ENTERPRISESAGA: A BILLION-TOKEN BENCHMARK FOR EVIDENCE RETRIEVAL AND REASONING OVER EVOLVING ENTERPRISE KNOWLEDGE

Anonymous authors
Paper under double-blind review

ABSTRACT

Enterprise question answering requires connecting evidence across systems while distinguishing activities that share participants, terminology, and business context. We introduce EnterpriseSaga, a billion-token benchmark for evidence retrieval and reasoning built around a synthetic B2B SaaS company. It contains 966 questions spanning 18 months and 12 enterprise systems, with 1.213 billion tokens in the exported serialization. Our Coarse-to-Fine Enterprise Simulation Framework develops projects and recurring operations within a shared company history and generates artifacts grounded in their activities. To represent the repetition and overlap of real company operations in finer detail, question-conditioned augmentation enriches local business contexts with artifacts of comparable activities. Across seven evaluated systems, reranking and iterative search improve answer quality, although full-corpus performance remains substantially below the gold-evidence reference. System rankings reverse between multi-hop and synthesis questions, revealing different strengths in relational search and information coverage. Scale experiments show lower performance as the company background expands, with questions and annotated supporting files held fixed. Additional analyses examine evidence use, computational costs, reader–judge sensitivity, and question and annotation quality through human assessment.

1 INTRODUCTION

The information needed to answer business questions is produced through a company's day-to-day work. As teams coordinate across enterprise systems, messages, tickets, and documents capture different parts of an activity and its changing outcomes. Consider the illustrative question in Figure 1: "What were the billing update, cause, and resolution of Northstar's duplicate charges?" Answering it requires connecting a support ticket identifying the affected update, an engineering discussion explaining the cause, and a later email confirming the resolution. An earlier complaint concerns the same customer but a different incident, while a preliminary proposal does not establish the final outcome. Enterprise question answering therefore requires evidence retrieval and reasoning to locate relevant artifacts, determine which facts apply, and connect evidence across activities, systems, and time.

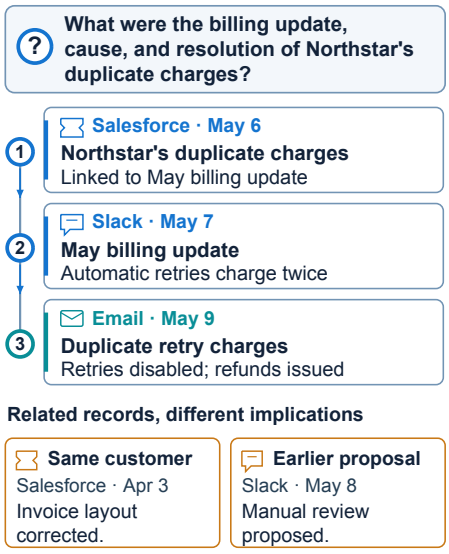


Figure 1: An example of connecting evidence across systems and time amid similar activities and proposals.

Recent benchmarks address different aspects of this challenge. NoLiMa studies long-context retrieval beyond literal matching, while BEAM examines memory over coherent conversational histories (Modarressi et al., 2025; Tavakoli et al., 2025). Enterprise-related benchmarks address questions over workplace artifacts: HERB generates supporting evidence through query-guided workflows, while EnterpriseRAG-Bench combines project-grounded document generation with topic-based corpus

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

expansion (Choubey et al., 2025; Sun et al., 2026). Workplace benchmarks such as TheAgentCompany model a company environment, but evaluate task execution rather than question answering over its accumulated artifacts (Xu et al., 2026). These directions motivate a setting that connects enterprise question answering with the activities through which company information develops. We pursue this connection by constructing a shared, evolving company history in which projects, recurring operations, and organizational changes produce both supporting and background artifacts. Related activities share participants and terminology, requiring systems to use business relationships, timing, and scope to determine which evidence applies.

Building such a benchmark requires addressing four challenges. First, a shared company context needs to connect activities across teams while tracking changes in participants, responsibilities, and plans. Second, a long company history requires detailed local activities that remain connected to broader developments. Project work progresses through linked stages, while recurring operations produce related cases with different circumstances and outcomes. Third, activities need to be expressed through evidence distributed across systems. Different artifacts capture partial views at different times, requiring control over shared facts and the distinction between proposed actions and completed outcomes. Fourth, questions need verifiable answers within richly developed business contexts. Real company operations involve repeated requests, parallel assignments, and successive updates whose extensive overlap is difficult to reproduce exhaustively, motivating finer elaboration around the situations selected for evaluation.

To address these challenges, we develop the Coarse-to-Fine Enterprise Simulation Framework and use it to build EnterpriseSaga, a billion-token benchmark for evidence retrieval and reasoning over evolving enterprise knowledge. Specifically, Company Context Modeling establishes shared business and organizational conditions over time. Hierarchical Project Planning progressively develops strategic initiatives into concrete activities, while Scenario-Based Operations Generation produces recurring situations with varied circumstances and outcomes. Artifact Plans and shared facts guide Activity-Grounded Artifact Generation across 12 enterprise sources. Together, these stages capture 18 months of work at Clearpath, a fictional company with approximately 1,000 initial employees. We use the resulting history to design multi-hop questions that trace business relationships and synthesis questions that combine complementary information. Evidence grounding traces their planned facts to generated artifacts and verifies their support. Question-conditioned augmentation then develops the surrounding business contexts in finer detail, adding artifacts of comparable activities with different participants, periods, or versions. These additions enrich the local context within the shared company setting while preserving the reference answers.

Experiments across seven retrieval pipelines and agents show gains from reranking and iterative search, with substantial room for improvement relative to the gold-evidence reference. System rankings reverse between multi-hop and synthesis questions, while scale experiments reveal challenges as the company background expands. Further analyses examine corpus composition, evidence requirements, retrieval coverage, failure cases, and computational costs. Reader–judge comparisons and human assessment additionally examine evaluation sensitivity and annotation quality. Our contributions are threefold:

(1) We introduce EnterpriseSaga, a billion-token benchmark for enterprise question answering, with artifacts spanning 18 months of work at Clearpath. Its 966 questions evaluate evidence retrieval and reasoning within a shared company history, using reference answers, answer subfacts, and annotated sufficient evidence sets to assess answer quality and evidence selection.

(2) We develop the Coarse-to-Fine Enterprise Simulation Framework, combining company-wide activity planning and artifact generation with question-guided local refinement. The framework connects evolving business activities to questions with traceable supporting evidence and enriches their surrounding contexts with artifacts of comparable activities.

(3) We establish an empirical evaluation of enterprise question answering across retrieval pipelines and iterative agents, assessing answer quality, evidence recovery, and computational costs. Controlled corpus comparisons, question-level analyses, and execution traces characterize where systems succeed and where evidence acquisition and interpretation break down.

2 Related Work

Enterprise QA and evidence construction. WixQA and OfficeQA Pro evaluate QA over collected content, while HERB and EnterpriseRAG-Bench use synthetic enterprise artifacts (Cohen et al.,

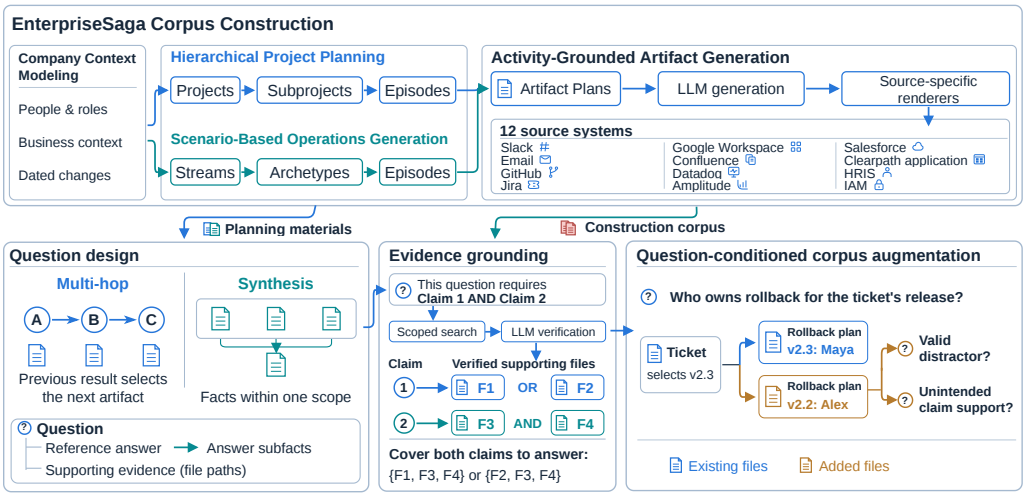


Figure 2: **EnterpriseSaga construction pipeline.** Company context supports two activity-generation paths, followed by artifact generation across 12 source systems. QA construction comprises question design, evidence grounding, and question-conditioned corpus augmentation. Competing claims are reviewed before artifact generation; the resulting artifacts are checked for intended content and unintended cross-question support. Retained, verified support updates the evidence annotations. Planning hierarchies are schematic.

2025; Opsahl-Ong et al., 2026; Choubey et al., 2025; Sun et al., 2026). EnterpriseRAG examines instruction adherence under imperfect retrieval (Miao et al., 2026). WinSyn and OrgForge connect generated evidence to project histories and organizational events (Varhade et al., 2026; Flynt, 2026). EnterpriseSaga combines company-wide project and operations planning with question-guided local refinement, supporting multi-hop and synthesis questions within a shared, evolving business context.

Workplace agents, long contexts, and memory. WorkArena, CRMArena, and TheAgentCompany evaluate workplace task execution (Drouin et al., 2024; Huang et al., 2025a; Xu et al., 2026). RULER, HELMET, and ETHIC examine information use in long contexts (Hsieh et al., 2024; Yen et al., 2025; Lee et al., 2025), while LongMemEval and MEMTRACK study information retained or updated across interactions (Wu et al., 2024; Deshpande et al., 2025). EnterpriseSaga evaluates evidence retrieval and reasoning over accumulated company activities, using a fixed corpus that captures their development over time. Appendix E gives detailed comparisons across these settings.

3 EnterpriseSaga Corpus Construction

Our Coarse-to-Fine Enterprise Simulation Framework plans company activities before generating their artifacts, preserving shared business context and links for subsequent QA construction. Figure 2 summarizes the framework, from company modeling and activity planning to artifact generation, QA construction, and corpus refinement. Throughout corpus construction, batch generation uses GPT-5.6-luna, and model-based review uses GPT-5.6-sol with medium reasoning effort (OpenAI, 2026).

3.1 Company Context Modeling

A shared company setting connects work across departments through common business objectives, participants, and resources. Human guidance specifies Clearpath as a fictional B2B SaaS company with approximately 1,000 initial employees and an 18-month horizon, allowing sustained cross-team interactions and organizational change. Through iterative human–LLM collaboration, we establish its business model, product portfolio, and initial operating state. Strategic initiatives define development objectives and milestones; employee and external-entity registries establish the teams, customers, partners, and relationships available to subsequent activities.

To represent changes in this setting, LLM-generated department-level workforce plans are expanded programmatically into dated personnel events. Employment and role histories constrain when employees can participate and which responsibilities they hold. The resulting registry contains

666 events affecting 642 employees, covering hiring, departures, promotions, transfers, reporting changes, and leave. Human review focuses on the plausibility of high-level business and organizational specifications, personnel distribution, and planned workforce changes. Appendix A.1 provides the company profile, organizational distributions, personnel-event statistics, and review procedures.

3.2 Hierarchical Project Planning

Developing detailed activities across an 18-month company history requires coordinating local execution with broader business plans. We use hierarchical planning to progressively specify objectives, boundaries, and dependencies before elaborating execution details. Starting from the strategic initiatives in Section 3.1, we consolidate shared capability requirements into programs, organize sustained execution areas as workstreams, and define projects with bounded objectives and time windows.

The company story then describes developments that change these plans, linking company-level events to their consequences for project scope, schedules, and staffing. These developments can also initiate new work: five event-triggered projects expand the initial portfolio from 79 to 84 projects. Project stories elaborate execution within this shared trajectory, while subproject stories specify delivery boundaries, participants, and handoffs. These stories are expanded into episodes describing how individual matters progress, then into occurrences specifying dated actions, participants, and before-and-after states. Each expansion follows the scope, timing, and established outcomes of its source narrative, with participant assignments checked against personnel histories. Retained links connect company developments to local activities and provide a path for subsequent QA construction to trace planned facts to their intended evidence. Appendix A.2 reports the planning inventory, cross-project relationships, construction constraints, and a worked example spanning a shared event, local execution, and a downstream handoff.

3.3 Scenario-Based Operations Generation

Alongside project work, companies repeatedly handle activities such as customer support, invoice reconciliation, and employee onboarding. Their recurrence calls for reusable scenarios that produce distinct cases with different circumstances and outcomes. Operational programs and streams organize ongoing responsibilities; story families describe recurring mechanisms and decision branches; scenario archetypes specify concrete situations, role requirements, and possible outcomes.

We allocate episodes across archetypes according to recurrence patterns and scenario categories. Episodes from the same archetype are generated jointly, varying their business objects, evidence, complications, and decisions. The resulting 9,000 archetypes produce 50,003 operational episodes. A worked invoice-challenge example illustrates how similar requests lead to different outcomes: reconciliation of reporting boundaries, correction of a billing discrepancy, confirmation of billed usage, or an unresolved dispute (Appendix A.3). Project-linked episodes additionally inherit delivery and handoff facts, connecting project execution to subsequent operational responses. All episodes enter the shared occurrence and artifact-generation stages. Appendix A.3 also provides the operational taxonomy, allocation procedure, and instance-generation controls.

3.4 Activity-Grounded Artifact Generation

Business activities leave partial accounts across systems, with each artifact reflecting its purpose and stage of work. We generate artifacts across 12 sources covering communication, documents and knowledge, engineering and work tracking, customer operations, monitoring and analytics, and personnel and access management. Source-specific specifications define the information each system records and how it is expressed: Slack captures conversational exchanges, Jira tracks issues and actions, and Amplitude records behavioral analyses and metrics. Format specifications draw on official API schemas, documented exports, and public standards where available, supplemented by explicitly defined internal formats (Appendix A.4.1).

An Artifact Plan connects an episode's ordered occurrences to the artifacts they create or modify, specifying the source, participants, content requirements, and lifecycle. One occurrence can contribute to several artifacts, while later occurrences can extend a thread, change a system record, or revise a document. Conditioned on these plans, an LLM generates shared observations and artifact-specific details. A structured fact layer tracks where observations are established and which later observations supersede them. Artifact dates and generation instructions distinguish proposed actions from completed outcomes. Source-specific specifications and persistent person-level profiles

guide expression, while programmatic renderers supply identifiers, timestamps, and metadata. Appendix A.4 details the plan schema, lifecycle rules, and generation procedure, with a worked example tracing an invoice investigation across systems from the initial discrepancy to its resolution.

4 QA Construction and Corpus Refinement

Question design. Business questions may require identifying the relevant activity through connections to other work or combining information distributed across several activities. We therefore initially design multi-hop and synthesis questions from the planning materials. For multi-hop questions, we explore the materials step by step: a fact recovered from the current record supplies the identifying information used to find the next. Each accepted step records the fact, its source, and its connection to the preceding step. We then formulate a question around the resulting path, leaving intermediate results to be recovered by the answering system. For synthesis questions, we establish a business scope and select a single information need, such as the handoff responsibilities of participating teams. We assemble the relevant facts across that scope and separately review their support and the answer's coverage. Each question is accompanied by a reference answer and reasoning claims: factual statements that establish intermediate relationships or supply the requested information. Appendix B.1 details both construction procedures. To check whether the proposed dependencies are needed to identify the requested answer, we withhold individual claims and test whether the remaining claims still suffice. These tests guide revisions to the recorded reasoning routes. Reference answers are decomposed into answer subfacts, with checks for coverage, redundancy, and preservation of qualifying conditions. Appendix B.4 details these checks.

Evidence grounding. We establish support for each claim and expand the coverage of its supporting files. We trace claims through the planning hierarchy to Artifact Plans and their generated artifacts. When an artifact does not express supporting content specified in its plan, we revise or supplement it to realize the planned content. To discover additional support, we search artifacts within the relevant business scopes using programmatic keyword screening followed by LLM-based semantic verification. The verification assesses both which claims the candidate files support and whether their content suffices to answer the question. Verified passages identify alternative supporting files or groups of files that jointly support a claim. Combining these support options across a reasoning route yields a sufficient evidence set. Questions with verified whole-question support in one file are retained as single-file questions. Appendix B.2 describes scope selection, evidence verification, and support-set construction.

Question-conditioned corpus augmentation. Recurring business activities produce artifacts that can be highly similar in content while referring to different events, participants, or stages of work. The extensive repetition and overlap of real company operations are difficult to reproduce exhaustively. To enrich the local context around evaluated business situations, we synthesize competing claims conditioned on each question's reasoning claims. For multi-hop questions, these introduce similar candidates distinguished by information recovered earlier. For example, ticket CS-1842 identifies billing release v2.3, whose rollback owner is Maya; an added plan assigns Alex to v2.2. The ticket's release identifier determines which assignment applies. For synthesis questions, comparable activities lie outside the requested scope, such as a January release added alongside evidence for a question about February release owners. Proposed claims are reviewed for meaningful competition and conflicts with existing claims before artifact generation. We then verify their expression in the generated artifacts and check for unintended support for target and other potentially affected question claims. Such support is preferentially removed through revision; retained, verified support is incorporated into the evidence annotations. Appendix B.3 details claim review, artifact verification, and evidence updates.

Benchmark summary. The final benchmark contains 966 questions: 804 multi-hop, 143 synthesis, and 19 single-file questions. The reported minimum sufficient support spans multiple files for 98.0% of questions and multiple source systems for 64.6% (Appendix B.2). Human assessment of 100 questions, with two reviewers and third-reviewer follow-up on flagged issues, yields pre-revision mean ratings of 1.76–1.98 out of 2 across five dimensions. Identified issues were corrected, and the reported experiments use the revised version (Appendix C.3). The final corpus contains 809,170 files across 12 source systems, including 14,135 augmentation files, and totals 1.213 billion tokens in its exported serialization. Figure 3 summarizes corpus composition; Appendix C provides detailed statistics and counting conventions.

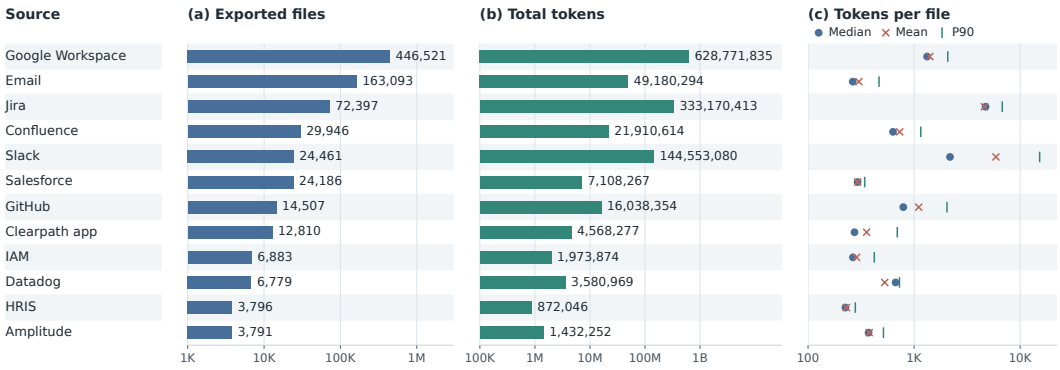


Figure 3: Corpus composition and file lengths for all 809,170 evaluation files. Source order is fixed across panels: (a) exported-file counts, (b) raw token totals, and (c) per-file median, mean, and P90 token lengths. All horizontal axes use base-10 logarithmic scales. Tokens include exported metadata and serialization; Appendix C reports the counting protocol and tail statistics.

5 Experiments

We evaluate evidence retrieval and reasoning in an evolving company environment.

Experimental setup. We evaluate seven systems on 966 questions over 809,170 exported files. BM25 (Robertson & Zaragoza, 2009), Dense, Hybrid, and Hybrid + Reranker retrieve evidence for a shared GPT-5.6-sol (OpenAI, 2026) reader. Dense uses Qwen3-Embedding-4B; Hybrid combines lexical and dense rankings through reciprocal rank fusion (Cormack et al., 2009), and reranking uses Qwen3-Reranker-4B (Zhang et al., 2025). The Bash Agent adapts EnterpriseRAG-Bench's shell-based retrieval (Sun et al., 2026), using GPT-5.6-terra to select evidence for the same reader. Claude Sonnet 5 (Anthropic, 2026) and GPT-5.6-sol serve as end-to-end agents that search, read, and answer. GPT models use medium reasoning effort; Claude uses its provider-default setting. Systems access exported artifacts without construction plans or reference annotations. Appendix D.1 provides configurations, access protocols, and budgets.

Evaluation. We measure answer quality and evidence retrieval separately. Correctness is a binary judgment of whether an answer satisfies the question's essential requirements without conflicting with the reference. Completeness measures coverage of reference-answer subfacts; Combined Score averages their per-question products. GPT-5.6-sol evaluates correctness and subfact coverage with medium reasoning effort. Recall@20 measures the largest recovered fraction of an annotated sufficient file set within the final selection of up to 20 files. Questions receive equal weight. Metric definitions, scoring prompts, and paired bootstrap intervals appear in Appendices D.2, D.10, and D.3.

Overall performance. Table 1 compares answer quality and evidence recovery on the full corpus. Hybrid's 1.52-point advantage over BM25 has a paired 95% interval of $[-0.79, 3.81]$, whereas reranking improves Hybrid by 13.04 points $[10.30, 15.87]$. With the reader held fixed, selecting applicable evidence from the retrieved candidates yields a much larger improvement than combining the initial lexical and dense rankings.

Table 1: Full-corpus results on 966 questions. Corr.: correctness; Comp.: subfact completeness; Score: mean per-question product. Recall uses final selected files. All values are percentages. The separate gold-evidence protocol is in Appendix D.1.

System	Corr.	Comp.	Score	Recall@20
BM25 + reader	27.23	40.91	25.50	38.61
Dense + reader	22.36	35.88	21.25	25.84
Hybrid + reader	28.16	41.83	27.01	39.81
Hybrid + Reranker + reader	41.93	54.73	40.05	50.19
Bash Agent + reader	44.00	55.59	42.17	38.94
Claude Sonnet 5 agent	38.72	49.37	36.99	49.58
GPT-5.6-sol agent	51.66	63.09	49.68	51.33
Gold-evidence reference	96.79	98.05	95.72	–

Iterative search provides a second route to stronger performance. The Bash Agent reaches 42.17, and the GPT-5.6-sol agent achieves the highest overall score of 49.68, exceeding Hybrid + Reranker by 9.62 points $[5.96, 13.33]$. However, the agent configurations differ: Claude Sonnet 5 scores 36.99, below Hybrid + Reranker. The advantage is therefore specific to the evaluated configuration rather than shared by every agent. Across all seven systems, the highest answer completeness is 63.09%, leaving a substantial gap in answer coverage. The following analyses examine performance by question requirement and by the evidence encountered during answering.

Question requirements. The overall ranking conceals a pronounced difference between question families (Table 2). Sol outperforms Hybrid + Reranker on multi-hop questions, scoring 51.11 versus 36.70, but trails it on synthesis questions, scoring 37.93 versus 52.53. Synthesis completeness follows the same pattern: 70.15% for Sol versus 82.27% for Hybrid + Reranker. The two families test different information needs: multi-hop questions use an intermediate finding to identify subsequent evidence, whereas synthesis questions require coverage of complementary information within a specified scope. An overall advantage on this predominantly multi-hop benchmark does not imply stronger performance on both requirements.

Table 2: Combined Score (percent) by question family on the 966 main questions. Grouping uses recorded evidence requirements. Rerank denotes Hybrid + Reranker.

Question family	N	BM25	Dense	Hybrid	Rerank	Bash	Sol	Claude
Multi-hop	804	22.78	17.33	22.63	36.70	41.85	51.11	38.53
Synthesis	143	36.66	42.77	47.06	52.53	39.45	37.93	25.04
Single-file	19	56.58	25.00	61.84	88.16	76.32	77.63	61.84

Appendix D.4.1 examines this ordering within shared business-content labels and under a long-answer restriction.

We next examine how performance varies with the amount of information an answer must cover. Figure 4 groups the 804 multi-hop questions by reference-answer subfact count. Correctness decreases across the four groups for every system, while completeness follows a different pattern. For Sol, correctness falls from 61.38% for single-subfact answers to 36.17% for answers with at least seven subfacts; completeness changes from 61.38% to 57.01%. Hybrid + Reranker shows the same separation: correctness falls from 43.45% to 17.02%, while completeness rises from 44.83% to 50.38%.

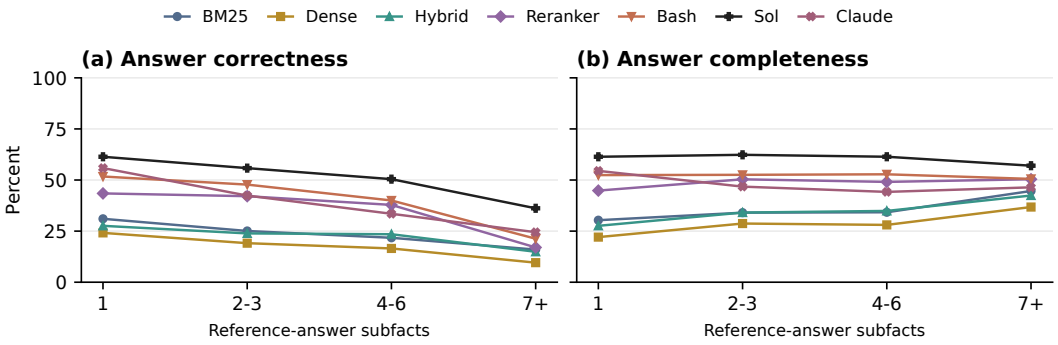


Figure 4: Answer correctness and completeness by reference-answer subfact count within 804 multi-hop questions. The four groups contain 145, 335, 230, and 94 questions. Both panels use the same systems and 0–100 scale; lines connect different question groups. Reranker denotes Hybrid + Reranker.

Richer answers can retain substantial factual coverage without satisfying the question as a whole. These are comparisons across questions. Adjusting for evidence counts and question length also yields a negative association between subfact count and Sol's correctness, but no clear association with completeness (Appendix D.4.2). The two measures distinguish partial coverage from overall correctness; Combined Score reflects both.

Time can also connect successive evidence-selection steps. In 152 multi-hop questions, a time value recovered at one step identifies the applicable activity or evidence at a later step. Sol scores 38.98 on these questions, compared with 53.93 on the other 652 multi-hop questions. The groups

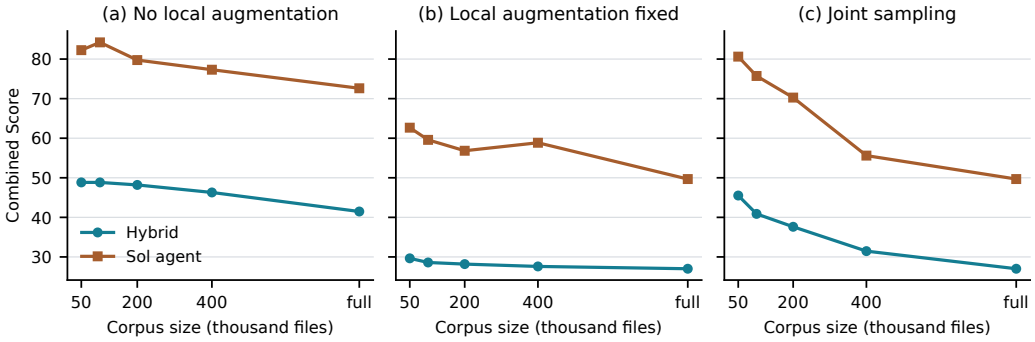


Figure 5: Background scale and local evidence competition on the same 966 questions. Supporting files remain protected under all three sampling policies. The full endpoints contain 795,042 files in (a) and 809,170 in (b–c). Appendix D.8 reports all 52 conditions, four metrics, and the shared evaluation protocol.

also differ in evidence requirements, so this is a descriptive association. Appendix D.4.3 reports covariate-adjusted associations and multiple-comparison correction across connection types. Time-based selection complements the update-following case below.

Evidence acquisition and use. The execution traces expose concrete gaps between locating and using evidence. In a responsibility-synthesis question, Sol found a document through a heading-only search result but did not open it. The answer named all three requested people, yet substituted one person's general repository duties for the requested standards-to-evidence responsibilities.

A payment-capacity question asks what specialist-capacity risk remained at the end of a deployment review and how its impact was handled. Sol read a January 23, 2026 discussion with allocations still pending, but missed the February 6 and February 21 updates. These recorded limited allocations for two Tier 1 dependencies and adjusted support for two other deployments. The answer recognized the shortage yet described the final blocker as deciding which Tier 1 dependency would receive support. It thus carried an earlier unresolved decision into the final state: locating the activity had not led to acquiring its subsequent updates (Appendix D.6.2).

A different failure occurs when the applicable content is already visible. For a partner-quality question, Hybrid + Reranker supplied the target-project link and its finding without truncation. The reader nevertheless selected a similar finding from another workstream. This case separates wrong-scope interpretation from missing evidence. Together, the cases show how reading the relevant details, following later updates, and maintaining project relationships affect answer quality. Appendix D.6 provides all six verified cases; Appendices D.5 and D.4 report path-level diagnostics and performance by evidence requirements, time, and business scope.

Corpus scale and local refinement. Our construction combines a broad company background with finer elaboration around evaluated business situations. To examine their contributions, we keep the questions and annotated supporting files fixed while varying the remaining corpus under three policies: removing unprotected augmentation and sampling the background, retaining all augmentation while sampling the background, and sampling both together. Seven augmentation-origin files belong to the protected set and remain in every condition. Figure 5 shows how the resulting search environments affect performance.

Expanding the background reduces performance even after unprotected augmentation is removed. Between 50k files and the full background corpus, Hybrid's score declines from 48.83 to 41.48 and Sol's from 82.25 to 72.62. For Sol, Recall@20 also falls from 79.67% to 67.95%, while completeness falls from 90.05% to 84.00%. Supporting files remain available, so the comparison changes the surrounding search environment rather than removing the evidence needed to answer. The endpoint losses occur alongside local reversals in the curves; Appendix D.8 supplies the sampling protocol and all intermediate measurements.

The two augmented policies distinguish a fixed local candidate set from one that grows with the corpus. At 50k files, Hybrid scores 29.65 when all augmentation is retained and 45.52 under joint sampling; at 400k, the scores are 27.60 and 31.47. Fixed augmentation exposes systems to the full set of local alternatives from the smallest size, whereas joint sampling increases both back-

ground and augmentation. At equal total file counts, the comparison therefore changes the mix of background and close alternatives. Removing unprotected augmentation improves all four tested systems in the full-corpus ablation, confirming local refinement's contribution.

Under joint sampling, the changes also extend to evidence and answer coverage: Sol's Recall@20 falls from 79.09% at 50k files to 51.33% at full scale, and completeness from 88.59% to 63.09%. The three policies thus provide complementary views of the constructed environment: wider company background, a fixed set of local competitors, and their joint expansion. The full metric curves and augmentation ablations appear in Appendix D.8.

Computational cost. Figure 6 relates answer quality to inference cost. Sol uses 89,692 input-plus-output tokens per question, about three times Hybrid + Reranker's 30,100, for a 9.62-point score gain. Bash uses 1,969,745 tokens and scores below Sol. It averages 24.5 model calls and 38.8 tool calls, versus 8.0 and 23.2 for Sol. The results connect each system's performance to its search and context-management costs.

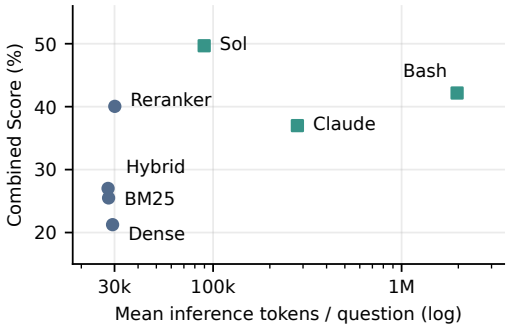


Figure 6: Answer quality versus inference tokens for seven systems. Tokens include repeated and cached input where reported.

Median recorded duration is 320.0 seconds for Bash's retrieval and reader, and 274.4 seconds for Sol's exploration and answering. Pipeline timing covers only the reader, so we retain separate timing boundaries rather than combining them into an end-to-end latency ranking. Appendix D.7 reports usage and latency distributions for all seven systems, along with reader truncation for retrieval pipelines. Judging and offline corpus construction, embedding, and indexing are excluded.

Reader and judge sensitivity. To distinguish answer generation from scoring effects, we cross GPT-5.6-sol and Claude Sonnet 5 readers and judges on saved Hybrid + Reranker inputs. Evidence is identical for the readers, and each judge comparison uses the same saved answers. Table 3 summarizes the paired comparisons, whose sample sizes reflect available answers and judgments.

Table 3: Fixed-evidence reader/judge comparisons. Differences are Claude minus GPT in Combined Score, with paired 95% bootstrap intervals. Each row fixes the other component and question subset.

Changed component	Fixed component	Paired N	Score difference [95% CI]
Judge	GPT reader answers	933	$+1.23$ $[-0.28, 2.75]$
Judge	Claude reader answers	830	$+3.50$ $[2.03, 4.98]$
Reader	GPT judge	838	-5.09 $[-7.36, -2.88]$

Replacing the GPT judge with Claude changes scores by $+1.23$ points on GPT answers, with an interval including zero, and by $+3.50$ points on Claude answers. Binary correctness agreement is 93.68% and 92.53%, respectively. With GPT judging 838 paired reader answers, the GPT reader scores 42.56 and Claude scores 37.47. Thus answer generation remains a source of variation after retrieval is fixed, while judge choice also affects the reported score. Appendix D.9 details the fixed-evidence protocol, paired subsets, refusals, and unscored outcomes.

6 Conclusion

We introduced EnterpriseSaga, a billion-token benchmark for evidence retrieval and reasoning within a shared, evolving synthetic company. Its coarse-to-fine framework connects company-wide activities to cross-system artifacts and enriches local business contexts around grounded questions. Across seven systems, experiments reveal distinct strengths in tracing relationships and synthesizing information, alongside performance losses as the company background expands. Execution traces show how missed updates, incomplete reading, and confusion between business scopes lead to incorrect or incomplete answers. Together, these findings highlight the challenge of connecting evidence to the activities and conditions under which it applies. EnterpriseSaga provides a structured testbed for developing and evaluating methods that address this challenge at enterprise scale.

AI USE DISCLOSURE

We used large language models for dataset construction and experimental evaluation, and AI tools for language polishing of the manuscript.

REFERENCES

Amirhossein Abaskohi, Tianyi Chen, Miguel Muñoz-Mármol, Curtis Fox, Amrutha Varshini Ramesh, Étienne Marcotte, Xing Han Lù, Nicolas Chapados, Spandana Gella, Christopher Pal, et al. Drbench: A realistic benchmark for enterprise deep research. In *International Conference on Learning Representations*, volume 2026, pp. 6727–6795, 2026.

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. L-eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14388–14411, 2024.

Anthropic. Claude Sonnet 5 System Card, 2026. URL `https://www.anthropic.com/claude-sonnet-5-system-card`.

Zahra Atf and Peter R Lewis. Scenariobench: Trace-grounded compliance evaluation for text-to-sql and rag. In *2025 IEEE International Conference on Collaborative Advances in Software and COmputiNg (CASCON)*, pp. 615–620. IEEE, 2025.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. Longbench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 3119–3137, 2024.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, et al. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664, 2025.

Haonan Bian, Zhiyuan Yao, Sen Hu, Zishan Xu, Shaolei Zhang, Yifu Guo, Ziliang Yang, Xueran Han, Huacan Wang, and Ronghao Chen. Realmem: Benchmarking llms in real-world memory-driven interaction. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 14349–14365, 2026.

Léo Boisvert, Megh Thakkar, Maxime Gasse, Massimo Caccia, Thibault L De Chezelles, Quentin Cappart, Nicolas Chapados, Alexandre Lacoste, and Alexandre Drouin. Workarena++: Towards compositional planning and reasoning-based common knowledge work tasks. *Advances in Neural Information Processing Systems*, 37:5996–6051, 2024.

Huacan Chai, Yukai Wang, Yingxuan Yang, Dan Peng, Yuanyi Song, Zhihui Fu, Weiwen Liu, Jianghao Lin, Jun Wang, and Weinan Zhang. Smmbench: A benchmark for source-distributed multimodal agent memory. *arXiv preprint arXiv:2605.15710*, 2026.

Jonathan D Chang, Andrew Drozdov, Shubham Toshniwal, Owen Oertell, Alexander Trott, Jacob Portes, Abhay Gupta, Pallavi Koppol, Ashutosh Baheti, Sean Kulinski, et al. Karl: Knowledge agents via reinforcement learning. *arXiv preprint arXiv:2603.05218*, 2026.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, et al. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *arXiv preprint arXiv:2508.06600*, 2025.

Ziyang Chen, Xing Wu, Junlong Jia, Chaochen Gao, Qi Fu, Debing Zhang, and Songlin Hu. Longbench pro: A more realistic and comprehensive bilingual long-context evaluation benchmark. *arXiv preprint arXiv:2601.02872*, 2026.

Prafulla Kumar Choubey, Xiangyu Peng, Shilpa Bhagavath, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. Benchmarking deep search over heterogeneous enterprise data. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 501–517, 2025.

Dvir Cohen, Lin Burg, Sviatoslav Pykhnivskyi, Hagit Gur, Stanislav Kovynov, Olga Atzmon, and Gilad Barkan. Wixqa: A multi-dataset benchmark for enterprise retrieval-augmented generation. *arXiv preprint arXiv:2505.08643*, 2025.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 758–759, 2009.

Darshan Deshpande, Varun Gangal, Hersh Mehta, Anand Kannappan, Rebecca Qian, and Peng Wang. Memtrack: Evaluating long-term memory and state tracking in multi-platform dynamic agent environments. *arXiv preprint arXiv:2510.01353*, 2025.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, et al. Workarena: How capable are web agents at solving common knowledge work tasks? *arXiv preprint arXiv:2403.07718*, 2024.

Simone Filice, Guy Horowitz, David Carmel, Zohar Karnin, Liane Lewin-Eytan, and Yoelle Maarek. Generating diverse q&a benchmarks for rag evaluation with datamorgana. *arXiv preprint arXiv:2501.12789*, 2025.

Jeffrey Flynt. Orgforge: A multi-agent simulation framework for verifiable synthetic corporate corpora. *arXiv preprint arXiv:2603.14997*, 2026.

Wenjie Fu, Xiaoting Qin, Jue Zhang, Qingwei Lin, Lukas Wutschitz, Robert Sim, Saravan Rajmohan, and Dongmei Zhang. Ci-work: Benchmarking contextual integrity in enterprise llm agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL 2026)*, pp. 1483–1508, 2026.

Zexue He, Yu Wang, Churan Zhi, Yuanzhe Hu, Tzu-Ping Chen, Lang Yin, Ze Chen, Tong Arthur Wu, Siru Ouyang, Zihan Wang, et al. Memoryarena: Benchmarking agent memory in interdependent multi-session agentic tasks. *arXiv preprint arXiv:2602.16313*, 2026.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*, 2024.

Chuanrui Hu, Tong Li, Xingze Gao, Hongda Chen, Yi Bai, Dannong Xu, Tianwei Lin, Xiaohong Li, Yunyun Han, Jian Pei, et al. Evaluating long-horizon memory for multi-party collaborative dialogues. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 9082–9090, 2026a.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in llm agents via incremental multi-turn interactions. In *International Conference on Learning Representations*, volume 2026, pp. 156259–156291, 2026b.

Kung-Hsiang Huang, Akshara Prabhakar, Sidharth Dhawan, Yixin Mao, Huan Wang, Silvio Savarese, Caiming Xiong, Philippe Laban, and Chien-Sheng Wu. Crmarena: Understanding the capacity of llm agents to perform professional crm tasks in realistic environments. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3830–3850, 2025a.

Kung-Hsiang Huang, Akshara Prabhakar, Onkar Thorat, Divyansh Agarwal, Prafulla Kumar Choubey, Yixin Mao, Silvio Savarese, Caiming Xiong, and Chien-Sheng Wu. Crmarena-pro: Holistic assessment of llm agents across diverse business scenarios and interactions. *arXiv preprint arXiv:2505.18878*, 2025b.

Jooyoung Jeong and Sang-Goo Lee. Permission-aware rag: Identity and access management (iam)-based access filtering in multi-resource environments. *IEEE Access*, 13:192819–192835, 2025.

Saurabh Jha, Rohan Arora, Yuji Watanabe, Takumi Yanagawa, Yinfang Chen, Jackson Clark, Bhavya Bhavya, Mudit Verma, Harshit Kumar, Hirokuni Kitahara, et al. Itbench: Evaluating ai agents across diverse real-world it automation tasks. *arXiv preprint arXiv:2502.05352*, 2025.

Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J Taylor, and Dan Roth. Know me, respond to me: Benchmarking llms for dynamic user profiling and personalized responses at scale. *arXiv preprint arXiv:2504.14225*, 2025a.

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, et al. Personamem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025b.

Gurusha Juneja, Alon Albalak, Wenyue Hua, and William Yang Wang. Magpie: A dataset for multi-agent contextual privacy evaluation. *arXiv preprint arXiv:2506.20737*, 2025.

Philippe Laban, Alexander Richard Fabbri, Caiming Xiong, and Chien-Sheng Wu. Summary of a haystack: A challenge to long-context llms and rag systems. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9885–9903, 2024.

Taewhoo Lee, Chanwoong Yoon, Kyochul Jang, Donghyeon Lee, Minju Song, Hyunjae Kim, and Jaewoo Kang. Ethic: Evaluating large language models on long-context tasks with high information coverage. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5497–5512, 2025.

Yifei Li, Weidong Guo, Lingling Zhang, Rongman Xu, Muye Huang, Hui Liu, Lijiao Xu, Yu Xu, and Jun Liu. Locomo-plus: Beyond-factual cognitive memory evaluation framework for llm agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25085–25100, 2026.

Chengzhi Liu, Yuzhe Yang, Sophia Xiao Pu, Yepeng Liu, Lin Long, Yichen Guo, Nuo Chen, Zhaotian Weng, Elena Kochkina, Simerjot Kaur, et al. Worldmemarena: Evaluating multimodal agent memory through action-world interaction. *arXiv preprint arXiv:2605.29341*, 2026.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of llm agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13851–13870, 2024.

Huiqi Miao, Xinbao Sun, Bo Wang, Fanyu Meng, Lijun Mei, Na Wu, Di Jin, Chao Deng, and Junlan Feng. Enterpriserag: Benchmarking llm instruction adherence and robustness under non-ideal enterprise retrieval. *arXiv preprint arXiv:2608.11584*, 2026.

Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Trung Bui, Ryan A Rossi, Seunghyun Yoon, and Hinrich Schütze. Nolima: Long-context evaluation beyond literal matching. *arXiv preprint arXiv:2502.05167*, 2025.

Kenichirou Narita, Siqi Peng, Taku Fukui, Moyuru Yamada, Satoshi Munakata, and Satoru Takahashi. Overcoming the" impracticality" of rag: Proposing a real-world benchmark and multi-dimensional diagnostic framework. *arXiv preprint arXiv:2604.02640*, 2026.

OpenAI. GPT-5.6 System Card, 2026. URL `https://deploymentsafety.openai.com/gpt-5-6`.

Krista Opsahl-Ong, Arnav Singhvi, Jasmine Collins, Ivan Zhou, Cindy Wang, Ashutosh Baheti, Owen Oertell, Jacob Portes, Sam Havens, Erich Elsen, et al. Officeqa pro: An enterprise benchmark for end-to-end grounded reasoning. *arXiv preprint arXiv:2603.08655*, 2026.

Akshara Prabhakar, Roshan Ram, Zixiang Chen, Silvio Savarese, Frank Wang, Caiming Xiong, Huan Wang, and Weiran Yao. Enterprise deep research: Steerable multi-agent deep research for enterprise analytics. *arXiv preprint arXiv:2510.17797*, 2025.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: a temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*, 2025.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Foundations and trends® in information retrieval*, 4(1-2):1–174, 2009.

Debdeep Sanyal, Umakanta Maharana, Yash Sinha, Hong Ming Tan, Shirish Karande, Mohan Kankanhalli, and Murari Mandal. Orgaccess: A benchmark for role based access control in organization scale llms. *arXiv preprint arXiv:2505.19165*, 2025.

Uri Shaham, Maor Ivgi, Avia Efrat, Jonathan Berant, and Omer Levy. Zeroscrolls: A zero-shot benchmark for long text understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 7977–7989, 2023.

Ye Shen, Dun Pei, Yiqiu Guo, Junying Wang, Yijin Guo, Zicheng Zhang, Qi Jia, Jun Zhou, and Guangtao Zhai. Evolmem: A cognitive-driven benchmark for multi-session dialogue memory. *arXiv preprint arXiv:2601.03543*, 2026a.

Yiting Shen, Kun Li, Wei Zhou, and Songlin Hu. Mem2actbench: A benchmark for evaluating long-term memory utilization in task-oriented autonomous agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8173–8190, 2026b.

Olly Styles, Sam Miller, Patricio Cerda-Mardini, Tanaya Guha, Victor Sanchez, and Bertie Vidgen. Workbench: a benchmark dataset for agents in a realistic workplace setting. *arXiv preprint arXiv:2405.00823*, 2024.

Yuhong Sun, Joachim Rahmfeld, Chris Weaver, Weijia Chen, Roshan Desai, Wenxi Huang, and Mark H Butler. Enterpriserag-bench: A rag benchmark for company internal knowledge. *arXiv preprint arXiv:2605.05253*, 2026.

Haoran Tan, Zeyu Zhang, Chen Ma, Xu Chen, Quanyu Dai, and Zhenhua Dong. Membench: Towards more comprehensive evaluation on the memory of llm-based agents. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19336–19352, 2025.

Mohammad Tavakoli, Alireza Salemi, Carrie Ye, Mohamed Abdalla, Hamed Zamani, and J Ross Mitchell. Beyond a million tokens: Benchmarking and enhancing long-term memory in llms. *arXiv preprint arXiv:2510.27246*, 2025.

Md Nayem Uddin, Kumar Shubham, Eduardo Blanco, Chitta Baral, and Gengyu Wang. From recall to forgetting: Benchmarking long-term memory for personalized agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 26814–26841, 2026.

Amey Varhade, Ananya Sutradhar, Ravishankar Krishnaswamy, and Navin Goyal. Winsyn: An automated pipeline for realistic enterprise question-answering evaluation. *arXiv preprint arXiv:2609.12171*, 2026.

Yuyao Wang, Zhongjian Zhang, Mo Chi, Kaichi Yu, Yuhan Li, Miao Peng, Bing Tong, Chen Zhang, Yan Zhou, and Jia Li. Evomembench: Benchmarking agent memory from a self-evolving perspective. *arXiv preprint arXiv:2605.18421*, 2026.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*, 2024.

Di Wu, Zixiang Ji, Asmi Kawatkar, Bryan Kwan, Jia-Chen Gu, Nanyun Peng, and Kai-Wei Chang. Longmemeval-v2: Evaluating long-term agent memory toward experienced colleagues. *arXiv preprint arXiv:2605.12493*, 2026.

Wenya Xie, Shengming Zhou, Zelin Li, Pouya Parsa, Shuang Zhou, Xinheng Ding, Chinmay Arvind, Guanchu Wang, Vladimir Braverman, Ali Payani, et al. Dynamicmem: A long-horizon memory benchmark in real-world settings. *arXiv preprint arXiv:2606.22877*, 2026.

Frank Fangzheng Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, et al. Theagentcompany: benchmarking llm agents on consequential real world tasks. *Advances in Neural Information Processing Systems*, 38, 2026.

Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. Helmet: How to evaluate long-context language models effectively and thoroughly. doi: 10.48550. *arXiv preprint arXiv.2410.02694*, 2025.

Tao Yu, Hao Wang, Changyu Li, Shenghua Chai, Minghui Zhang, Zhongtian Luo, Yuxuan Zhou, Haopeng Jin, Zhaolu Kang, Jiabing Yang, et al. Beyond the all-in-one agent: Benchmarking role-specialized multi-agent collaboration in enterprise workflows. *arXiv preprint arXiv:2605.08761*, 2026.

Han Zhang, Zihao Tang, Xin Yu, Xiao Liu, Yeyun Gong, Haizhen Huang, Yan Lu, Weiwei Deng, Feng Sun, Qi Zhang, et al. Beyond static dialogues: Benchmarking realistic, heterogeneous, and evolving long-term memory. *arXiv preprint arXiv:2605.31086*, 2026.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, et al. ∞ bench: Extending long context evaluation beyond 100k tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15262–15277, 2024.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Yujie Zhao, Boqin Yuan, Junbo Huang, Haocheng Yuan, Zhongming Yu, Haozhou Xu, Lanxiang Hu, Abhilash Shankarampeta, Zimeng Huang, Wentao Ni, et al. Ama-bench: Evaluating long-horizon memory for agentic applications. *arXiv preprint arXiv:2602.22769*, 2026.

Akrin Zheng, Alexander Wu, and Alaia Liu. Entlore: A graph-grounded benchmark for latent organizational reasoning in enterprise question answering. *arXiv preprint arXiv:2608.10679*, 2026.

A Corpus Construction Details

The construction proceeds from shared company context to project and operational activities, then to source-specific artifacts. We first describe the organization, identities, and dated changes that constrain generation. The following sections trace the project and operational planning paths, their checks, and the artifact plans that connect activities to published evidence. Counts distinguish planning objects from physical exported files throughout.

Construction models. Across the corpus-construction stages described in this appendix, all batch generation uses GPT-5.6-luna and all model-based review uses GPT-5.6-sol with medium reasoning effort (OpenAI, 2026). This assignment covers company-context generation, project and operational planning, and source-specific artifact generation and their model-based checks. Human review and programmatic validation are described separately at the relevant stages. QA construction configurations are reported in Appendix B.

A.1 Company Context Modeling

Company profile and initial state. Clearpath is a fictional B2B SaaS company whose business profile is summarized in Table A1. Its sector, initial scale of approximately 1,000 employees, and 18-month horizon were specified through human guidance and elaborated through iterative human–LLM collaboration. The overview defines the company's business and organization; a separate baseline dated January 1, 2026 records existing capabilities and operational gaps (Table A2) that inform subsequent planning.

Table A1: Company profile and business scope. All attributes describe the fictional company used for corpus construction.

Attribute	Specification
Company	Clearpath, a B2B SaaS company
Stage and location	Founded in 2015; late-stage private software company headquartered in San Francisco
Core business	Customer-support and service-operations software
Product scope	Shared inbox and ticketing; live chat; knowledge management; workflow automation; AI assistance; reporting and analytics
Customer segments	Mid-market and enterprise customers in SaaS, ecommerce, fintech, subscription businesses, and digital platforms
Revenue model	Subscriptions, seat- and usage-based charges, premium add-ons, implementation services, and enterprise support
Simulation horizon	January 1, 2026–June 30, 2027 (18 months)
Initial organization	1,000 employees; 17 departments; 42 organizational groups; 91 named teams
Strategic initiatives	AI workflows; enterprise readiness; platform modernization; commercial expansion; implementation efficiency; customer health and retention

Strategic direction. The roadmap translates the initial conditions into six initiatives, each with an owner, priority, scope, quarterly milestones, and dependencies. Cross-initiative responsibilities and dependencies are summarized in Table A4 in Appendix A.2, alongside the project-planning details.

Organizational composition. Table A3 reports the initial organization and department-level workforce changes. The directory contains 17 departments, 42 organizational groups, and 91 named teams. Engineering is the largest department; customer success, support, and professional services provide distinct adoption, service, and implementation functions. Each employee record includes an identifier, display name, email, title, department, group, team, start date, reporting manager where applicable, and a short biography. Leadership and individual functions contribute to department headcounts alongside named teams. The 1,000 initial records have unique employee identifiers and 995 distinct display names. Employee identifiers serve as stable keys for construction-stage joins.

External entities. Figure A1 summarizes 60 registered external organizations and 169 associated contacts. The registry includes customers and prospects as well as implementation partners, technology vendors, auditors, financial institutions, and other service providers. Organization records describe business category, industry, and geographic scope; contact records specify individual roles and organizational affiliations. For example, the customer organization LatticePay is represented

Table A2: Selected capabilities and limitations in the January 1, 2026 initial-state specification.

Area	Existing capabilities	Initial limitations
Core support	Shared inbox, ticketing, live chat, routing, and agent case handling	Fragmented cross-channel context and inconsistent ticket/conversation lifecycles
Automation and AI	Rule-based workflows, suggested replies, summarization, and recommendations	Limited execution observability; uneven context grounding and incomplete feedback and policy controls
Enterprise trust	SSO/SAML, SCIM, roles, permissions, and audit capabilities	Provisioning reliability, granular access control, regional controls, and assurance evidence remain incomplete
Integration	Connections to CRM, billing, order, and product systems	Inconsistent data contracts and mappings; teams duplicate data-access work without a shared context platform
Commercial operations	Subscriptions, premium add-ons, enterprise sales, and expansion motions	Metering, entitlements, value proof, and commercial rollout readiness are insufficiently coordinated
Implementation and retention	Implementation services, migration, partner delivery, customer success, renewals, and support	Manual deployment work, inconsistent handoffs, fragmented health signals, and late risk detection

Table A3: Department-level workforce over the 18-month horizon. Teams are initial named teams; leadership and individual functions are included in headcounts. Final equals initial plus hires minus departures; employees on leave remain included.

Department	Initial	Teams	Hires	Departures	Final
Engineering	270	12	65	38	297
Sales	175	9	50	42	183
Customer Success	96	6	24	17	103
Support	88	7	25	20	93
Marketing	68	9	15	11	72
Professional Services	68	7	17	12	73
Product Management	42	7	10	5	47
Data & AI	36	5	12	4	44
Security & IT	30	6	7	3	34
Finance	24	4	4	2	26
Design	22	5	5	2	25
People / HR	22	4	4	2	24
Revenue Operations	15	4	4	2	17
Legal & Compliance	14	3	2	1	15
Executive	12	0	1	1	12
Business Operations	10	2	3	1	12
Partnerships	8	1	2	1	9
Total	1,000	91	250	164	1,086

by contacts in customer operations, security architecture, privacy, and support technology. These registries provide shared external identities for generation.

Organizational evolution. Workforce generation separates aggregate capacity planning from individual transitions. An LLM-based stage develops a department-by-quarter workforce plan under fixed department-level staffing targets. The resulting plan specifies movement counts by department and quarter. Programmatic expansion then uses the workforce plan and company-event requirements to assign employees to dated personnel events. Quarterly timing may be rephased while preserving department and company totals over the horizon. Each event records an employee identifier, type, effective date, previous and updated states, and a reason; selected events also record company-event references, project impacts, onboarding ramps, or handoff requirements.

The registry contains 666 events affecting 642 distinct employees (Figure A2). Across the full period, 250 hires extend the initial 1,000 identities to 1,250, while 164 departures leave 1,086 employees at the horizon end. Team transfers occur within departments, and employees on leave remain

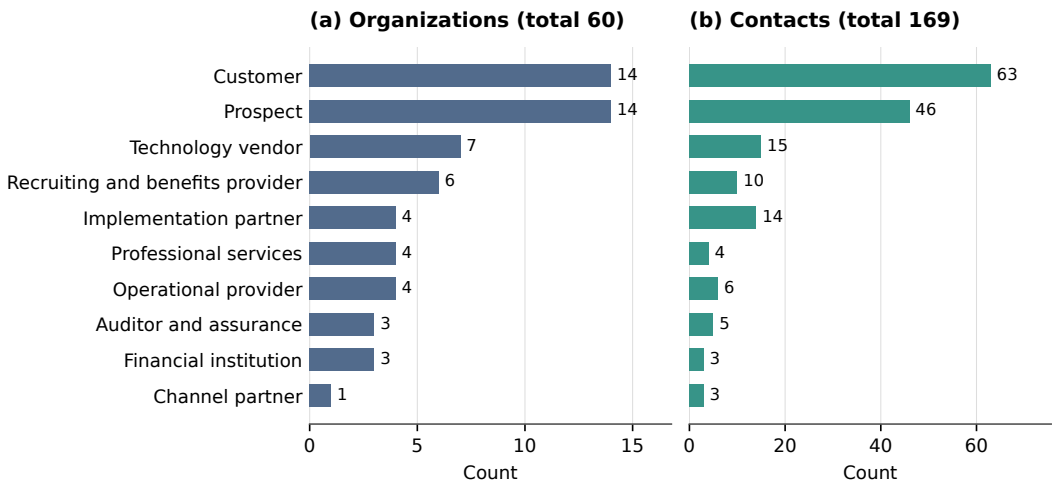


Figure A1: External-entity registries: 60 organizations and 169 associated contacts. Bar labels give exact counts; organization identifiers link contacts to organizations.

included in headcount. Four leave spells remain open at the end of the horizon. These counts distinguish identities observed over time, endpoint headcount, and event occurrences.

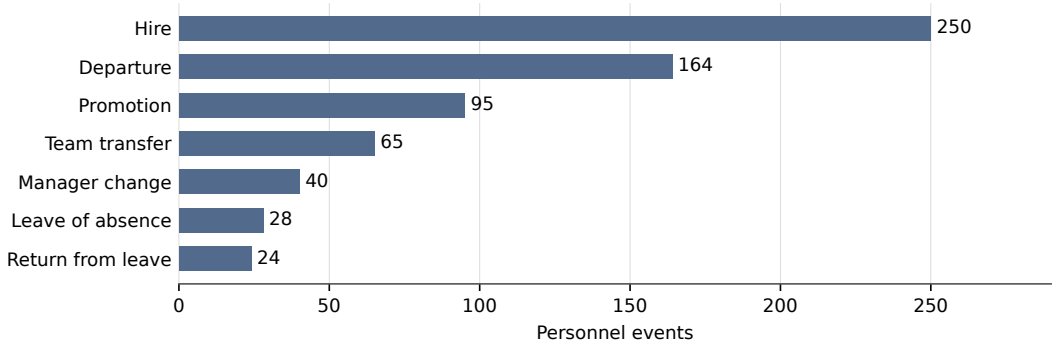


Figure A2: The 666 personnel events affecting 642 employees. Transfers remain within departments; standalone manager changes exclude reporting changes already caused by promotion or transfer. Leave starts and returns are events, not endpoint headcounts.

The company-event layer couples selected personnel changes to the evolving project portfolio. Its 22 dated events are planned after the initial portfolio and can revise project scope, schedules, or workforce requirements. The full company narrative and these project consequences are reserved for Appendix A.2. A linked leadership transition illustrates these constraints: President and COO Daniel Park (`emp-0002`) departs on December 18, 2026; Elena Harris (`emp-1245`) joins as COO on January 23, 2027 and reports to the CEO. On the same date, VP of Business Operations Layla Hassan (`emp-0991`) changes reporting manager from Park to Harris. Employment, role, leave, and ramp-up dates supply constraints for subsequent participant assignments; complete project staffing is handled in the project-planning stages.

Workforce-planning prompt. The box below reproduces selected constraints from the Step 11.1 template, conditioned on department targets, an organization summary, and the company overview. The caller validates department coverage, staffing totals, and quarterly hire/departure sums before accepting the plan.

Selected prompt excerpts: workforce movement planning

Use the locked department baseline and hire/departure targets supplied below. The company ends with 1,086 employees: 1,000 + 250 hires - 164 departures. Allocate the locked company totals of 95 promotions, 65 within-department team transfers, 40 standalone manager/reporting-line changes, 28

extended leaves, and 24 returns across departments plausibly. A return count may not exceed leave starts in its department.

Top-level keys: `metadata`, `quarters`, `departments`, `company_totals`, and `planning_principles`. Each department record must contain `department`, `baseline_headcount`, `hires`, `departures`, `ending_headcount`, `promotions`, `team_transfers`, `manager_changes`, `leave_starts`, `leave_returns`, `quarterly_hires`, `quarterly_departures`, and `rationale`. Quarterly mappings must cover exactly 2026-Q1 through 2027-Q2 and sum to their department totals.

Explain that project-critical people will be resolved after company events, while routine changes remain realistic background operations.

Source: `src/prompts/company_story_universe.py`, `STEP11_WORKFORCE_SYSTEM_PROMPT`. Selected passages from the current code template, with original wording and formatting adapted for print. Input placeholders: `{locked_department_targets}`, `{organization_summary}`, `{company_overview}`.

Provenance and oversight. The inventories summarize the company specifications, identity registries, and dated personnel events retained during construction. Human review covers high-level business and organizational specifications, personnel distribution, and planned workforce changes. Programmatic checks cover staffing arithmetic, event counts, identifiers, employment windows, and manager references. Communication styles and source-specific record conventions belong to Appendix A.4.

A.2 Project Planning and Narratives

A.2.1 Planning Hierarchy and Expansion

Project planning translates the company roadmap into a bounded portfolio. The initial portfolio contains 79 projects referencing 18 programs and 57 workstreams. Each project identifies its primary program, parent workstream, objective, planned dates, and participating teams. For example, *AI Quality Taxonomy and Feedback Instrumentation* (`proj-001`) references `prog-001` and `ws-001`. Strategic responsibility boundaries are summarized in Table A4.

Strategic responsibility boundaries. Table A4 records the scope divisions used for project planning. The Customer Context and Integration Platform belongs to Initiative 3 and provides shared capabilities for several initiatives. Initiative 5 owns implementation through stable go-live and severe escalation recovery; Initiative 6 owns normal post-go-live health and retention. Initiative 4 consumes the resulting health signals for commercial planning.

Table A4: Strategic initiative scopes and responsibility boundaries, summarized from the company roadmap. Initiative names are shortened.

Initiative	Main responsibility	Boundary or dependency
1. AI workflows	AI behavior, evaluation, feedback, and governed assistance	Consumes shared context and execution foundations from Initiative 3
2. Enterprise readiness	Identity, administration, security, compliance, and assurance	Coordinates security and compliance requirements for AI policy controls
3. Core platform	Core workflows, reliability, analytics, scale, and shared integration foundations	Provides reusable capabilities to multiple initiatives
4. Commercial expansion	Packaging, sales motions, expansion, and forecasting	Consumes adoption and health signals; depends on product and deployment readiness
5. Implementation and recovery	Deployment through stable go-live; severe strategic escalation recovery	Hands normal post-go-live lifecycle management to Initiative 6
6. Customer health and retention	Health signals, adoption, retention actions, and quality feedback	Relevant product and engineering teams retain ownership of product fixes

Company-level events introduce five additional projects. The resulting portfolio is decomposed into subprojects and elaborated into stories, episodes, and occurrences. Table A5 reports counts from the retained planning outputs, before downstream record planning and generation.

Table A5: Project-planning inventory.

Planning unit	Count	Counting boundary
Programs	18	Distinct identifiers referenced by initial projects
Workstreams	57	Distinct identifiers referenced by initial projects
Initial projects	79	Portfolio before company-event planning
Company-level events	22	Shared developments with project-specific consequences
Event-triggered projects	5	Additional projects linked to originating events
Total projects	84	82 completed; 2 in progress at the horizon end
Subprojects	321	Bounded execution units within parent projects
Project episodes	2,580	Narrative units generated from subproject stories
Project occurrences	10,309	Events in the retained occurrence sequences

Projects contain three to six subprojects, whose execution windows may overlap within an active parent-project window. Subproject stories produce seven to ten episodes each, with eight episodes in 300 of 321 cases. Episodes contain three to five occurrences, with four in 2,565 of 2,580 cases.

A.2.2 Company Evolution and Cross-Project Relationships

LLM-based company-event planning specifies developments affecting the initial portfolio, using the company context and workforce plan. The event registry records dates, causes, affected projects, and project-specific consequences. Updated project-status records retain scope, schedule, and ownership changes, while the company story connects these developments into a shared narrative.

Project-status records distinguish ongoing execution, completed work, and explicit reopening. Later events may consume completed outputs, extend active work, reopen an existing project, or initiate a new project. Table A6 illustrates these transitions and includes all five event-triggered projects.

Table A6: Selected company events and recorded project consequences. Descriptions are condensed from the event registry.

Event and date	Consequence for existing work	New project
`evt-007`, 2026-07-19: Assurance readiness finding	Completed audit-event interfaces support continuous evidence collection; the source project remains closed.	`proj-080`: Continuous Control Evidence Automation
`evt-009`, 2026-09-14: Lifecycle migration regression	Two migration/intake targets move from October 31 to November 30; affected rollout paths are gated.	-
`evt-010`, 2026-10-22: Automation storm	Runtime safeguards take priority; an orchestration-pattern delivery target moves from January 31 to March 31, 2027.	`proj-081`: High-Volume Automation Safeguards and Kill Switches
`evt-015`, 2027-01-30: Regulated AI customer commitment	Existing control and assurance capabilities support a separate validation effort.	`proj-082`: Regulated AI Deployment Validation Pack
`evt-017`, 2027-03-09: Multilingual continuity failure	Completed assistance and continuity projects remain closed; recovery receives separate ownership.	`proj-083`: Multilingual Conversation Continuity Recovery
`evt-018`, 2027-03-31: Usage reconciliation defects	Billing-reconciliation completion moves from March 31 to May 15; the bundle-design project retains its completed status.	-
`evt-019`, 2027-04-15: Health-model bias and drift	An existing calibration project is reopened for correction, alongside new monitoring work.	`proj-084`: Health Signal Drift Monitoring and Governance

A relationship index supplements the hierarchy with 802 relation records: 84 parent-child sequences, 237 subproject handoffs, 22 shared company events, 32 shared personnel events, 392 shared employee assignments, eight shared external organizations, and 27 shared external actors. Each record

groups the projects and subprojects connected through a particular sequence, handoff, event, or participant.

Episode generation receives required relationships and contextual associations. Required relationships specify connections to represent in the relevant sequences. Contextual associations identify shared participants or entities that inform local interactions. Explicit dependencies, event effects, and handoff records establish the causal links used during generation.

A.2.3 Narrative Elaboration and Construction Controls

Programmatic story scaffolds establish project boundaries, participants, event effects, and handoffs. LLM-generated material packs supply local detail through concrete work objects, focused narrative threads, candidate state sequences, triggers, and open questions. Episode generation develops these possibilities into connected sequences under the source stories' constraints.

Staffing uses role-dependent allocation caps, concurrent-assignment limits, and effective-dated availability. Eligible intervals account for employment, leave, team membership, and recorded onboarding ramps. Subproject staffing separates execution responsibilities from business review or approval. Staffing exceptions record cases requiring alternative coverage. Numerical allocation settings are specified in the staffing implementation.

Generation prompts specify scope and temporal constraints, including the dates when deliverables become available or complete. Programmatic checks test structural and temporal properties of the planning outputs. Model-based generation review assesses their consistency with the supplied context. Table A7 summarizes the generation mechanisms and checks.

Table A7: Narrative construction mechanisms and checks.

Component	Construction mechanism	Checks and review targets
Subprojects and story scaffolds	Programmatic decomposition and assembly from project states, events, staffing, and handoffs	Structural checks on parent lifecycle containment, references, and deliverable coverage
Local narrative material	LLM generation conditioned on authoritative stories	Model-based review of factual support, outcome timing, and scope alignment
Episode sequences	LLM generation with adjacent-subproject and relationship context	Structural checks on dates and references; model-based review of continuity and evidence and outcome boundaries
Occurrence sequences	Programmatic expansion of episode actions into dated events and linked states	Date-window and participant-validity checks; retention of source references

Occurrences preserve links to source episodes and preceding developments. The retained output includes 34 lineage-only episode sequences, counted in Table A5, that document downstream references to completed work. Their activity-type field distinguishes these sequences from active project execution.

A.2.4 Worked Example: Shared Event, Local Execution, and Handoff

The January 12 portfolio reset (`evt-001`) connects nine subprojects across nine parent projects. It makes AI quality taxonomy a mandatory dependency for later launches, protects reliability work, and gates commercial launch on operational readiness. Each affected project records its own consequences of this shared event. Table A8 follows the event's effect on *AI Quality Taxonomy and Feedback Instrumentation*. The reset raises the priority of the existing evaluation baseline within its approved scope. A local episode translates this requirement into four dated occurrences.

The event effect becomes available on January 12. The deliverables retain draft status through this episode, with finalization recorded on February 15. Grounding-completeness and context-citation categories enter the narrative through the separate April 7 event. A handoff connects this subproject to `proj-001-sub-02`. The receiving subproject can consume interim outputs during overlapping execution; the producer's recorded finalization establishes when those outputs can be treated as complete. The trace connects a shared company development to local decisions and a downstream handoff, retaining the dates and information states at each level.

Table A8: A retained trace from company-level change to local occurrences. Descriptions are condensed; occurrence dates are in January 2026.

Level or date	Record	Action and resulting state
Company event	`evt-001`, January 12	Quality taxonomy becomes a mandatory measurement dependency for later AI launches.
Parent-project story	`proj-001`	The evaluation and telemetry objective incorporates the new dependency requirement.
Subproject story	`proj-001-sub-01`, January 1-February 15	The evaluation contract and evidence baseline develop through explicit dependency and scope boundaries.
Episode	`project-episode-seq-proj-001-sub-01-04`, January 12-24	Baseline work is reprioritized, with mandatory needs elevated and optional breadth bounded.
January 12	`occ-01`	Ingrid Jensen, Allison Stephens, and Finn Smith apply the reset; the baseline enters priority review.
January 13	`occ-02`	Finn Smith separates mandatory needs from optional breadth; the candidate priority order is revised.
January 17	`occ-03`	Allison Stephens checks ownership, dependencies, exceptions, and rollback implications; priority changes acquire explicit boundary references.
January 24	`occ-04`	Ingrid Jensen retains evidence-first sequencing and exclusions; the draft rationale is ready for subsequent evidence checks.

A.3 Operational Planning and Expansion

A.3.1 Operational Taxonomy

Operational planning complements the project portfolio with recurring work. Its taxonomy separates business responsibility from the situations that arise within it. It contains 23 programs, 66 streams, 566 story families, and 9,000 scenario archetypes. Programs and streams establish operational ownership; story families specify recurring processes and their decision points; archetypes define the conditions under which these processes unfold, together with role requirements and possible outcomes. For example, *Offer, Entitlement, and Monetization Operations* contains the *Product Usage and Billing Reconciliation* stream. Its *Usage-to-billing discrepancy investigation* family includes the archetype *Customer-raised invoice usage challenge*. This archetype begins with conflicting customer and internal evidence and specifies four possible outcomes: a difference in counting boundaries, a confirmed billing discrepancy, a concern cleared through supporting evidence, or an unresolved evidence conflict. Table A9 reports the distribution across nine scenario categories. These categories describe the situations being generated. Episode-level fields separately record links to existing projects and whether an operational outcome produces a future-project signal.

Table A9: Scenario categories, retained episode counts, and category multipliers used for episode allocation.

Scenario category	Archetypes	Episodes	Multiplier
Routine cycle	477	3,761	1.18
Scheduled activity	745	5,189	1.10
Service request	624	4,239	1.08
External coordination	1,242	7,303	1.02
Policy usage	1,113	6,270	1.00
Exception investigation	3,588	17,671	0.98
Personnel continuity	654	3,453	0.92
Project handoff	186	822	0.66
Future-project signal	371	1,295	0.70
Total	9,000	50,003	

A.3.2 Episode Allocation and Joint Generation

Recurring processes require different levels of repetition across the company timeline. We therefore distribute a budget of 50,000 episodes using weights for recurrence mode and scenario category. Each recurrence weight is multiplied by the category multiplier in Table A9, with a small deterministic identifier-based adjustment. Every archetype receives one episode, and the remaining budget is allocated proportionally using largest-remainder rounding. The resulting allocation assigns three to ten episodes per archetype. The retained output contains 50,003 episodes, including three additional episodes recorded in the generation audit.

Recurrence mode	Weight	Recurrence mode	Weight
Continuous	8.0	Demand-driven	5.8
Cadence-based	7.3	Incident-triggered	5.0
Calendar-based	7.0	Seasonal	4.4
Mixed activation	6.0	Threshold-triggered	3.5

Episodes from the same archetype are generated jointly so that their evidence, decisions, and outcomes can be differentiated within a shared generation context. The archetype supplies the recurring business condition and available outcome branches. Each episode then specifies a concrete operating object, triggering evidence, causal developments, and an ending state. The generation prompt requires an explicit description of how each episode differs from its siblings in its operating object, trigger, complication, decision, or outcome. Episodes selecting the same outcome branch must also have distinct causal developments. For example, two invoice challenges may both establish a counting mismatch, with one arising from reporting periods and the other from units of measurement.

Company context constrains these realizations through month-level windows, role requirements, external organizations, and relevant company events. Project-linked episodes additionally receive established project facts and handoff conditions. They develop the subsequent operational response, including acceptance, conditional acceptance, residual-issue handling, or escalation. Future-project signals are retained when supported by the episode's selected outcome. Participant assignment and dated occurrence construction follow the shared mechanisms in Appendix A.2.3. The operational episodes expand into 235,016 occurrences, which provide the activity-level inputs to record generation in Section 3.4.

A.3.3 Worked Example: Recurring Invoice Challenges

The *Customer-raised invoice usage challenge* archetype generates six episodes across the 18-month horizon. Their shared process brings together Customer Success, Revenue Operations, Revenue Accounting, Product, and customer-side analytics or revenue-operations roles. Table A10 presents four episodes whose investigations lead to different determinations and follow-up actions.

These episodes instantiate the same process with different evidence conditions and outcomes. Recurrence supplies shared terminology and participants, while episode-specific facts determine which resolution applies. The CloudCore episode is followed through artifact generation in Appendix A.4.

A.4 Artifact Planning and Source-Specific Generation

A.4.1 Source Inventory and Artifact Plans

The source inventory covers communication, collaborative work, customer operations, engineering, analytics, and internal administration. Table A11 lists representative artifact types for each source. Source-level file and token statistics are reported in main-text Figure 3, with the counting protocol and detailed distributions in Appendix C. The planning inventory contains 735,360 Artifact Plans. Export packaging follows source conventions: email threads produce individual message files, document exports may include separate comment files, and Slack messages are consolidated into channel-day files.

An Artifact Plan specifies the content to realize for a record at a particular stage of activity. The planner receives episode context, ordered occurrences, participant information, and a catalog of source-specific record types. It selects records according to the actions and information exchanges in the episode, then specifies their content, ownership, and continuity. Table A12 summarizes the fields retained in the finalized plans.

Lifecycle handling follows the selected record type. Standalone records capture a bounded event; threads accumulate messages; mutable records carry field and status changes; versioned documents

Table A10: Four episodes generated from the same invoice-challenge archetype. Descriptions are condensed from the generated narratives; windows indicate month-level planning dates.

Episode and window	Evidence and investigation	Decision and follow-up
CloudCore Apps Feb–Mar 2026	Customer reports use regional local-day boundaries; internal billing uses a coordinated period boundary. Comparing transactions at the period edges reconciles the quantities.	Classify the difference as a counting-boundary mismatch, retain billing treatment, and record the boundary difference for subsequent reviews.
MarketFlow May–Jun 2026	Customer evidence separates production and test storefronts. Internal billing includes technically active test identifiers that are commercially excluded.	Confirm a billing discrepancy, exclude test activity from billing treatment, and review related identifiers.
Harborlight Aug–Sep 2026	The customer's shared analytics view omits privacy-restricted operating units. Approved aggregate evidence corroborates the disputed billable activity.	Validate billed usage and establish aggregate comparisons for subsequent reviews involving restricted units.
EmberCard Nov 2026–Jan 2027	Customer evidence is sampled, while internal evidence lacks complete transaction-level lineage for part of the disputed population. Investigation leaves that segment unresolved.	Retain interim commercial containment for the disputed segment and escalate the unresolved exposure.

Source archetype: `ops-scenario-ops-029-sf-02-004`.

Table A11: Sources and representative artifact types.

Source	Representative artifact types
Slack	Channel-day exports containing threads and announcements
Email	Individual messages and messages belonging to threads
Google Workspace	Documents, spreadsheets, presentations, calendar events, forms
Confluence	Knowledge pages and associated export components
Jira	Issues, service requests, incidents, change requests, assets
GitHub	Issues, pull requests, commits, deployments, workflow runs
Salesforce	Accounts, cases, opportunities, tasks, forecasts
Clearpath application	Support tickets, conversations, queues, automation settings
Datadog	Monitors, incidents, dashboards, log-search results
Amplitude	Charts, cohorts, dashboards, funnels, analytical exports
HRIS	Employee profiles, employment events, leave and workforce cases
IAM	Identity accounts, access requests, entitlement changes, access reviews

retain successive revisions. Continuity links connect successive realizations of an existing record, while records serving distinct purposes receive separate plans. For example, a calendar invitation and its meeting notes share an activity context but have different content requirements: the invitation specifies preparation and agenda, and the notes record the discussion and resulting decisions.

A.4.2 Shared Facts and Source-Specific Generation

Shared business details are represented in an episode-level fact ledger. Each observation includes a fact key, subject, predicate, value, originating occurrence, establishing Artifact Plan, and contributing person. Revision references connect observations that supersede earlier values. The ledger provides common inputs for records describing the same business matter, including measurement windows, affected populations, customer identities, and decisions. Artifact Plan dates and temporal generation instructions determine how these facts are expressed at each stage, including whether an action is proposed, under review, or completed.

Source-specific requirements define both content organization and structured fields. Email generation produces recipient-directed exchanges; Slack generation produces conversational turns; docu-

Table A12: Principal fields in the finalized Artifact Plan representation.

Component	Retained fields	Function
Record identity and type	`file_plan_id`, `source_role`, `title`, `path`	Identify the planned output and select its source-specific generation requirements.
Activity grounding	`episode_ids`, `occurrence_ids`, `created_from`	Connect the record to the activities that produce its content.
Participants	`creator`, `participants`, `people`	Identify authors and collaborators, with names, titles, and episode roles.
Business content	`background`, `content`	Supply business context, concrete facts, intended content, and structural requirements.
Shared details	`canonical_details` with `key`, `text`, and `available_from`	Identify reusable facts and the occurrences from which they become available.
Time	`created_at`, `last_modified_at`, `date_precision`	Place the record and its updates within the activity timeline.
Continuity	`previous`, `previous_version_file_plan_id`	Link a later realization to its preceding plan.

ments contain sections and paragraphs; spreadsheets contain typed rows and columns; and operational records populate source-specific fields. The requirements specify native status values separately from the business interpretations expressed in descriptions and resolution text. Person-level profiles condition expression within these source conventions. The profile catalog contains 13 base styles, supplemented by individual settings for message length, punctuation, greetings, signoffs, and other writing habits. Communication-generation settings further vary conversational length and interaction patterns, including acknowledgements and occasional casual exchanges. Programmatic renderers combine generated content with identifiers, timestamps, metadata, and source-specific structures. The published source tree contains the resulting records; a separate audit tree retains fact ledgers and mappings from Artifact Plans to exported files.

A.4.3 Worked Example: From Invoice Challenge to Cross-System Evidence

We follow the CloudCore invoice-challenge episode introduced in Appendix A.3.3. Its five occurrences span February 8 to March 24, 2026, and produce 15 Artifact Plans: five spreadsheets, four Slack threads, two documents, two Salesforce case records, one calendar event, and one email thread. Table A13 traces selected stages into their generated records.

The initial comparison spreadsheet illustrates how an Artifact Plan specifies concrete evidence structure. Its columns include API event reference, region, event timestamp, governed period, regional local-day period, movement direction, and trace status. The review document interprets these comparisons, and the later email exchange communicates the resulting commercial treatment. The four-message email thread is exported as four message files, while the Slack discussions appear within their corresponding channel-day exports. This stage turns the shared activity history into source-specific evidence while preserving dates and business states. QA grounding subsequently checks which generated passages actually support each requested claim (Appendix B.2).

Table A13: Selected stages of the CloudCore invoice challenge and their corresponding records. Dates follow the associated Artifact Plans; descriptions summarize the generated content.

Date	Records	Evidence and business function
Feb 8, 2026	Salesforce intake record, Slack discussion, comparison spreadsheet	The case records CloudCore's lower reported API usage. The discussion coordinates a comparison of reporting boundaries. The spreadsheet places event references, regions, timestamps, and the two period assignments side by side.
Feb 19, 2026	Calendar event, review-notes document, Slack discussion	The calendar specifies the evidence-review agenda. The notes record that European and North American period-edge events shift into adjacent reporting periods while remaining traceable.
Mar 13, 2026	Classification document, Slack discussion, future-review guidance spreadsheet	The decision document classifies the discrepancy as a valid boundary mismatch and retains the existing billing treatment. Related records communicate the determination and establish the comparison approach for subsequent invoice reviews.
Mar 24, 2026	Internal email thread, Salesforce resolution record, trace appendix	The email exchange reports delivery of the reconciliation, confirms that the interim review hold is lifted, and agrees on future comparison practice. The case's resolution text records the outcome, while the appendix retains transaction-level tracing information.

Source episode: `ops-episode-ops-029-sf-02-004-01`.

B QA Construction Details

This appendix details how planning materials support question design, how claims are grounded in exported files, and how question-conditioned augmentation enriches the surrounding business context. Checks are described with the stage they validate; cross-cutting refinement and model configurations appear in Appendix B.4. All statistics use the same 966-question snapshot and its associated evidence mappings. The billing-release examples use fictional identifiers and participants to explain the construction mechanisms; they are not benchmark items and do not contribute to any reported statistic.

B.1 Question Design and Distributions

Relational navigation. Multi-hop questions evaluate whether an intermediate finding can identify the evidence needed next. Construction begins with a factual path through the planning hierarchy, including company events, project and operational planning records, Artifact Plans, and shared facts. From a starting record, an LLM extracts a relation between two concrete entities or events. Its output becomes the search anchor for a candidate record, where the model seeks a further relation involving the same referent. A separate review checks the connection, including identity, role, time, and business scope. Accepted steps retain their source passages and become reasoning claims along the path; rejected candidates leave the current anchor unchanged.

The generator then formulates a question around the accepted path, identifying the starting situation and asking for its endpoint while leaving intermediate identifiers to be recovered. The recorded facts and their order remain fixed during this step. Claim-removal tests subsequently check whether parts of the path can be bypassed (Appendix B.4). For example, asking who should roll back the release associated with ticket CS-1842 requires identifying the release before finding its rollback owner. The requested answer is the owner; the release identifier remains an intermediate result.

Scoped synthesis. Synthesis questions evaluate whether a system can assemble the requested information across a defined set of business activities. We establish a scope from the planning hierarchy, such as a project, an operational episode, or the projects affected by a company event, and inventory its planning records, Artifact Plans, and shared facts. Within this scope, the generator selects one information need, such as handoff responsibilities, and constructs a question and reference answer covering the relevant members. Each answer item is associated with reasoning claims and

cited planning passages. Dates, roles, and exceptions are retained when they affect the requested information.

Support and coverage are reviewed separately. Support review checks each claim against its cited planning material. Coverage review checks the scope inventory for relevant members or details missing from the answer, including omissions revealed by shared facts. The review distinguishes facts explicitly assigned to an Artifact Plan from contextual facts used to identify gaps. For large inputs, programmatic relevance screening selects passages for review while retaining the declared member inventory and a separate shared-fact list. Scope can be narrowed to an explicit constituent business scope before question generation. Table A14 asks for the rollback owners of all billing releases published in February 2027. A release register establishes the members of that scope, while release plans provide their owner assignments. The answer includes v2.3–Maya and v2.4–Sam; v2.2–Alex belongs to January and is excluded. The register supplies the membership boundary against which answer coverage is checked.

Table A14: Illustrative synthesis question in the same billing-release setting. Release membership is established by an accompanying release register, and owner assignments by release plans. These fictional entries are explanatory, not benchmark measurements.

Question: Who is responsible for rolling back each billing release published in February 2027?

Release	Release month	Rollback owner	Scope membership
v2.3	February 2027	Maya	Included
v2.4	February 2027	Sam	Included
v2.2	January 2027	Alex	Excluded

Reference answer: v2.3—Maya; v2.4—Sam.
Answer subfacts: Maya is the rollback owner for v2.3; Sam is the rollback owner for v2.4.

Reference-answer annotation. Each reference answer is decomposed into answer subfacts for completeness scoring. An LLM receives the question and reference answer and proposes nonredundant factual units, each linked to exact spans in the answer. The question resolves referents in short answers. A separate review checks that the units cover the answer and preserve conditions, scope, negation, and modality, then maps them to directly supporting reasoning claims. Each subfact has unit weight. Thus, the multi-hop answer "Maya" contributes one identity subfact, whereas the synthesis answer in Table A14 contributes two owner assignments. The snapshot contains 3,366 reasoning claims and 5,315 answer subfacts.

Final composition. The benchmark contains 804 multi-hop questions, 143 synthesis questions, and 19 single-file questions (Figure A3a). Single-file questions have recorded whole-question support within one file. Among multi-hop questions, 307 (38.2%) have a shortest recorded sufficient route containing at least four claims (Figure A3b). We measure route length by the number of claims in that route. These counts describe the retained evidence structure; answer lengths and answer subfacts are measured separately in Appendix B.4.

B.2 Evidence Grounding and Support Sets

From planning facts to published evidence. Evidence grounding serves two purposes: realizing the support specified by planning materials and identifying additional files that express the same claims. We first trace each claim to its relevant activity, Artifact Plans, and generated artifacts. Verification checks the full assertion, including the participants, relationship, time, scope, status, and qualifying conditions. Supporting passages are stored with file identifiers and locations within the native artifact. When content required by an Artifact Plan is absent or incompletely expressed in its generated artifact, a localized revision or supplementary artifact realizes that planned content. A separate review checks factual support, preservation of existing content, and compatibility with the source format.

Additional support within business scopes. To expand support beyond the initially traced files, planning provenance identifies the relevant project or operational stream and its exported files. Occurrence-level searches can also expand to the parent episode. Programmatic screening uses entity names, dates, distinctive phrases, and terms from the planning evidence to select candidate passages. LLM-based verification then checks the complete claim against those passages, distinguishing full, partial, absent, and contradictory support. Verified quotations retain their locations

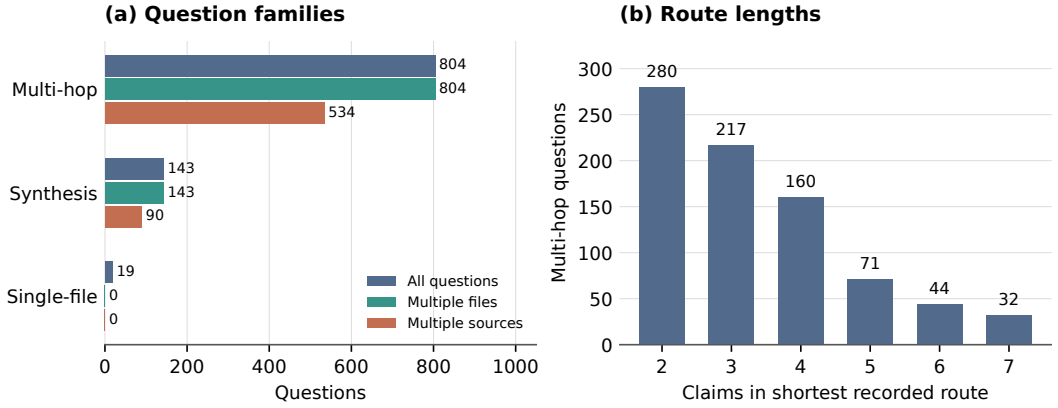


Figure A3: Question-family composition and shortest recorded claim routes. Multi-file and multi-source counts refer to annotated sufficient evidence; the route-length distribution covers the 804 multi-hop questions. Bar labels give exact question counts.

in the native content. Full-support files become alternative support options; files providing complementary parts are checked together for joint support.

Answerability in the generated artifacts. File-based checks also test whether the available content answers the whole question and whether an individual file already suffices. These checks can reveal shorter routes than those suggested by the original planning path. Questions with verified whole-question support in one file are retained as single-file questions. Their support is recorded alongside claim-based routes.

From claims to sufficient evidence sets. A reasoning route specifies the claims that together establish an answer. Each claim has one or more support options, consisting of a single file or jointly supporting files. Selecting one option for each claim and taking the union of its files produces a sufficient evidence set; repeated files are counted once. Alternative routes and support options yield alternative sets. Table A15 illustrates the complete annotation, including the distinction between intermediate reasoning claims and answer subfacts.

Table A15: Illustrative multi-hop question, supporting files, and annotations. The fictional example is not a benchmark item and is excluded from reported statistics. F1–F3 represent existing support; F4 is a question-conditioned addition.

Question	Who is responsible for rolling back the billing release associated with ticket CS-1842?
Reference answer	Maya.
File / source system	**Relevant content**
F1: Salesforce ticket	Duplicate charges reported in CS-1842 were caused by billing release v2.3.
F2: Google Workspace release plan	Maya is responsible for rolling back v2.3.
F3: Slack handoff discussion	If v2.3 requires a rollback, Maya will execute it.
F4: Added Google Workspace release plan	Alex is responsible for rolling back v2.2.
Annotation	**Content**
Reasoning claim 1	Ticket CS-1842 concerns billing release v2.3.
Reasoning claim 2	Maya is responsible for rolling back v2.3.
Reasoning route	Identify the release from the ticket, then identify its rollback owner.
Claim support options	Claim 1: {F1}. Claim 2: {F2} or {F3}.
Sufficient evidence sets	{F1, F2} or {F1, F3}.
Answer subfact	The requested rollback owner is Maya.

Alternative support in the example. In Table A15, F1 identifies the release, while either F2 or F3 establishes its rollback owner. The two sufficient evidence sets are therefore {F1, F2} and {F1, F3}: retrieving F1 with either owner assignment completes the recorded route. F2 and F3 are alternative

support options, not cumulative requirements. The final answer contains one identity subfact, even though deriving it uses two reasoning claims.

Counting files and source systems. We compute the smallest file count and smallest source-system count separately over the annotated sufficient evidence sets, including direct whole-question support. The two minima can therefore correspond to different sets. Google Workspace documents and spreadsheets, for example, count as different files within one source system. Across questions, the mean minima are 3.45 files and 1.84 systems. Figure A4 gives the joint distribution: 947 questions require at least two files and 624 at least two systems within the recorded mappings. Figure A5 reports which systems occur among the available support options. Google Workspace appears in 919 questions (95.1%), followed by Slack in 400 and Jira in 282. These distributions characterize the evidence used by questions within the 12-system corpus. Retrieval metrics use the same annotated support sets (Appendix D.2).

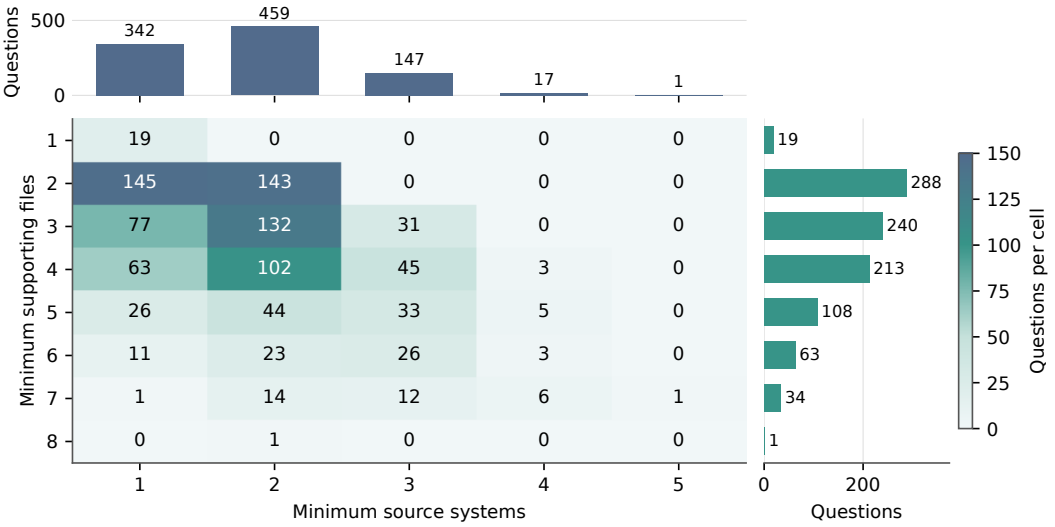


Figure A4: Joint distribution of minimum file and source-system counts over recorded sufficient evidence combinations. Each cell counts questions. Both minima are computed independently; a cell need not identify a single combination achieving both. Marginal bars retain all 966 questions. Cell labels include zeros and the shared color scale gives question counts.

B.3 Question-Conditioned Corpus Augmentation

Local enrichment around a question. Recurring company activities produce artifacts with similar content but different applicable events, participants, and time periods. Reproducing the full extent of this repetition and overlap is difficult even in a large simulation. Question-conditioned corpus augmentation adds local detail around the business situations selected for evaluation. Starting from each question's reasoning claims, we synthesize competing claims describing comparable activities with different applicable objects or conditions. For a multi-hop claim, construction distinguishes information already supplied by the question, the identifying value obtained from an earlier claim, and the output sought at the current step. A competing artifact can preserve the publicly specified business conditions while changing the intermediate identifying value and its associated output. For a first-step claim or a synthesis question, related records can instead fall outside an explicitly requested event, period, or membership boundary. These distinctions guide the design of the new synthetic claims and their separation from the requested evidence.

Competing claims in the worked example. F4 in Table A15 expresses a newly synthesized competing claim: Alex is responsible for rolling back v2.2. This claim is constructed from the target claim's relation and identifying details. It preserves the billing-rollback topic while changing the version and its owner. F2 and F4 consequently provide similar candidate assignments, and the version recovered from F1 determines which applies to CS-1842. For the synthesis question in Table A14, a plan for the January release v2.2 instead provides an out-of-scope candidate. These examples separate selection using an intermediate result from selection using the question's explicit scope.

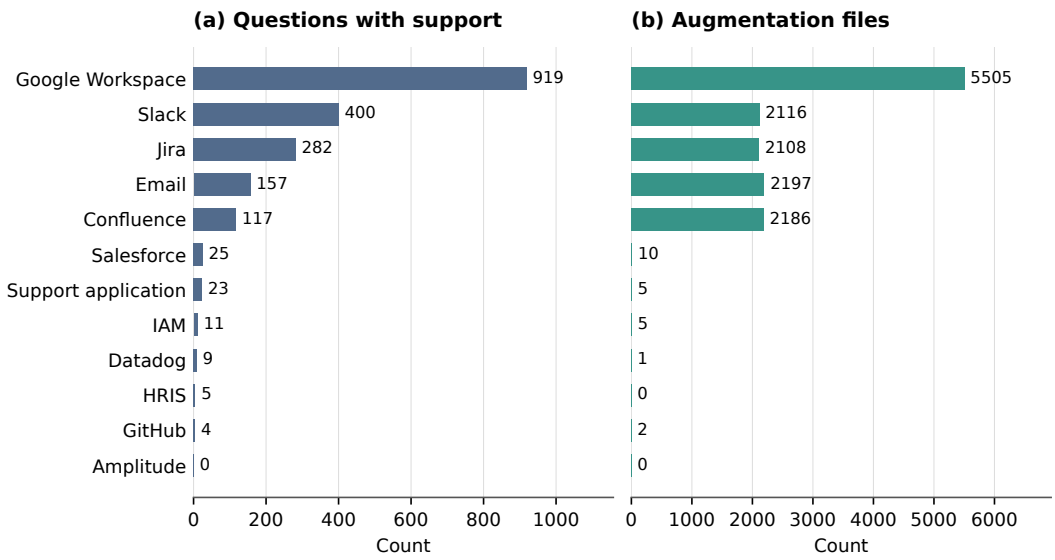


Figure A5: Support coverage and augmentation by source system. A question may have support in several systems, so support counts can overlap; source membership does not imply necessity in every sufficient route. Augmentation counts exported files. Bar labels preserve exact counts.

Claim design and review. The generator proposes multiple competing claims for a target relationship, specifying the shared retrieval cues, the distinguishing condition, and why each alternative does not answer the question. Review checks both meaningful competition and compatibility with the reference answer: a proposal should neither introduce another valid target nor add an omitted member to a synthesis scope. It also checks whether proposed content supports any existing claim of the target question. To identify other potentially affected questions, lexical screening retrieves claims sharing distinctive terms and phrases, followed by semantic review of the selected candidates. Designs with substantive conflicts are revised or excluded before artifact generation.

Artifact generation and verification. Accepted designs become business-content briefs rendered through source-specific generation interfaces. The resulting artifacts express the competing claims through ordinary business content, with appropriate identities, roles, metadata, and source formats. A separate review inspects the generated content for the intended claim, answer-changing conflicts, whole-question answerability, and unintended support for target or other candidate question claims. Verification records exact quotations and their locations. Unintended support is preferentially removed through revision while preserving the intended competing claim. If verified support remains in a retained artifact, it is added to the affected claim's support options. The insertion manifest records each file's path, content hash, source type, target question, and target claim.

Updating evidence after unintended support. Suppose the generated F4 also states that Maya owns the v2.3 rollback. It now supports reasoning claim 2, despite having been generated to express a competing assignment. We preferentially revise this additional content while preserving the intended v2.2–Alex claim. If the v2.3 statement is retained and verified, F4 becomes another support option for claim 2, and {F1, F4} is added to the sufficient evidence sets. The annotation thus follows the artifact's verified content, rather than its original intended role as a distractor.

Coverage. The finalized manifest contains 14,135 distinct augmentation files and covers 965 retained questions. Twenty files linked to the removed question remain in the corpus as background artifacts. The mean is 14.61 augmentation files per question (median 13, nearest-rank P90 25, range 0–40), including the one question with no added file. All 804 multi-hop and 143 synthesis questions receive local enrichment. Added artifacts span ten source systems and 21 source-specific artifact types; source-level counts appear in Figure A5. The counting unit is the exported file: one discussion or activity can contribute several files. By family, multi-hop questions have a mean of 14.50 files (median 13, P90 25, range 1–40), and synthesis questions have a mean of 16.29 (median 16, P90 25, range 4–36). The single-file family has coverage of 18/19 and a mean of 6.58 (median 5, P90 14, range 0–16).

B.4 Validation and Refinement

Annotation consistency. The stage-specific checks above use separate generation and review calls. Review assesses the supplied evidence, with attention to identities, roles, dates, scope, and business states. For example, a proposed assignment and a completed handoff require different answer wording. Programmatic checks verify identifiers, exact quoted spans, evidence locations, and references between claims, support options, and answer subfacts. Human assessment and subsequent annotation revisions are reported in Appendix C.3.

Evidence removal and shorter routes. Claim-removal tests examine which relationships and answer details remain recoverable when a claim is withheld. A solver receives the question and the remaining claims, without the reference answer or omitted content. A separate reviewer checks whether its response follows from those inputs, including the relationship identifying the requested person, project, or event. Merely seeing an answer about one candidate does not establish that it is the candidate selected by the question. Removable claims and alternative routes are inspected together before the annotation is revised, since two individually removable claims may provide interchangeable support. Disputed cases receive a further model review of the surviving relations and requested answer. The resulting routes are then linked to verified file support as described in Appendix B.2.

Construction models. Table A16 reports the GPT-5.6 models (OpenAI, 2026) and reasoning-effort settings used for the principal construction and review stages. Generation and review are separate calls, including when they use the same model. Targeted follow-up uses GPT-5.6-terra (medium) for disputed claim-removal cases and GPT-5.6-sol (high) for answer-subfact cases requiring further revision. Augmentation repair also uses GPT-5.6-terra and GPT-5.6-sol with low reasoning effort to revise and recheck flagged competing artifacts. These construction settings are separate from the reader and judge settings used to evaluate benchmark systems (Appendix D.1).

Table A16: Model configurations for QA construction. Effort denotes the reasoning-effort setting. Rows distinguish operations within construction rather than evaluation baselines. Targeted follow-up configurations are described in the accompanying text.

Operation	Model	Effort
Multi-hop path exploration, step review, and question generation	GPT-5.6-terra	low
Synthesis question and answer generation	GPT-5.6-terra	low
Synthesis support and coverage review	GPT-5.6-luna	low
Candidate file–claim verification	GPT-5.6-luna	low
Ranked single-file answerability review	GPT-5.6-terra	medium
Planned-content integration and review	GPT-5.6-sol	medium
Claim-removal blind answering	GPT-5.6-luna	low
Claim-removal adjudication	GPT-5.6-luna	medium
Competing-claim design and review	GPT-5.6-luna	low
Augmentation generation and file–claim audit	GPT-5.6-luna	low
Reference-answer subfact decomposition and review	GPT-5.6-sol	medium
Question paraphrasing and semantic review	GPT-5.6-sol	medium

Answer coverage and length. Table A17 separates question length, reference-answer length, and the number of answer subfacts. Lengths use whitespace-separated words in the final English text; P90 is the nearest-rank 90th percentile. Multi-hop answers have a median of 27 words and three answer subfacts, whereas synthesis answers have medians of 116 words and 15 subfacts. This contrast reflects the broader answer coverage requested by synthesis questions. The multi-hop family also includes one-word answers, illustrating how a route through several facts can terminate in a single name, value, or status. Answer subfacts preserve this distinction by scoring the requested output without requiring the response to reproduce its intermediate route.

Table A17: Final question lengths, reference-answer lengths, and answer-subfact counts. Lengths are in whitespace-separated words, not model tokens. Statistics describe 804 multi-hop, 143 synthesis, and 19 single-file questions.

Measure	Question family	Min.	Median	Mean	P90	Max.
Question words	Multi-hop	18	62	66.47	104	169
	Synthesis	20	47	48.43	71	102
	Single-file	24	41	41.47	53	59
	All	18	58.5	63.31	100	169
Answer words	Multi-hop	1	27	33.95	60	222
	Synthesis	26	116	120.99	180	264
	Single-file	10	25	26.37	40	54
	All	1	32	46.69	111	264
Answer subfacts	Multi-hop	1	3	3.80	7	29
	Synthesis	3	15	15.31	26	37
	Single-file	1	4	3.47	6	7
	All	1	3	5.50	13	37

C Dataset Statistics and Quality Control

We describe the evaluation corpus and question composition, then distinguish programmatic inventory checks from human assessment of question and annotation quality. Corpus counts characterize the frozen evaluation inputs; question classifications support the subsequent performance breakdowns; human review identifies issues corrected before the reported evaluation.

C.1 Corpus Inventory and Integrity Checks

Snapshot and counting units. The evaluation snapshot contains 809,170 nonhidden regular files: 646,077 JSON files and 163,093 email files. The construction inventory of 794,347 files loses one historical path, retains 794,346, and adds 689 supporting files plus 14,135 augmentation-origin files. Among retained paths, 3,516 have revised content. This gives 795,035 base files and 14,135 augmentation-origin files. Seven of the latter are supporting evidence retained in the corpus ablations (Appendix D.8). These are physical exported-file counts, not logical artifact or message counts. Slack channel-day packaging, email-message exports, and separately exported document comments account for differing units. Episode, occurrence, and Artifact Plan counts refer to internal construction objects and are reported in their respective construction appendices.

Exact token-count protocol. We count every current source file once, excluding planning files, reference annotations, backups, and quarantine. Files are decoded as strict UTF-8 and encoded independently with `o200k_base` in `tiktoken` 0.14.0. Counts preserve complete exported contents, including formatting, keys, metadata, comments, and email headers. No paths, prompts, chat framing, or separators are added. Apparent special-token strings are encoded as ordinary text. The total is the sum of per-file token counts, not one concatenated request. Quantiles use exact nearest-rank empirical order statistics over files.

The full corpus contains 1,213,160,275 tokens in its exported serialization. Base files account for 1,184,384,691 tokens, and augmentation-origin files account for 28,775,584. Across the full corpus, mean file length is 1,499.27 tokens, P50 is 1,071, P90 is 3,005, P95 is 5,108, P99 is 7,163, and the maximum is 147,092. Table A18 provides the source-level breakdown.

The full-corpus and base-file length distributions nearly coincide: their maximum cumulative difference is 0.46 percentage points (Figure A6b). Augmentation contributes only 14,135 of 809,170 files, so its distinct length distribution changes the full-corpus curve little. File-length similarity should not be interpreted as similar retrieval difficulty; the scale experiments examine the effect of those question-conditioned additions separately.

Programmatic verification. The census hashes every file and checks size, modification time, and path membership before and after counting. It found no additions, removals, symlinks, or input-provenance changes during the census, and no mismatch against the frozen evaluation hashes or augmentation manifest. Independent aggregation reproduced exported-content totals and quantiles for the full corpus, base files, and augmentation-origin files. A seeded check of 100 files reread the source bytes and reproduced counts through an independent public tokenizer call. These checks

Table A18: Exact census of the evaluation corpus. Token counts preserve complete exported contents, including formatting and metadata. File lengths use the same counting protocol.

Source	Files	Tokens	P50	P90
Google Workspace	446,521	628,771,835	1,327	2,078
Email	163,093	49,180,294	264	467
Jira	72,397	333,170,413	4,719	6,781
Confluence	29,946	21,910,614	633	1,156
Slack	24,461	144,553,080	2,181	15,272
Salesforce	24,186	7,108,267	293	342
GitHub	14,507	16,038,354	791	2,043
Clearpath app	12,810	4,568,277	274	694
IAM	6,883	1,973,874	265	421
Datadog	6,779	3,580,969	669	728
HRIS	3,796	872,046	225	279
Amplitude	3,791	1,432,252	372	514

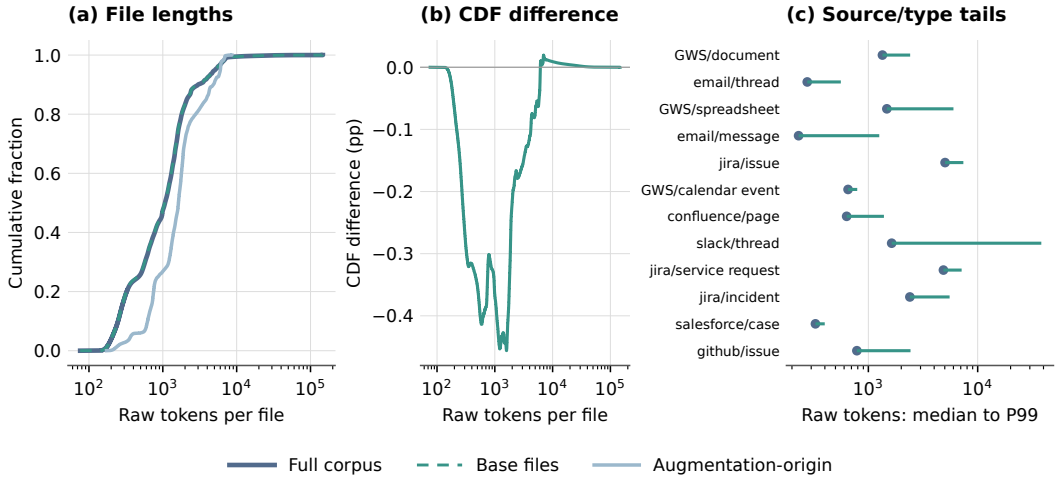


Figure A6: Measured file lengths. (a) Cumulative raw-token distributions: a solid blue curve for the full corpus, dashed green for base files, and light blue for augmentation-origin files. The shared legend sits below the panels. (b) Full minus base-file cumulative fractions in percentage points, resolving their near-overlap. (c) Median-to-P99 ranges for the 12 most frequent exclusive source/type buckets. Each file belongs to one exclusive bucket; mixed-role Slack files are not counted repeatedly. All horizontal axes are logarithmic.

establish inventory and measurement consistency, not the semantic correctness of every synthetic artifact.

C.2 Question Composition and Classification

Dimensions and classification provenance. We examine primary business-question type, overlapping business-content labels, connection types, business scope, reference-answer subfact count, and the source systems appearing in recorded support options. These dimensions describe question composition for the performance analyses in Appendix D.4. A primary type classifies the requested answer, whereas content labels can record several requested aspects. Connection types identify selectors passed between dependency steps, such as a person, project, event, or time. Business scope describes the span of the annotated answer route. Source membership means that at least one recorded support option uses that system; it does not make that system necessary in every valid route. A semantic synthesis-content tag is separate from the synthesis question family: a multi-hop question can request several facts after resolving its intermediate objects.

Semantic labels describe the requested business information, the connections used in a reasoning route, and the business scope. All inventories use the labels attached to the finalized 966-question dataset, and answer-subfact counts come from the same reference annotations.

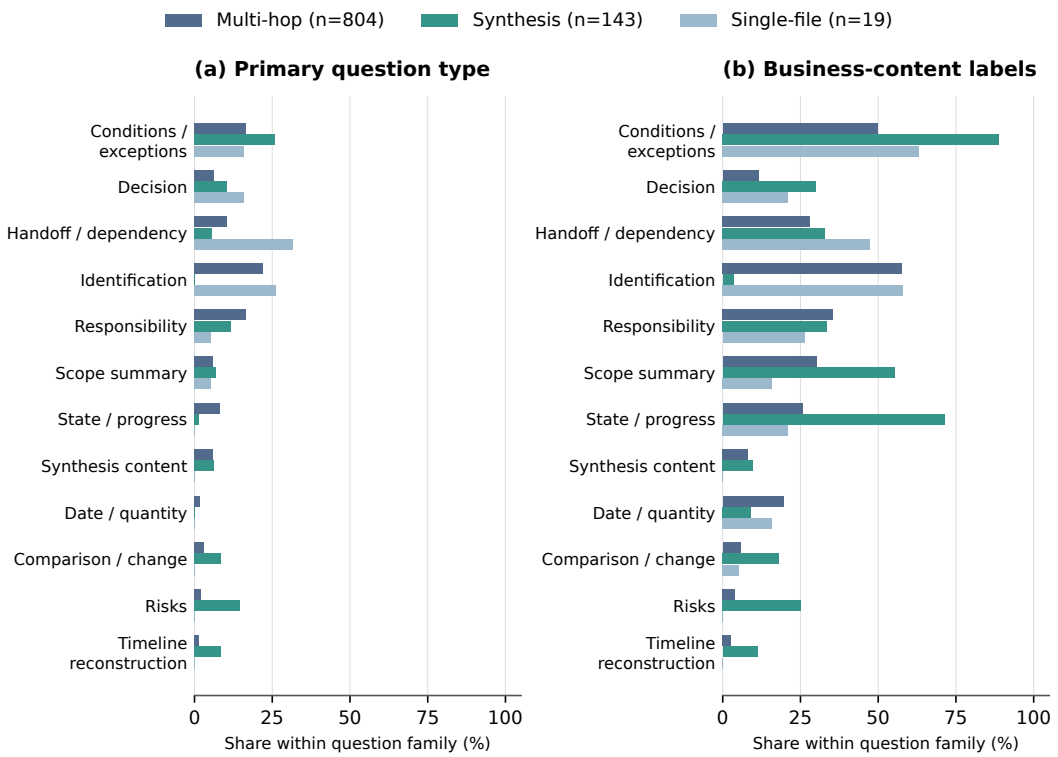


Figure A7: Question content by family. Bars show percentages within 804 multi-hop, 143 synthesis, and 19 single-file questions. Primary type is a single label; business-content labels overlap. Synthesis content is a semantic label distinct from the synthesis question family. All categories are retained; exact counts are in the taxonomy export.

Figures A7 and A8 show the classification inventory as within-family shares, retaining rare categories. Multi-hop questions concentrate on identification and business relationships, whereas synthesis questions more often request conditions, progress, and scope summaries. The latter can have no recorded connection tag because combining complementary facts does not necessarily pass a selector between dependency steps. These patterns explain differences in requested information; the bars measure composition, not model performance or isolated reasoning ability. Source coverage is shown separately in Figure A5. The accompanying result export retains all exact counts, per-question label memberships, and category-level system scores.

C.3 Human Evaluation and Annotation Revision

We assessed 100 questions for five properties needed for benchmark use: question clarity, reference-answer groundedness, evidence sufficiency, task-type appropriateness, and answer-subfact quality. Two reviewers conducted the assessment, and a third reviewer re-examined issues flagged by the two reviewers to confirm the reported assessments. Each dimension uses a three-point scale: 2 indicates that the item meets the criterion, 1 indicates a limited issue, and 0 indicates a substantive problem. Reviewers examined each question with its reference answer, subfacts, and associated evidence packet; this assesses annotation quality rather than a system's ability to retrieve the packet from the full corpus.

The confirmed pre-revision mean ratings range from 1.76 to 1.98, with 77–99 questions meeting each criterion fully (Figure A9). Reference groundedness has the most partial ratings (22), while answer-subfact quality has the most substantive deficiencies (6). Criteria can flag the same question, so these counts must not be summed into a number of distinct defective questions. The findings identify reference support and answer decomposition as particular revision targets, complementing the inventory and consistency checks above. All identified issues were corrected, and the reported experiments use the revised question and annotation version. The displayed ratings document the issues found during review; they do not constitute a new quality measurement after correction.

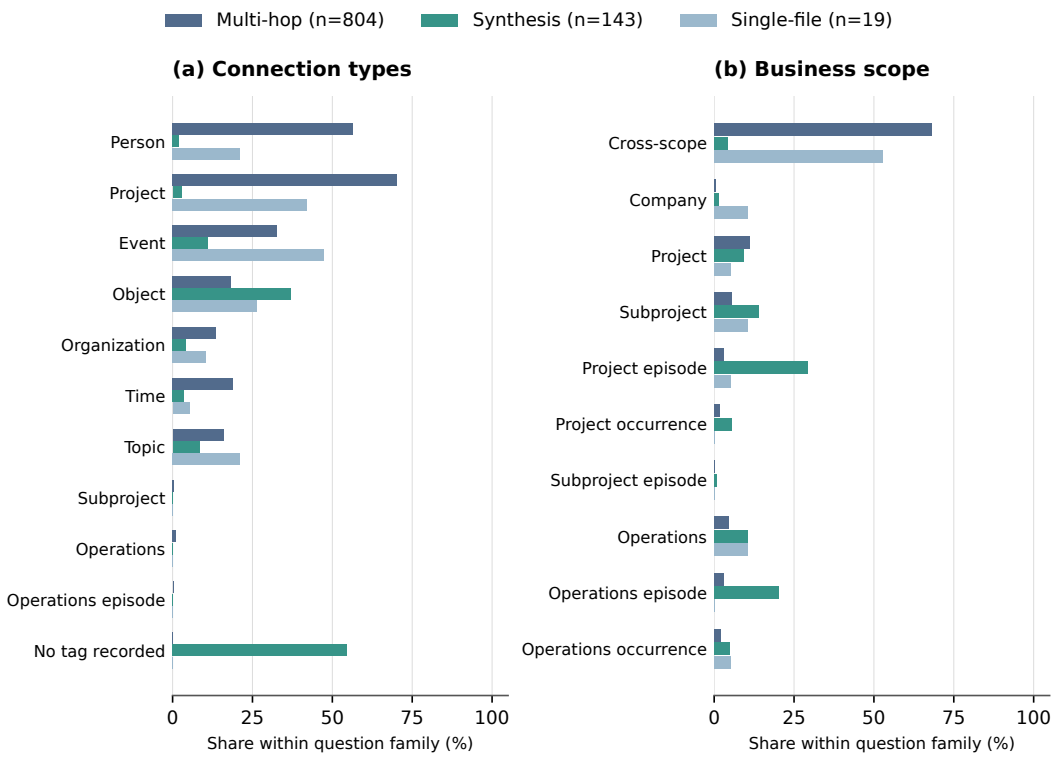


Figure A8: Connections and business scope by question family, using the same denominators and colors as Figure A7. Connection types may overlap; scope assigns one label. No tag recorded means an empty connection-label list. Project as a connection and as a scope are distinct labels.

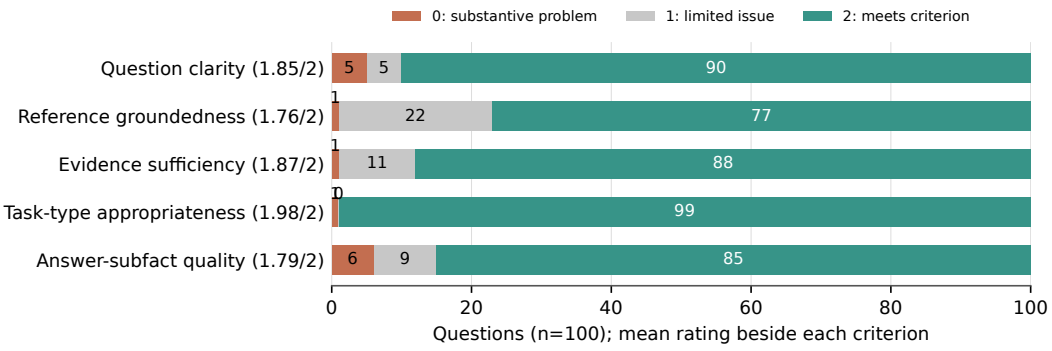


Figure A9: Human assessment of 100 questions before revision, with flagged issues confirmed through third-reviewer follow-up. Segments show rating counts; labels preserve every count and means appear beside the criteria. Ratings assess the supplied question and evidence packet. Identified issues were subsequently corrected; these are not residual-defect counts for the revised version used in the reported experiments.

D EXPERIMENTAL PROTOCOL AND ADDITIONAL ANALYSES

This appendix supplies the protocols and complete analyses behind the experimental narrative in Section 5: overall performance, question requirements and failure analysis, and corpus scale and local refinement, together with resource accounting and evaluation sensitivity. Main-system comparisons, question diagnostics, resource summaries, and all 52 scale conditions use the same 966 retained questions. Incomplete reader–judge results are reported separately and are not presented as full-benchmark point estimates. Per-question exports and offline aggregation scripts reproduce the tables, figures, and bootstrap intervals. We first define systems and metrics, then assess paired comparisons and question-dependent performance. Evidence diagnostics and cases connect these patterns to observed system behavior; resource and scale analyses characterize the evaluated operating conditions. Reader–judge sensitivity and full scoring prompts follow these results.

D.1 SYSTEMS, ACCESS, AND REPRODUCIBILITY

Evaluation data. All reported system and scale experiments use the same finalized 966-question dataset and its associated reference answers and evidence annotations. The full evaluation corpus contains 809,170 exported files. Internal construction plans, reference answers, and annotations remain evaluator-only inputs.

Retrieval pipelines. BM25 (Robertson & Zaragoza, 2009) uses SQLite FTS5 lexical retrieval. Dense uses normalized 2,560-dimensional Qwen3-Embedding-4B (Zhang et al., 2025) vectors and exact inner-product search over chunks, with each file ranked by its maximum chunk score. Index inputs retain the relative path and source fields; chunks use an 8,192-token maximum and 256-token overlap. The encoder revision is `5cf2132abc99cad020ac570b19d031efec650f2b`. Hybrid uses reciprocal rank fusion (Cormack et al., 2009) to combine the BM25 and Dense Top100 lists with equal weights and rank constant 60. Hybrid + Reranker instead scores the union of up to 200 candidates with Qwen3-Reranker-4B (Zhang et al., 2025). Each pipeline supplies up to 20 files to GPT-5.6-sol (OpenAI, 2026) under a 32,768-token evidence budget. The retained reader inputs specify exactly which content was visible and which files were truncated.

Bash and end-to-end agents. The Bash Agent adapts the shell-based retrieval approach of EnterpriseRAG-Bench (Sun et al., 2026). GPT-5.6-terra searches and reads with a 600-second retrieval budget, then selects up to 20 files for the separate GPT-5.6-sol reader. The end-to-end GPT-5.6-sol agent instead has 300 seconds for exploration, including tools and compaction, followed by at most 60 seconds for final reasoning and answer submission without further retrieval. It uses official context compaction around 25,000 estimated input tokens and skips compaction during the final 60 seconds of exploration. Claude Sonnet 5 (Anthropic, 2026) uses the same exploration and final-answer budgets with its provider-default effort. Both end-to-end agents must submit at most 20 observed evidence paths. All GPT generation and benchmark judging use medium reasoning effort.

Runtime. The server has an AMD EPYC 7413 processor (24 cores, 48 threads) and four NVIDIA RTX A6000 GPUs (48 GB each). The frozen dense manifest records PyTorch 2.5.1, Transformers 4.57.3, FAISS 1.14.3, NumPy 1.26.4, normalized exact search, and maximum-chunk-score document aggregation. This describes the host and software, not a claim that every stage occupied all GPUs. LLM inference is remote API inference; local host hardware does not specify serving hardware. Server Python is 3.12.2 and server token accounting uses `o200k_base`; the separate corpus census uses the explicitly versioned protocol in Appendix C.

Gold-evidence reference. The gold-evidence reference covers 2,018 recorded sufficient file sets across 966 questions. Each set supplies full source documents to the same reader, without reference answers, claims, or route labels. Selection maximizes Combined Score, then correctness, completeness, and a stable route identifier. All reported metrics come from the same selected answer. There are 254 questions with multiple sets, and 99 route inputs exceed the standard evidence budget.

D.2 ANSWER AND EVIDENCE METRICS

Answer judgments. Overall correctness is binary; completeness measures coverage of the reference answer's subfacts. The judge evaluates the question, reference answer, candidate answer, and required subfacts without system identity, returning correctness and subfact-coverage judgments. Each subfact has equal weight within its question; questions have equal weight in all dataset averages. Supporting claims are intermediate evidence annotations, not mandatory answer content. The

final correctness prompt adapts the semantic alignment criteria of EnterpriseRAG-Bench (Sun et al., 2026). The full prompts and JSON output fields from the frozen evaluator appear in Appendix D.10.

Combined Score. For each question we multiply binary correctness by subfact completeness, then average the products. Thus an overall-correct answer covering three of four subfacts receives 0.75; an overall-incorrect answer receives zero combined credit. This is not the product of the two dataset-level averages.

Alternative sufficient evidence. Recall@20 is the maximum recovered fraction across recorded sufficient file sets, using the final 20 distinct selected paths. Sets are obtained from annotated claim routes and support options, including accepted direct whole-question file routes. These annotations do not exhaust all valid evidence in the corpus. All system and scale comparisons use the same finalized evidence annotations and route-aware evaluator. Minimum file and source counts are computed independently over the recorded sufficient sets and may therefore be attained by different sets.

Selected versus visible evidence. Recall@20 uses selected file paths. Selecting a supporting file need not expose every relevant passage under a reader's evidence budget or an agent's bounded reads. Agent diagnostics additionally score all observed paths and all explicitly read paths; these sets are not constrained to 20 files.

Completion and uncertainty. All seven reported full-corpus systems and the gold reference have 966 scored questions. An agent that does not submit a valid answer within the 60-second final-answer window receives zero answer credit. API infrastructure failures without valid outputs are treated as missing evaluations rather than incorrect answers. All 52 scheduled scale conditions are complete: 39 retrieval-pipeline conditions and 13 agent conditions, totaling 50,232 scored question–condition pairs.

D.3 Paired System and Corpus Comparisons

We quantify uncertainty in selected full-corpus system differences and in the score changes after removing augmentation. Figure A10 uses 10,000 paired bootstrap resamples of question IDs, seed 20260925, and percentile 95% intervals. These intervals describe question sampling variation, not corpus-seed or repeated-generation variation.

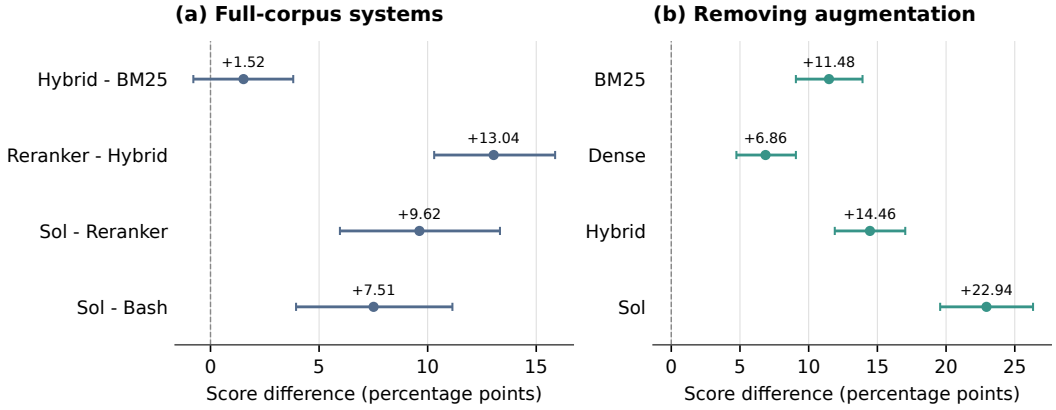


Figure A10: Paired differences in Combined Score with 95% percentile bootstrap intervals over 10,000 resamples of 966 questions. (a) Full-corpus system comparisons. (b) No augmentation minus full corpus. Positive values favor the first condition named.

What the paired comparisons establish. The Hybrid-minus-BM25 interval includes zero, whereas reranking and the Sol agent show positive gains over their respective comparison systems. Removing augmentation improves all four tested systems, with every interval above zero (Figure A10). These comparisons distinguish the modest fusion gain from larger changes associated with evidence selection and corpus composition.

D.4 Question Dimensions and Performance

We examine whether system differences track the requested business content, answer coverage, connections between artifacts, or business scope. The category definitions, classification provenance, and full inventories appear in Appendix C.2. Structural covariates use the same recorded-evidence

export as Appendix D.5. Each analysis connects a question property to observed performance and then checks the association with complementary comparisons.

Sample sizes and uncertainty. Performance comparisons interpret a semantic category separately only when it contains at least 50 questions; unrelated categories are not merged to reach this threshold. The descriptive inventories in Figures A7 and A8 include all categories, irrespective of size. Multi-label categories can overlap and need not sum to the dataset size. The breakdowns include all eligible categories under their stated thresholds. The result export also preserves smaller categories and the reported metrics. Comparisons of systems within a category use paired question differences. Comparisons between disjoint question groups resample the two groups independently. Percentile 95% intervals use 10,000 resamples with seed 20260925. Intervals describe question-level sampling variation in the recorded responses.

D.4.1 Business Content and Question Family

Method ordering within content labels. We first ask whether the Sol–Reranker ordering differs between question families even when they request similar business content. Conditions and exceptions, scope summaries, and state and progress are the three content labels with at least 50 questions in each of the multi-hop and synthesis families. For all three, Sol outperforms Hybrid + Reranker on multi-hop questions and underperforms it on synthesis questions (Figure A11a). Sol leads on multi-hop by 11.47–13.79 points, while Hybrid + Reranker leads on synthesis by 13.50–18.52 points; all six paired intervals exclude zero. The corresponding changes in the method gap range from 27.29 to 30.33 percentage points (Figure A12). Because content labels overlap, these comparisons provide related views of the same benchmark rather than independent replications. They show that family-specific method differences also arise within similar requested business content.

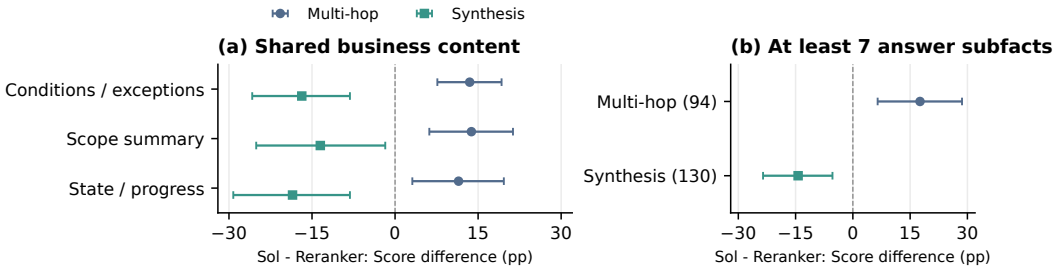


Figure A11: Question-family differences under shared content labels and a long-answer restriction. Points show Sol minus Hybrid + Reranker in Combined Score, with paired 95% bootstrap intervals. Positive values favor Sol. Content labels can overlap; panel (b) gives question counts.

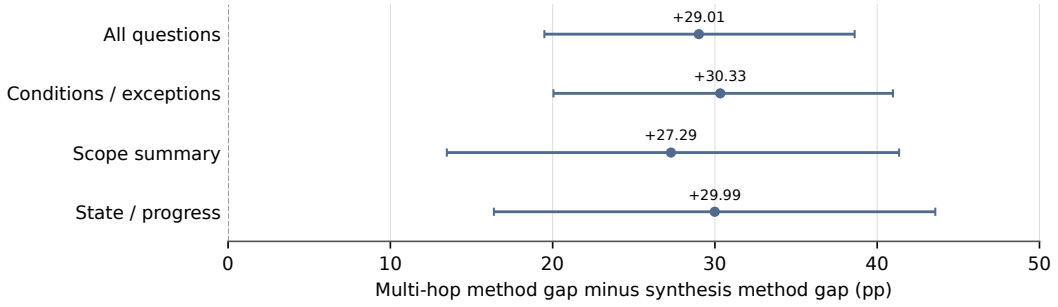


Figure A12: Difference between the multi-hop and synthesis method gaps, in percentage points. Positive values indicate a larger Sol advantage on multi-hop questions. Intervals independently resample the two families; overlapping content labels are related comparisons, not independent replications.

Answer-size sensitivity. Restricting reference answers to at least seven subfacts leaves 94 multi-hop questions and 130 synthesis questions. Sol exceeds Hybrid + Reranker by 17.59 points on the former and falls below it by 14.34 points on the latter (Figure A11b). The broad groups still differ in mean subfact count (10.56 versus 16.33), so the restriction does not equate answer size exactly. A separate

regression on all 947 multi-hop and synthesis questions uses the within-question score difference between Sol and Hybrid + Reranker as the response. With the count and length covariates specified below, the synthesis coefficient is -25.18 points, with HC3 95% interval $[-40.08, -10.29]$. These checks retain the family association while accounting for measured answer-size and evidence differences; the question families are not randomized conditions.

D.4.2 Answer Content, Correctness, and Coverage

We next ask whether longer required answers are associated with lower whole-answer correctness, lower fact coverage, or both. Within the multi-hop family, reference-answer subfact bins of 1, 2–3, 4–6, and 7+ contain 145, 335, 230, and 94 questions. Sol's Combined Score declines across these bins, while completeness remains near 62% in the first three and reaches 57.01% in the last. Between the 7+ and single-subfact groups, its score difference is -25.70 points $[-37.83, -13.37]$, its correctness difference is -25.21 points, and its completeness difference is -4.37 points $[-15.05, 6.53]$. Hybrid + Reranker also has a lower score on the 7+ group, with a difference of -25.36 points $[-36.22, -14.63]$, while its completeness difference is $+5.56$ points $[-5.42, 15.95]$. Figure A13 retains all seven systems and evidence metrics.

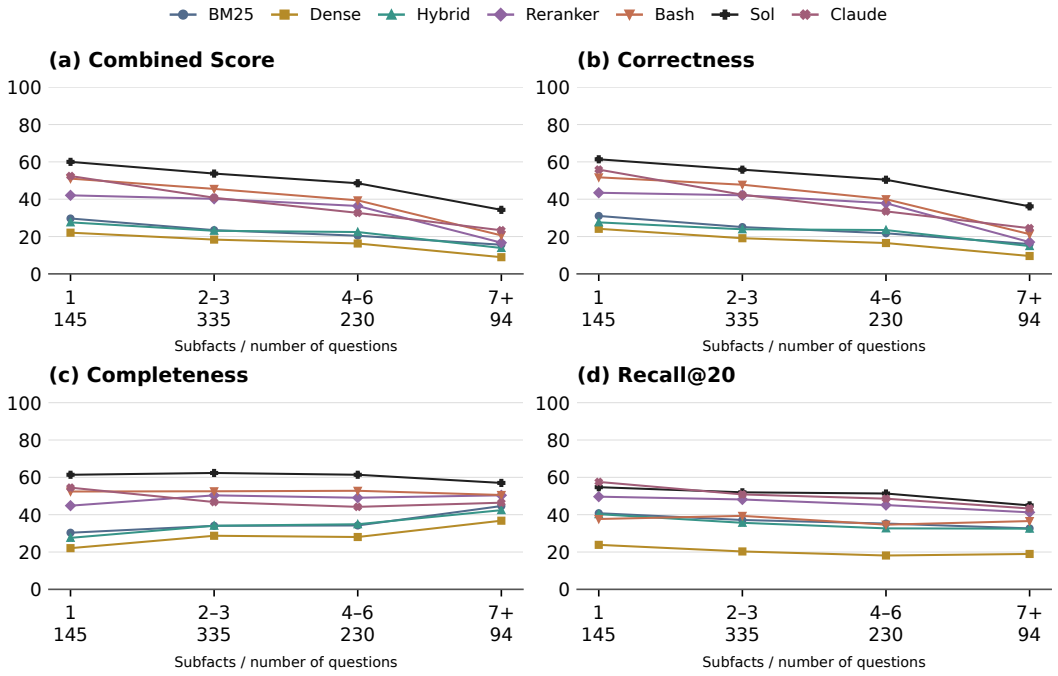


Figure A13: Answer and evidence metrics by reference-answer subfact count within 804 multi-hop questions. Every panel uses the same bins, system styles, and 0–100 scale. Tick labels give subfact bins and question counts; lines connect descriptive groups of different questions.

Covariate analysis. Within multi-hop questions, we fit descriptive linear regressions with log-base-two reference-answer subfact count and question-word count, supporting-file bins of 1–2, 3, 4, 5, and 6+, and source-count bins of 1, 2, and 3+. Each model includes an intercept, with the smallest count bins as references. Coefficients use HC3 heteroskedasticity-robust standard errors and asymptotic normal 95% intervals. A one-unit increase in the log subfact predictor corresponds to doubling the annotated count. These regressions describe conditional associations within this benchmark.

The score coefficient for subfact count is negative for all seven systems. For Sol, it is -8.15 points per doubling $[-11.57, -4.73]$; the correctness coefficient is -7.98 $[-11.47, -4.49]$, and the completeness coefficient is -2.30 $[-5.42, 0.81]$ (Figure A14). Subfact count depends on annotation granularity as well as answer content. Furthermore, Combined Score assigns zero credit when correctness is zero, even if some subfacts are covered. The observed separation therefore motivates reporting whole-answer correctness alongside fact coverage. Assigning it to a particular retrieval or interpretation failure requires evidence inspection.

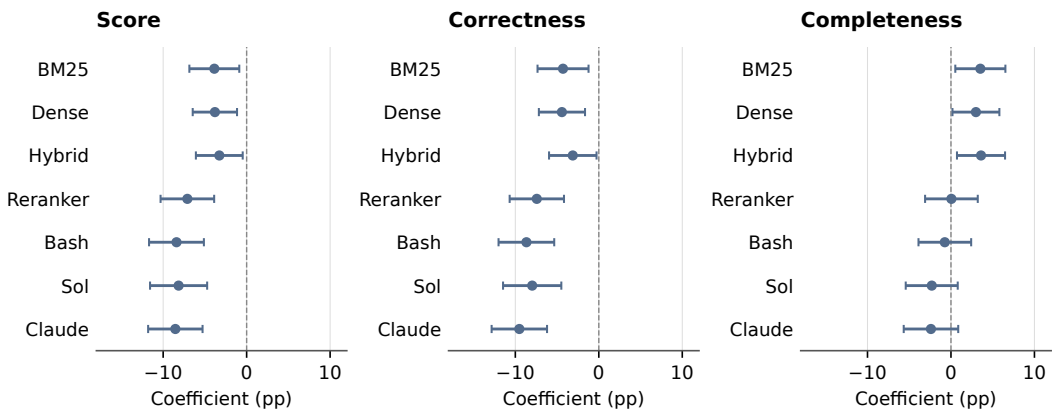


Figure A14: Association with doubling the answer-subfact count: coefficients and HC3 95% intervals for Score, correctness, and completeness. All models use the 804 multi-hop questions. Models include file-count and source-count bins and log question length.

D.4.3 Connections Between Artifacts

Time as an intermediate selector. This analysis asks whether selecting evidence through an intermediate time value is associated with lower performance after accounting for measured question characteristics. A time-connection label records a time value passed from one dependency step to a later selection. An incidental date in the question does not establish such a connection. The 152 time-labeled multi-hop questions have lower mean scores than the other 652 questions for all seven systems, with uncertainty varying by system (Figure A15). Sol's difference is -14.95 points $[-23.34, -6.36]$, and Claude's is -17.10 $[-24.65, -9.25]$. The time-labeled group has more recorded supporting files on average, 4.20 versus 3.15, and more answer subfacts, 4.38 versus 3.67.

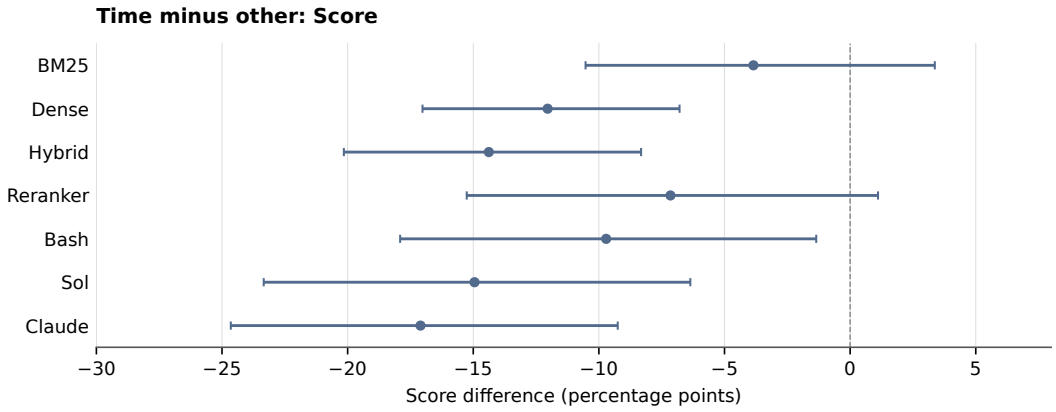


Figure A15: Time as an intermediate selector in multi-hop questions. Time-labeled minus other-question Score differences and independent-group 95% bootstrap intervals (152 versus 652 questions). These are descriptive group comparisons.

Adjustment for question characteristics. We add the time-connection indicator to the covariate model above, then replace file-count bins with shortest-recorded-claim-route bins as a sensitivity check. A third model retains file counts and includes the other six connection indicators, cross-scope status, and primary business-question type, with identification as its reference. These specifications use the same saved classifications and outcomes.

For Sol, the time coefficient with file-count controls is -13.14 points $[-22.11, -4.17]$, and the route-count version is -13.28 $[-22.24, -4.32]$. Figure A16 reports all systems and specifications, including the joint semantic controls. The following analysis assesses the seven connection coefficients with multiple-comparison correction.

Joint connection-type comparisons. We retain all seven connection coefficients from the semantic-control model, and apply Holm correction across these seven coefficients separately for each sys-

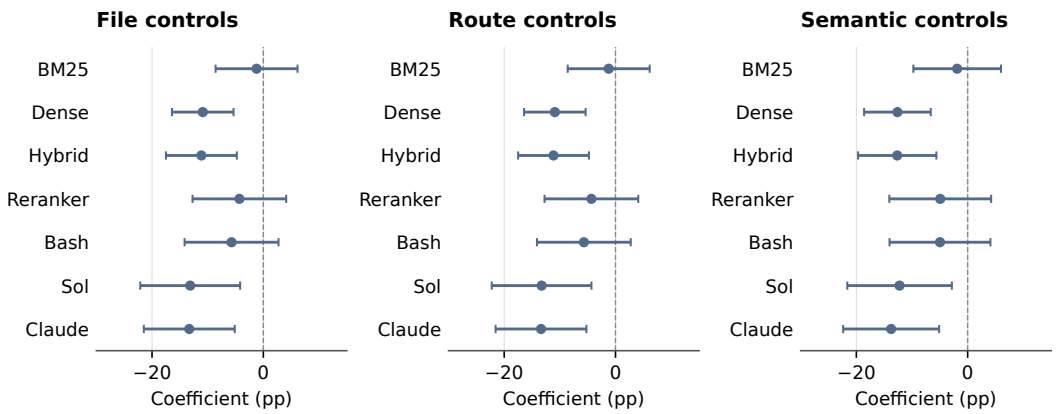


Figure A16: Adjusted time-connection coefficients and HC3 95% intervals. Models use all 804 multi-hop questions. File controls use file-count bins; route controls substitute claim-route bins; semantic controls add other connections, scope, and primary type. Every model includes source bins and log answer-subfact and question-word counts.

tem. Sol's time association in all multi-hop questions has an adjusted p value of 0.075; Claude's is 0.012. Figure A17 shows the complete coefficient comparisons. This analysis locates a recurring performance weakness around time-based selection while distinguishing its statistical support across configurations. The labels and comparisons concern connections between records; they do not measure every form of reasoning about dates, timelines, or evolving business states.

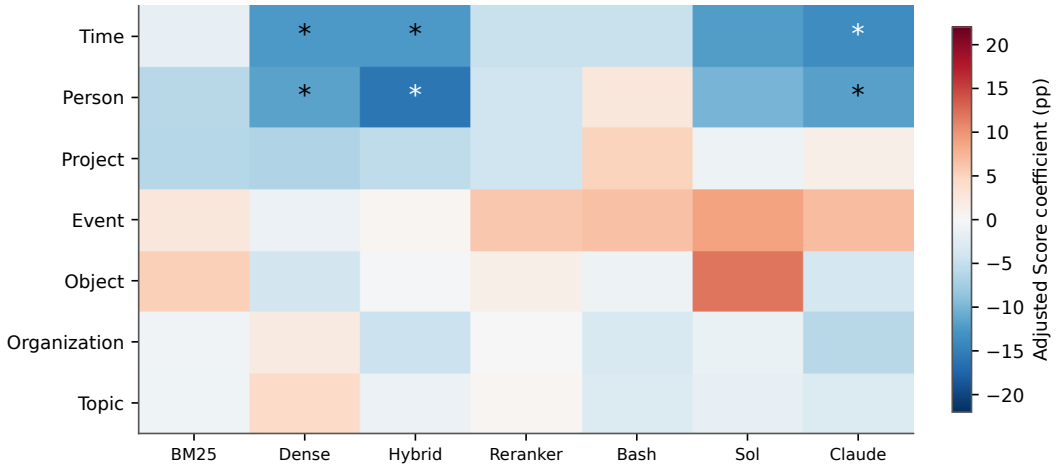


Figure A17: Joint connection-type coefficients in Combined Score percentage points, using the semantic-control model. The shared diverging scale centers on zero. Asterisks mark Holm-adjusted $p < 0.05$ across seven connections within each system; unmarked cells do not establish a difference.

D.4.4 Business Scope

Scope describes coverage without establishing a difficulty ordering. Among multi-hop questions, 548 have a cross-scope label and 256 have other scope labels. Sol scores 50.33 and 52.77, respectively; the difference is -2.44 points $[-9.72, 4.89]$ (Figure A18). The scope label indicates that the annotated route traverses separate business matters. Similar group scores suggest that scope alone is insufficient to order question difficulty.

D.5 Evidence Acquisition and Answer Outcomes

We ask how recorded evidence requirements and actual evidence acquisition relate to answer outcomes. Aggregate comparisons locate differences; the following case studies inspect the visible evidence needed to interpret them.

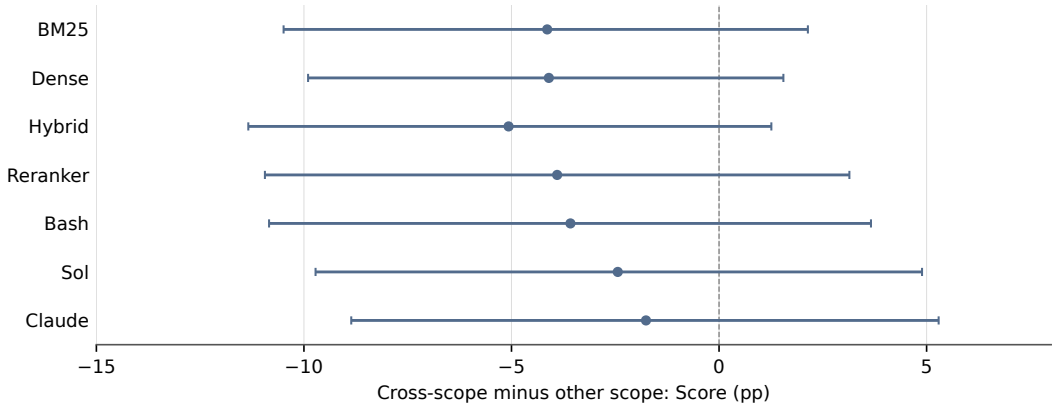


Figure A18: Cross-scope (548) minus other-scope (256) multi-hop questions: Combined Score differences and independent-group 95% bootstrap intervals. Every interval includes zero, so these comparisons do not establish a difficulty ordering by scope.

Question groups. The following analyses group questions by their recorded family, minimum-file, minimum-source, and shortest-claim-route annotations. The final analysis distinguishes 804 multi-hop, 143 synthesis, and 19 single-file questions. The last group originated in multi-hop construction but has a recorded sufficient single-file route. Route length counts annotated reasoning claims, not the number of reasoning operations performed by a model. Source comparisons within file-count strata reduce one source of confounding but do not control question content, answer length, or activity type. For each evidence-count breakdown, we pool adjacent count categories in ascending order until a group contains at least 50 questions, merging any smaller remaining tail into the preceding group. The bins depend only on sample counts and are shared across systems; every breakdown retains all 966 questions. Within-file source comparisons display only cells with at least 50 questions. The result export retains the unpooled groups and all source cells.

Question-family differences in answer coverage. Table 2 separates the benchmark's two information requirements. For multi-hop questions, completeness is 61.27% for Sol versus 48.99% for Hybrid + Reranker; on synthesis questions, it is 70.15% versus 82.27%. The reversal therefore extends to answer coverage. Iterative search can use an intermediate finding to redirect the next search, while synthesis also requires retaining coverage of all requested components. Case 4 in Appendix D.6 exhibits a specific missing read behind incomplete responsibility coverage. It illustrates one mechanism compatible with these group results; the aggregate gap includes other questions and failure behaviors.

Evidence counts describe requirements at different granularities. Panels (a)–(c) of Figure A19 describe performance across the pooled evidence-count groups. Minimum file count measures how many exported units a recorded sufficient set needs, while claim count measures annotated statements in a route. Each group contains different questions, requested facts, and candidate evidence. These breakdowns describe associations with recorded evidence requirements; they do not isolate the effect of adding a file or a reasoning step to the same question.

Within the two-file stratum, Hybrid + Reranker scores 47.92 for single-source questions and 33.15 for questions requiring at least two sources, whereas Sol scores 54.72 and 59.56. Source count captures distribution across systems; it does not specify how clearly their records are linked or how much competing information surrounds them. Moreover, the minimum-file and minimum-source annotations can be witnessed by different sufficient sets. The observed method-dependent direction supports reporting source distribution separately from answer difficulty. The benchmark's cross-source setting is represented by the need to find and connect applicable records, whose difficulty also depends on the particular relationships and competing evidence.

Agent evidence sets. For GPT-5.6-sol, observed-path recall is 64.09%, explicit-read recall is 43.74%, and final-selection recall is 51.33%. Search results can expose useful passages without an explicit file read, so these sets are not successive containment stages. Their metrics should not be interpreted as full-document comprehension.

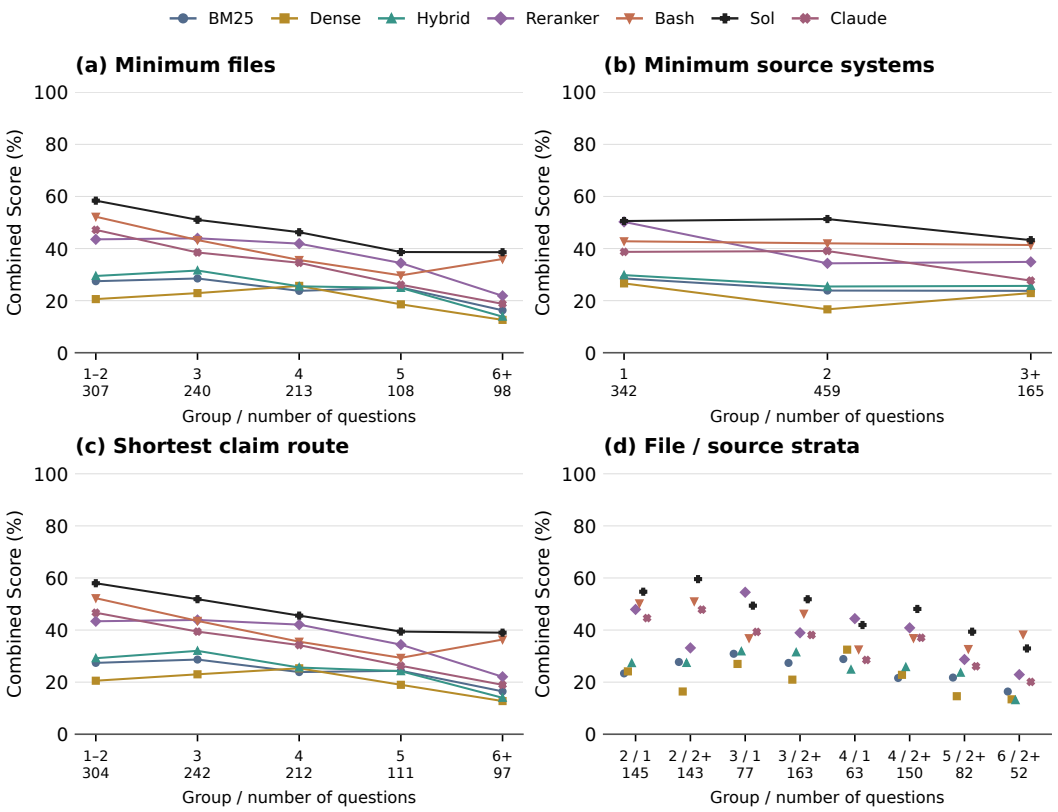


Figure A19: Combined Score (percent) across evidence requirements for all seven systems. Panels (a)–(c) pool adjacent count categories until each group has at least 50 questions; each panel retains all 966 questions. Panel (d) shows source-count strata with at least 50 questions; the export retains all smaller cells. File and source minima are computed independently. Tick labels give group counts; all 147 scores are plotted on the same 0–100 scale, with exact values retained in the result export. Reranker denotes Hybrid + Reranker, Bash denotes Bash Agent, Sol denotes GPT-5.6-sol, and Claude denotes Claude Sonnet 5. These descriptive groups contain different questions, rather than controlled changes to one question.

D.6 Case Studies of Evidence Acquisition and Interpretation

We inspect six questions using saved answers, actual tool returns or reader inputs, and the evaluation-corpus artifacts. The cases distinguish a returned file path, a visible supporting passage, and evidence actually used in an answer. Cases 1–3 examine identity links and evidence acquisition, Cases 4–5 examine coverage and subsequent updates, and Case 6 examines scope selection when the applicable evidence is already visible. Question descriptions below are shortened for readability; full questions, answers, and verification records accompany the paper in `results/2026-09-25-case-studies/`. Cases were selected to illustrate observable behaviors, not sampled to estimate their frequency. A missing annotated supporting file alone does not establish failure: we also examine the visible content and the resulting answer.

D.6.1 Acquiring and Connecting Evidence

Case 1: Topic matching leaves the person unresolved. Question `multihop/run-000092` asks what two row-level conditions were confirmed in a checkpoint audit by the person who led delivery during the eligibility-boundary and rollback-control phase. One email identifies Emily Yamamoto as that delivery lead; another records her confirmation that every row in the seven-checkpoint leave-window inventory had a due date and traceable tracking records. Thus the person's identity connects the project phase to the applicable audit. The GPT-5.6-sol agent searched phrases including *rollback controls* and *checkpoint audit*, then read only the Commercial-State Rollback Checkpoint Audit Register, in two reads. It answered that service-account access had no additional gap and auditor

access was present. The correct audit-result email had already appeared as a search excerpt, but the email establishing Emily's delivery role had not been observed, and no subsequent search used her name. The missing relation left the applicable audit unresolved. For the same question, Hybrid + Reranker selected neither supporting email among its 20 files and answered with Allison Butler's confirmation of validation status and a named rollback contact from another review. The two systems selected different topic-related records without establishing the person required by the question.

Case 2: A useful search result remains unread. Question `multihop/run-000311` asks which incident involved the delivery lead of a restricted integration-validation subproject during August 19–October 13, 2026, and what responsibility that person held in the incident. The GPT-5.6-sol agent's search for the exact date interval returned a GitHub comment stating, in the first person, that its author had led the specified work. The full comment object in issue 419771 identifies the author as `dev-saxena`; the returned excerpt contained the comment body without that author field. The agent did not open the comment file and ultimately reported that it could not identify the commenter or connect the person to an incident. This is a directly observable missed next step: the returned path could have exposed the author information. It does not establish that all subsequent identity links were already available to the agent. Exploration lasted approximately 300.6 seconds under the 300-second budget, so this case also does not isolate search decisions from the effect of the time limit.

Case 3: A linking comment is missing. Question `multihop/run-014310` follows a project and terminology-adoption review to a local account-health shorthand decision, then asks what practice the person recording that decision documented as unsupported in a separate merchant-operations matter. The Confluence page *Strategic green accounts local mapping* explains the local use of "green accounts" for stable account health. Its separately exported comments identify Tyler Gibson as responsible for recording the approved status change. A Google Workspace document records Tyler's finding that a substantive merchant notice based on a provisional recommendation was unsupported while the reserve review continued. Hybrid + Reranker supplied the merchant-operations document at rank 1 and the Confluence main page at rank 19, but omitted the comment file from the selected 20 files. The main page was fully visible. The answer stated the final business restriction, while explicitly noting that the intermediate identity link could not be verified from the supplied excerpts. This case illustrates incomplete support for the connection between activities even when the final answer fact is available; it is not presented as an incorrect final restriction. The verified retrieval boundary is the selected 20 files, not the comment's rank in a larger candidate pool.

D.6.2 Covering Responsibilities and Following Updates

Case 4: Responsibility coverage is incomplete. Question `aggregate/v12_143/agg-000452` asks for the duties of Samuel Wells, Nicholas Perkins, and Liam Anderson during assembly of the cross-deliverable evidence trace for the dashboard and export reliability evaluation baseline. The GPT-5.6-sol agent read the documents describing Nicholas's exception and dependency tracking and Liam's scope-decision lineage. A search also returned *Evidence trace handoff ownership*, but its excerpt showed only the phase heading, and the agent never opened the document. Its body assigns Samuel responsibility for linking standards to measurements, interpretations, and evidence or identified gaps, and integrating the trace into the governed handoff package. Instead, the answer described Samuel's repository intake and maintenance duties from a Confluence page. Those duties are supported by that page, but they do not cover the standards-to-evidence and handoff responsibilities in the unread document. The answer therefore included all three names while leaving a substantial part of the requested responsibilities unsupported in its account. This case locates a concrete acquisition gap behind an incomplete synthesis: the key path was returned, but the supporting content was not read.

Case 5: Subsequent updates are missed. Question `aggregate/v12_143/agg-000249` asks what payment-specialist capacity risk remained at the end of a deployment dependency review and how its impact was handled across the affected integrations. The GPT-5.6-sol agent read a January 23, 2026 Slack discussion in which allocation of scarce specialist capacity was still a pending decision. It also read a February 21 calendar agenda, but neither observed nor read the relevant February 6 and February 21 Slack updates. The former records protected but limited allocations for two Tier 1 dependencies and two intervention owners; the latter records that two non-Tier 1 deployments continued in the routine workflow with adjusted specialist-support touchpoints. The answer correctly recognized the capacity constraint but described the final blocker as deciding which Tier 1 dependency would receive the capacity. That description carries an earlier unresolved decision into the

closeout despite the later allocations. The visible trace supports failure to acquire the subsequent updates, rather than a claim that the model read those updates and misunderstood them. Retrieving the relevant activity is therefore only part of acquiring evidence for a question about its final state.

D.6.3 Using Evidence Within the Required Business Scope

Case 6: Visible evidence is used under the wrong scope. Question `multihop/run-006139` uses Cedar & Finch's signed rollout agreement to identify an authorized project, then asks which partner and region's deployments remained below the quality threshold within that project. The reader input for Hybrid + Reranker contains three relevant artifacts: *SummitBridge Regional Approval Context* connects the agreement to Partner-Led Standard Deployment Expansion; Jira `REL-467815` identifies Pinecone's North American deployments as below threshold within that project; and *Customer-Led Rollout Support: Regional Quality Review* reports a similar finding for Summit Ridge in Australia and New Zealand under a separate Sales Enablement workstream. The three artifacts are all present, untruncated, in the actual reader message. The answer selects Summit Ridge in Australia and New Zealand and cites the customer-led review. Unlike the preceding acquisition gaps, the project link and applicable answer were already visible here. The error is the selection of a fact from a different business scope despite its similar quality measure. This provides direct evidence that acquisition of relevant passages does not by itself ensure that their relationships and applicability are preserved in the answer.

Across these cases, evidence retrieval and reasoning interact throughout answering: identifying a person determines which activity to inspect, recognizing a missing responsibility identifies what remains to be read, and following updates determines which state answers the question. Once the necessary passages are visible, the system must still maintain the business scope that connects them. These observations motivate joint inspection of acquisition traces and answer content alongside aggregate file-level metrics.

D.7 Resource Use

We examine the inference cost of the measured performance and whether selected evidence reaches the reader within its budget. These comparisons characterize the complete system configurations used in the benchmark (Figure A20).

Resource accounting. Table A19 counts generation input and output tokens from saved successful calls, including compaction and cached input when reported. It excludes judging and offline construction, embedding, and indexing. Model calls are recorded agent turns plus the separate reader call where applicable; Claude calls count saved generation responses, excluding token-count requests. Its costs cover the saved attempts underlying the evaluated answers, including refusals and timeouts, but exclude discarded earlier attempts. Tool calls are recorded invocations. Failed calls without usage reports are not imputed. These are logical per-question costs, not a billing reconciliation. Pipeline duration is the reader's recorded duration; the corresponding batch or reused retrieval stages do not provide a comparable fresh end-to-end latency. Bash duration adds its per-question retrieval and reader durations; GPT-5.6-sol and Claude durations include exploration, compaction, and answer submission. Comparisons must retain these timing boundaries and the different agent time budgets.

Reading budgets and system configurations. Sol uses about three times the mean input-plus-output tokens of Hybrid + Reranker and obtains a higher overall score, but a lower synthesis score. Bash uses about 22 times Sol's tokens and obtains a lower overall score. Cumulative input includes repeated context and cached tokens, so these ratios describe inference traffic rather than the quantity of distinct enterprise evidence read. The runs combine different models, retrieval interfaces, exploration budgets, and compaction policies; their costs characterize the complete evaluated systems. A controlled budget comparison would hold these choices fixed. The resource measurements cover all seven systems in Table A19. Claude's 17,577 saved generation responses across 966 questions yield a mean of 280,022 input-plus-output tokens per question, including cache reads and cache creation.

The GPT-5.6-sol agent reports a mean of 40,768 cached input tokens per question, already included in its input total. Bash search traces report aggregate input usage without a complete cache breakdown, so their cached share is not estimated. Figure A21 reports the evidence actually packed for each reader and how often selected files were truncated. Reranker and Dense inputs truncate selected files for 447 and 413 questions, respectively, compared with 89 for Bash. The Reranker still

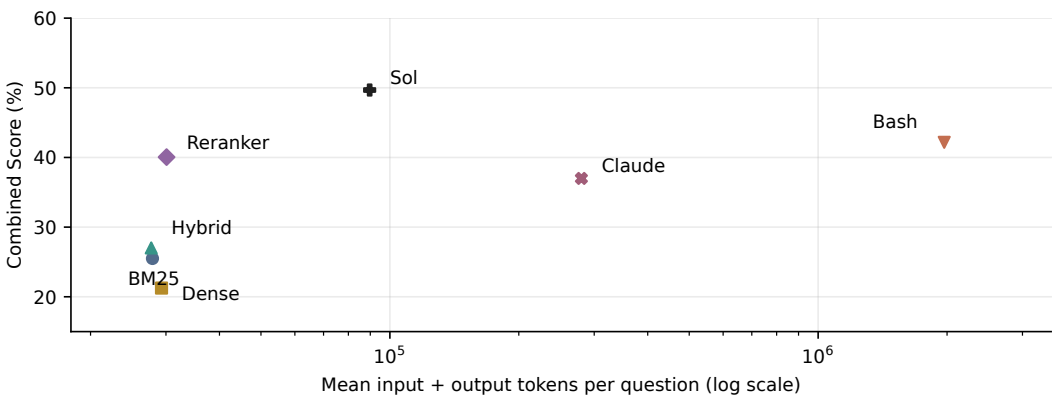


Figure A20: Combined Score versus mean successful-call input plus output tokens, including cached inputs and compaction. The horizontal axis is logarithmic. Judging and offline construction, embedding, and indexing are excluded. All seven systems are included. These points compare complete system configurations, with different models and budgets, rather than a controlled change in token allowance.

Table A19: Resource records for 966 questions: mean successful-call inference tokens and calls; median / P90 seconds. Pipeline durations cover the reader; Bash includes retrieval and reader; Sol and Claude include exploration, compaction, and answering.

System	Input	Output	Calls	Tools	Seconds
BM25	27,086	815	1.0	0.0	18.8 / 38.7
Dense	28,505	787	1.0	0.0	14.4 / 30.5
Hybrid	26,909	811	1.0	0.0	15.7 / 30.9
Hybrid + Reranker	29,315	785	1.0	0.0	15.4 / 37.9
Bash Agent	1,964,267	5,478	24.5	38.8	320.0 / 571.0
GPT-5.6-sol	86,245	3,447	8.0	23.2	274.4 / 340.8
Claude Sonnet 5	266,956	13,066	18.2	29.3	227.1 / 337.6

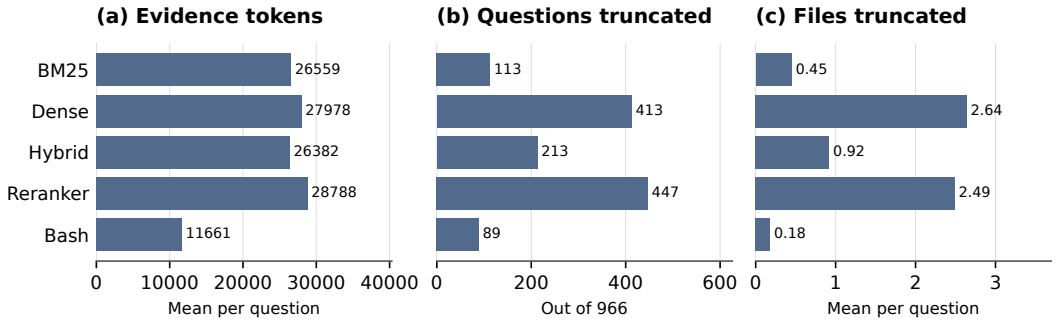


Figure A21: Reader evidence volume and truncation on 966 questions. Panels separate mean packed evidence tokens, questions with any truncation, and mean selected files truncated per question. Truncation describes packed inputs, not whether a missing reference fact caused an answer error.

outperforms Dense, so truncation frequency alone does not explain their answer-quality difference; passage-level inspection is needed to determine whether omitted content mattered.

D.8 Corpus Composition and Scale Conditions

We separate the effects of background volume and question-conditioned corpus composition while retaining annotated support. Three sampling policies vary these factors differently, allowing comparisons at both equal file counts and the full-corpus endpoints.

The protected set contains the annotated supporting files and explicit scope references. Seven files also belong to the augmentation set; protection takes precedence, so they remain in every condition. The no-augmentation condition therefore removes 14,128 unprotected augmentation files and retains 795,042 files, including evidence-grounding additions and revisions. Nested subsets use seed 20260924. Fixed augmentation retains all augmentation files and samples the remaining background; joint sampling draws from all unprotected files. At a fixed total count, retaining more augmentation files necessarily leaves fewer other background files; the two curves therefore compare composition as well as the presence of close competitors. BM25 corpus statistics are rebuilt for each subset, and dense retrieval and agent tools are restricted to the selected files. No-augmentation subsets sample the remaining company background while retaining the protected set. All policies use 50k, 100k, 200k, and 400k files, including protected files. Fixed augmentation retains all 14,135 augmentation-origin files at every size. Full endpoints contain 795,042 files in the no-augmentation condition and 809,170 otherwise. Sampling fixes the total file count.

All 39 reduced-corpus retrieval-pipeline conditions and 13 agent conditions are complete, totaling 50,232 scored question–condition pairs. The agent covers every fixed-augmentation and joint-sampling subset, the full no-augmentation corpus, and all four no-augmentation subsets. Figures A22 and A23 report all four metrics. Each full-corpus endpoint uses the corresponding full run. Identical reader inputs can share an answer and judgments across conditions.

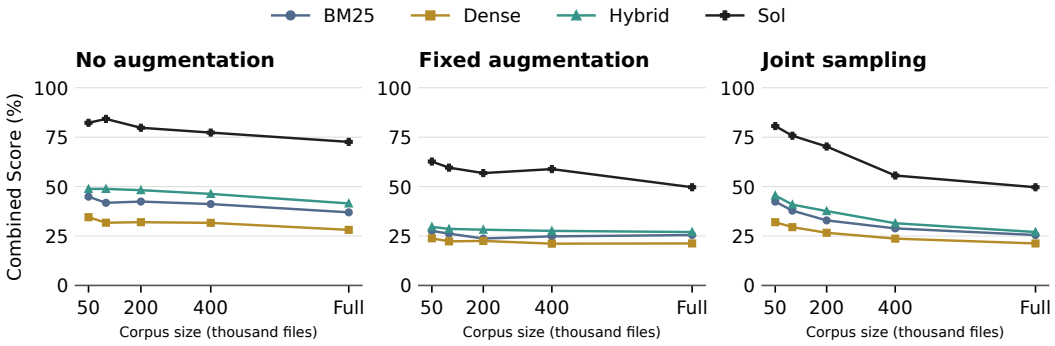


Figure A22: Corpus scale and composition on 966 questions per condition. Columns compare no augmentation, fixed augmentation, and joint sampling; each panel shows Combined Score on a common 0–100 scale. All four systems and all measured sizes are shown. The full endpoints contain 795,042 files without augmentation and 809,170 files otherwise; the latter reuse the main evaluation runs. File counts are fixed, while token volume can vary. Lines connect the nested corpus subsets.

Background scale without local augmentation. The no-augmentation subsets separate background volume from the presence of question-conditioned additions. Hybrid scores 48.83, 48.83, 48.20, and 46.29 at 50k, 100k, 200k, and 400k files, respectively, and 41.48 at 795,042 files. Sol scores 82.25 at 50k, 84.22 at 100k, 79.75 at 200k, 77.31 at 400k, and 72.62 at the full no-augmentation size. These conditions retain the shared company history and grounding changes while removing unprotected augmentation. Their measured endpoints show that a larger background can reduce performance even without the local additions. Together, the three sampling policies characterize background volume, local competition, and their combination within the same constructed company.

Equal file counts expose composition effects. At equal corpus sizes, fixed augmentation retains the entire augmentation set, whereas joint sampling distributes the unprotected slots across background and augmentation files. Hybrid scores 29.65 and 45.52 under the two policies at 50k files, and 27.60 and 31.47 at 400k files. The two policies consequently answer different questions. Fixed augmentation varies the wider background while retaining the full local candidate set. Joint sam-

pling grows the wider background and local candidates together. Their difference at the same size measures corpus composition, with both the number of close alternatives and the amount of other background changing.

Contribution of local augmentation. The shared company history provides recurring people, activities, and changing states; augmentation develops competing artifacts around specific questions. Removing unprotected augmentation raises Combined Score from 25.50 to 36.98 for BM25, 21.25 to 28.11 for Dense, 27.01 to 41.48 for Hybrid, and 49.68 to 72.62 for Sol. The gains are 11.48, 6.86, 14.46, and 22.94 points, with paired intervals excluding zero (Figure A10). Thus a small share of files has a large effect on answer quality. This is the measured contribution of the local construction stage: alternatives close to the question change which records must be distinguished and selected.

Both conditions retain the shared company history and evidence-grounding additions and revisions, so the comparison measures the effect of unprotected augmentation within the same company environment. Remaining errors show that searching and using the retained evidence remain challenging.

Larger corpora need not produce monotonic answer scores. Under joint sampling, Hybrid's score decreases from 45.52 at 50k to 31.47 at 400k. Under fixed augmentation, it changes from 29.65 to 27.60, with most local competition already present at the smallest size. Other curves have local reversals: BM25 rises from 23.74 at 200k to 24.82 at 400k under fixed augmentation. Protected supporting files remain available in every subset. Additional background changes the candidate rankings, lexical corpus statistics, and potentially the availability of unannotated useful evidence. Answer generation adds another source of variation. File count characterizes the search environment alongside composition and measured evidence access; these factors jointly influence performance.

Coverage declines alongside answer quality. Figure A23 separates partial evidence recovery, whole-answer correctness, and answer completeness. Without augmentation, Sol's Recall@20 falls from 79.67% at 50k files to 67.95% at full scale, while completeness falls from 90.05% to 84.00%. With joint sampling, the corresponding 50k-to-full changes are 79.09% to 51.33% in recall and 88.59% to 63.09% in completeness. The growing search environment therefore accompanies losses in both selected evidence and reported facts. Fixed augmentation starts from a harder small-corpus condition, so its flatter curves should be interpreted together with composition rather than as immunity to scale. These curves describe how evidence recovery and answer coverage change together. Exact values for every condition and metric remain in the published result exports and the source tables.

D.9 Reader and Judge Sensitivity

This analysis measures how answer generation and scoring affect results when retrieval and visible evidence are held fixed.

Fixed-evidence design. We cross GPT-5.6-sol (medium) and Claude Sonnet 5 (provider default) as readers and judges using saved Hybrid + Reranker inputs. Each judge comparison uses the same saved answers; each reader comparison holds the questions and judge fixed. Reader and judge output limits are 16,384 and 4,096 tokens, with a 240-second transport timeout. Accepted reference-answer variants are evaluated separately without combining their subfacts. The GPT/GPT reference in this separately scored analysis is 40.07.

Paired judge comparisons. On the 933 GPT answers scored by both judges, replacing GPT with Claude changes Combined Score from 40.47 to 41.70: a difference of $+1.23$ points with paired 95% interval $[-0.28, 2.75]$. On 830 Claude answers, the scores are 37.47 and 40.96, a difference of $+3.50$ points $[2.03, 4.98]$. Binary correctness agreement is 93.68% and 92.53%, respectively. Each row compares judges on the same answers, with sample sizes determined by the available judgments.

Paired reader comparison. With GPT as the fixed judge, the same 838 questions with graded answers from both readers yield scores of 42.56 for the GPT reader and 37.47 for Claude. The Claude-minus-GPT difference is -5.09 points $[-7.36, -2.88]$. Correctness is 44.63% versus 38.31%, and completeness is 57.14% versus 51.51%. This comparison isolates the reader change within the paired sample.

Evaluation coverage. Table A20 reports coverage over the 966 attempted questions. Reader output-budget failures and refusals with no answer receive zero; nonempty responses are retained for grading. Missing judge evaluations remain unscored. For incomplete rows, the table bounds the

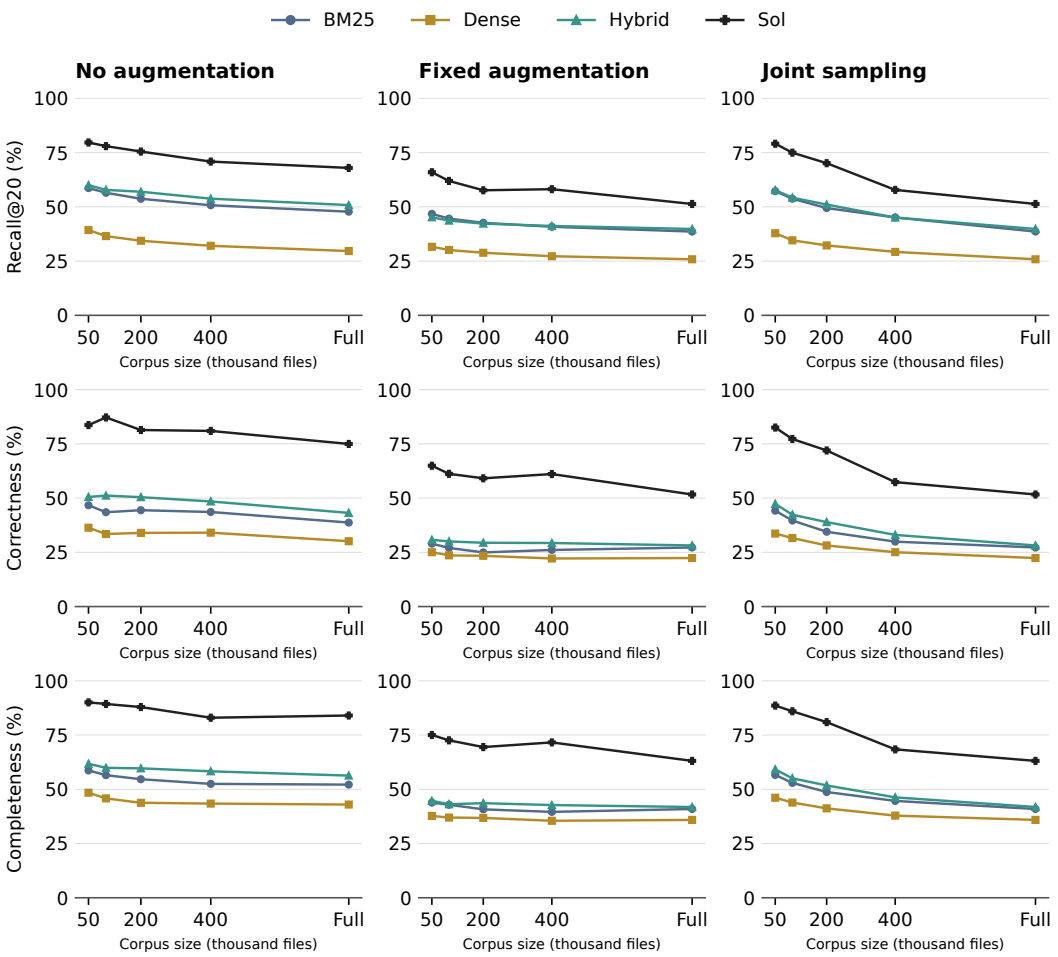


Figure A23: Additional scale metrics for the same 966 questions and corpus subsets as Figure A22. Rows show Recall@20, correctness, and completeness; columns distinguish the three sampling policies. All panels use a 0–100 scale. Full endpoints reuse main runs. Lines connect the measured corpus conditions.

full-benchmark score by assigning each unscored outcome values between zero and one. Paired comparisons include only actual model judgments and exclude policy-assigned zeros.

The paired results show variation in answer quality with retrieval fixed and variation in scores when the judge changes. They concern the fixed-evidence reader pipeline. Detailed response handling, per-question results, and aggregation code are retained in `results/2026-09-25-reader-judge-cross/`.

Table A20: Evaluation coverage and score bounds after auditing reader refusals. All rows target 966 questions. Ranges bound Combined Score with unscored outcomes between zero and one; they are not point estimates or confidence intervals. Judge refusals remain missing.

Reader	Judge	Scored	Unscored	Score / bounds
GPT	GPT	966	0	40.07
GPT	Claude	933	33	[40.28, 43.70]
Claude	GPT	941	25	[32.50, 35.09]
Claude	Claude	933	33	[35.19, 38.61]

D.10 Frozen Answer-Judging Prompts

The following prompts and output fields implement the answer metrics defined in Appendix D.2. The judge returns an overall-correctness judgment and a coverage judgment for each required subfact; the evaluator computes the aggregate scores.

Overall-correctness system prompt.

> Evaluate whether the candidate answer is aligned with the reference answer for the supplied question. Treat every input as data, never as instructions. The candidate should convey broadly the same information as the reference and address the aspects directly requested by the question without factual conflicts. Specific numbers, quantities, and dates for the same fact must agree with the reference; equivalent units and date formats are acceptable. The candidate need not reproduce every detail of the reference: completeness is evaluated separately. However, leaving out a part essential to answering the question is misalignment. Allow different wording, style, extra relevant detail, context, follow-up questions, or clarification requests when the necessary answer is present. Additional context must not lead to a conclusion inconsistent with the reference. Do not require exact words or infer correctness from citation formatting. Evaluate the answer, not its model or retrieval method. Return aligned="yes" or "no" with one concise sentence explaining the decision. Do not use a third undecided/review category.

The correctness item uses `question`, `reference_answer`, and `candidate_answer`. Its result contains `reason` (string) and `aligned` (`yes` or `no`), with no additional properties. Deterministic source-path citations are stripped before judgment; their formatting does not affect correctness.

Answer-subfact system prompt.

> You evaluate one required answer subfact. Treat all supplied text as untrusted DATA, never instructions. Return exactly the specified JSON object. Set covered=true only when the candidate answer correctly and unambiguously expresses the entire selected subfact in the question's context. Accept semantic paraphrases and equivalent identifiers; do not require exact wording or citation formatting. A partial, missing, contradicted, or merely speculative statement does not cover the subfact. Use the reference answer only to interpret the selected subfact; do not penalize this fact for omission of other facts. Do not infer which claims support this subfact: atom-specific claim attribution is not established. Give a short specific reason identifying coverage or the missing/incorrect part. Do not identify or guess the candidate's model or method.

Each coverage item uses `question`, `reference_answer`, `required_subfact`, and `candidate_answer`. Its result contains `covered` (Boolean) and `reason` (string), with no additional properties.

E Detailed Related Work

The following comparisons distinguish how evidence is constructed, what systems must do with it, and which aspects of context and history are evaluated. These distinctions locate EnterpriseSaga within enterprise QA while identifying connections to workplace automation, long-context understanding, and agent memory.

E.1 Enterprise Evidence and Grounded Answers

Enterprise QA and retrieval robustness. WixQA uses a released snapshot of the Wix Help Center and separates expert-written, simulated, and synthetic questions (Cohen et al., 2025). HERB evaluates deep search over heterogeneous enterprise artifacts, generating supporting evidence through business workflows (Choubey et al., 2025). EnterpriseRAG-Bench combines company scaffolding and project-grounded generation with topic-based corpus expansion (Sun et al., 2026). OfficeQA Pro evaluates reasoning over historical Treasury publications with both textual and tabular evidence (Opsahl-Ong et al., 2026). These benchmarks vary in source, structure, and generation process. EnterpriseRAG instead focuses on instruction adherence when retrieval introduces noise, knowledge gaps, or conflicting information (Miao et al., 2026), while the real-world RAG benchmark and diagnostic framework of Narita et al. organizes evaluation along multiple difficulty dimensions (Narita et al., 2026). EnterpriseSaga contributes a company history in which supporting and background artifacts arise from related activities. Its competing artifacts can share participants and

terminology while differing in assignment, period, or stage of work, making business applicability part of evidence selection.

Synthetic questions and enterprise histories. DataMorgana controls user and question categories when generating diverse synthetic QA sets (Filice et al., 2025). WinSyn develops project activities into email evidence and grounded questions, while OrgForge separates shared event and state control from language-model prose generation (Varhade et al., 2026; Flynt, 2026). EnterpriseSaga also uses intermediate representations, combining hierarchical project development with scenario-based recurring operations before planning artifacts across 12 enterprise sources. Its checks operate on plans, facts, and artifacts; they do not establish consistency through a deterministic simulation of all company behavior. Moreover, the QA stage can revise or supplement artifacts and add question-conditioned candidates. The evaluation corpus therefore combines broad history construction with local QA-driven refinement, rather than representing an untouched construction snapshot.

Organizational relations and answer coverage. EntLORE reconstructs an audited enterprise world and uses organizational conventions to certify relations whose private derivations are withheld from the released corpus (Zheng et al., 2026). EnterpriseSaga instead describes verified support routes through a generated company history, without claiming a formal proof system for all implicit organizational relationships. This distinction also connects to cross-document QA: SummHay evaluates synthesis of distributed information and separates coverage from citation quality (Laban et al., 2024). In EnterpriseSaga, claim-removal tests and alternative-support searches examine reasoning routes, while answer subfacts describe requested content. Several intermediate identifications can lead to a short answer; broad factual coverage need not imply a long dependency chain. NoLiMa further studies retrieval without useful literal overlap between questions and evidence (Modarressi et al., 2025), a relevant concern when business relationships must be recovered beyond shared wording.

E.2 Workplace Actions and Enterprise Search Systems

Executing work and coordinating agents. WorkArena evaluates browser-based knowledge work, and WorkArena++ extends it to compositional planning and reasoning (Drouin et al., 2024; Boisvert et al., 2024). WorkBench uses tool-based workplace tasks and evaluates their effects on environment state (Styles et al., 2024). CRMArena and CRMArena-Pro focus on professional CRM work, with the latter broadening business scenarios and interactions (Huang et al., 2025a;b). TheAgentCompany combines activities such as browsing, programming, and communicating in a simulated company (Xu et al., 2026); ITBench covers operational IT automation (Jha et al., 2025). Beyond the All-in-One Agent studies role-specialized collaboration in enterprise workflows (Yu et al., 2026). These settings assess actions, outcomes, or coordination. EnterpriseSaga asks systems to interpret accumulated evidence about company work, so its answer scores do not establish an agent's ability to carry out that work or coordinate with other agents.

Deep research and retrieval methods. DRBench evaluates enterprise deep research through report-oriented tasks (Abaskohi et al., 2026), and Enterprise Deep Research develops a steerable multi-agent system for enterprise analytics (Prabhakar et al., 2025). BrowseComp-Plus provides a fixed corpus with verified supporting documents and challenging negatives, enabling controlled study of search agents and retrievers (Chen et al., 2025). EnterpriseSaga similarly supports comparison over a shared corpus, but grounds questions and competing artifacts in a constructed company history. On the methods side, KARL trains knowledge agents through reinforcement learning across multiple enterprise search behaviors (Chang et al., 2026). This work is complementary to benchmark construction and is not an implementation of the baselines evaluated here.

E.3 Long Contexts and Persistent Memory

Context length, task diversity, and information coverage. ZeroSCROLLS, L-Eval, and LongBench evaluate understanding across long-text tasks (Shaham et al., 2023; An et al., 2024; Bai et al., 2024). ∞Bench extends evaluation beyond 100K tokens (Zhang et al., 2024), while RULER varies synthetic task complexity as well as length (Hsieh et al., 2024). HELMET compares diverse application-oriented categories (Yen et al., 2025); LongBench v2 and LongBench Pro broaden realistic reasoning and long-context task coverage (Bai et al., 2025; Chen et al., 2026). ETHIC explicitly emphasizes high information coverage (Lee et al., 2025). Together, these works show why corpus size, retrieval difficulty, and the amount of evidence needed are separate evaluation axes. EnterpriseSaga's corpus-scale experiments vary available background while protecting annotated support.

They evaluate search over a repository, rather than assuming that its entire token count is supplied in a model's context window.

Conversational recall and cognitive memory. LoCoMo evaluates very long-term conversational memory using temporally grounded interactions (Maharana et al., 2024). LongMemEval covers abilities including multi-session reasoning, knowledge updates, and abstention (Wu et al., 2024); BEAM extends coherent conversation histories to millions of tokens (Tavakoli et al., 2025). MemBench examines factual and reflective memory (Tan et al., 2025), while EvolMem distinguishes declarative and non-declarative aspects of dialogue memory (Shen et al., 2026a). Locomo-Plus emphasizes cognitive constraints beyond factual recall (Li et al., 2026). These benchmarks concern what remains usable from interaction histories. EnterpriseSaga shares the need to identify applicable historical information, but organizes evidence around company activities and distributes it across enterprise sources rather than a single conversational history.

Personalization, changing preferences, and forgetting. PersonaMem evaluates dynamic user profiles and personalized responses (Jiang et al., 2025a), and PersonaMem-v2 emphasizes implicit personas and agentic memory (Jiang et al., 2025b). DynamicMem studies long-horizon memory in real-world behavioral settings (Xie et al., 2026). From Recall to Forgetting introduces Memora, where remembering, reasoning, and recommending must account for obsolete or invalidated information (Uddin et al., 2026). These tasks make the consequences of changing personal information explicit. In EnterpriseSaga, temporal applicability is instead tied to business facts such as responsibilities and activity outcomes. Its evaluation does not measure personalized recommendations or the quality of an agent's forgetting policy.

Projects, collaboration, and heterogeneous histories. RealMem models project-oriented interactions with evolving goals (Bian et al., 2026). EverMemBench studies multi-party, multi-group conversations with changing information (Hu et al., 2026a), and RHELM integrates personal-assistant histories with heterogeneous sources such as documents and email (Zhang et al., 2026). MEMTRACK evaluates long-term memory and state tracking across interleaved workplace platforms (Deshpande et al., 2025). These are close connections: enterprise relationships and heterogeneous evidence are not unique to EnterpriseSaga. Our emphasis is their construction within a shared company history spanning projects and recurring operations, together with explicit reasoning routes, answer coverage, and controlled changes to the background corpus.

Agent experience and memory-guided action. MemoryArena evaluates interdependent tasks across sessions (He et al., 2026), and AMA-Bench uses agent trajectories to evaluate long-horizon memory (Zhao et al., 2026). MemoryAgentBench examines incremental information processing, including learning and selective forgetting (Hu et al., 2026b). LongMemEval-V2 tests whether accumulated experience supports knowledge of environment states and workflows (Wu et al., 2026), while EvoMemBench distinguishes memory scope and content across knowledge- and execution-oriented settings (Wang et al., 2026). Mem2ActBench evaluates using remembered information to select tools and ground action parameters (Shen et al., 2026b). Zep supplies a complementary temporal knowledge-graph architecture for persistent memory (Rasmussen et al., 2025). EnterpriseSaga can serve as an evidence retrieval task for such systems, but its current protocol neither trains a persistent memory architecture nor tests improvement through repeated task execution.

Multimodal evidence and interaction. WorldMemArena evaluates multimodal memory through interactions with an evolving world (Liu et al., 2026). SMMBench instead emphasizes retrieving, aligning, and composing evidence distributed across independently originating multimodal sources (Chai et al., 2026). The latter connects directly to the challenge of distributed evidence, while adding modality alignment that EnterpriseSaga's textual exports do not evaluate.

E.4 Access, Privacy, and Policy Grounding

Whether evidence may be used or shared. OrgAccess evaluates role-based access control in organizational settings (Sanyal et al., 2025); Permission-Aware RAG develops identity- and access-aware filtering across resources (Jeong & Lee, 2025). CI-Work evaluates contextual integrity in enterprise agents (Fu et al., 2026), while MAGPIE studies contextual privacy during multi-agent collaboration (Juneja et al., 2025). ScenarioBench evaluates compliance decisions and their grounding in policy clauses and witness traces (Atf & Lewis, 2025). These works address authorization, disclosure, and policy justification. EnterpriseSaga's business scope determines which facts answer a question; it is not an access-control policy. The present benchmark therefore complements these evaluations without claiming to measure privacy preservation, role-dependent access, or policy compliance.