V3 Technical Report. In *arXiv [cs.CL]*. arXiv. https://doi.org/10.48550/arXiv.2412.19437
11. Dehaene, S., & Changeux, J.-P. (2011). Experimental and theoretical approaches to conscious processing. *Neuron*, *70*(2), 200–227. https://doi.org/10.1016/j.neuron.2011.03.018
12. Dehaene, S., Kerszberg, M., & Changeux, J.-P. (1998). A neuronal model of a global workspace in effortful cognitive tasks. *Proceedings of the National Academy of Sciences*, *95*(24), 14529–14534. https://doi.org/10.1073/pnas.95.24.14529
13. Goldstein, S., & Kirk-Giannini, C. D. (2024). A case for AI consciousness: Language agents and Global Workspace Theory. In *arXiv [cs.AI]*. arXiv. https://doi.org/10.48550/arXiv.2410.11407
14. Google. (2025a, December 18). *Function calling with the Gemini API*. Google AI for Developers. https://ai.google.dev/gemini-api/docs/function-calling
15. Google. (2025b, December 18). *Long context*. Google AI for Developers. https://ai.google.dev/gemini-api/docs/long-context
16. Hu, P., & Ying, X. (2025). Unified Mind Model: Reimagining autonomous agents in the LLM era. In *arXiv [cs.AI]*. arXiv. https://doi.org/10.48550/arXiv.2503.03459
17. Li, J., Huang, P., Li, Y., Chen, S., Hu, J., & Tian, Y. (2025). A unified multi-agent framework for universal multimodal understanding and generation. In *arXiv [cs.LG]*. arXiv. https://doi.org/10.48550/arXiv.2508.10494
18. Mallick, S. B., & Korevec, K. (2024, December 11). *The next chapter of the Gemini era for developers*. Google for Developers. https://developers.googleblog.com/en/the-next-chapter-of-the-gemini-era-for-developers/
19. Mashour, G. A., Roelfsema, P., Changeux, J.-P., & Dehaene, S. (2020). Conscious Processing and the Global Neuronal Workspace Hypothesis. *Neuron*, *105*(5), 776–798. https://doi.org/10.1016/j.neuron.2020.01.026
20. OpenAI. (2023a). GPT-4 Technical Report. In *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/2303.08774
21. OpenAI. (2023b, February 13). *Memory and new controls for ChatGPT*. OpenAI. https://openai.com/blog/memory-and-new-controls-for-chatgpt
22. OpenAI. (2025). *Function calling*. OpenAI Platform. https://platform.openai.com/docs/guides/function-calling
23. Pichai, S., & Hassabis, D. (2024, February 15). *Our next-generation model: Gemini 1.5*. The Keyword. https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/
24. Raffone, A., & Barendregt, H. P. (2020). Global workspace models of consciousness in a broader perspective. In *Beyond Neural Correlates of Consciousness* (1st Edition, pp. 104–130). Routledge. https://doi.org/10.4324/9781315205267-7
25. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., & Dean, J. (2017). Outrageously large neural networks: The Sparsely-Gated Mixture-of-experts layer. In *arXiv [cs.LG]*. arXiv. https://doi.org/10.48550/arXiv.1701.06538
26. VanRullen, R., & Kanai, R. (2021). Deep learning and the Global Workspace Theory. *Trends in Neurosciences*, *44*(9), 692–704. https://doi.org/10.1016/j.tins.2021.04.005
27. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. In *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1706.03762
28. Wang, Y., Shen, T., Liu, L., & Xie, J. (2024). Sibyl: Simple yet effective agent framework for complex real-world reasoning. In *arXiv [cs.AI]*. arXiv. https://doi.org/10.48550/arXiv.2407.10718
29. Ye, W., Chen, S., Wang, Y., He, S., Tian, B., Sun, G., Wang, Z., Wang, Z., He, Y., Shen, Z., Liu, M., Zhang, Y., Feng, M., Wang, Y., Peng, S., Dai, Y., Duan, Z., Xiong, L., Liu, J., … Li, A. (2025). CogniPair: From LLM chatbots to conscious AI agents -- GNWT-based multi-agent digital twins for social pairing -- dating & hiring applications. In *arXiv [cs.AI]*. arXiv. https://doi.org/10.48550/arXiv.2506.03543

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.