DON'T WASTE YOUR SEARCH-AND-VISIT SIGNALS IN DEEP RESEARCH

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep research agents repeatedly search and visit the web, yet task-relevant signals exposed or derived during these interactions may not be retained in the updated research state. Representative approaches improve planning, exploration, or evidence organization, while systematic reuse of such unretained signals remains underexplored. We therefore introduce **WasteHub**, a plug-and-play mechanism that stores Unvisited-result, Visit-content, and Source-appraisal waste as typed, provenance-preserving residuals and selectively reuses them through a stage-aware gate without modifying the underlying framework's planner or search/visit interfaces. With ReAct as the underlying framework, WasteHub outperforms the Base setting and two baselines that augment Base with process reward model feedback (PRM-F) and post-answer exploration (PAE), respectively, on the aggregate metrics of DeepResearch Bench and DeepResearch Bench II. WasteHub uses fewer cumulative input tokens and has lower end-to-end latency than both PRM-F and PAE. Ablations show that all three residual types and both reuse phases contribute to the full method. Cross-framework experiments with Self-Manager and WebWeaver show that WasteHub achieves the best aggregate performance on both benchmarks.

1 INTRODUCTION

Deep research agents conduct multi-round web research for complex, open-ended questions and produce long-form, citation-supported reports through web search, webpage reading, and evidence synthesis (Li et al., 2026b; Dong et al., 2026). We focus on web-based deep research frameworks built around `search` and `visit`. Here, `search` returns candidate sources with search-result metadata, while `visit` reads selected pages and distills them into a goal-conditioned summary for the current sub-goal (Zhang et al., 2026; Li et al., 2025). This goal-conditioned processing is intended to reduce full-page context overhead while emphasizing information relevant to the current sub-goal (Liu et al., 2024; Jiang et al., 2024; Yoon et al., 2024).

Prior work has advanced deep research agents along several directions. Representative methods focus on long-horizon planning and trajectory control (Li et al., 2026b; Yan et al., 2026; Fan et al., 2026), scale exploration through parallel search or multi-agent collaboration (Lin et al., 2026; Xu et al., 2026; Ko et al., 2026), or organize research materials with structured representations and persistent workspaces (Zhang et al., 2026; Zhu et al., 2026; Hu et al., 2026). However, potential information waste in the search-and-visit process has received limited systematic attention. We define ***information waste*** as task-relevant signals not retained in the updated research state, excluding signals irrelevant to the original research task or already covered by that state. We focus on three forms distinguished by when they arise and their information status, as illustrated in Figure 1.

- ▷ **Unvisited-result waste.** Unvisited search candidates not retained for later exploration; their search-result metadata remains an unverified cue.
- ▷ **Visit-content waste.** Task-relevant page content represented in neither the goal-conditioned visit summary nor the updated research state.
- ▷ **Source-appraisal waste.** After a successful visit, the available source information may indicate whether and under what conditions a source is suitable for use. Such task-relevant information constitutes Source-appraisal waste when it is not retained for later reuse.

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.

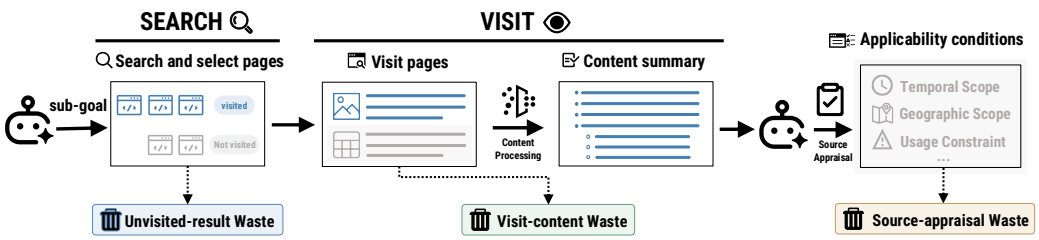

Figure 1: Three forms of information waste arising in the search-and-visit process.

The three forms arise at different stages and, when stored as residuals, retain their respective information statuses and intended uses. Unvisited-result residuals remain unverified cues for source selection, Visit-content residuals provide page-derived content for reuse, while Source-appraisal residuals describe the conditions under which visited sources can be used. Reintroducing these signals without distinction may blur their evidential boundaries and interfere with subsequent exploration, evidence use, and synthesis. The key is therefore to preserve these distinctions and selectively reuse each signal according to the current research stage.

To this end, we introduce **WasteHub**, a plug-and-play mechanism that maintains an auxiliary residual state alongside the underlying framework's native research state and selectively reuses stored signals at appropriate research stages while leaving its planner and search/visit interfaces unchanged. WasteHub stores the three forms of information waste as typed residuals that preserve their provenance and information status, and uses a stage-aware gate to select residuals that are valid for the current stage, relevant to its inputs, and not already covered. Unvisited-result residuals provide non-evidential cues for subsequent source selection. Visit-content residuals support same-goal evidence reuse and pre-finalization synthesis, while Source-appraisal residuals guide the reuse of previously explored sources and provide source-linked scope annotations during report finalization. WasteHub reuses only signals already exposed or derived during the existing search-and-visit process and does not independently initiate additional search or visit calls.

We evaluate WasteHub on DeepResearch Bench (DRB) and DeepResearch Bench II (DRB-II), two public report-level benchmarks for deep research agents. Under ReAct, we compare WasteHub with the original framework implementation without additional guidance (Base), process reward model feedback (PRM-F), and post-answer exploration (PAE) using report-quality, fine-grained, and resource-use metrics. Residual-type and reuse-phase ablations evaluate the effects of removing residual types or reuse phases, while experiments with two additional research frameworks, Self-Manager and WebWeaver, assess cross-framework generalization. WasteHub improves aggregate scores over the corresponding Base settings across all evaluated framework-benchmark combinations and outperforms the two enhancement baselines under ReAct.

Our contributions are summarized as follows.

1. **Definition.** We define information waste in the search-and-visit process and distinguish the three forms considered here: Unvisited-result, Visit-content, and Source-appraisal.

2. **Method.** WasteHub stores these signals as typed, provenance-preserving residuals and reuses them through a stage-aware gate while leaving the planner and search/visit interfaces unchanged.

3. **Evaluation.** We evaluate WasteHub on two report-level benchmarks and three research frameworks, demonstrating the highest aggregate scores among evaluated methods while analyzing resource efficiency and component contributions.

2 METHODOLOGY

This section first formalizes the search-and-visit process considered in this work and then introduces how **WasteHub** collects and reuses signals that are not fully retained during research.

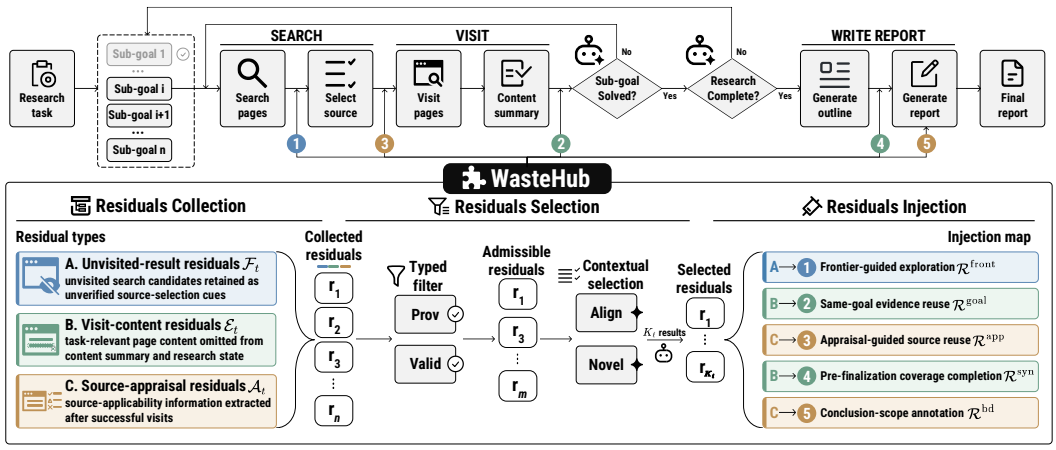

Figure 2: WasteHub pipeline. Three types of residual signals are collected, contextually selected, and injected at five stage-specific reuse points across research and report finalization.

2.1 Vanilla Formulation

We consider a local search-and-visit interaction as the basic research step. Given a research task x, the agent maintains a research state s_t summarizing the information retained before step t, determines a current sub-goal g_t, and formulates a natural-language search query q_t. The search tool returns a set of candidate sources

$$\mathcal{C}_t = \text{Search}(q_t) = \{c_{t,i}\}_{i=1}^{n_t}, \qquad c_{t,i} = (u_{t,i}, m_{t,i}), \tag{1}$$

where n_t is the number of returned candidates, $u_{t,i}$ identifies the i-th source, and $m_{t,i}$ contains search-result metadata such as the title, snippet, source domain, and publication date. The agent selects an index set $I_t \subseteq \{1, \ldots, n_t\}$ for further inspection. For each $i \in I_t$, the visit tool obtains the webpage content $d_{t,i}$ and produces a goal-conditioned summary

$$e_{t,i} = \Phi(d_{t,i}; g_t), \tag{2}$$

where Φ denotes the goal-conditioned summarization performed by visit and $e_{t,i}$ is the resulting visit summary returned to the agent. Let $\mathcal{B}$ denote the underlying research framework, $o_t^{\mathcal{B}}$ its tool observation at step t, and $\mathcal{U}_{\mathcal{B}}$ its native state-update rule. The state is updated as $s_{t+1} = \mathcal{U}_{\mathcal{B}}(s_t, o_t^{\mathcal{B}})$.

To characterize task-relevant signals that are not carried forward into the next research state, let $\text{Rel}_x(z)$ indicate that signal z is relevant to the original research task x. We use $\text{Cov}(z, r)$ to indicate that the semantic content of z is represented in context r. We further let $\text{Units}(d)$ denote the semantic content units of page d. The three forms of information waste are defined as

$$\begin{aligned}
\mathcal{W}_t^{\text{unv}} &= \{c_{t,i} \mid i \notin I_t,\ \text{Rel}_x(c_{t,i}),\ \neg\,\text{Cov}(c_{t,i}, s_{t+1})\}\,, \\
\mathcal{W}_t^{\text{con}} &= \{z \mid i \in I_t,\ z \in \text{Units}(d_{t,i}),\ \neg\,\text{Cov}(z, e_{t,i}),\ \text{Rel}_x(z),\ \neg\,\text{Cov}(z, s_{t+1})\}\,, \\
\mathcal{W}_t^{\text{app}} &= \{a_{t,i} \mid i \in I_t,\ a_{t,i} \neq \varnothing,\ \text{Rel}_x(a_{t,i}),\ \neg\,\text{Cov}(a_{t,i}, s_{t+1})\}\,.
\end{aligned} \tag{3}$$

Here, $a_{t,i}$ denotes task-relevant applicability information, if any, for the visited source $u_{t,i}$.

2.2 Our Method: WasteHub

WasteHub (see Figure 2) maintains an auxiliary state that stores the three forms of information waste in Section 2.1 as typed residuals, and selectively reuses them according to the current stage of research. WasteHub thus maintains Unvisited-result, Visit-content, and Source-appraisal residuals.

$$\mathcal{H}_t = \mathcal{F}_t \uplus \mathcal{E}_t \uplus \mathcal{A}_t, \tag{4}$$

where $\mathcal{F}_t$, $\mathcal{E}_t$, and $\mathcal{A}_t$ are typed candidate stores for Unvisited-result, Visit-content, and Source-appraisal waste, respectively; stage-specific alignment and novelty are enforced before reuse. Each residual retains its provenance and the type-specific information required for subsequent selection. WasteHub leaves the underlying framework's planner and search/visit interfaces unchanged and does not initiate new search or visit calls on its own.

The residual stores are cumulative. Let $\widetilde{s}_t$ denote the research state produced with WasteHub, where $\widetilde{s}_0 = s_0$. At process step t, same-goal and appraisal reuse access only records available before the current visit, whereas finalization uses the terminal stores.

2.2.1 Residual Collection

Unvisited-result residuals. After $\mathcal{C}_t$ is returned, WasteHub stores each task-relevant candidate $c_{t,i}$ not visited at step t in $\mathcal{F}_t$ with its search-result metadata. Its status is updated upon a later visit. These residuals contain only search-returned information and remain unverified.

Visit-content residuals. For each $i \in I_t$, the underlying framework receives the original visit summary $e_{t,i}$. WasteHub also extracts residual content from the page content obtained by that visit

$$\widetilde{\mathcal{E}}_{t,i} = \Psi_{\mathrm{con}}\left(x,\, g_t,\, d_{t,i},\, e_{t,i},\, \widetilde{s}_t\right), \tag{5}$$

where Ψ_{con} denotes the residual extractor and $\widetilde{\mathcal{E}}_{t,i}$ the resulting candidate residuals. A candidate is added to $\mathcal{E}_t$ if it is task-relevant, is not covered by $e_{t,i}$ or the current WasteHub state $\widetilde{s}_t$, and is extracted from the visited page content. Each stored residual retains its provenance.

Source-appraisal residuals. After each successful visit, WasteHub extracts task-relevant source-applicability information from the available source information and visit summary

$$\widetilde{\mathcal{A}}_{t,i} = \Psi_{\mathrm{app}}\left(x,\, u_{t,i},\, m_{t,i},\, e_{t,i}\right), \tag{6}$$

where Ψ_{app} denotes the Source-appraisal extractor and $\widetilde{\mathcal{A}}_{t,i}$ the resulting candidate records. It captures task-relevant usage conditions such as temporal scope, geographic scope, and definitional conventions. Each resulting record is keyed to the visited source, retains its source identifier, and is accumulated in $\mathcal{A}_t$ for subsequent reuse.

2.2.2 Stage-Aware Gate

The three residual types are used under stage-specific constraints. Let $\ell \in \{\mathrm{front}, \mathrm{goal}, \mathrm{app}, \mathrm{syn}, \mathrm{bd}\}$ denote frontier guidance, same-goal evidence reuse, appraisal guidance, pre-finalization synthesis, and source-linked scope annotation, respectively. Let $\kappa_{\ell,t}$ denote the context of the current stage. WasteHub first forms a provenance-preserving, type-valid pool and then selects aligned and novel residuals

$$\begin{aligned} \mathcal{V}_{\ell,t} &= \{r \in \mathcal{H}_t \mid \mathrm{Prov}(r) \wedge \mathrm{Valid}_\ell(r)\}, \\ \mathcal{R}_{\ell,t} &= \mathrm{Select}_\ell^{K_\ell}\left(\{r \in \mathcal{V}_{\ell,t} \mid \mathrm{Align}_\ell(r; \kappa_{\ell,t}) \wedge \mathrm{Novel}_\ell(r; \kappa_{\ell,t})\}\right). \end{aligned} \tag{7}$$

Here, $\mathrm{Prov}(r)$ requires a nonempty source identifier carried with the residual. The interface treats this identifier as immutable and copies it to every delivered form of the residual. In addition, $\mathrm{Valid}_\ell(r)$ admits only the residual type and information status allowed at stage ℓ. Writing $\tau(r) \in \{\mathsf{F}, \mathsf{E}, \mathsf{A}\}$ for the residual type, $\mathrm{Valid}_\ell(r)$ requires $\tau(r) \in \mathcal{T}_\ell$, where $\mathcal{T}_{\mathrm{front}} = \{\mathsf{F}\}$, $\mathcal{T}_{\mathrm{goal}} = \mathcal{T}_{\mathrm{syn}} = \{\mathsf{E}\}$, and $\mathcal{T}_{\mathrm{app}} = \mathcal{T}_{\mathrm{bd}} = \{\mathsf{A}\}$. The context $\kappa_{\ell,t}$ contains the current decision inputs, against which Align_ℓ checks relevance and Novel_ℓ excludes information already covered. $\mathrm{Select}_\ell^{K_\ell}$ retains at most K_ℓ residuals.

Frontier-guided exploration. Before a subsequent source-selection decision, WasteHub selects Unvisited-result residuals that align with the current unresolved sub-goal and have not previously been supplied as frontier cues. They are exposed only as unverified source-selection cues; WasteHub neither treats them as factual evidence nor lets them independently trigger tool calls.

Same-goal evidence reuse. After a visit, WasteHub selects residual evidence from $\mathcal{E}_{t-1}$ that aligns with the current sub-goal g_t, is not covered by the current visit summaries, and comes from a different source. Let $E_t^{\mathrm{obs}} = \{e_{t,i}\}_{i \in I_t}$ denote the current visit summaries and $U_t = \{u_{t,i}\}_{i \in I_t}$ their source identifiers. The selected residuals are

$$\mathcal{R}_t^{\mathrm{goal}} = \mathrm{Select}_{\mathrm{goal}}^{K_{\mathrm{goal}}} \left(\left\{ r \in \mathcal{V}_{\mathrm{goal},t} \cap \mathcal{E}_{t-1} \;\middle|\; \mathrm{Align}_{\mathrm{goal}}(r; g_t) \wedge \neg\, \mathrm{Cov}(r, E_t^{\mathrm{obs}}) \wedge \mathrm{src}(r) \notin U_t \right\} \right), \tag{8}$$

where $\mathrm{src}(r)$ denotes the source identifier stored with residual r. Selected residuals are retained as residual content records with their original provenance.

Appraisal-guided source reuse. When the agent selects a source previously explored in the trajectory, WasteHub retrieves its source-keyed applicability record from $\mathcal{V}_{\mathrm{app},t} \cap \mathcal{A}_{t-1}$, if available. The record is presented as an applicability note only if it remains valid, matches the intended use in the current step, and adds scope information not already reflected in the current WasteHub state $\widetilde{s}_t$.

Pre-finalization synthesis. Before final report generation, WasteHub obtains a coverage sketch $\mathcal{O} = \{o_j\}_{j=1}^{M}$ from the task x and the terminal WasteHub state $\widetilde{s}_T$, where each o_j denotes a semantic topic or report slot. An existing outline is reused when available, otherwise the slots are derived from the terminal WasteHub state. Residual content extracted from visited pages that matches an existing slot and is not already covered by $\widetilde{s}_T$ is selected as

$$\mathcal{R}^{\mathrm{syn}} = \mathrm{Select}_{\mathrm{syn}}^{K_{\mathrm{syn}}} \left(\{ r \in \mathcal{V}_{\mathrm{syn},T} \cap \mathcal{E}_T \mid \exists o_j \in \mathcal{O},\, \mathrm{Match}(r, o_j) \wedge \neg\, \mathrm{Cov}(r, \widetilde{s}_T) \} \right), \tag{9}$$

where $\mathrm{Match}(r, o_j)$ indicates that residual r matches the information need of slot o_j. Only residual content extracted from visited pages is admissible at this stage.

2.2.3 Residual-Augmented State Update and Report Finalization

Let $\mathcal{R}_t^{\mathrm{front}}$ and $\mathcal{R}_t^{\mathrm{app}}$ denote the outputs of the frontier and appraisal gates. The native state update becomes

$$\widetilde{s}_{t+1} = \mathcal{U}_{\mathcal{B}} \left(\widetilde{s}_t,\, o_t^{\mathcal{B}} \oplus \mathrm{Cue}\big(\mathcal{R}_t^{\mathrm{front}}\big) \oplus \mathcal{R}_t^{\mathrm{goal}} \oplus \mathrm{Constraint}(\mathcal{R}_t^{\mathrm{app}}) \right), \tag{10}$$

where $\oplus$ appends typed residual observations without replacing $o_t^{\mathcal{B}}$. Here, $\mathrm{Cue}(\mathcal{R}_t^{\mathrm{front}})$ denotes unverified frontier cues, $\mathcal{R}_t^{\mathrm{goal}}$ denotes residual content records, and $\mathrm{Constraint}(\mathcal{R}_t^{\mathrm{app}})$ denotes typed source-applicability notes. Inactive terms are empty.

During report finalization, the coverage sketch and selected residual evidence are first supplied to the framework-specific generator to produce a report

$$\widehat{y}^{(0)} = \mathrm{Gen}_{\mathcal{B}} \left(x,\, \widetilde{s}_T,\, \mathcal{O},\, \mathcal{R}^{\mathrm{syn}} \right), \tag{11}$$

where $\mathrm{Gen}_{\mathcal{B}}$ is the framework-specific report generator and $\widehat{y}^{(0)}$ is its output before boundary annotation. The boundary gate then selects source-applicability records that align with this report

$$\mathcal{R}^{\mathrm{bd}} = \mathrm{Select}_{\mathrm{bd}}^{K_{\mathrm{bd}}} \left(\left\{ r \in \mathcal{V}_{\mathrm{bd},T} \;\middle|\; \mathrm{Align}_{\mathrm{bd}} \left(r; \widehat{y}^{(0)} \right) \wedge \mathrm{Novel}_{\mathrm{bd}} \left(r; \widehat{y}^{(0)} \right) \right\} \right). \tag{12}$$

where $\mathrm{Align}_{\mathrm{bd}}(r; \widehat{y}^{(0)})$ requires the record to concern a source cited in the draft and a scope condition applicable to one of its claims, while $\mathrm{Novel}_{\mathrm{bd}}$ excludes scope information already stated in the draft. Each selected record is rendered as a source-linked scope annotation. The finalized report is

$$\widehat{y} = \mathrm{Append}\left(\widehat{y}^{(0)}, \mathrm{Scope}(\mathcal{R}^{\mathrm{bd}})\right), \tag{13}$$

where Append adds one boundary paragraph without changing the preceding report content. If no boundary record is selected, it returns $\widehat{y}^{(0)}$ unchanged.

Appendix J derives a local risk trade-off and a direct-channel invariant for residual gating.

3 EXPERIMENT

This section presents the datasets, implementation details, and baselines used in our experiments.

3.1 DATASETS

We evaluate our method on two public report-level benchmarks for deep research agents, namely DeepResearch Bench (DRB) (Du et al., 2026) and DeepResearch Bench II (DRB-II) (Li et al., 2026a). Both benchmarks evaluate end-to-end open-web research and long-form report generation, which closely matches the search-and-visit setting studied in this work. DRB evaluates overall report and citation quality, while DRB-II provides fine-grained assessment using expert-derived criteria. Detailed dataset statistics are reported in Table 5 of **Appendix B**, while **Appendix C** provides the definitions and scoring procedures for all evaluation metrics.

3.2 IMPLEMENTATION

All methods share the same search/visit interfaces, tool backends, output formats, timeout/retry settings, content extraction pipeline, and summarization configuration. The `search` tool sends each query to a **Zhipu** search endpoint and returns candidate sources, each with a URL and available metadata such as a title, snippet, publication date, and source domain. The `visit` tool takes a webpage URL and the current research sub-goal, extracts the page content, and produces a goal-conditioned summary. The shared extraction pipeline uses, in fallback order, **Mozilla Readability**, **Jina Reader**, and **Readability** after dynamic rendering with Playwright. All methods use the same fixed summarization model, prompt, input-length limit, truncation policy, and fallback criteria. WasteHub leaves the shared tool pipeline unchanged and processes only signals not retained by the current research step. All reported token and latency metrics include method-specific processing.

The main experiments use **Qwen3.5-35B-A3B** as the agent backbone, while appendix evaluations additionally use **Qwen3.5-397B-A17B**, **Kimi-K2.5**, and **DeepSeek-V4-Flash** (**Appendix D**), all deployed on our cluster of **32 NVIDIA H800 GPUs** with 80GB memory each. Benchmark evaluation uses **GPT-5.6 Sol** through the OpenAI API. Under the same backbone setting, all methods use identical model weights, decoding hyperparameters, and context budgets.

3.3 BASELINES

We use **ReAct** (Yao et al., 2023) as the default research framework for Secs. 4.1–4.3, and evaluate **Self-Manager** (Xu et al., 2026) and **WebWeaver** (Li et al., 2026b) for cross-framework generalization (Sec. 4.4). Within each framework, we evaluate **Base**, **PRM-style feedback**, **Post-answer exploration**, and **WasteHub** using the same agent backbone, tool stack, and decoding settings:

Base. It follows the original framework for querying, source selection, and exploration.

PRM-style feedback (PRM-F). Motivated by process reward modeling for agent trajectories (Xi et al., 2026; Chae et al., 2025), a process reward model evaluates the trajectory and provides guidance on evidence sufficiency, exploration completeness, and premature conclusions.

Post-answer exploration (PAE). Motivated by test-time scaling for search agents (Ou et al., 2026), we instantiate additional test-time exploration by treating the initial report as an intermediate draft, allocating a fixed additional research budget for continued search and visit, and then revising the report with newly acquired evidence.

Table 1: Main results on DRB and DRB-II under ReAct, averaged over five independent runs. Best results are shown in bold and second-best results are underlined.

	DRB					DRB-II			
Method	**Overall** ↑	**Comp.** ↑	**Insight** ↑	**IF** ↑	**Read.** ↑	**Total** ↑	**Info.R.** ↑	**Anal.** ↑	**Pres.** ↑
Base	0.4713	0.4577	0.4614	0.4922	0.4862	0.3096	0.2419	0.3405	0.8730
+PRM-F	0.4736	0.4585	0.4657	0.4934	<u>0.4873</u>	0.3136	0.2449	0.3444	<u>0.8874</u>
+PAE	<u>0.4772</u>	<u>0.4676</u>	<u>0.4685</u>	<u>0.4957</u>	0.4857	<u>0.3219</u>	<u>0.2540</u>	<u>0.3588</u>	0.8747
+WasteHub	**0.4861**	**0.4743**	**0.4814**	**0.4968**	**0.4906**	**0.3380**	**0.2691**	**0.3772**	**0.8952**
Δ vs. 2nd Best	**↑0.0089**	**↑0.0067**	**↑0.0129**	**↑0.0011**	**↑0.0033**	**↑0.0161**	**↑0.0151**	**↑0.0184**	**↑0.0078**

WasteHub. Unlike PRM-F, which adds process-level feedback, and PAE, which acquires new evidence through extra exploration, WasteHub reuses task-relevant signals already exposed or derived during the existing search-and-visit process but not retained.

4 Experimental Results

This section evaluates WasteHub in terms of overall effectiveness, resource efficiency, component effects, and generalization across research frameworks.

4.1 Main Results

We first evaluate the overall effectiveness of WasteHub under the default ReAct framework by comparing it with the Base setting and two alternative research-enhancement strategies, PRM-F and PAE, under the DRB and DRB-II evaluation protocols. Table 1 reports the main report-quality and fine-grained research metrics. Remaining metrics from the same runs, including DRB citation metrics and the DRB-II blocked-source rate, are reported in **Appendix A**. Aggregate-metric run-to-run stability and task-level significance of WasteHub versus PAE are reported in **Appendix E**.

As shown in Table 1, WasteHub achieves the best performance on all metrics reported across DRB and DRB-II. On DRB, the gains over the second-best results are concentrated in Comprehensiveness and Insight, while the improvements in Instruction Following and Readability are relatively small. On DRB-II, the gains are also larger in Information Recall and Analysis than in Presentation. Across both benchmarks, the larger improvements occur in dimensions related to information coverage and analytical depth, consistent with WasteHub's design objective of selectively reusing task-relevant information that would otherwise not be retained.

4.2 Efficiency and Resource Usage

We further characterize the research process and resource usage of different methods on DRB and DRB-II. For each dataset, we report cumulative input tokens, the numbers of search and visit calls, the number of main-agent rounds, the character length of the final report, and the end-to-end latency from receiving the query to completing the final report, all averaged over its queries and five independent runs. As shown in Table 2, WasteHub uses the second-fewest cumulative input tokens on both benchmarks, behind only Base, and has the lowest and second-lowest end-to-end latency on DRB and DRB-II, respectively. Relative to PRM-F, it reduces input tokens by 8.4% and 6.0%, and latency by 14.1% and 7.0%; relative to PAE, the reductions are 30.3% and 18.2%, and 25.6% and 29.7%, respectively. It also uses fewer search and visit calls and fewer main-agent rounds than PRM-F and PAE on both benchmarks, while producing longer reports.

4.3 Ablation Studies

We ablate WasteHub by residual type and reuse phase. Type ablations remove Unvisited-result, Visit-content, or Source-appraisal residuals from all stages where they are used. Phase ablations keep collection fixed but restrict reuse to the research process (*Process Only*: frontier guidance, same-goal evidence reuse, and appraisal guidance) or report finalization (*Finalization Only*: pre-

Table 2: Research process and resource use on DRB and DRB-II under ReAct. Cum. Tok. and Rep. Len. are in thousands, and E2E Lat. is in seconds. Best results are shown in bold and second-best results are underlined.

Dataset	Method	Cum. Tok. ↓	E2E Lat. ↓	Search	Visit	Rounds	Rep. Len.
DRB	Base	**377.7**	2245.7	8.10	7.32	17.33	12.4
	+PRM-F	482.5	2562.1	9.53	7.40	20.68	15.2
	+PAE	634.0	2954.6	9.88	8.61	22.59	17.5
	+WasteHub	441.9	**2199.6**	8.02	7.18	18.05	18.7
DRB-II	Base	**469.7**	**2911.3**	12.36	7.36	21.13	15.0
	+PRM-F	585.3	3265.8	12.70	7.82	22.71	15.3
	+PAE	672.7	4318.7	12.80	9.30	25.57	18.9
	+WasteHub	550.2	3037.4	12.51	7.70	20.81	19.3

Table 3: Ablation of residual types and reuse phases under ReAct. Cum. Tok. is in thousands and E2E Lat. in seconds. Values are means over five independent runs. Bars are min–max normalized within each column, with longer bars indicating better values.

	DRB			DRB-II		
Variant	Overall ↑	Cum. Tok. ↓	E2E Lat. ↓	Total ↑	Cum. Tok. ↓	E2E Lat. ↓
Full	0.4861	441.9	2199.6	0.3380	550.2	3037.4
w/o Unvisited-result	0.4821	431.4	2261.8	0.3312	537.9	2995.2
w/o Visit-content	0.4778	405.9	2076.4	0.3227	506.0	2858.7
w/o Source-appraisal	0.4829	436.5	2183.1	0.3328	543.8	3018.4
Process Only	0.4809	426.9	2128.5	0.3293	531.6	2941.3
Finalization Only	0.4800	410.6	2178.9	0.3284	514.8	2992.8
Base	0.4713	377.7	2245.7	0.3096	469.7	2911.3

finalization synthesis and source-linked scope annotation). We report DRB Overall, DRB-II TotalScore, cumulative input tokens, and end-to-end latency. Base is ReAct without WasteHub.

As shown in Table 3, the ablation results reveal contributions at both the residual-type and reuse-phase levels. For residual-type ablations, removing any of the three residual types reduces aggregate performance on both benchmarks, indicating that each type contributes to WasteHub. Among them, Visit-content has the largest effect, with its removal causing the largest decrease in both DRB Overall and DRB-II TotalScore. All three type ablations use fewer cumulative input tokens than Full, while end-to-end latency does not decrease consistently across the two benchmarks. For reuse-phase ablations, both Process Only and Finalization Only remain above Base but below Full on both benchmarks. This indicates that both reuse phases contribute, while their combination yields the strongest aggregate performance. Both phase ablations also use fewer cumulative input tokens and have lower end-to-end latency than Full on both benchmarks. **Appendix G** reports finer stage-level ablations, and **Appendix I** a noise ablation with random residual selection.

4.4 Cross-Framework Generalization

To test whether WasteHub's gains extend beyond the default ReAct setting, we evaluate it with Self-Manager and WebWeaver. Within each framework, Base, PRM-F, PAE, and WasteHub use the same backbone, tools, and evaluation protocol. Figure 3 reports absolute score changes from Base on DRB Overall and DRB-II TotalScore.

WasteHub improves over Base in all six framework-benchmark combinations, with gains from 0.0112 to 0.0148 on DRB and from 0.0169 to 0.0284 on DRB-II. It also exceeds the better-performing of PRM-F and PAE in every setting, by 0.0057 to 0.0091 on DRB and by 0.0109 to 0.0221 on DRB-II. PRM-F falls below Base on DRB-II for Self-Manager and WebWeaver, while PAE falls below Base on DRB for WebWeaver. This pattern suggests that WasteHub provides more robust gains across research frameworks than PRM-F and PAE within the evaluated settings.



Figure 3: Cross-framework results on DRB and DRB-II. Bars show absolute score changes from the Base score reported in each panel title.

5 Related Work

Deep research agents conduct multi-round planning, web search, reading, and evidence synthesis to produce long-form, citation-supported reports (Li et al., 2026b; Xu et al., 2026). Report-level benchmarks evaluate the final report and its use of evidence (Du et al., 2026; Li et al., 2026a), while work on citation-supported generation focuses on output verifiability (Gao et al., 2023). Neighboring information-seeking tasks typically target bounded answers in multi-hop question answering, general-assistant, and web-traversal settings (Yang et al., 2018; Mialon et al., 2024; Wu et al., 2025). Deep research uses similar search and reasoning operations but must organize evidence collected over an extended research process into a coherent report. The distinction lies in output scope and evidence organization rather than retrieval itself.

At the method level, RAG conditions generation on retrieved passages and DPR retrieves supporting passages for open-domain question answering (Lewis et al., 2020; Karpukhin et al., 2020), while chain-of-thought supports multi-step reasoning (Wei et al., 2022). Agent methods interleave reasoning with retrieval or tool use, or learn when and how to invoke APIs (Trivedi et al., 2023; Yao et al., 2023; Press et al., 2023; Schick et al., 2023); Reflexion retains verbal feedback across trials (Shinn et al., 2023), and web-agent research studies interaction with realistic sites (Deng et al., 2023; Zhou et al., 2024). FLARE and Self-RAG retrieve adaptively (Jiang et al., 2023; Asai et al., 2024), RECOMP compresses retrieved documents (Xu et al., 2024), and Search-o1 refines them before use (Li et al., 2025). Together, these methods determine when external information is acquired and what reaches the current step. Such selectivity is particularly relevant because models do not use long contexts reliably (Liu et al., 2024).

Report-oriented agents manage information across long trajectories. WebWeaver co-evolves an outline and evidence memory and retrieves evidence by section (Li et al., 2026b); Self-Manager coordinates parallel threads with isolated contexts (Xu et al., 2026); persistent workspaces retain notes and raw sources beyond the context window (Zhu et al., 2026). We study a retention boundary: task-relevant candidates, page content, and source-applicability information exposed or derived during search and visit but omitted from the updated state. WasteHub stores these signals as typed, provenance-preserving residuals and selectively reuses them in research and report finalization.

6 Conclusions

We study task-relevant signals exposed or derived during search and visit interactions but not retained in the updated research state. To reuse these signals, we introduce WasteHub, a plug-and-play mechanism that stores them as three provenance-preserving residual types (Unvisited-result, Visit-content, and Source-appraisal) and selectively reuses them through a stage-aware gate. Under ReAct, WasteHub achieves higher aggregate scores than Base, PRM-F, and PAE on DRB and DRB-II. Removing any residual type or either reuse phase lowers aggregate scores on both benchmarks. With Self-Manager and WebWeaver, WasteHub likewise achieves higher aggregate scores than Base, PRM-F, and PAE on both benchmarks.

AI USE STATEMENT

Outside the generative models used as components of our experimental systems and evaluation pipeline, generative AI tools were used only to assist with literature search and discovery, including identifying potentially relevant related work, and to polish writing for improved readability, grammar, and phrasing. For author-side research activities, generative AI tools were not used for the development of theoretical or conceptual frameworks, formulation of hypotheses or mathematical claims, research methodology or experimental design, method implementation, data generation or analysis, or interpretation of experimental results. All literature identified with AI assistance was independently verified against the original sources, and all AI-assisted edits were reviewed by the authors. We take full responsibility for the final content of this work, including its text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide an anonymous repository containing a reference implementation of WasteHub on ReAct, the shared search/visit tool stack, and the DRB and DRB-II evaluation pipelines: `https://anonymous.4open.science/r/WasteHub-D742`. All materials are anonymized in accordance with the ICLR double-blind review requirements.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *Proceedings of the International Conference on Learning Representations*, 2024.

Hyungjoo Chae, Seonghwan Kim, Junhee Cho, Seungone Kim, Seungjun Moon, Gyeom Hwangbo, Dongha Lim, Minjin Kim, Yeonjun Hwang, Minju Gwak, Dongwook Choi, Minseok Kang, Gwanhoon Im, ByeongUng Cho, Hyojun Kim, Jun Han, Taeyoon Kwon, Minju Kim, Beongwoo Kwak, Dongjin Kang, and Jinyoung Yeo. Web-shepherd: Advancing prms for reinforcing web agents. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Proceedings of the Conference on Neural Information Processing Systems*, volume 36, pp. 28091–28114, 2023.

Yao Dong, Xinglin Xiao, Liwei Dong, Xinlong Jin, Zhengbo Li, Heng Zhang, Duyun Wang, and Nan Xu. S1-deepresearch: Beyond search, toward real-world long-horizon research agents. *arXiv preprint arXiv:2606.15367*, 2026.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Licheng Zhang, Xiaorui Wang, and Zhendong Mao. DeepResearch Bench: A comprehensive benchmark for deep research agents. In *The International Conference on Learning Representations*, 2026.

Wei Fan, Wenlin Yao, Zheng Li, Feng Yao, Xin Liu, Liang Qiu, Qingyu Yin, Yangqiu Song, and Bing Yin. DeepPlanner: Scaling planning capability for deep research agents via advantage shaping. In *Findings of the Association for Computational Linguistics*, pp. 7510–7525, 2026.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, 2023.

Yusong Hu, Runmin Ma, Yue Fan, Jinxin Shi, Zongsheng Cao, Yuhao Zhou, Jiakang Yuan, Shuaiyu Zhang, Shiyang Feng, Xiangchao Yan, Shufei Zhang, Wenlong Zhang, Lei Bai, and Bo Zhang. FlowSearch: Advancing deep research with dynamic structured knowledge flow. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 21212–21245, 2026.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, volume 1, pp. 1658–1677. Association for Computational Linguistics, 2024.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, 2023.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, 2020.

Dayoon Ko, Jihyuk Kim, Haeju Park, Sohyeon Kim, Dahyun Lee, Yongrae Jo, Gunhee Kim, Moontae Lee, and Kyungjae Lee. Hybrid deep searcher: Scalable parallel and sequential search reasoning. In *Proceedings of the International Conference on Learning Representations*, volume 2026, pp. 38580–38616, 2026.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the Conference on Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Ruizhe Li, Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. DeepResearch Bench II: Diagnosing deep research agents via rubrics from expert report. *arXiv preprint arXiv:2601.08536*, 2026a.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, 2025.

Zijian Li, Xin Guan, Bo Zhang, Shen Huang, Houquan Zhou, Shaopeng Lai, Ming Yan, Yong Jiang, Pengjun Xie, Fei Huang, Jun Zhang, and Jingren Zhou. WebWeaver: Structuring web-scale evidence with dynamic outlines for open-ended deep research. In *Proceedings of the International Conference on Learning Representations*, 2026b.

Xiaoqiang Lin, Jun Hao Liew, Silvio Savarese, and Junnan Li. W&d: Scaling parallel tool calling for efficient deep research agents. *arXiv preprint arXiv:2602.07359*, 2026.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. In *Proceedings of the International Conference on Learning Representations*, 2024.

Litu Ou, Kuan Li, Huifeng Yin, Liwen Zhang, Zhongwang Zhang, Xixi Wu, Rui Ye, Zile Qiao, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Browseconf: Confidence-guided test-time scaling for web agents. In *Findings of the Association for Computational Linguistics*, pp. 446–465, 2026.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics*, pp. 5687–5711, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Proceedings of the Conference on Neural Information Processing Systems*, volume 36, pp. 68539–68551, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Proceedings of the Conference on Neural Information Processing Systems*, volume 36, pp. 8634–8652, 2023.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 10014–10037, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the Conference on Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022.

Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yulan He, Deyu Zhou, Pengjun Xie, and Fei Huang. WebWalker: Benchmarking LLMs in web traversal. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 10290–10305, 2025.

Zhiheng Xi, Chenyang Liao, Guanyu Li, Zhihao Zhang, Wenxiang Chen, Binghai Wang, Senjie Jin, Yuhao Zhou, Jian Guan, Wei Wu, Tao Ji, Tao Gui, Qi Zhang, and Xuanjing Huang. Agentprm: Process reward models for llm agents via step-wise promise and progress. In *Proceedings of the ACM Web Conference*, pp. 4184–4195, 2026.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. RECOMP: improving retrieval-augmented LMs with context compression and selective augmentation. In *Proceedings of the International Conference on Learning Representations*, 2024.

Yilong Xu, Zhi Zheng, Xiang Long, Yujun Cai, and Yiwei Wang. Self-manager: Parallel agent loop for long-form deep research. *arXiv preprint arXiv:2601.17879*, 2026.

Lingyong Yan, Can Xu, Yukun Zhao, Wenxuan Li, Qingyang Chen, Jiulong Wu, Wenli Song, Xiangnan Li, Weixian Shi, Yiqun Chen, Xuchen Ma, Yuchen Li, Jiashu Zhao, Shuaiqiang Wang, Jianmin Wu, and Dawei Yin. Dumate-deepresearch: An auditable multi-agent system with recursive search and rubric-grounded reasoning. *arXiv preprint arXiv:2606.07299*, 2026.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, 2018.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *Proceedings of the International Conference on Learning Representations*, 2023.

Chanwoong Yoon, Taewhoo Lee, Hyeon Hwang, Minbyul Jeong, and Jaewoo Kang. CompAct: Compressing retrieved documents actively for question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 21424–21439. Association for Computational Linguistics, 2024.

Zhen Zhang, Liangcai Su, Zhuo Chen, Xiang Lin, Haotian Xu, Simon Shaolei Du, Kaiyu Yang, Bo An, Lidong Bing, and Xinyu Wang. Argus: Evidence assembly for scalable deep research agents. *arXiv preprint arXiv:2605.16217*, 2026.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *Proceedings of the International Conference on Learning Representations*, 2024.

Chiwei Zhu, Benfeng Xu, Mingxuan Du, Shaohan Wang, Xiaorui Wang, Zhendong Mao, and Yongdong Zhang. FS-researcher: Test-time scaling for long-horizon research tasks with file-system-based agents. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 6353–6373, 2026.

APPENDIX CONTENTS

Appendix	Contents	Page
A	Remaining Metrics from the Main Evaluation	13
B	Dataset Statistics	13
C	Evaluation Metrics	14
D	Additional Backbone Results	16
E	Stability and Significance	17
F	Budget Sensitivity	18
G	Fine-Grained Stage Ablation	19
H	Effect of PAE Exploration Strength	21
I	Noise Ablation for Residual Selection	22
J	Properties of Residual Gating	24
K	Shared Execution Configuration	26
L	Cross-Evaluator Consistency	26
M	Case Study	27

A REMAINING METRICS FROM THE MAIN EVALUATION

We report the remaining metrics from the same evaluation runs used for the main results in Table 4. These metrics are moved to the appendix only for a more compact presentation of the main table. For DRB, Citation Accuracy (C.Acc.) and Effective Citations (Eff.C.) measure citation quality. For DRB-II, Blocked Rate (B.Rate) records the proportion of rubric judgments affected by blocked source articles. Higher values are better for C.Acc. and Eff.C., while lower B.Rate indicates fewer affected rubric judgments. Detailed metric definitions are provided in Appendix C.

Table 4: Remaining metrics from the main evaluation.

Method	C.Acc. (%)	Eff.C.	B.Rate
Base	56.90	3.52	0.0016
+PRM-F	54.15	7.24	0.0017
+PAE	48.78	6.67	0.0008
+WasteHub	63.95	8.19	0.0011

B DATASET STATISTICS

Table 5 summarizes the main statistics, construction basis, and evaluation protocols of the two report-level deep research benchmarks used in our experiments. DRB contains expert-authored tasks whose topic distribution follows 44,019 deep research queries filtered from 96,147 real-world queries. It evaluates report quality with four RACE dimensions and citation quality with FACT. DRB-II derives its tasks and evaluation criteria from expert-written reports and provides 9,430 binary rubrics organized into *Information Recall*, *Analysis*, and *Presentation*. Detailed definitions and scoring procedures for the evaluation metrics are provided in Appendix C.

Table 5: Statistics of the two report-level deep research benchmarks.

Dataset	# Tasks	Zh / En	# Domains	Construction Basis	Evaluation Protocol
DRB	100	50 / 50	22	44,019 filtered queries from 96,147 raw queries	RACE (4 dims.) + FACT (2 metrics)
DRB-II	132	66 / 66	22	132 expert-written source reports	9,430 binary rubrics (3 dims.)

C Evaluation Metrics

We follow the official evaluation protocols of DeepResearch Bench (DRB) and DeepResearch Bench II (DRB-II). DRB uses RACE to evaluate report quality and FACT to evaluate citation quality. DRB-II uses fine-grained rubrics derived from expert reports to assess Information Recall, Analysis, and Presentation. The metrics used by DRB and DRB-II are summarized in Tables 6 and 7, respectively.

RACE scoring. For each task q, the evaluator generates task-specific rubrics and scores both the generated (target) report and a high-quality reference report against them. For a report $a \in \{\mathrm{tgt}, \mathrm{ref}\}$, let $s^a_{q,d,r} \in [0, 10]$ denote the score of rubric r under dimension d, and let $\alpha_{q,d,r}$ denote its normalized weight, shared across the two reports. The weighted score of report a on dimension d, and the corresponding report-level aggregate, are

$$S^a_{q,d} = \sum_{r \in \mathcal{R}_{q,d}} \alpha_{q,d,r} s^a_{q,d,r}, \qquad S^a_q = \sum_{d \in \mathcal{D}} \beta_{q,d} S^a_{q,d}, \tag{14}$$

with $\sum_{r \in \mathcal{R}_{q,d}} \alpha_{q,d,r} = 1$ and $\sum_{d \in \mathcal{D}} \beta_{q,d} = 1$, where $\mathcal{D}$ contains Comprehensiveness, Insight, Instruction Following, and Readability.

RACE is reference-based: each dimension and the overall quality are reported relative to the reference report. The reported dimension score and overall RACE score are

$$\begin{aligned} \tilde{S}_{q,d} &= \frac{S^{\mathrm{tgt}}_{q,d}}{S^{\mathrm{tgt}}_{q,d} + S^{\mathrm{ref}}_{q,d}}, \\ S^{\mathrm{RACE}}_q &= \frac{\sum_{d \in \mathcal{D}} \beta_{q,d} S^{\mathrm{tgt}}_{q,d}}{\sum_{d \in \mathcal{D}} \beta_{q,d} S^{\mathrm{tgt}}_{q,d} + \sum_{d \in \mathcal{D}} \beta_{q,d} S^{\mathrm{ref}}_{q,d}}. \end{aligned} \tag{15}$$

A value of 0.5 indicates parity with the reference report. The rubric weights $\alpha_{q,d,r}$ and dimension weights $\beta_{q,d}$ are generated for each task. Note that S^{RACE}_q is a ratio of weighted sums and is therefore not a weighted average of the per-dimension scores $\tilde{S}_{q,d}$.

FACT scoring. FACT extracts and deduplicates claim–URL pairs from each report and verifies whether each cited source supports its corresponding claim. Let N^{verified}_q denote the number of citations that are successfully verified for task q, and let $N^{\mathrm{supported}}_q$ denote the number that support their associated claims. Citation Accuracy is defined as

$$\mathrm{CitationAccuracy}_q = \frac{N^{\mathrm{supported}}_q}{N^{\mathrm{verified}}_q}. \tag{16}$$

Effective Citations is defined as

$$\mathrm{EffectiveCitations}_q = N^{\mathrm{supported}}_q. \tag{17}$$

Citations that cannot be verified are excluded from the denominator of Citation Accuracy. Dataset-level results are aggregated across tasks following the official evaluation protocol. For reporting, Citation Accuracy is multiplied by 100 and shown as a percentage, while Effective Citations is reported as the corresponding citation count.

DRB-II scoring. DRB-II evaluates each report using atomic and verifiable rubrics derived from expert-written reports. Each rubric belongs to Information Recall, Analysis, or Presentation. For each rubric r, the evaluator assigns a label $y_{q,r} \in \{-1, 0, 1\}$. A label of 1 indicates that the rubric is satisfied, 0 indicates that it is not satisfied, and -1 indicates that the relevant content depends on a blocked source article. Only rubrics assigned 1 are counted as passed.

Let $\mathcal{R}_{q,d}$ denote the rubrics associated with dimension d for task q. The dimension score is

$$S^{\mathrm{DRB-II}}_{q,d} = \frac{1}{|\mathcal{R}_{q,d}|} \sum_{r \in \mathcal{R}_{q,d}} \mathbb{I}(y_{q,r} = 1). \tag{18}$$

Table 6: Evaluation metrics used by DeepResearch Bench.

Metric	Definition
RACE	
Comprehensiveness	Coverage of the key topics and important aspects required by the task.
Insight	Depth and value of the report's analysis, explanations, and conclusions.
Instruction Following	Degree to which the report satisfies the requested questions, scope, and output requirements.
Readability	Clarity of the report's structure, language, and information presentation.
Overall	Reference-normalized aggregation of the four RACE dimensions, scoring the target report's weighted quality relative to the reference report.
FACT	
Citation Accuracy	Proportion of verified citations whose sources support the corresponding claims.
Effective Citations	Number of verified citations whose sources support the corresponding claims.

Table 7: Evaluation metrics used by DeepResearch Bench II.

Metric	Definition
Information Recall	Ability to identify information needs and retrieve the required information accurately and sufficiently from the open web.
Analysis	Ability to synthesize evidence into comparisons, trends, explanations, and conclusions.
Presentation	Ability to organize and communicate research findings clearly, coherently, and verifiably.
TotalScore	Proportion of all rubrics that are judged to be satisfied.
Blocked Rate	Proportion of rubric judgments for which the relevant content depends on a blocked source article.

Let $\mathcal{R}_q$ denote all rubrics for task q. The TotalScore is

$$S_q^{\text{Total}} = \frac{1}{|\mathcal{R}_q|} \sum_{r \in \mathcal{R}_q} \mathbb{I}(y_{q,r} = 1). \tag{19}$$

Blocked Rate records the fraction of rubric judgments assigned the blocked-source label:

$$B_q^{\text{blocked}} = \frac{1}{|\mathcal{R}_q|} \sum_{r \in \mathcal{R}_q} \mathbb{I}(y_{q,r} = -1). \tag{20}$$

Dataset-level results are aggregated across tasks following the benchmark evaluation procedure. Although DRB-II scores can equivalently be expressed as percentages by multiplying them by 100, throughout this paper we report TotalScore, Information Recall, Analysis, Presentation, and Blocked Rate as proportions in $[0, 1]$. This convention is used consistently in both the main results and the appendix.

Table 8: Results with three additional agent backbones under ReAct, averaged over five independent runs. C.Acc., Eff.C., and B.Rate denote Citation Accuracy, Effective Citations, and Blocked Rate, respectively. C.Acc. is reported as a percentage. Best and second-best results are shown in bold and underlined, respectively, for the DRB RACE and DRB-II quality metrics.

	DRB							DRB-II				
Method	**Overall**	**Comp.**	**Insight**	**IF**	**Read.**	**C.Acc.**	**Eff.C.**	**Total**	**Info.R.**	**Anal.**	**Pres.**	**B.Rate**
Qwen3.5-397B-A17B												
Base	0.4964	0.4925	0.4936	0.5042	0.5011	66.85	26.55	0.3269	0.2601	0.3599	0.8789	0.0012
+PRM-F	0.4962	0.4922	0.4924	0.5047	0.5033	62.37	22.27	0.3168	0.2495	0.3475	0.8771	0.0010
+PAE	0.5062	0.5032	0.5049	0.5140	0.5048	65.00	22.30	0.3280	0.2607	0.3647	0.8753	0.0005
+WasteHub	**0.5196**	**0.5188**	**0.5250**	**0.5217**	**0.5077**	66.88	28.85	**0.3402**	**0.2724**	**0.3824**	**0.8792**	0.0008
Kimi-K2.5												
Base	0.5023	0.4979	0.4990	0.5071	0.5090	73.50	15.04	0.3325	0.2640	0.3640	0.9022	0.0015
+PRM-F	0.5004	0.4958	0.4967	0.5080	0.5075	71.20	13.88	0.3302	0.2635	0.3615	0.8834	0.0015
+PAE	0.5110	0.5079	0.5091	0.5193	0.5094	70.38	15.30	0.3360	0.2671	0.3711	0.9012	0.0011
+WasteHub	**0.5216**	**0.5093**	**0.5374**	**0.5205**	**0.5098**	75.52	17.67	**0.3522**	**0.2834**	**0.3931**	**0.9029**	0.0014
DeepSeek-V4-Flash												
Base	0.5137	0.5085	0.5165	0.5192	0.5055	78.19	15.06	0.3362	0.2688	0.3688	0.8933	0.0010
+PRM-F	0.5145	0.5082	0.5182	0.5170	0.5090	80.50	15.25	0.3350	0.2674	0.3664	0.8960	0.0014
+PAE	0.5164	0.5108	0.5198	0.5195	0.5088	77.90	19.88	0.3397	0.2708	0.3758	**0.9021**	0.0015
+WasteHub	**0.5238**	**0.5224**	**0.5315**	**0.5228**	**0.5120**	81.16	22.54	**0.3538**	**0.2854**	**0.3944**	0.9015	0.0009

D Additional Backbone Results

We repeat the ReAct comparison with Qwen3.5-397B-A17B, Kimi-K2.5, and DeepSeek-V4-Flash. All results in Table 8 are averaged over five independent runs using the same evaluation setup as the main experiments.

Across the three additional backbones, WasteHub achieves the highest DRB Overall and DRB-II TotalScore. Relative to Base, the improvements range from 0.0101 to 0.0232 on DRB and from 0.0133 to 0.0197 on DRB-II. PAE is the stronger of the two enhancement baselines on both aggregate metrics for each backbone. Relative to PAE, WasteHub improves DRB Overall by 0.0074 to 0.0134 and DRB-II TotalScore by 0.0122 to 0.0162.

The dimension-level results are consistent with the pattern in the main ReAct experiment. On DRB, the differences between WasteHub and PAE are larger in Comprehensiveness and Insight than in Instruction Following and Readability for all three backbones. On DRB-II, WasteHub has higher Information Recall and Analysis than PAE in all three settings, while Presentation changes little and is slightly lower with DeepSeek-V4-Flash. WasteHub also has higher Citation Accuracy and Effective Citations than PAE for each backbone. Thus, the aggregate advantage of WasteHub observed in the main ReAct experiment is also present with the three additional backbones tested here, although the size of the difference varies across backbones.

Table 9: Run stability under the main ReAct configuration. Results are reported as mean ± sample standard deviation over five independent runs.

Method	DRB Overall	DRB-II TotalScore
Base	0.4713 ± 0.0020	0.3096 ± 0.0023
+PRM-F	0.4736 ± 0.0024	0.3136 ± 0.0028
+PAE	0.4772 ± 0.0027	0.3219 ± 0.0030
+WasteHub	$\mathbf{0.4861 \pm 0.0022}$	$\mathbf{0.3380 \pm 0.0026}$

Table 10: Task-level paired significance analysis of WasteHub against PAE. Δ denotes the mean aggregate-score difference between WasteHub and PAE.

Dataset	Comparison	Δ	95% CI	p
DRB	WasteHub vs. PAE	$+0.0089$	$[0.0020, 0.0157]$	0.028
DRB-II	WasteHub vs. PAE	$+0.0161$	$[0.0065, 0.0256]$	0.004

E Stability and Significance

The main results in Table 1 are averaged over five independent runs. For each method, the model weights, decoding configuration, search and visit backends, method-specific research budget, and evaluation protocol are held fixed across runs. To further quantify run-to-run variation, Table 9 reports the mean and sample standard deviation for the primary aggregate metric of each benchmark.

Across the five runs, all methods exhibit relatively limited variation, while WasteHub maintains the highest mean aggregate score on both benchmarks. Compared with PAE, the strongest baseline on the aggregate metrics, WasteHub improves DRB Overall by 0.0089 and DRB-II TotalScore by 0.0161 on average.

We further assess the difference between WasteHub and PAE at the task level. For each task, we first average its score over the five runs and then perform paired comparisons on the same benchmark tasks. We estimate 95% confidence intervals using paired bootstrap resampling and evaluate statistical significance using a two-sided paired permutation test. The results are reported in Table 10.

The 95% confidence intervals do not include zero on either benchmark, and both two-sided tests yield $p < 0.05$. Together with the limited run-to-run variation in Table 9, these results indicate that the aggregate performance gains of WasteHub over PAE are stable across independent runs and statistically significant in the task-level paired analysis.

F BUDGET SENSITIVITY

We examine whether WasteHub depends sensitively on the stage-wise selection budgets used by the stage-aware gate. These budgets are denoted by K_ℓ in Eq. (7), where $\ell \in \{\mathrm{front}, \mathrm{goal}, \mathrm{app}, \mathrm{syn}, \mathrm{bd}\}$. We study the four stages that involve a multi-record capacity tradeoff. K_{front} limits the unverified frontier cues supplied before a later source-selection decision. K_{goal} limits the cross-source evidence reused after a visit when it matches the current sub-goal. K_{syn} limits the number of Visit-content residuals selected for pre-finalization synthesis, while K_{bd} limits the source-applicability records used for source-linked scope annotation. In our implementation, appraisal guidance is source-specific and retrieves at most one matched applicability record. We therefore fix $K_{\mathrm{app}} = 1$.

The main ReAct experiments use the same budget configuration on DRB and DRB-II,

$$(K_{\mathrm{front}}, K_{\mathrm{goal}}, K_{\mathrm{syn}}, K_{\mathrm{bd}}) = (2, 2, 4, 2).$$

We use small fixed caps to keep the auxiliary residual context bounded. The frontier, same-goal, and boundary stages use a capacity of two, while pre-finalization synthesis uses a capacity of four because it aggregates residual content extracted from visited pages across the research trajectory. These values are fixed implementation choices rather than values selected from the sensitivity sweeps. The analysis below therefore evaluates local robustness around the main configuration.

We vary one budget at a time while keeping all remaining settings fixed. The tested values are $K_{\mathrm{front}}, K_{\mathrm{goal}}, K_{\mathrm{bd}} \in \{1, 2, 3, 4, 5\}$ and $K_{\mathrm{syn}} \in \{1, 2, 4, 6, 8\}$. Because the four sweeps share the same default configuration, the 20 plotted sweep points correspond to 17 unique configurations. Each configuration is evaluated under the main ReAct setting with five independent runs on all 100 DRB tasks and all 132 DRB-II tasks. The default configuration obtains 0.4861 ± 0.0022 DRB Overall and 0.3380 ± 0.0026 DRB-II TotalScore.

Figure 4 reports the relative change from the default configuration,

$$\Delta_m(K) = 100 \times \frac{\overline{S}_m(K) - \overline{S}_m(K^{\mathrm{def}})}{\overline{S}_m(K^{\mathrm{def}})},$$

where $\overline{S}_m$ denotes the mean score over five independent runs. Error bars show one sample standard deviation under the same normalization.

The default configuration lies in a stable region for all four budgets. Increasing K_{front}, K_{goal}, or K_{bd} from 2 to 3 changes the relative score on either benchmark by no more than 0.12%. Increasing K_{syn} from 4 to 6 changes DRB and DRB-II by $+0.02\%$ and $+0.15\%$, respectively. These changes are smaller than the observed run-to-run variation, indicating that the default configuration does not correspond to a sharp isolated optimum.

The largest observed degradation occurs when the pre-finalization budget is reduced to $K_{\mathrm{syn}} = 1$, which decreases DRB and DRB-II by 0.76% and 2.07%, respectively. Among the remaining budgets, K_{goal} shows the largest variation on DRB-II, where $K_{\mathrm{goal}} = 1$ and $K_{\mathrm{goal}} = 5$ reduce TotalScore by 1.12% and 1.24%. The largest absolute deviations over the tested ranges of K_{front} and K_{bd} are 0.62% and 0.38%, respectively. Overall, very small budgets reduce the available residual capacity, while increasing the budgets beyond the default provides no consistent additional gain. WasteHub therefore remains stable over a neighborhood of the shared default configuration.

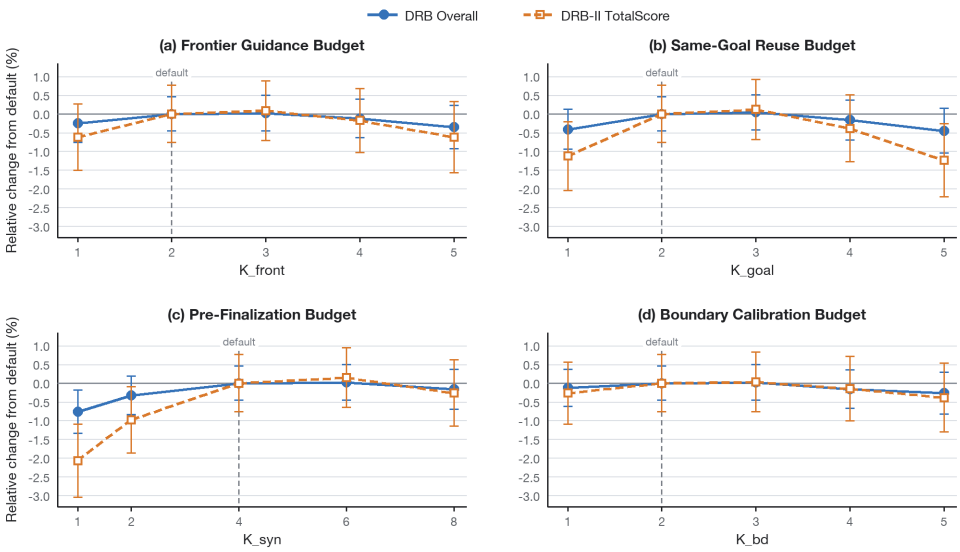

Figure 4: Sensitivity of WasteHub to four stage-specific selection budgets. Each panel varies one budget while keeping all other settings fixed. The vertical dashed line marks the default value. The vertical axis reports the relative change from the default configuration. Points show the mean over five independent runs, and error bars show one sample standard deviation under the same normalization.

G Fine-Grained Stage Ablation

The residual-type ablations in Table 3 remove every use of one residual type, while Process Only and Finalization Only compare two broad reuse phases. We further evaluate the five reuse stages by disabling one gate at a time. The residual-collection mechanism remains enabled and unchanged, the other four gates retain their default budgets, and unused capacity is not reassigned. Disabling Goal or Synthesis preserves the other use of Visit-content residuals. For w/o Synthesis, only $\mathcal{R}^{\mathrm{syn}}$ is suppressed, while the coverage sketch $\mathcal{O}$ is retained unchanged. Disabling Appraisal or Boundary likewise preserves the other use of source-applicability records. Since Unvisited-result residuals are admissible only at the Frontier gate, w/o Frontier Gate is identical to w/o Unvisited-result in Table 3 and is included here for completeness. Each difference from Full is a conditional end-to-end effect with the other four gates active, and the five differences are not additive.

We follow the main ReAct protocol and evaluate all 100 DRB tasks and all 132 DRB-II tasks with five independent runs. Table 11 reports aggregate performance, cumulative input tokens, and end-to-end latency.

Removing any gate lowers the aggregate score on both datasets, although the magnitude varies by stage. Removing Goal produces the largest decrease, reducing DRB Overall by 0.0058 and DRB-II TotalScore by 0.0116. Removing Synthesis yields the next-largest decreases of 0.0051 and 0.0093, followed by Frontier at 0.0040 and 0.0068. Goal and Synthesis therefore represent two distinct uses of Visit-content residuals, during research and before final report generation, while Frontier contributes through subsequent source-selection decisions. Removing Appraisal and Boundary produces smaller decreases of 0.0016 and 0.0015 on DRB and 0.0024 and 0.0029 on DRB-II. The Goal, Synthesis, Appraisal, and Boundary variants each remain above the corresponding residual-type ablation in Table 3, which removes both uses of the associated residual type. Frontier exactly matches w/o Unvisited-result because it is the only stage that admits Unvisited-result residuals.

Table 12 reports the usage of each gate in the full configuration. Active is the percentage of task-run pairs that receive at least one nonempty message from the gate. Sel. Rec./Run is the mean number of selected-record occurrences per task run. For process-time gates, this count is summed over all

Table 11: Fine-grained ablation of the five reuse stages under ReAct. Cum. Tok. is measured in thousands of tokens and E2E Lat. in seconds. Bars are min–max normalized within each column, with longer bars indicating better values.

	DRB			DRB-II		
Variant	Overall	Cum. Tok.	E2E Lat.	TotalScore	Cum. Tok.	E2E Lat.
Full	**0.4861**	441.9	2199.6	**0.3380**	550.2	3037.4
w/o Frontier Gate	0.4821	431.4	2261.8	0.3312	537.9	2995.2
w/o Goal Gate	0.4803	423.7	2134.2	0.3264	525.4	2924.6
w/o Appraisal Gate	0.4845	439.0	2192.3	0.3356	546.4	3024.8
w/o Synthesis Gate	0.4810	430.2	2154.8	0.3287	535.8	2968.5
w/o Boundary Gate	0.4846	438.4	2187.9	0.3351	546.8	3021.7
Base	0.4713	377.7	2245.7	0.3096	469.7	2911.3

Table 12: Usage statistics for the five reuse stages in the full WasteHub configuration, averaged over the same five independent runs. Active is the task-run-level nonempty activation rate. Sel. Rec./Run is the mean number of selected-record occurrences per task run, and Inj. Tok./Run is measured in thousands of tokens.

	DRB			DRB-II		
Stage	Active (%)	Sel. Rec./Run	Inj. Tok./Run	Active (%)	Sel. Rec./Run	Inj. Tok./Run
Frontier	73.6	3.08	1.66	78.0	3.78	2.15
Goal	86.4	4.72	5.52	89.1	5.84	7.48
Appraisal	39.2	0.55	0.33	44.1	0.64	0.44
Synthesis	80.0	2.72	2.96	84.1	3.12	3.76
Boundary	64.0	1.14	0.76	69.7	1.29	0.97

invocations; Synthesis and Boundary are invoked once per run, so their counts reflect the records selected in that single invocation. Inj. Tok./Run is the mean number of tokens in the complete typed gate messages delivered to downstream contexts or the final report, summed over all invocations and measured in thousands. It includes type labels and provenance. Inj. Tok./Run need not equal the cumulative input-token difference because the latter also includes the inputs used for stage processing and, for process-time gates, later contexts that retain an injected message.

Goal has the highest activation rate, selected-record count, and direct token volume on both datasets, consistent with its removal producing the largest aggregate decrease and resource reduction. Removing Goal decreases cumulative input tokens by 18.2k and 24.8k and latency by 65.4 and 112.8 seconds on DRB and DRB-II. Synthesis ranks second in both direct token volume and aggregate decrease. Appraisal has the lowest activation rate and direct token volume, consistent with its smaller aggregate change. Overall, the stage-level results refine the type-level ablations without changing their interpretation. Visit-content residuals support both Goal and Synthesis, Source-appraisal records support both Appraisal and Boundary, and Unvisited-result residuals are confined to Frontier.

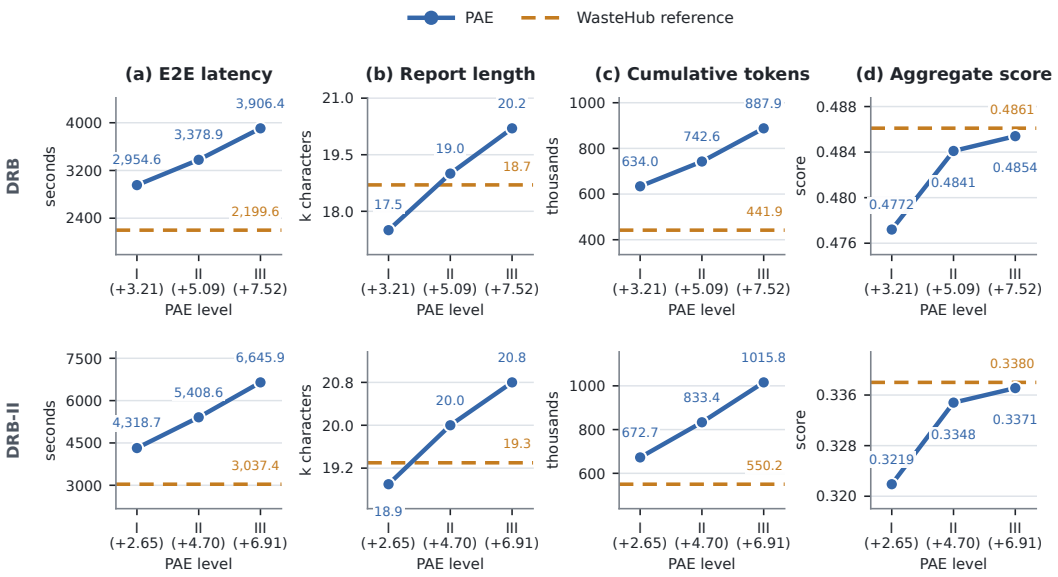

Figure 5: Effect of PAE exploration strength under ReAct on DRB and DRB-II. The upper row reports DRB and the lower row reports DRB-II. Values in parentheses below each PAE level are the mean numbers of additional exploration rounds after the initial report. Solid lines connect the three PAE levels, and dashed lines mark the WasteHub result from the main experiment. Report length is measured in thousands of characters, and cumulative input tokens are measured in thousands. Aggregate score denotes Overall on DRB and TotalScore on DRB-II. All values are means over five independent runs.

H Effect of PAE Exploration Strength

PAE treats the initial report as an intermediate draft and supplements the preceding research with additional exploration before finalization. We study how extending this process changes performance and resource use. We implement three exploration levels whose prompt templates have a maximum pairwise length difference below 50 characters. The only substantive change is the requested persistence of post-answer exploration. PAE-I is the setting used in the main paper. PAE-II and PAE-III request progressively more extensive exploration. The ReAct framework, backbone model, tool interface, context limits, and evaluation protocol remain unchanged.

An additional exploration round is one completed main-agent interaction round in the PAE stage after the initial report. This quantity covers only the PAE stage and is distinct from Avg. Rounds in Table 2, which covers the full research trajectory. PAE-I, PAE-II, and PAE-III add 3.21, 5.09, and 7.52 rounds on DRB and 2.65, 4.70, and 6.91 rounds on DRB-II. Figure 5 reports end-to-end latency, report length, cumulative input tokens, and the aggregate score. The aggregate score is Overall on DRB and TotalScore on DRB-II. All statistics follow the protocol in the main experiments and are averaged over five independent runs.

Figure 5 shows that more extensive PAE exploration consistently increases resource use. From PAE-I to PAE-III on DRB, end-to-end latency rises from 2954.6 to 3906.4 seconds, cumulative input tokens rise from 634.0k to 887.9k, and report length rises from 17.5k to 20.2k characters. The corresponding changes on DRB-II are from 4318.7 to 6645.9 seconds, from 672.7k to 1015.8k tokens, and from 18.9k to 20.8k characters. After normalization by the observed increase in exploration rounds, the two DRB-II intervals add 531.7 and 559.9 seconds per round and 78.4k and 82.5k input tokens per round. The second slope is therefore slightly larger for both resources. As a local scale check, extrapolating the first interval to zero additional rounds gives 2909.8 seconds and 465.0k tokens. These values are close to the ReAct Base results of 2911.3 seconds and 469.7k tokens in the main experiment. Report growth becomes smaller as exploration is extended. The two increments are 1.5k and 1.2k characters on DRB and 1.1k and 0.8k characters on DRB-II. Additional

exploration therefore continues to expand the report, although less new material is incorporated by later rounds.

Performance shows the same diminishing returns. DRB Overall improves from 0.4772 to 0.4841 and then reaches 0.4854. DRB-II TotalScore improves from 0.3219 to 0.3348 and then reaches 0.3371. PAE-II closes 77.5% and 80.1% of the respective score gaps between PAE-I and WasteHub. Moving from PAE-II to PAE-III adds only 0.0013 on DRB and 0.0023 on DRB-II. The PAE-III scores are within 0.0007 and 0.0009 of WasteHub, while its reports are 1.5k characters longer on both datasets. Its cumulative input tokens are 100.9% and 84.6% higher, and its latency is 77.6% and 118.8% higher.

These results show diminishing returns as post-answer exploration is extended. PAE-II already captures most of the observed score gain, while PAE-III incurs substantially higher resource use for a limited additional improvement. Similar final scores therefore do not imply similar computational efficiency. The comparison also indicates that the advantage of WasteHub cannot be explained solely by a longer exploration trajectory or a longer report.

I Noise Ablation for Residual Selection

The existing ablations show that different residual types and reuse stages contribute to WasteHub. However, they do not isolate whether the improvement comes from residual injection itself or from selecting residuals that match the current information need. We therefore introduce a noise ablation setting that replaces the stage-aware semantic selection with random residual selection.

Ablation setting. For each gate invocation at stage ℓ, we form the same stage-specific admissible residual pool as in WasteHub. The pool preserves the provenance, residual-type, temporal, and source restrictions described in the Stage-Aware Gate subsection, while excluding the semantic alignment and novelty criteria. Let this pool be denoted by $\mathcal{V}^{\mathrm{adm}}_{\ell,t}$.

Noise-WasteHub replaces the original alignment- and novelty-based selection with uniform random sampling:

$$\mathcal{R}^{\mathrm{noise}}_{\ell,t} \sim \mathrm{UniformSubset}(\mathcal{V}^{\mathrm{adm}}_{\ell,t}, m_{\ell,t}). \tag{21}$$

Here, $m_{\ell,t}$ is the number of records selected by the standard gate on the current Noise-WasteHub trajectory. At each invocation, the standard gate is evaluated only to obtain this count. Its selected identities and payloads are discarded. Noise-WasteHub then samples $m_{\ell,t}$ records from $\mathcal{V}^{\mathrm{adm}}_{\ell,t}$ without replacement. Thus the standard gate contributes its activation and selected count, but not the selected identities or payloads.

For a realized invocation, let $M_{\ell,t}$ be the records in $\mathcal{V}^{\mathrm{adm}}_{\ell,t}$ classified as aligned and novel by the standard gate, and write $n = |\mathcal{V}^{\mathrm{adm}}_{\ell,t}|$ and $a = |M_{\ell,t}|$. Since the standard gate selects only from $M_{\ell,t}$, $m_{\ell,t} \le a \le n$. Conditional on the realized pool, classification, and selected count, and treating stored records as distinct by identity, uniform sampling gives

$$\mathbb{E}\big[\big|\mathcal{R}^{\mathrm{noise}}_{\ell,t} \cap M_{\ell,t}\big| \,\big|\, \mathcal{V}^{\mathrm{adm}}_{\ell,t}, M_{\ell,t}, m_{\ell,t}\big] = m_{\ell,t}\frac{a}{n}, \tag{22}$$

for $n > 0$; when $n = 0$, both outputs are empty. The control is therefore identity-randomized conditional on the standard gate's count; matching that count does not ensure matching satisfaction of the alignment and novelty criteria. In particular, the conditional expectation in Eq. (22) is strictly below $m_{\ell,t}$ whenever $m_{\ell,t} > 0$ and $a < n$.

The random payload selection does not use alignment, novelty, or ranking scores. Residual type constraints remain unchanged. Frontier reuse samples Unvisited-result residuals, evidence reuse samples Visit-content residuals, and appraisal reuse samples Source-appraisal records. All other components, including provenance information, temporal access rules, message formatting, token limits, and tool interfaces, remain unchanged. Noise-WasteHub does not introduce additional search or visit operations.

Table 13: Noise ablation results on DRB and DRB-II under ReAct.

	DRB					DRB-II			
Method	**Overall**	**Comp.**	**Insight**	**IF**	**Read.**	**Total**	**Info.R.**	**Anal.**	**Pres.**
Base	0.4713	0.4577	0.4614	0.4922	0.4862	0.3096	0.2419	0.3405	0.8730
Noise-WasteHub	0.4709	0.4582	0.4587	0.4925	0.4860	0.3083	0.2426	0.3348	0.8747
WasteHub	**0.4861**	**0.4743**	**0.4814**	**0.4968**	**0.4906**	**0.3380**	**0.2691**	**0.3772**	**0.8952**

Table 14: Research-process and resource-usage statistics for the noise ablation under ReAct. Cum. Tok. and Rep. Len. are reported in thousands, and E2E Lat. in seconds.

Dataset	**Method**	**Cum. Tok.**	**Search**	**Visit**	**Rounds**	**Rep. Len.**	**E2E Lat.**
DRB	Base	377.7	8.10	7.32	17.33	12.4	2245.7
	Noise-WasteHub	439.6	7.98	7.15	17.94	18.0	2105.4
	WasteHub	441.9	8.02	7.18	18.05	18.7	2199.6
DRB-II	Base	469.7	12.36	7.36	21.13	15.0	2911.3
	Noise-WasteHub	548.1	12.46	7.62	20.67	18.7	2878.6
	WasteHub	550.2	12.51	7.70	20.81	19.3	3037.4

Evaluation protocol. We evaluate Base, Noise-WasteHub, and WasteHub under the default ReAct setting. All methods use the same backbone model, tool backends, decoding configuration, context budget, and evaluation procedure. Each setting is averaged over five independent runs.

As shown in Table 13, WasteHub improves over Base by 0.0148 on DRB Overall and 0.0284 on DRB-II TotalScore. Noise-WasteHub remains close to Base but does not reproduce these gains. On DRB-II, its Information Recall is slightly above Base, whereas its Analysis is below Base; on DRB, it shows the same mixed pattern across fine-grained dimensions. The gaps between WasteHub and Noise-WasteHub are 0.0152 on DRB Overall and 0.0297 on DRB-II TotalScore. Thus preserving the admissible pool, residual types, provenance constraints, and the standard gate's selected count is insufficient to reproduce WasteHub's gains. The difference is associated with the identities and payloads chosen by the standard alignment, novelty, and ranking procedure; this experiment does not separately identify these three components.

As shown in Table 14, Noise-WasteHub uses slightly fewer resources than WasteHub. It uses 2.3k and 2.1k fewer input tokens on DRB and DRB-II, respectively. Search calls decrease by 0.04 and 0.05, visit calls decrease by 0.03 and 0.08, and average rounds decrease by 0.11 and 0.14. These are endogenous trajectory statistics rather than fixed budgets. End-to-end latency is lower by 94.2 and 158.8 seconds and coincides with the shorter trajectories. Since latency also depends on model generation and tool response times, we do not attribute the difference to a single component. The performance gap is therefore not explained by an intentionally smaller configured research budget.

Overall, the noise ablation results indicate that preserving residual type, provenance, and access constraints alone is insufficient to reproduce WasteHub's improvements. They are associated with the identities and payloads retained by the standard selection procedure rather than with matched-count residual injection alone.

J PROPERTIES OF RESIDUAL GATING

We study one gate invocation at a time and do not model the downstream trajectory that follows.

Local gate model Index an invocation by $j = (t, \ell)$, where $\ell \in \{\text{front}, \text{goal}, \text{app}, \text{syn}, \text{bd}\}$. Let C_j contain the realized decision context, stage, and budget. For a process-time gate, C_j also includes the other messages passed to the same state update; these messages are held fixed in the comparison. Let V_j be the finite residual pool remaining after provenance, type, temporal, and source restrictions; it is the invocation-level counterpart of $\mathcal{V}^{\text{adm}}_{\ell,t}$ in the noise ablation. Alignment, novelty filtering, budgeted selection, and the stage-specific wrapper map (C_j, V_j) to the delivered message Q_j.

Let (Y_j, C_j, V_j) be jointly distributed over these invocations, with Y_j denoting a pre-existing target for the local decision. WasteHub does not observe Y_j, and Y_j is neither the action produced after delivery nor an end-to-end outcome. Conditional on (C_j, V_j), the gate constructs Q_j using only an independent random seed. Hence

$$Y_j \perp Q_j \mid (C_j, V_j). \tag{23}$$

This condition allows arbitrary dependence among Y_j, C_j, and V_j. All variables take values in standard Borel spaces. We assume that the Bayes envelopes are measurable, the displayed conditional information is finite, and measurable ϵ-optimal decision rules exist for every $\epsilon > 0$. Information quantities use natural logarithms, and $\|P - P'\|_{\text{TV}} = \sup_A |P(A) - P'(A)|$.

Risk trade-off Let a downstream decision take values in $\mathcal{D}$ and be evaluated by a measurable loss $L : \mathcal{Y} \times \mathcal{D} \to [0, 1]$. For an input Z, define its Bayes risk as

$$\mathcal{R}^*_L(Z) = \inf_d \ \mathbb{E}[L(Y_j, d(Z))], \tag{24}$$

where the infimum is over measurable, possibly randomized rules and the expectation includes their internal randomness. Write $Z^{\text{full}}_j = (C_j, V_j)$ and $Z^{\text{gate}}_j = (C_j, Q_j)$. We use Z^{full}_j only as a reference observation; WasteHub does not deliver the full pool. Fix an encoding for each observation and let d^{full}_θ and d^{gate}_θ be the rules induced by the same downstream model under the two interfaces. Any randomness in these rules is included in the expectation and is conditionally independent of Y_j given the corresponding input. Define

$$\begin{aligned} \mathcal{R}^b_{\theta,L} &= \mathbb{E}[L(Y_j, d^b_\theta(Z^b_j))], \\ \text{Reg}^b_{\theta,L} &= \mathcal{R}^b_{\theta,L} - \mathcal{R}^*_L(Z^b_j), \qquad b \in \{\text{full}, \text{gate}\}. \end{aligned} \tag{25}$$

Proposition J.1 (Risk under residual gating)**.** *Under Eq. (23), the Bayes penalty of replacing the full admissible pool by the gate message,*

$$D^L_j = \mathcal{R}^*_L(Z^{\text{gate}}_j) - \mathcal{R}^*_L(Z^{\text{full}}_j), \tag{26}$$

satisfies

$$0 \le D^L_j \le \min\left\{1, \sqrt{\frac{1}{2} I(Y_j; V_j \mid C_j, Q_j)}\right\}. \tag{27}$$

For this downstream model under the two fixed encodings,

$$\begin{aligned} &\mathcal{R}^{\text{full}}_{\theta,L} - \mathcal{R}^{\text{gate}}_{\theta,L} \\ &\qquad = \text{Reg}^{\text{full}}_{\theta,L} - \text{Reg}^{\text{gate}}_{\theta,L} - D^L_j. \end{aligned} \tag{28}$$

Proof. Eq. (23) makes (C_j, Q_j) a garbling of (C_j, V_j). A rule observing (C_j, V_j) can simulate the gate and then apply any rule for (C_j, Q_j), hence $D_j^L \geq 0$.

For a distribution P on Y_j, let $b(P) = \inf_{a \in \mathcal{D}} \mathbb{E}_P[L(Y_j, a)]$. Since $0 \leq L \leq 1$, its Bayes envelope is one-Lipschitz in total variation: $|b(P) - b(P')| \leq \|P - P'\|_{\mathrm{TV}}$. Using the posterior representation of Bayes risk and taking expectation over (C_j, V_j, Q_j),

$$D_j^L = \mathbb{E}\big[b(P_{Y_j|C_j,Q_j}) - b(P_{Y_j|C_j,V_j})\big]\,. \tag{29}$$

Applying the Lipschitz property to Eq. (29), followed by Pinsker's inequality and Jensen's inequality, gives

$$\begin{aligned} D_j^L &\leq \mathbb{E}\Big[\big\|P_{Y_j|C_j,V_j} - P_{Y_j|C_j,Q_j}\big\|_{\mathrm{TV}}\Big] \\ &\leq \sqrt{\frac{1}{2}\mathbb{E}\big[D_{\mathrm{KL}}\left(P_{Y_j|C_j,V_j} \,\|\, P_{Y_j|C_j,Q_j}\right)\big]} \\ &= \sqrt{\frac{1}{2} I(Y_j; V_j \mid C_j, Q_j)}, \end{aligned} \tag{30}$$

where the last equality uses Eq. (23). Also $D_j^L \leq 1$ because the loss is bounded in $[0, 1]$. Expanding both diagnostic risks as Bayes risk plus excess risk proves Eq. (28). □

For a Bayes rule, replacing V_j with Q_j can only increase risk. For the specified downstream system, gating lowers risk only when its reduction in excess risk exceeds D_j^L. The benchmark results do not estimate these local quantities because gate choices alter subsequent trajectories.

Direct-channel invariant Let Ev project the payload admitted to the direct residual-content channel. Formally,

$$\begin{aligned} \mathrm{Ev}(R) &= \{r \in R : \tau(r) = \mathsf{E}\}, \\ \mathrm{Ev}(\mathrm{Cue}(R)) = \mathrm{Ev}(\mathrm{Constraint}(R)) &= \mathrm{Ev}(\mathrm{Scope}(R)) = \varnothing. \end{aligned} \tag{31}$$

Let $\mathrm{src}_{\mathrm{store}}(e)$ and $\mathrm{src}_{\mathrm{out}}(e)$ be the source fields of residual e in the store and after delivery, respectively. The former is $\mathrm{src}(e)$ in Eq. (8). The gate definitions imply

$$\mathrm{Ev}(Q_j) = \begin{cases} \mathcal{R}_t^{\mathrm{goal}}, & \ell = \mathrm{goal}, \\ \mathcal{R}^{\mathrm{syn}}, & \ell = \mathrm{syn}, \\ \varnothing, & \ell \in \{\mathrm{front}, \mathrm{app}, \mathrm{bd}\}. \end{cases} \tag{32}$$

Every $e \in \mathrm{Ev}(Q_j)$ has type $\tau(e) = \mathsf{E}$ and

$$\mathrm{src}_{\mathrm{out}}(e) = \mathrm{src}_{\mathrm{store}}(e) \neq \varnothing. \tag{33}$$

Moreover, Goal delivery requires $e \in \mathcal{E}_{t-1}$ and $\mathrm{src}_{\mathrm{store}}(e) \notin U_t$, while Synthesis delivery requires $e \in \mathcal{E}_T$. These statements follow from the type map after Eq. (7), the Goal and Synthesis restrictions in Eqs. (8) and (9), and the delivery interfaces in Eqs. (10), (11), and (13). The provenance claim follows from the Prov contract.

This invariant applies only to the delivery interface and does not make claims about factuality or downstream use.

Table 15: Shared execution configuration for the experiments. All settings are held constant across methods.

Category	Setting	Value
Decoding	Main-agent sampling	Temperature 0.8, top-p 0.95
Evaluation	Evaluator model and inference	GPT-5.6 Sol, reasoning effort `medium`
Auxiliary	Summarization, PRM-F feedback, and WasteHub operations	Qwen3.5-35B-A3B
Generation budget	Maximum output per main-agent call	40,960 tokens
Context	Main-agent context window	262,144 tokens
Generation budget	Maximum tokens generated per visit-summary call	8,192 tokens
Research budget	Maximum research rounds per task	50 rounds

Table 16: Primary aggregate scores under three evaluators, averaged over the same five report-generation runs.

Dataset	Evaluator	Base	PRM-F	PAE	WasteHub	WasteHub − PAE
DRB	GPT-5.6 Sol	0.4713	0.4736	0.4772	**0.4861**	+0.0089
	DeepSeek-V4-Pro	0.4669	0.4708	0.4756	**0.4829**	+0.0073
	Claude-Opus-5	0.4740	0.4767	0.4801	**0.4905**	+0.0104
DRB-II	GPT-5.6 Sol	0.3096	0.3136	0.3219	**0.3380**	+0.0161
	DeepSeek-V4-Pro	0.3049	0.3098	0.3195	**0.3331**	+0.0136
	Claude-Opus-5	0.3138	0.3181	0.3230	**0.3424**	+0.0194

K Shared Execution Configuration

For reproducibility, Table 15 lists the key execution settings used throughout the experiments. All settings are held constant across methods. The summarization model, the PRM-F feedback model, and all WasteHub operations use Qwen3.5-35B-A3B.

As of September 21, 2026, Zhipu provides its Web Search API through multiple supported search engines; pricing is available from the official Web Search documentation.[1] Jina Reader meters API-key usage by output tokens. Jina's Search Foundation token packages list one billion tokens for USD 50 and eleven billion tokens for USD 500, corresponding to USD 0.050 and USD 0.045 per million tokens, respectively.[2] Mozilla Readability and Playwright are open-source libraries and incur no per-call API charge.

L Cross-Evaluator Consistency

We test whether the main results depend on GPT-5.6 Sol by rescoring the reports from all five ReAct runs with DeepSeek-V4-Pro and Claude-Opus-5. The audit covers all 100 DRB tasks, all 132 DRB-II tasks, and the four methods in the main comparison. Reports, reference reports, rubrics, weights, prompts, and parsing rules are fixed across evaluators. DRB Overall is computed with the same task-specific rubrics. DRB-II TotalScore is computed with the released expert rubrics. Each report is evaluated once by each model at a requested reasoning effort of `medium`. We use the API identifiers `deepseek-v4-pro` and `claude-opus-5`.

As shown in Table 16, all evaluators produce the same method ordering, with WasteHub followed by PAE, PRM-F, and Base. The report-level Spearman correlations of DeepSeek-V4-Pro and Claude-Opus-5 with GPT-5.6 Sol are 0.962 and 0.972 on DRB, and 0.948 and 0.962 on DRB-II. The corresponding absolute-agreement $\mathrm{ICC}(A,1)$ values across all three evaluators are 0.957 and 0.947. Score levels and method gaps vary across evaluators, but WasteHub remains ahead of PAE in every setting.

[1] `https://docs.bigmodel.cn/cn/guide/tools/web-search`
[2] `https://jina.ai/reader/`

M CASE STUDY

We examine five traces spanning all three residual types and all five reuse gates. Each case follows one residual from collection to its gate decision and downstream status. These qualitative examples illustrate mechanism behavior rather than event frequency or the causal effect of an individual reuse.

Table 17: Case-study overview. The five traces cover all residual types and reuse gates.

Case	Residual path	Observable outcome
A	Unvisited-result → Frontier → agent-selected visit	A visited lead remains unused when its content is insufficient
B	Visit-content → Goal	A source-linked design parameter is selected under the current sub-goal
C	Source-appraisal → Appraisal	A revisited local study remains bounded to its population and measurement setting
D	Visit-content → Synthesis	An omitted material example appears with its original citation
E	Source-appraisal → Boundary	A cited national source receives an explicit geographic qualification

M.1 A FRONTIER CUE THAT REMAINS NON-EVIDENTIAL

DRB question 19 asks how high time-series churn affects Prometheus, how it can be mitigated, and what managed alternatives are available. An early search returns *Scaling Prometheus: Handling Large-Scale Deployments*, but the page is not visited at that step. WasteHub stores its title, URL, and snippet as an Unvisited-result residual. The snippet remains explicitly marked *unverified* and *not citable*, so it can inform a later source choice but cannot support a report claim.

In a later source-selection step, the Frontier gate returns the result because it matches the unresolved high-churn sub-goal and has not previously been surfaced. The cue does not trigger a tool call. The agent chooses and visits the page, which discusses general scaling limitations involving metric cardinality, ingestion rate, and query latency, but does not provide the detailed high-churn analysis required by the current goal. The final report does not cite the page. The visit changes the status of the residual, not that of the earlier search snippet, and the visited content is still unused because it does not satisfy the active information need. The frontier record therefore affects source selection without becoming evidence, while the later visit remains subject to ordinary relevance checks.

M.2 RECOVERING A DESIGN CONSTRAINT

DRB question 35 asks for an operational reform plan for municipal wastewater collection and treatment. At round 2, source S_1 describes placing one high-precision water-level gauge every 500 meters along an open-channel segment to avoid redundant investment and reduce construction and maintenance costs. The visit summary retains broader observations about system operation and maintenance, including collaborative operation, layered control, and data-driven maintenance, but not this placement rule. WasteHub stores the omitted rule as a Visit-content residual linked to S_1, together with its cost-control role and source identifier.

At round 4, the agent pursues a cost-control sub-goal while visiting a different source S_2, whose summary does not cover sensor placement. The Goal gate selects the earlier record because it is aligned, not covered by the current visit, and comes from a different source. The augmented state therefore combines the current evidence from S_2 with the complementary parameter from S_1 without merging their provenance. The final report places the 500-meter spacing in its construction-stage cost-control discussion, retains the cost rationale and citation to S_1, and does not state the spacing as a universal engineering standard.

M.3 CONSTRAINING THE REUSE OF A LOCAL ESTIMATE

DRB-II question 123 requests a global account of non-compliant and adaptive housing practices within otherwise legal urban areas using research available before May 2024. One visited study

estimates $1{,}045 \pm 226$ additional informal detached accessory dwelling units in San José from 2016 to 2020. It derives the estimate from a computer-vision model, human annotations on a stratified parcel sample, and public satellite imagery. After the visit, WasteHub stores a Source-appraisal record identifying the supported use as a model-based estimate for one unit type, city, and period, rather than a measure of global prevalence. The record preserves the measured population, estimation method, and reported uncertainty because each condition affects how the local number may be interpreted.

When the agent later selects the same study while investigating the global extent of informal housing, the Appraisal gate retrieves the source-keyed record. The note accompanies the new visit as an applicability constraint and does not enter the evidence channel. The current visit supplies the evidence, while the note limits its interpretation to additional informal detached ADUs in San José during 2016–2020. The updated state may retain the study as a local example, but requires other sources for claims about other unit types, cities, periods, or global prevalence. Appraisal therefore constrains how a revisited source is used without adding a new housing estimate or replacing the content of the current visit.

M.4 Completing a Material Example Before Drafting

DRB question 11 surveys common corrosion inhibitors for carbon steel and their Raman and infrared activity. A visited review states that encapsulating benzotriazole in nanocontainers can improve the protective properties and environmental safety of steel coatings. The visit summary covers benzotriazole, adsorption, surface layers, and general effects across pH conditions, but omits the nanocontainer example. WasteHub stores the example as a source-linked Visit-content residual. The record is not selected by Goal during the remaining research steps and therefore remains available at finalization.

Before report generation, the coverage sketch contains a slot for trends and future prospects, while the terminal state does not contain this example. The Synthesis gate selects the record for that slot. The final report includes the nanocontainer example in its discussion of encapsulated composite inhibitors and connects it to coating protection and environmental safety. The original source citation is retained. The selected record fills an existing report slot rather than opening a new research direction, and no unvisited source is used as evidence.

M.5 Adding a Source-Linked Boundary

DRB-II question 4 requests a comparison of retirement-savings protection in Indonesia, Malaysia, Pakistan, the Philippines, Sri Lanka, Thailand, and Vietnam using data and policies available by early 2025. During research, the trace visits an Indonesian government portal for information about the BPJS Ketenagakerjaan scheme. WasteHub stores a Source-appraisal record stating that the portal is applicable to Indonesia, but cannot establish benefit formulas or accrual rates for the other six countries. The record is keyed to the portal and records a geographic usage condition rather than pension evidence. This distinction matters because the requested report also compares mandatory benefit formulas across all seven countries.

The complete report draft cites the portal in its Indonesia profile, while the source's country scope remains unstated. The Boundary gate selects the appraisal record because it concerns a cited source, applies to claims in the draft, and supplies a qualification not already present. WasteHub appends one source-linked paragraph stating that the Indonesian portal cannot support benefit formulas or accrual rates for Malaysia, Pakistan, the Philippines, Sri Lanka, Thailand, or Vietnam. The preceding report text is unchanged. The paragraph adds no pension evidence and does not revise the country profiles or comparison tables. It only makes explicit how the cited source may be interpreted, while preserving the portal's valid use for the Indonesia profile.

Cross-case pattern. Frontier cues remain non-evidential. Goal and Synthesis deliver source-linked Visit-content evidence, while Appraisal and Boundary deliver source-applicability conditions only. Gate timing determines whether a residual guides source selection, constrains source reuse, augments research evidence, completes draft coverage, or qualifies a cited claim. These traces complement the aggregate and ablation results by showing the corresponding mechanism behavior.