

Orchestrating Specialized Agents for Trustworthy Enterprise RAG

Xincheng You
xyou@atlassian.com
Atlassian
Cambridge, USA

Huayi Li
hli8@atlassian.com
Atlassian
Mountain View, USA

Qi Sun
qsun@atlassian.com
Atlassian
Mountain View, USA

Shubham Goel
sgoel8@atlassian.com
Atlassian
San Francisco, USA

Neha Bora
nbora@atlassian.com
Atlassian
Washington DC, USA

Kang Li
kli3@atlassian.com
Atlassian
Bellevue, USA

Sean Culatana
sculatana@atlassian.com
Atlassian
Mountain View, USA

Abstract

Retrieval-Augmented Generation (RAG) shows promise for enterprise knowledge work, yet it often underperforms in high-stakes decision settings that require deep synthesis, strict traceability, and recovery from underspecified prompts. One-pass retrieval-and-write pipelines frequently yield shallow summaries, inconsistent grounding, and weak mechanisms for completeness verification. We introduce ADORE (Adaptive Deep Orchestration for Research in Enterprise), an agentic framework that replaces linear retrieval with iterative, user-steered investigation coordinated by a central orchestrator and a set of specialized agents. ADORE’s key insight is that a structured Memory Bank (a curated evidence store with explicit claim–evidence linkage and section-level admissible evidence) enables traceable report generation and systematic checks for evidence completeness.

Our contributions are threefold: (1) *Memory-locked synthesis*—report generation is constrained to a structured Memory Bank (Claim–Evidence Graph) with section-level admissible evidence, enabling traceable claims and grounded citations; (2) *Evidence-coverage-guided execution*—a retrieval–reflection loop audits section-level evidence coverage to trigger targeted follow-up retrieval and terminates via an evidence-driven stopping criterion; (3) *Section-packed long-context grounding*—section-level packing, pruning, and citation-preserving compression make long-form synthesis feasible under context limits.

Across our evaluation suite, ADORE ranks first on DeepResearch Bench (52.65) and achieves the highest head-to-head preference win rate on DeepConsult (77.2%) against commercial systems.

CCS Concepts

- Information systems → Users and interactive retrieval; Question answering;
- Computing methodologies → Natural language processing.

Keywords

retrieval-augmented generation, agentic information retrieval, evidence grounding, claim traceability, evidence coverage, long-form report generation, enterprise search

1 Introduction

Large Language Models (LLMs) are increasingly embedded in enterprise knowledge workflows, reshaping how employees discover, summarize, and operationalize institutional information. Retrieval-Augmented Generation (RAG) [9] is the dominant grounding pattern: it ties generation to a retrieved evidence set, aiming to reduce hallucination and improve factuality on proprietary corpora [6, 8]. In practice, however, enterprise queries are rarely clean “one-shot” QA. They are underspecified, multi-faceted, and often require iterative decomposition, cross-document synthesis, and auditable traceability.

Despite strong performance on short-form QA, many “retrieve-then-generate” pipelines fall short for executive decision support. First, they can confidently answer the wrong question when the prompt is ambiguous, producing plausible but misaligned narratives [6, 8]. Second, linear pipelines are brittle: when early retrieval misses a critical subtopic, downstream generation often compounds the omission. Third, long contexts are not a panacea; even when relevant evidence is present, models can ignore or underuse it (the “lost in the middle” effect) [11]. These issues become acute for long-form reporting, where completeness, citation stability, and section coverage matter as much as fluency.

Limitations of static RAG in enterprise. Our analysis of real-world enterprise deployment reveals three recurring failure modes:

- **Prompt ambiguity & cold start.** Users seldom phrase precise, decontextualized queries. Requests like “Analyze our Q3 risks” are informationally underspecified. Without clarification behaviors [1], systems retrieve against vague lexical cues, reducing precision and recall.
- **Workflow rigidity.** Treating retrieval as a single pre-processing step limits recovery from early mistakes. Static “chains” often lack the agency to adjust strategy when intermediate evidence indicates a mismatch or missing dimension [14].
- **Contextual myopia.** Long-context ingestion does not guarantee utilization; important details can be ignored even when retrieved [11]. Moreover, one-pass synthesis lacks self-correction loops that can systematically target missing or weakly supported sections [2, 12].

Positioning vs. prior deep-research agents. ADORE is inspired by recent deep-research systems, but it differs in what drives iteration and what constitutes convergence. WebWeaver emphasizes outline-centric evidence structuring and section-level writing with a memory bank [10]. TTD-DR frames long-form generation as test-time diffusion over drafts, improving quality via iterative denoising steps [7]. In contrast, ADORE introduces *Memory-locked synthesis*: we treat the Memory Bank as a hard constraint on generation (section-scoped admissible evidence with explicit claim–evidence linkage), then use section-level evidence coverage audits to diagnose gaps and drive targeted follow-up retrieval with an evidence-driven stopping rule. This mechanism is operationalized via Evidence-coverage-guided execution, with an explicit evidence-coverage stopping rule rather than a fixed iteration budget or outline-only convergence.

ADORE overview. ADORE (Adaptive Deep Orchestration for Research in Enterprise) replaces linear “retrieve-then-write” with an agentic workflow that mirrors human research: scoping via clarification, plan formation, iterative evidence collection, and verification. The core design choice is a structured Memory Bank that stores evidence with explicit claim–evidence linkage, enabling traceable report generation and systematic checks for coverage and citation stability.

Contributions. We make the following contributions (using consistent terminology throughout the paper):

- (1) **Memory-locked synthesis.** We generate long-form reports *constrained to a structured Memory Bank* (Claim–Evidence Graph with section-level admissible evidence), ensuring that claims are traceable and citations are grounded in retrieved sources.
- (2) **Evidence-coverage-guided execution.** We use section-level evidence coverage audits to localize weak or missing sections, trigger targeted follow-up retrieval, and terminate via an evidence-driven stopping criterion [2, 13].
- (3) **Section-packed long-context grounding.** We make long-form grounded reporting tractable by packing section-specific evidence, pruning redundant context, and compressing retrieved content into citation-preserving summaries so synthesis stays within context limits [3, 11].
- (4) We implement an adaptive orchestrator that routes between fast retrieval and deep multi-agent investigation, and we demonstrate state-of-the-art results on both public and internal benchmarks [4, 5].

2 The ADORE Framework

ADORE abandons a single linear pipeline in favor of a hub-and-spoke architecture governed by a central **Orchestrator Agent** (Figure 1). The orchestrator continuously interprets user intent, conversation context, and intermediate evidence signals to select an execution path: a low-latency retrieval response for simple informational queries, or a multi-agent deep-research workflow for complex analytical tasks.

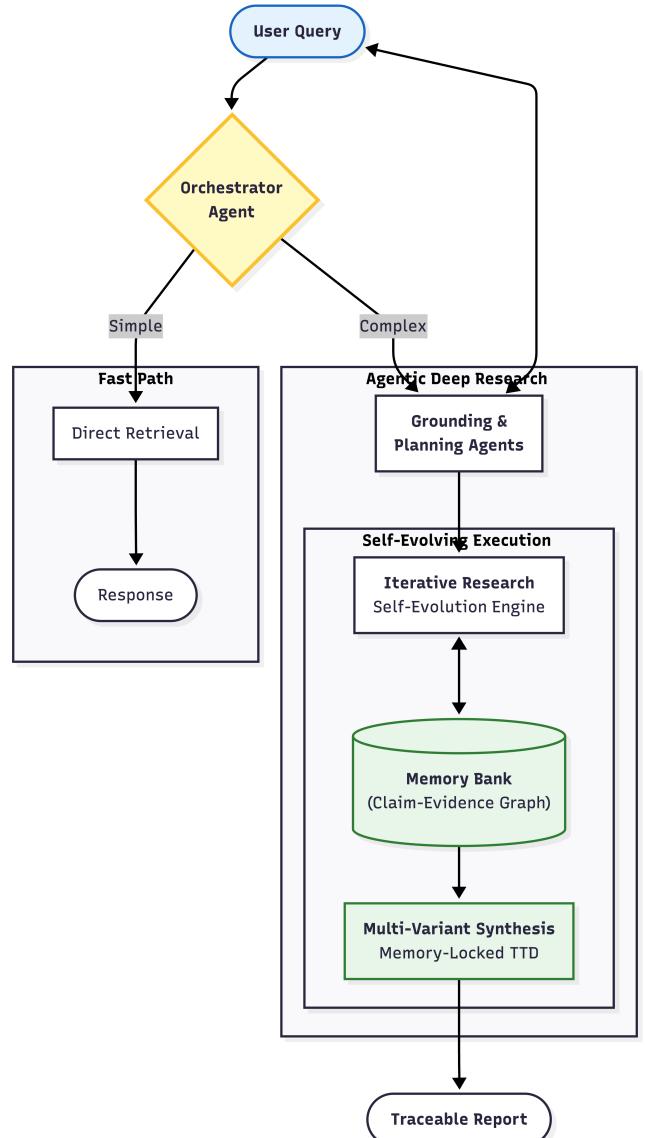


Figure 1: The ADORE framework. The Orchestrator routes complex tasks to an agentic workflow that replaces linear retrieval with Memory-locked synthesis. Unlike standard RAG, ADORE constrains generation to a structured Memory Bank (Claim–Evidence Graph), enabling iterative, evidence-coverage-guided execution for traceable enterprise reporting.

2.1 Adaptive Orchestration and Routing

The Orchestrator classifies incoming requests by *research complexity* (e.g., ambiguity, expected depth, number of sub-questions, and required traceability). Simple factoid-style requests follow a standard retrieval path. Complex requests trigger a multi-phase workflow:

- **Clarification (Grounding):** resolve ambiguity and establish success criteria before heavy retrieval starts.
- **Planning:** produce a structured research plan that the user can accept or edit.
- **Execution:** run iterative retrieval, synthesis, and verification until evidence coverage criteria are satisfied.

A key principle is that *iteration is evidence-driven*. Rather than iterating a fixed number of times, ADORE uses Evidence-coverage-guided execution signals (coverage and stability) to decide whether to retrieve more, refine the outline, or finalize.

2.2 Specialized Agents

ADORE delegates responsibilities to specialized agents:

- **Grounding Agent:** mitigates cold start by formulating clarification questions (and proposing initial scope assumptions) to convert underspecified prompts into a concrete brief [1].
- **Planning Agent:** converts the refined brief into a structured plan (sections, priorities, success criteria). The plan is presented as a shared artifact so users can edit or prioritize sections, keeping humans in the loop.
- **Execution Agent:** manages retrieval and synthesis. It uses a **Self-Evolution Engine** to evolve search queries based on intermediate findings and detected evidence gaps, enabling multi-hop recovery [13].
- **Report Generation Agent:** produces (i) an initial outline, (ii) outline revisions based on retrieved evidence, and (iii) final reports with traceable citations.
- **WebSearch Agent:** retrieves and filters high-authority public sources when external context is required, discarding low-signal content before ingestion.

2.3 Execution Workflow

The **Execution Agent** orchestrates the research loop as follows:

- (1) **Outline draft:** The Report Generation Agent proposes a report outline aligned with the user-confirmed plan.
- (2) **Iterative retrieval:** The Self-Evolution Engine proposes search queries over internal and external sources. A reflection step audits retrieved evidence against the outline to identify missing sections and weak support; queries are refined and re-executed.
- (3) **Outline update:** The Report Generation Agent revises the outline using newly retrieved evidence.
- (4) **Evidence-driven stopping:** The Execution Agent checks whether evidence coverage and citation stability satisfy the plan. If not, it returns to retrieval; otherwise, it triggers final report generation.

2.4 Synthesis and Traceability

ADORE draws inspiration from diffusion-style iterative drafting for long-form reports [7] and outline-grounded writing with structured evidence memories [10]. Our departure is to make the Memory Bank a *hard constraint* on generation and to use explicit section-level evidence coverage signals to drive targeted retrieval and determine convergence.

2.4.1 Memory Bank (Claim–Evidence Graph). The Memory Bank is a persistent evidence store with explicit claim–evidence linkage. For each planned section, it maintains the admissible evidence set (sources and excerpts) and tracks coverage requirements (what must be supported). This structure ensures that report writing is grounded and that any claim can be traced to supporting documents.

2.4.2 Memory-locked synthesis. Given a fixed Memory Bank, ADORE generates a report under a strict admissible-evidence constraint: each section is written using only the section-scoped evidence stored in the claim–evidence graph. This enables traceability by construction (each claim must map to supporting citations in the Memory Bank) and supports citation auditing by verifying that cited sources and quoted spans appear in the admissible evidence store.

2.4.3 Evidence-coverage-guided execution Criterion. ADORE drives iteration via section-level evidence coverage audits. After each retrieval–reflection cycle, the system evaluates whether each planned section is sufficiently supported by admissible evidence in the Memory Bank. When coverage indicates insufficient support, the system triggers targeted follow-up retrieval and focused rewriting for the affected sections. This yields an *evidence-driven stopping criterion*: the loop terminates when coverage requirements satisfy the plan, rather than after a fixed number of refinement steps [2, 13].

2.4.4 Section-packed long-context grounding Strategy. Long-form grounded reporting is context-intensive. Section-packed long-context grounding makes it tractable by: (i) packing only section-relevant evidence into each generation call, (ii) pruning redundant or low-salience retrieved text, and (iii) compressing long sources into citation-preserving summaries so synthesis remains grounded without exceeding context budgets [3, 11]. This preserves traceability while keeping long-form generation practical at enterprise scale.

3 Evaluation Methodology

We employ a dual evaluation strategy combining reference-based metrics and reference-free preference judgments.

3.1 Reference-Based: The RACE Framework

We utilize the **RACE** framework (Readability, Instruction Following, Comprehensiveness, Insight) to benchmark report quality [5]. RACE generates a prompt-specific weight vector W_T tailored to task requirements.

For a given research task T , RACE generates a weight vector W_T tailored to the specific requirements of the prompt. For example, a task analyzing “Provide a comprehensive market analysis of Strategic Portfolio Management (SPM) technology solutions” assigns higher weight to Comprehensiveness:

$$W_{Insight} = 0.3, \quad W_{Comp} = 0.35, \quad W_{Instruct} = 0.2, \quad W_{Read} = 0.15 \quad (1)$$

Within the **comprehensiveness** dimension, RACE generated detailed sub-criteria with differentiated weights. For instance, for the SPM market-analysis task the generated rubrics and their assigned weights were:

Table 1: Comprehensiveness sub-criteria and weights for the SPM market-analysis task.

Sub-criterion	Weight
Definition and Conceptual Framework of SPM	10%
Market Landscape Analysis	15%
Technical Features and Capabilities Coverage	15%
Vendor Assessment Comprehensiveness	15%
Selection Decision Factors Analysis	10%
Evaluation Framework Robustness	15%
Implementation and Best Practices Coverage	15%

3.2 Reference-free: Side-by-Side Evaluation

In addition to reference-based benchmarks, we run side-by-side evaluations against competitor deep research products. For each research task, we:

- (1) Generate a report using ADORE.
- (2) Obtain a competing report (e.g., ChatGPT Deep Research).
- (3) Compare the two reports using an LLM-judge win/tie/lose framework.

This setup estimates how often users or judges would prefer ADORE when viewing reports side by side.

3.3 Evaluation Datasets

We evaluate ADORE on three datasets:

- (1) **DeepResearch Bench (Public):** 100 PhD-level research tasks authored by domain experts (reference-based) [5].
- (2) **Internal Enterprise Benchmark:** proprietary tasks derived from high-quality internal documents via reverse prompt engineering (reference-based).
- (3) **Deep Consult Dataset (Public):** business/consulting research tasks (reference-free) [4].

4 Experimental Results

4.1 RACE Performance

On the public **DeepResearch Bench**, ADORE achieved a score of **52.65**, with the following component scores:

- Comprehensiveness: **52.22**
- Insight: **54.37**
- Instruction Following: **51.11**
- Readability: **52.18**

This places it at rank #1 on the leaderboard [5]. Table 2 compares our performance against top industry baselines.

On the **Internal Enterprise Benchmark**, the system achieved a RACE score of **64.11**, with the following component scores:

- Comprehensiveness: **65.91**
- Insight: **66.76**
- Instruction Following: **60.35**
- Readability: **61.95**

Given that the benchmark is calibrated such that a score of 50.0 represents the human-written reference, ADORE demonstrates a **14.11%** improvement in quality over human baselines for these specific enterprise tasks.

4.2 Side-by-Side Preference

We conducted blind side-by-side evaluation on **DeepConsult** [4]. We compared ADORE against a leading competitor (ChatGPT Deep Research) using an LLM-as-a-Judge paradigm.

As shown in Table 3, ADORE achieves a win rate of **77.21%** with a lose rate of **4.41%**.

5 Conclusion

We presented ADORE, an agentic framework for deep enterprise research that moves beyond rigid RAG pipelines toward adaptive, evidence-driven investigation. ADORE’s central novelty is operational: Memory-locked synthesis constrains report generation to a structured Memory Bank for traceability; Evidence-coverage-guided execution turns evidence coverage into targeted retrieval decisions with a stopping rule; and Section-packed long-context grounding keeps long-form grounded synthesis practical under long-context constraints. Across public and internal benchmarks, ADORE achieves state-of-the-art results while maintaining traceable, section-level evidence grounding.

Future work will focus on:

- **Actionability:** extending planning to propose decision options and trade-offs, not only synthesis.
- **Factual guardrails:** developing granular citation-accuracy metrics and automatic audits for technical domains.
- **Multimodal synthesis:** generating visual artifacts (charts, plots) aligned with evidence-backed report sections.

Acknowledgments

We thank the GenAI platform, AI fundamental, assistant evaluation teams for their support on the project.

References

- [1] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 2019. Asking Clarifying Questions in Open-Domain Information-Seeking Conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 475–484.
- [2] Akari Asai, Zeqin Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *International Conference on Learning Representations (ICLR)*.
- [3] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv preprint arXiv:2004.05150* (2020).
- [4] Deep Consult. 2025. Deep Consult. <https://github.com/Su-Sea/ydc-deep-research-evals>
- [5] Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. 2025. DeepResearch Bench: A Comprehensive Benchmark for Deep Research Agents. *arXiv preprint* (2025).
- [6] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997* (2023).
- [7] Rujun Han, Yanfei Chen, Zoey CuiZhu, Lesly Miculicich, Guan Sun, Yuanjun Bi, Weiming Wen, Hui Wan, Chunfeng Wen, Solène Maître, George Lee, Vishy Tirumalachetty, Emily Xue, Zizhao Zhang, Salem Haykal, Burak Göktürk, Tomas Pfister, and Chen-Yu Lee. 2025. Deep Researcher with Test-Time Diffusion. *arXiv preprint arXiv:2507.16075* (2025). doi:10.48550/arXiv.2507.16075
- [8] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of Hallucination in Natural Language Generation. *Comput. Surveys* 55, 12 (2023), 1–38.
- [9] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 33. 9459–9474.
- [10] Zijian Li, Xin Guan, Bo Zhang, Shen Huang, Houquan Zhou, Shaopeng Lai, Ming Yan, Yong Jiang, Pengjin Xie, Fei Huang, Jun Zhang, and Jingren Zhou. 2025. WebWeaver: Structuring Web-Scale Evidence with Dynamic Outlines for

Table 2: RACE scores on DeepResearch Bench: ADORE compared with leading deep-research systems.

Rank	Model	Overall	Comp.	Insight	Inst.	Read.
1	ADORE (Ours)	52.65	52.22	54.37	51.11	52.18
2	Tavily Research	52.44	52.84	53.59	51.92	49.21
3	ThinkDepthAI	52.43	52.02	53.88	52.04	50.12
4	CellCog	51.94	52.17	51.90	51.37	51.94
5	Salesforce Air	50.65	50.00	51.09	50.77	50.32
6	LangChain (GPT-5)	50.60	50.06	50.76	51.31	49.72
7	Gemini 2.5 Pro	49.71	49.51	49.45	50.12	50.00
8	LangChain (Tavily)	49.33	49.80	47.34	51.05	48.99
9	OpenAI Deep Research	46.45	46.46	43.73	49.39	47.22
10	Claude Research	45.00	45.34	42.79	47.58	44.66

Table 3: DeepConsult side-by-side evaluation: win/tie/lose rates and aggregate score for ADORE vs. agentic baselines.

Rank	Agentic System	Win (%)	Tie (%)	Lose (%)	Score
1	ADORE (Ours)	77.21	18.38	4.41	7.03
2	Enterprise Deep Research	71.57	19.12	9.31	6.82
3	WebWeaver (Claude-sonnet)	66.86	10.47	22.67	6.96
4	WebWeaver (gpt-oss-120b)	65.31	11.22	23.47	6.64
5	Gemini-2.5-pro	61.27	31.13	7.60	6.70
6	WebWeaver (qwen3-235b)	54.74	28.61	16.67	6.47
7	openai-deepresearch	0.00	100.00	0.00	5.00
8	Perplexity Deep Research	32.00	-	-	-
9	doubao-research	29.95	40.35	29.70	5.42
10	Claude-research	25.00	38.89	36.11	4.60
11	WebWeaver (qwen3-30b)	28.65	34.90	36.46	4.57
12	WebShaper (32B)	3.25	3.75	93.00	1.63

- Open-Ended Deep Research. *arXiv preprint arXiv:2509.13312* (2025). doi:10.48550/arXiv.2509.13312
- [11] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics* 12 (2024), 157–173.
- [12] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-Refine: Iterative Refinement with Self-Feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 36.
- [13] Harsh Trivedi, Tushar Khot, Ashish Sabharwal, and Peter Clark. 2023. IRCoT: Iterative Retrieval and Chain-of-Thought Generation for Long-Form Question Answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*.
- [14] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations (ICLR)*.

