# OFFICETOWN: GENERATIVE ORGANIZATIONS FOR EVOLVING AGENT INTELLIGENCE

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

AI agents increasingly use files and software tools to produce professional deliverables. Such work draws on materials, decisions, and feedback accumulated across assignments. We introduce OfficeTown, a generative simulation framework for professional organizations in which tasks and their context arise from continuing organizational work. From a natural-language description of an organization's purpose and activities, OfficeTown initializes an organization whose members coordinate work, delegate bounded assignments to assistant agents, and review their deliverables. Artifacts, decisions, and unresolved matters persist and shape subsequent work, while responsibilities and permissions determine what each member can see and do. We curate work requests, their context, and review feedback into executable resources for evaluation and learning. We construct OfficeTown Bench with 406 tasks and 4,836 evaluation criteria from 23 simulated organizations across eight professional domains, incorporating domain-expert feedback. Frontier-model evaluations reveal substantial remaining headroom. Supervised fine-tuning and on-policy distillation on OfficeTown data substantially improve Qwen3.5-9B on external professional-work benchmarks, including GDPval. Supplementary evaluations show that the principal–assistant harness achieves higher scores than the single- and multi-agent baselines tested. These results highlight the value of organizational simulation in generating work-grounded data and environments for the continued development of generalist agents.

## 1 INTRODUCTION

Large language models are increasingly expected to produce professional deliverables using files, software tools, and organizational knowledge. Recent benchmarks (Patwardhan et al., 2025) and environment-synthesis methods (Dong et al., 2026) provide valuable resources for evaluating these capabilities, but typically treat tasks as isolated instances. Real professional work is rarely isolated. In organizations, assignments arise from ongoing activity rather than appearing independently. Deliverables are reviewed and revised, decisions create follow-up requests, and materials accumulated in one project become inputs to later work. Tasks therefore emerge from a continuing process in which work products and work context co-evolve.

This observation suggests a different perspective. Instead of generating standalone tasks, we may seek to model the system that continuously produces professional work. Such a system would naturally generate new assignments, contextual materials, review feedback, and long-range dependencies, offering a scalable source of both evaluation and training data.

Professional organizations are precisely such systems. We therefore introduce OfficeTown, a generative simulation framework for professional organizations. From a natural-language description of an organization's purpose and activities, OfficeTown constructs persistent organizational structures, roles, resources, and authority boundaries, then simulates ongoing work as professionals coordinate, delegate, review, and follow up on assignments.Organizational state is persistent but not globally observable: each member has a permission-filtered view determined by role, ownership, and sharing rules, and acts within their role-specific authority. Figure 1 illustrates a Juniper instance.

These professionals act as principals who coordinate work, delegate bounded assignments to assistant agents, and review their deliverables. Assistants execute within authorized workspaces. Arti-



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.

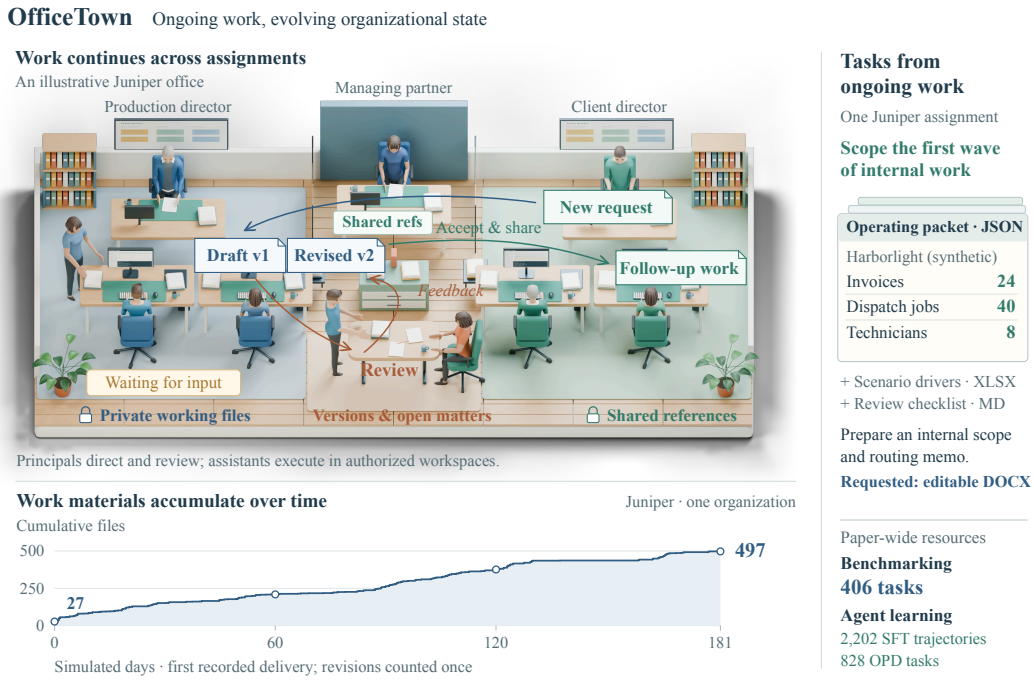



Figure 1: OfficeTown models ongoing work in a persistent, permissioned organization. Professionals coordinate requests and review deliverables; assistants execute within authorized workspaces. Private drafts, revisions and unresolved matters carry context into subsequent work. The ten-role Juniper scene and its work links are illustrative, not a recorded task trajectory. The lower plot tracks one Juniper organization over 181 simulated days, not wall-clock runtime: 27 initial materials and 470 subsequent logical outputs yield 497 cumulative files. All recorded time points are retained; revisions and exact copies are deduplicated (Appendix A.7). These counts measure material accumulation, not quality or task success. The right-hand example is a retained scope-and-routing-memo assignment, showing synthetic input records, supporting materials and the requested deliverable, not a model answer.

facts and decisions update organizational state, while unresolved matters prompt follow-up work. We package selected assignments with their task-time context into executable resources for evaluation and learning. Figure 1 (right) shows a concrete assignment with supporting materials and a requested internal scope and routing memo.

We construct OfficeTown Bench with 406 tasks and 4,836 evaluation criteria from 23 simulated organizations across eight professional domains, incorporating domain-expert feedback into task and rubric refinement. Delegating professionals span 139 distinct job titles. Training resources comprise 2,202 demonstrations for supervised fine-tuning (SFT) and 828 task–environment pairs for on-policy distillation (OPD).

Frontier-model evaluations reveal substantial remaining headroom on OfficeTown Bench. Training on OfficeTown data improves Qwen3.5-9B on external professional-work benchmarks. On 79 shared GDPval tasks, its rubric score rises from 65.75 to 73.65 after SFT and OPD. Supplementary evaluations favor the principal–assistant harness over tested single- and multi-agent baselines.

Our contributions are:

- We introduce a generative simulation framework for professional organizations in which persistent, permissioned state shapes subsequent work.
- We construct an expert-informed professional-work benchmark and training resources from ongoing organizational work.

- We demonstrate that the resulting tasks challenge frontier models and that OfficeTown data improves performance on external benchmarks.

## 2 OFFICETOWN: A GENERATIVE ORGANIZATIONAL SIMULATION FRAMEWORK

OfficeTown initializes an organization with roles, goals, resources, and authority, then simulates ongoing professional work progressing asynchronously (Figure 2). Requests, messages, commitments, and unresolved matters activate professionals, who coordinate work and delegate bounded assignments to assistants. They review delivered artifacts, request revisions, and arrange follow-up work. Files, communications, and decisions update persistent, permissioned organizational state throughout this process, shaping subsequent assignments.

### 2.1 ORGANIZATIONS, ROLES, AND AUTHORITY

An OfficeTown instance represents a professional organization or team initialized from a natural-language description of its purpose and activities (Figure 2a). The description guides organizational objectives, professional roles, and reporting relationships. Each role defines continuing responsibilities and role-specific authority: which matters the professional handles and what they may execute, delegate, approve, or share. Figure 2d shows Juniper Financial Modelling, a twelve-role instance.

Each role is represented by a model-driven professional whose profile draws on O*NET and ESCO occupational descriptions. Professionals act as principals responsible for ongoing work across assignments; assistants perform delegated executions.

The organization starts with documents and data, termed the seed. These combine retrieved or adapted public sources with organization-specific synthetic records, preserving source references and the distinction between these origins. The seed provides initial working context that subsequent work can extend or revise. Appendices A.1–A.2 detail construction, role skills, and Juniper's setup.

### 2.2 PERSISTENT AND PERMISSIONED ORGANIZATIONAL STATE

Organizational state includes personal working files, shared materials, communications, commitments, and pending decisions. Each professional maintains a persistent workspace (Figure 2e) and structured work records separate from active conversation context, so refreshing a model session does not erase unresolved work. Each activation presents selected work records alongside current requests, prioritizing outstanding commitments and discussions awaiting a response.

This state is persistent but not globally observable. Role permissions, ownership, sharing rules, and confidentiality determine each member's information view; role-specific authority determines what they may do with it. Reporting relationships do not automatically grant file access. Assistants receive writable working files and permission-filtered, read-only shared references within the principal's authorization, not access to other professionals' private workspaces.

Saving, accepting, and sharing a deliverable are distinct actions. Saving retains files, including drafts and pending revisions; professional review determines acceptance; explicit sharing makes an accepted deliverable available to authorized recipients. Thus, saved work need not be accepted or visible to others. Acceptance, version, classification, and recipient permissions are checked before broader publication. Appendices A.4–A.6 cover workspace checks, sessions, and recovery.

### 2.3 COORDINATION, DELEGATION, AND REVIEW

Work follows a shared simulated calendar, with professionals activated asynchronously by incoming requests, messages, meetings, follow-ups, or assistant deliveries (Figure 2b). The simulation controls activation and permitted actions; the professional's model decides how to respond using current requests, commitments, and accessible materials. Several matters can therefore progress concurrently. Appendix A.6 details scheduling and clock control.

When responsibilities intersect, members exchange permitted materials, question assumptions, and seek colleagues' input. Team leads can meet to clarify objectives, identify missing inputs, and

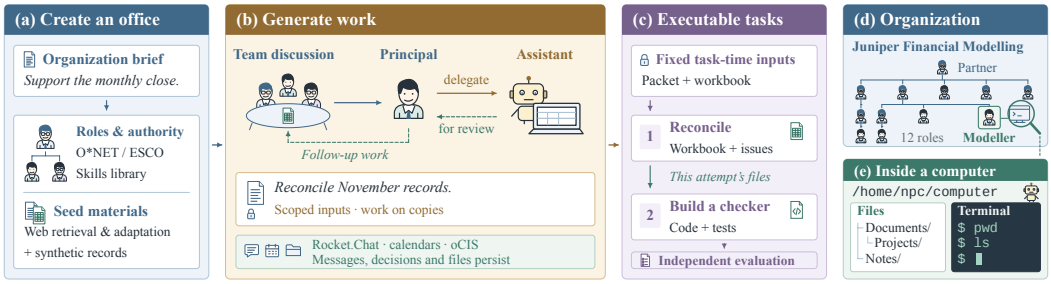



Figure 2: OfficeTown's organizational simulation. **(a)** Organizational objectives, roles, and seed materials establish an organization. **(b)** Coordination, delegation, execution, and review connect ongoing work through retained materials and follow-up assignments. **(c)** Tasks drawn from this activity provide fixed inputs and executable environments for training and evaluation. **(d)** The recorded twelve-role hierarchy of Juniper Financial Modelling; the senior modeller's computer is highlighted. **(e)** Filesystem and terminal views used by an assistant. The reconciliation and Juniper examples are separate. Directory names are verified; interfaces and terminal activity are schematic.

allocate responsibilities; meetings are not required for every assignment. Agreed work is recorded as requests specifying the intended outcome, responsible person, supporting materials, and deadline. Appendix A.3 provides a recorded example.

For delegated execution, the principal specifies the objective, available inputs, constraints, and expected deliverable. The assistant works within the authorized workspace and can ask the principal to clarify requirements or consult colleagues. On delivery, the principal inspects the actual artifact and decides whether to accept it, request revisions, or seek further information. Review can change both deliverable requirements and the work that other members need to undertake. These reviews are organizational decisions within the simulated organization, not independent benchmark judgments.

Revisions build on submitted artifacts and review comments. Accepted results can become inputs to authorized colleagues' work, while unresolved questions remain available for later investigation. These updates link assignments through accumulated context, extending work beyond the initial seed. Appendix A.7 illustrates material accumulation over six simulated months.

### 2.4 From Ongoing Work to Executable Tasks

An executable task pairs a work request with materials available to its responsible professional when the work was assigned (Figure 2c). This task-time snapshot preserves both temporal and permission boundaries: personal files, authorized shared resources, and existing artifacts needed for revision may be included, but later solutions and evaluator-only information are excluded.

Each attempt starts in a fresh workspace initialized from these fixed inputs, where the agent can inspect and edit files, execute commands, and use the provided interfaces. The continuing organization is the source of the assignment; individual evaluation or learning attempts do not require replaying its entire history. Section 3 explains selection, expert-informed refinement, and packaging into benchmark and training resources.

## 3 OfficeTown Data and Benchmark

We curate assignments and their task-time environments from ongoing organizational work into OfficeTown Bench and training resources. The benchmark measures professional task completion, while execution demonstrations and interactive task environments support learning.

### 3.1 Benchmark Design

OfficeTown Bench covers eight professional domains selected for economic relevance and digital executability, inspired by GDPval (Patwardhan et al., 2025). It contains 406 tasks and 4,836 evalu-

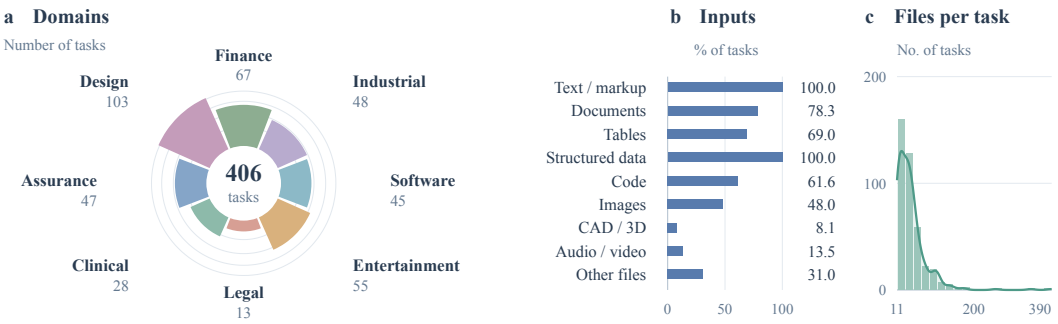
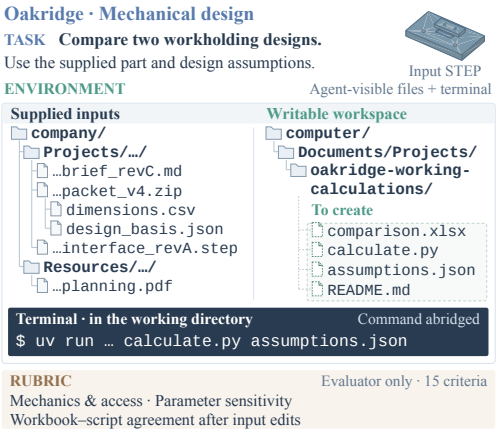



Figure 3: OfficeTown Bench composition. (a) Domains, with petal areas and labels indicating task counts. (b) Input-type coverage, including archive contents; types overlap. (c) Input-file counts: 20-file bins starting at 11, with a smoothed frequency curve. All panels use all 406 tasks; an archive counts as one file in (c).

Table 1: OfficeTown Bench statistics. Per-task rows: mean / median. Total counts file instances across tasks; unique counts deduplicate by content. Archives count as one file.

| Statistic | Value |
|---|---|
| Tasks | 406 |
| Professional domains | 8 |
| Source organizations | 23 |
| Distinct role titles | 139 |
| Input files (task-wise total) | 18,082 |
| Unique input files | 6,951 |
| Input files / task | 44.54 / 37 |
| Input types / task | 5.10 / 5 |
| Rubric criteria (total) | 4,836 |
| Rubric criteria / task | 11.91 / 11 |
| Contributing domain experts | 71 |

Figure 4: An OfficeTown Bench task. Dashed files are requested outputs; the rubric is evaluator-only.




ation criteria from 23 simulated organizations, involving delegating professionals with 139 distinct job titles. The benchmark includes 247 file-based tasks and 159 tasks involving service state. Figure 3 and Table 1 report its domain coverage and input-material composition.

Each sample combines a work request, an executable environment, and a task-specific rubric. Environments supply the files, tools, and authorized organizational state for the assignment; rubrics assess professional requirements but remain outside the agent's starting context.

Figure 4 illustrates a mechanical-design assignment. Given workpiece geometry, feature tables, design assumptions, and engineering references, the agent must compare two locating-and-clamping approaches in an editable workbook with reproducible calculation code. The rubric assesses mechanical plausibility, sensitivity to uncertain inputs, and agreement between the workbook and code.

### 3.2 Construction and Expert Review

We inspect delegated requests and their working materials, then edit them into self-contained evaluation tasks while retaining work objectives, constraints, and intended deliverables. Task environments preserve the relevant starting materials, including artifacts requiring revision, but exclude historical solutions to the evaluated task and evaluator-only information.

Rubrics are drafted by a language model from task requirements and inspected source materials. Human-authored examples guide criterion formulation without supplying task-specific facts, thresh-

Table 2: OfficeTown Bench Pass@1 (%) on 406 tasks, with high reasoning effort. Domain abbreviations follow Figure 3. 1st 2nd 3rd mark within-column ranks at displayed precision; ties share rank and skip positions. Weighted Scores and scoring definitions appear in Appendices D.1 and C.1.

| **Model** | Overall | Assur. | Clin. | Des. | Ent. | Fin. | Ind. | Legal | Soft. |
|---|---|---|---|---|---|---|---|---|---|
| GPT-6 Astra | 74.4 | 63.8 | 92.9 | 56.3 | 72.7 | 88.1 | 81.3 | 92.3 | 84.4 |
| GPT-5.6 Sol | 33.5 | 38.3 | 35.7 | 26.2 | 45.5 | 34.3 | 29.2 | 46.2 | 28.9 |
| GPT-5.6 Terra | 39.4 | 48.9 | 32.1 | 35.0 | 43.6 | 40.3 | 43.8 | 38.5 | 33.3 |
| GPT-5.6 Luna | 32.5 | 48.9 | 53.6 | 21.4 | 41.8 | 31.3 | 22.9 | 30.8 | 28.9 |
| GPT-5.4 | 30.5 | 31.9 | 39.3 | 25.2 | 41.8 | 32.8 | 29.2 | 30.8 | 20.0 |
| GPT-5.2 | 21.2 | 27.7 | 28.6 | 13.6 | 27.3 | 31.3 | 14.6 | 15.4 | 13.3 |
| Claude Opus 5.5 | 50.7 | 55.3 | 67.9 | 38.8 | 60.0 | 49.3 | 50.0 | 61.5 | 51.1 |
| DeepSeek V4-Pro | 18.5 | 23.4 | 14.3 | 14.6 | 30.9 | 19.4 | 12.5 | 0.0 | 20.0 |
| DeepSeek V4-Flash | 13.5 | 17.0 | 10.7 | 11.7 | 23.6 | 10.4 | 10.4 | 15.4 | 11.1 |
| Kimi-K2.5 | 4.9 | 10.6 | 7.1 | 1.0 | 14.5 | 1.5 | 2.1 | 7.7 | 2.2 |
| Grok-4.7 | 58.4 | 70.2 | 57.1 | 54.4 | 47.3 | 65.7 | 58.3 | 46.2 | 62.2 |
| Grok-4.3 | 8.1 | 19.1 | 14.3 | 6.8 | 12.7 | 4.5 | 0.0 | 7.7 | 4.4 |
| Mistral-Large-3 | 1.2 | 2.1 | 0.0 | 0.0 | 7.3 | 0.0 | 0.0 | 0.0 | 0.0 |

olds, or answers. A separate model-based review checks professional correctness, missing coverage, overlapping criteria, and unsupported restrictions. Revisions preserve professional requirements.

During benchmark construction, 71 domain experts contributed feedback on task realism, clarity, input sufficiency, and feasibility, as well as rubric correctness, coverage, and alignment. Expert assignment follows relevant specialization, occupational experience, and familiarity with working materials and applicable standards. Experts propose edits, removals, merges, or splits and rate importance as Critical (P0), Major (P1), Minor (P2), or Optional (P3). These annotations concern tasks and rubrics, not individual agent answers. The released task and criterion sets are fixed for model comparisons. Appendix B details construction, annotation instructions, and review coverage.

### 3.3 Training Data

The SFT resource comprises 2,202 complete execution trajectories distilled from DeepSeek-V4-Flash, pairing task inputs with teacher reasoning, tool calls, observations, and final responses. We retain complete histories without truncation or filtering by task-success score.

The OPD resource provides 828 task–environment pairs for the subsequent distillation stage. Each supplies a work request and its initial environment without a teacher response history, using a system message and tool interface aligned with SFT. Appendix E describes resource construction; Section 4 reports training and evaluation results.

## 4 Experiments

We evaluate organizational simulation through the work it produces: whether OfficeTown tasks challenge current agents, and whether learning from this work improves external-benchmark performance. A supplementary study tests the principal–assistant design on external tasks.

### 4.1 Evaluation on OfficeTown Bench

Agents execute work requests using the supplied files and tools, without access to the task-specific rubrics. We assess submitted artifacts and observable outcomes through executable checks and content-level judgment. Table 2 reports Pass@1 for 13 models on the same 406 tasks: a task passes only when all applicable required criteria are met. Weighted Score, reported in Appendix D.1, averages each task's fulfilled rubric-weight fraction. These metrics distinguish complete delivery from partial fulfillment; scoring and execution protocols are in Appendix C.

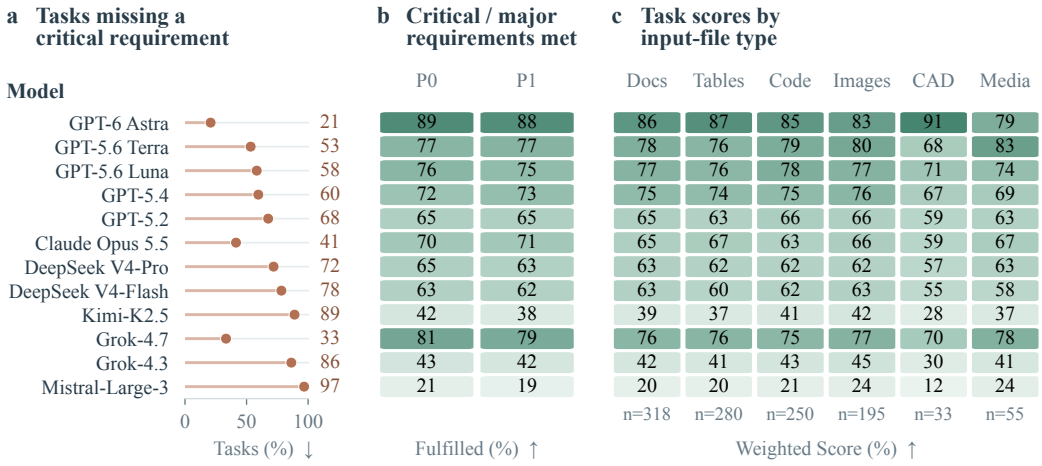



Figure 5: Professional requirements beyond aggregate scores. (a) Judged tasks missing at least one applicable critical (P0) requirement. (b) Task-averaged fulfillment of critical and major requirements (P0/P1). (c) Mean task scores grouped by supplied input-file types; groups overlap. Twelve models with bound item-level judgments are shown. Item-level panels exclude unjudged outcomes; file-group scores retain the main table's declared zeros. Definitions and full values are in Appendix D.

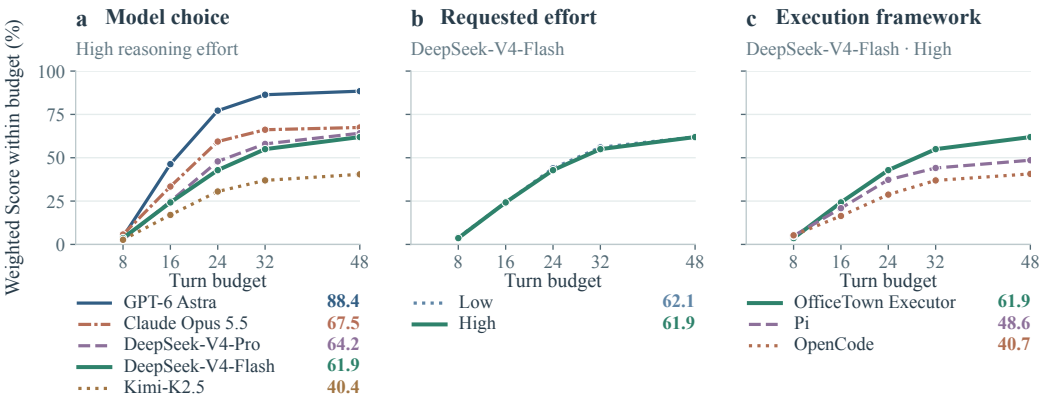



Figure 6: OfficeTown Bench at five recorded turn budgets. All panels use the same 406 tasks. Scores enter only after execution ends; terminal evaluation errors remain zero. (a) High-effort models with the same executor. (b) Low and High effort for DeepSeek-V4-Flash. (c) Execution frameworks for DeepSeek-V4-Flash High. Points show 8, 16, 24, 32, and 48 turns; lines guide the eye. Appendix C.2 specifies counting and execution settings.

GPT-6 Astra leads with 74.4% Pass@1, followed by Grok-4.7 at 58.4% and Claude Opus 5.5 at 50.7%. Even Astra leaves roughly one quarter of tasks incompletely delivered. The metrics also expose different performance profiles: Claude passes more tasks than GPT-5.6 Terra (50.7% versus 39.4%) despite a lower Weighted Score (67.5 versus 76.1). High average fulfillment therefore need not imply reliable complete delivery. Domain strengths vary: Grok-4.7 exceeds Astra in assurance (70.2% versus 63.8%) despite its lower overall pass rate.

This gap extends to critical requirements (Figure 5). Astra satisfies 88.8% of critical criterion weight on average, yet 84 of 404 eligible tasks (20.8%) miss at least one critical (P0) requirement. The miss rate rises to 53.3% for Terra and 78.3% for DeepSeek-V4-Flash, while Claude Opus 5.5 and Grok-4.7 miss critical requirements on 41.4% and 33.3% of eligible tasks. Failures are not limited to optional refinements. Input-file groups reveal different relative strengths: Astra leads Terra on CAD-containing tasks, but trails on media-containing tasks. These overlapping task populations do not isolate the effect of file format.

Table 3: Learning from OfficeTown: Qwen3.5-9B across training stages (%). Rubric and Weighted denote mean rubric and weighted rubric scores; Success denotes the native binary success rate. Subscripts show percentage-point differences from Base. Protocols: Appendix E.3.

| **Training** | GDPval Rubric | JobBench-Easy Weighted | APEX-Agents Pass@1 | Terminal-Bench 2.1 Success | OfficeTown Bench Pass@1 |
|---|---|---|---|---|---|
| Base | 65.75 | 40.41 | 9.40 | 42.31 | 1.48 |
| SFT | 70.30$_{+4.55}$ | 44.87$_{+4.46}$ | 16.16$_{+6.76}$ | 51.22$_{+8.91}$ | 3.45$_{+1.97}$ |
| SFT + OPD | **73.65**$_{+7.90}$ | **51.48**$_{+11.07}$ | **19.43**$_{+10.03}$ | **63.64**$_{+21.33}$ | **4.68**$_{+3.20}$ |

Table 4: Harness results on external benchmarks. All harnesses use GPT-5.5 with xhigh reasoning effort. Reported point estimates on a 0–100 scale; higher is better. GDPevo aggregates repeated runs by task group, not best-of-three success. Codex on GDPevo uses the few-shot reference setting. Terminal results use different reported budgets.

| **Harness** | PRDBench | | GDPevo | Terminal-Bench 2.1 | Research-code |
|---|---|---|---|---|---|
| | DEV | DEBUG | | | |
| Codex | 85.87 | 89.04 | 64.51 | 83.7 | 40.88 |
| MetaGPT | 84.86 | 86.51 | 63.44 | 81.1 | 41.15 |
| CAMEL | 76.74 | 78.93 | 63.45 | 85.1 | 39.95 |
| OfficeTown Harness | 89.67 | 92.62 | 74.70 | 87.4 | 46.40 |

Figure 6 summarizes five budgets on the same 406 tasks. In the full records, Astra reaches a Weighted Score of 40 within 16 turns, followed by Claude at 18, DeepSeek-V4-Pro at 22, DeepSeek-V4-Flash at 24, and Kimi-K2.5 at 42. Flash's Low and High settings finish close in Weighted Score (62.1 versus 61.9), whereas execution frameworks differ more: OfficeTown Executor reaches 61.9, Pi 48.6, and OpenCode 40.7. These curves summarize recorded executions, not fresh shorter-budget runs or equal-token comparisons; frameworks retain their native prompts and tools. OfficeTown Executor is single-agent, distinct from the principal–assistant harness below.

### 4.2 Learning from OfficeTown

To test learning value beyond the source setting, we fine-tune Qwen3.5-9B on DeepSeek-V4-Flash demonstrations, then continue with OPD (Agarwal et al., 2024) in OfficeTown environments using Qwen3.8-Flash-Next as teacher. Table 3 compares Base, SFT, and SFT + OPD. External benchmarks use their respective metrics; GDPval and Terminal-Bench exclude search-dependent tasks (Appendix E.3).

On 79 shared GDPval tasks, the rubric score rises from 65.75 to 70.30 after SFT and to 73.65 after OPD, gaining 7.90 percentage points over Base. JobBench-Easy (Li et al., 2026b) and APEX-Agents (Vidgen et al., 2026) also improve across the reported stages. Together, the results support the learning value of OfficeTown-generated work beyond its source organizations. The corresponding OfficeTown Bench Weighted Scores are 26.20, 29.20, and 31.33 for Base, SFT, and SFT + OPD, respectively. Training dynamics and retrospective budget curves appear in Appendices E.4–E.5; they do not establish data-size scaling or equal-resource gains.

### 4.3 Harness Evaluation

This supplementary study tests whether the principal–assistant design for delegation, clarification, and review is useful beyond the simulated organizations. We compare OfficeTown Harness with Codex, MetaGPT (Hong et al., 2024), and CAMEL (Li et al., 2023) using GPT-5.5 xhigh throughout on external tasks (Table 4). OfficeTown has the highest reported point estimate in all five columns, covering initial development and feedback-guided revision (PRDBench DEV/DEBUG; Fu et al.,

2026), business rules (GDPevo; Zhou et al., 2026), terminal work (Merrill et al., 2026), and research code (Hua et al., 2025). This is not a uniform efficiency advantage: on research code, OfficeTown scores 46.40 versus MetaGPT's 41.15 with 21.8% fewer reported tokens, but uses 3.49 times Codex's tokens. Terminal budgets differ, and Codex's GDPevo reference uses few-shot skills. These comparisons support the design's broader utility, not the necessity of persistent simulation.

# 5 Related Work

## 5.1 Professional Work Benchmarks

Executable benchmarks cover web and desktop interaction (Zhou et al., 2024a; Xie et al., 2024), enterprise knowledge work (Drouin et al., 2024; Boisvert et al., 2024), and cross-application, long-horizon, and file-dependent workflows (Wang et al., 2024; 2025; Tang et al., 2026). GDPval evaluates professional deliverables across occupations (Patwardhan et al., 2025). EnterpriseBench, CRMArena-Pro, and AutomationBench examine cross-functional workflows, access constraints, and policy-governed interaction (Vishwakarma et al., 2025; Huang et al., 2025; Shepard & Salimans, 2026). ClawMark includes multi-day workflows with externally changing service state (Meng et al., 2026). Closest to our setting, TheAgentCompany combines a simulated company, office applications, and model-driven colleagues with manually curated tasks (Xu et al., 2024).

## 5.2 Environment and Task Synthesis

Environment synthesis scales the construction of executable tasks and training resources. ScaleEnv generates tool environments and rollout-verified tasks, while EnvFactory builds stateful environments and training trajectories from API resources (Tu et al., 2026; Xu et al., 2026b). EnvScaler, AWM, and AutoForge automate environment and verifiable-task construction (Song et al., 2026; Wang et al., 2026; Cai et al., 2025). Agent-World combines synthesis with iterative reinforcement learning (Dong et al., 2026), and ClawEnvKit generates tasks, tools, and scoring criteria from natural-language requests (Li et al., 2026a). ScienceBuddy derives executable tasks and rubrics from researcher collaboration (Xue et al., 2026). Our focus is work generation, not recursive self-improvement (Duan et al., 2026).

## 5.3 Generative Agents and Organizational Simulation

Generative Agents uses memory, reflection, and planning to sustain social activity (Park et al., 2023). SOTOPIA studies goal-directed social interaction (Zhou et al., 2024b), while $\tau$-bench evaluates policy-constrained tool use with simulated users (Yao et al., 2024). Agent Hospital connects institutional simulation with professional capability development through accumulated medical cases and reflection, without updating the underlying models (Li et al., 2024).

OfficeTown connects organizational simulation with executable environment generation. Earlier artifacts, decisions, and unresolved matters shape subsequent work, while roles, ownership, and sharing rules scope the information and authority available for each assignment. This persistent, permissioned process generates tasks, context, and feedback for evaluation and training.

# 6 Conclusion and Future Work

OfficeTown models professional organizations as persistent, permissioned, and evolving work systems. Tasks, contextual materials, and feedback arise from ongoing work, rather than being constructed as isolated instances. Benchmark evaluations reveal a gap between partial fulfillment and complete professional delivery, while training results demonstrate the learning value of the generated work on external benchmarks.

Looking ahead, we will extend these simulations to more professional domains and institutional settings, with richer interactions, longer-running workflows, and collaboration across organizations. We will also investigate how persistent history, scoped access, and review shape task generation and learning outcomes. Improving simulation fidelity and efficiency, and validating simulated work against real professional practice, will be important as these systems expand.

**AI Use Statement.** We used LLM-based generative AI tools to assist with writing refinement and the creation of figures. All research ideas, methodology, experiments, analyses, and conclusions were developed and verified by the authors.

**Ethics Statement.** OfficeTown simulates professional organizations and synthetic customers; it does not use private customer records. Seed materials combine referenced public sources with adapted or synthetic records, and the distinction between these origins is retained. Domain experts contributed feedback on tasks and rubrics rather than personal or sensitive information, and we report their contributions only in aggregate. Because occupational taxonomies, public sources, and generative models can encode social or professional biases, the resulting organizations and tasks should not be interpreted as authoritative representations of occupations, institutions, or demographic groups. Tool-using agents may also create privacy, security, or operational risks if deployed without appropriate controls. Our simulations therefore use permission-scoped workspaces and synthetic organizational authority; they do not authorize real-world financial, legal, clinical, employment, or external-release decisions. Any release of tasks, environments, or execution traces should preserve source attribution, remove sensitive operational information, respect applicable licenses, and document residual limitations and intended use.

**Reproducibility Statement.** The main text specifies the OfficeTown simulation, permission model, task construction, benchmark, metrics, and training stages. Appendix A documents organization construction, scheduling, memory, workspaces, access control, recovery, and a longitudinal simulation; Appendix B reports resource membership, separation, expert-review coverage, domain and file-type composition, and benchmark cases. Appendices C and E provide scoring definitions, execution budgets, failure accounting, training-resource construction, complete SFT and OPD configurations, and external-evaluation protocols. The tables identify task counts, model and teacher versions, hardware, optimization settings, context limits, seeds, and checkpoint schedules. Supplementary artifacts will provide the corresponding task environments, rubrics, configuration files, and released execution components, subject to licensing, privacy, and security constraints.

## REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Léo Boisvert, Megh Thakkar, Maxime Gasse, Massimo Caccia, Thibault Le Sellier De Chezelles, Quentin Cappart, Nicolas Chapados, Alexandre Lacoste, and Alexandre Drouin. WorkArena++: Towards compositional planning and reasoning-based common knowledge work tasks. *arXiv preprint arXiv:2407.05291*, 2024. URL https://arxiv.org/abs/2407.05291v2.

Shihao Cai, Runnan Fang, Jialong Wu, Baixuan Li, Xinyu Wang, Yong Jiang, Liangcai Su, Liwen Zhang, Wenbiao Yin, Zhen Zhang, Fuli Feng, Pengjun Xie, and Xiaobin Wang. AutoForge: Automated Environment Synthesis for Agentic Reinforcement Learning. *arXiv preprint arXiv:2512.22857*, 2025. URL https://arxiv.org/abs/2512.22857v1.

Guanting Dong, Junting Lu, Junjie Huang, Wanjun Zhong, Longxiang Liu, Shijue Huang, Zhenyu Li, Yang Zhao, Xiaoshuai Song, Xiaoxi Li, Jiajie Jin, Yutao Zhu, Hanbin Wang, Fangyu Lei, Qinyu Luo, Mingyang Chen, Zehui Chen, Jiazhan Feng, Ji-Rong Wen, and Zhicheng Dou. Agent-World: Scaling real-world environment synthesis for evolving general agent intelligence. *arXiv preprint arXiv:2604.18292*, 2026. URL https://arxiv.org/abs/2604.18292v1.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H. Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. WorkArena: How Capable Are Web Agents at Solving Common Knowledge Work Tasks? *arXiv preprint arXiv:2403.07718*, 2024. URL https://arxiv.org/abs/2403.07718v5.

Yi Duan, Ying Liu, Zirui Tang, Haodong Chen, Jun Zhou, Yumou Liu, Bangrui Xu, Yukai Wu, Sidi Chen, Yuhan Zhou, Haoyu Wang, Xiaoyou Yu, Shaokun Han, Xuzhou Zhu, Le Zhou, Bolin Lu,

Wei Zhou, Jiachen Liu, Nuozhou Fang, Jiaxin Tian, Ruoyu Chen, Yuxuan Li, Kai Zuo, Kaiyan Zhang, Qianyu Yang, Zijie Wang, Jiantao Qiu, Conghui He, Guoliang Li, Bowen Zhou, Zhiyuan Liu, Zhoufutu Wen, Jihua Kang, Xuanhe Zhou, and Fan Wu. The last AI built by humans: Toward genuine recursive self-improvement. *arXiv preprint arXiv:2609.11873*, 2026. URL https://arxiv.org/abs/2609.11873v3.

Lingyue Fu, Bolun Zhang, Hao Guan, Yaoming Zhu, Lin Qiu, Weiwen Liu, Xuezhi Cao, Xunliang Cai, Weinan Zhang, and Yong Yu. Automatically benchmarking LLM code agents through agent-driven annotation and evaluation. In *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems*, 2026. doi: 10.65109/40MNNF. URL https://arxiv.org/abs/2510.24358v3.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Steven Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, volume 2024, pp. 23247–23275, 2024.

Tianyu Hua, Harper Hua, Violet Xiang, Benjamin Klieger, Sang T. Truong, Weixin Liang, Fan-Yun Sun, and Nick Haber. ResearchCodeBench: Benchmarking LLMs on implementing novel machine learning research code. *arXiv preprint arXiv:2506.02314*, 2025. URL https://arxiv.org/abs/2506.02314v1.

Kung-Hsiang Huang, Akshara Prabhakar, Onkar Thorat, Divyansh Agarwal, Prafulla Kumar Choubey, Yixin Mao, Silvio Savarese, Caiming Xiong, and Chien-Sheng Wu. CRMArena-Pro: Holistic assessment of LLM agents across diverse business scenarios and interactions. *arXiv preprint arXiv:2505.18878*, 2025. URL https://arxiv.org/abs/2505.18878v1.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper/2023/hash/a3621ee907def47c1b952ade25c67698-Abstract-Conference.html.

Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, and Yang Liu. Agent Hospital: A Simulacrum of Hospital with Evolvable Medical Agents. *arXiv preprint arXiv:2405.02957*, 2024. URL https://arxiv.org/abs/2405.02957v3.

Xirui Li, Ming Li, Ion Stoica, Cho-Jui Hsieh, and Tianyi Zhou. ClawEnvKit: Automatic Environment Generation for Claw-Like Agents. *arXiv preprint arXiv:2604.18543*, 2026a. URL https://arxiv.org/abs/2604.18543v4.

Yuetai Li, Yichen Feng, Zhangchen Xu, Zixian Ma, Kaiyuan Zheng, Fengqing Jiang, Xinghua Sun, Rulin Shao, Zichen Chen, Yue Huang, Xinyang Han, Brian Lee, Kayla Xu, Shenglai Zeng, Hang Hua, Xiangliang Zhang, Basel Alomair, Ranjay Krishna, Luke Zettlemoyer, Pang Wei Koh, Bhaskar Ramasubramanian, Luyao Niu, Xiang Yue, and Radha Poovendran. JobBench: Aligning agent work with human will. *arXiv preprint arXiv:2605.26329*, 2026b. URL https://arxiv.org/abs/2605.26329v1.

Fanqing Meng, Lingxiao Du, Zijian Wu, Guanzheng Chen, Xiangyan Liu, Jiaqi Liao, Chonghe Jiang, Zhenglin Wan, Jiawei Gu, Pengfei Zhou, Rui Huang, Ziqi Zhao, Shengyuan Ding, Ailing Yu, Bo Peng, Bowei Xia, Hao Sun, Haotian Liang, Ji Xie, Jiajun Chen, Jiajun Song, Liu Yang, Ming Xu, Qionglin Qiu, Runhao Fu, Shengfang Zhai, Shijian Wang, Tengfei Ma, Tianyi Wu, Weiyang Jin, Yan Wang, Yang Dai, Yao Lai, Youwei Shu, Yue Liu, Yunzhuo Hao, Yuwei Niu, Jinkai Huang, Jiayuan Zhuo, Zhennan Shen, Linyu Wu, Hannah Yao, Charles Chen, Cihang Xie, Yuyin Zhou, Jiaheng Zhang, Zeyu Zheng, Mengkang Hu, and Michael Qizhe Shieh. ClawMark: A Living-World Benchmark for Multi-Turn, Multi-Day, Multimodal Coworker Agents. *arXiv preprint arXiv:2604.23781*, 2026. URL https://arxiv.org/abs/2604.23781v2.

Mike A. Merrill, Alexander G. Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, E. Kelly Buchanan, Junhong Shen, Guanghao Ye, Haowei Lin, Jason Poulos, Maoyu Wang, Marianna Nezhurina, et al. Terminal-Bench: Benchmarking

agents on hard, realistic tasks in command line interfaces. *arXiv preprint arXiv:2601.11868*, 2026. URL https://arxiv.org/abs/2601.11868v1.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–22, 2023. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, Natalie S. Kim, Patrick Chao, Samuel Miserendino, Gildas Chabot, David Li, Michael Sharman, Alexandra Barr, Amelia Glaese, and Jerry Tworek. GDPval: Evaluating AI model performance on real-world economically valuable tasks. Technical report, OpenAI, 2025. URL https://cdn.openai.com/pdf/d5eb7428-c4e9-4a33-bd86-86dd4bcf12ce/GDPval.pdf.

Ewa Roszkowska. Rank ordering criteria weighting methods–a comparative overview. *Optimum. Studia Ekonomiczne*, (5 (65)):14–33, 2013.

Daniel Shepard and Robin Salimans. AutomationBench. *arXiv preprint arXiv:2604.18934*, 2026. URL https://arxiv.org/abs/2604.18934v1.

Xiaoshuai Song, Haofei Chang, Guanting Dong, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. EnvScaler: Scaling Tool-Interactive Environments for LLM Agent via Programmatic Synthesis. *arXiv preprint arXiv:2601.05808*, 2026. URL https://arxiv.org/abs/2601.05808v2.

Zirui Tang, Xuanhe Zhou, Yumou Liu, Linchun Li, Yukai Wu, Weizheng Wang, Hongzhang Huang, Wei Zhou, Jun Zhou, Jiachen Song, Shaoli Yu, Jinqi Wang, Zihang Zhou, Hongyi Zhou, Yuting Lv, Jinyang Li, Jiashuo Liu, Ruoyu Chen, Chunwei Liu, Guoliang Li, Jihua Kang, and Fan Wu. Workspace-Bench 1.0: Benchmarking AI agents on workspace tasks with large-scale file dependencies. *arXiv preprint arXiv:2605.03596*, 2026. URL https://arxiv.org/abs/2605.03596v4.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Ziwei Chai, Y Charles, HS Che, Cheng Chen, et al. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026.

Dunwei Tu, Hongyan Hao, Hansi Yang, Yihao Chen, Yi-Kai Zhang, Zhikang Xia, Yu Yang, Yueqing Sun, Xingchen Liu, Furao Shen, et al. Scaleenv: Scaling environment synthesis from scratch for generalist interactive tool-use agent training. *arXiv preprint arXiv:2602.06820*, 2026.

Bertie Vidgen, Austin Mann, Abby Fennelly, John Wright Stanly, Lucas Rothman, Marco Burstein, Julien Benchek, David Ostrofsky, Anirudh Ravichandran, Debnil Sur, Neel Venugopal, Alannah Hsia, Isaac Robinson, Calix Huang, Olivia Varones, Daniyal Khan, Michael Haines, Austin Bridges, Jesse Boyle, Koby Twist, Zach Richards, Chirag Mahapatra, Brendan Foody, and Osvald Nitski. APEX-Agents. *arXiv preprint arXiv:2601.14242*, 2026. URL https://arxiv.org/abs/2601.14242v3.

Harsh Vishwakarma, Ankush Agarwal, Ojas Patil, Chaitanya Devaguptapu, and Mahesh Chandran. Can LLMs help you at work? a sandbox for evaluating LLM agents in enterprise environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9167–9201. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.466. URL https://aclanthology.org/2025.emnlp-main.466/.

Weixuan Wang, Dongge Han, Daniel Madrigal Diaz, Jin Xu, Victor Rühle, and Saravan Rajmohan. OdysseyBench: Evaluating LLM agents on long-horizon complex office application workflows. *arXiv preprint arXiv:2508.09124*, 2025. URL https://arxiv.org/abs/2508.09124v1.

Zhaoyang Wang, Canwen Xu, Boyi Liu, Yite Wang, Siwei Han, Zhewei Yao, Huaxiu Yao, and Yuxiong He. Agent World Model: Infinity Synthetic Environments for Agentic Reinforcement Learning. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.10090v3.

Zilong Wang, Yuedong Cui, Li Zhong, Zimin Zhang, Da Yin, Bill Yuchen Lin, and Jingbo Shang. OfficeBench: Benchmarking language agents across multiple applications for office automation. *arXiv preprint arXiv:2407.19056*, 2024. URL https://arxiv.org/abs/2407.19056v1.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5d413e48f84dc61244b6be550f1cd8f5-Abstract-Datasets_and_Benchmarks_Track.html.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026a.

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, Shuyan Zhou, and Graham Neubig. TheAgentCompany: Benchmarking LLM Agents on Consequential Real World Tasks. *arXiv preprint arXiv:2412.14161*, 2024. URL https://arxiv.org/abs/2412.14161v3.

Minrui Xu, Zilin Wang, Mengyi Deng, Zhiwei Li, Zhicheng Yang, Xiao Zhu, Yinhong Liu, Boyu Zhu, Baiyu Huang, Chao Chen, Heyuan Deng, Fei Mi, Lifeng Shang, Xingshan Zeng, and Zhijiang Guo. EnvFactory: Scaling Tool-Use Agents via Executable Environments Synthesis and Robust RL. *arXiv preprint arXiv:2605.18703*, 2026b. URL https://arxiv.org/abs/2605.18703v1.

Shuhan Xue, Jianyuan Zhong, Ziyuan Nan, Wenbin Li, Zhaochen Yu, Jinchao Ding, Qiang Gao, Pengyu Zhan, Yuntong Zhang, Tian Cheng, Zhenfei Yin, Yingcheng Wu, and Ling Yang. ScienceBuddy: Recursive-in-Recursive Self-Improvement for Interactive Scientific Agents. *arXiv preprint arXiv:2609.17523*, 2026. URL https://arxiv.org/abs/2609.17523v1.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. $\tau$-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. *arXiv preprint arXiv:2406.12045*, 2024. URL https://arxiv.org/abs/2406.12045v1.

Leijun Zhou, Zhihao Liu, Xiang Qu, Chenxu Liu, Yifei Liu, Yanke Yu, Jingzhe Xu, Xuejun Wu, Buyue Qian, Xi Chen, Yaowei Zheng, and Junhao Hu. GDPevo: Evaluating agent self-evolution on real business tasks. *arXiv preprint arXiv:2608.03764*, 2026. URL https://arxiv.org/abs/2608.03764v1.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, volume 2024, pp. 15585–15606, 2024a.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. Sotopia: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, volume 2024, pp. 40975–41019, 2024b.

# APPENDIX CONTENTS

**A Organization Construction and Simulation** **15**

A.1 Creating an Organization from a Natural-Language Description . . . . . 15

A.2 Example Organization and Configuration . . . . . 16

A.3 Communication and Work Coordination . . . . . 17

A.4 Sessions and Memory . . . . . 17

A.5 Workspaces, Tools, and Access Control . . . . . 18

A.6 Simulation Time and Recovery . . . . . 19

A.7 A Six-Month Office Simulation . . . . . 21

**B Data Construction and Benchmark Cases** **22**

B.1 Resource Membership and Separation . . . . . 22

B.2 Expert Review . . . . . 22

B.3 Domain and File-Type Coverage . . . . . 23

B.4 Benchmark Case Studies . . . . . 25

B.4.1 Financial Reconciliation . . . . . 25

B.4.2 Mechanical Fixture Design . . . . . 26

**C OfficeTown Bench Evaluation** **27**

C.1 Metrics and Scoring . . . . . 27

C.2 Reasoning and Interaction Budgets . . . . . 27

C.3 Coverage and Resource Accounting . . . . . 28

**D OfficeTown Bench Analysis** **29**

D.1 Weighted Scores by Domain . . . . . 29

D.2 Requirement Importance . . . . . 29

D.3 Input-File Groups . . . . . 30

D.4 Pass Rates within a Turn Budget . . . . . 31

**E Training Data and Model Adaptation** **32**

E.1 Training Resources . . . . . 32

E.2 SFT and On-Policy Distillation . . . . . 32

E.3 External Evaluation Protocol . . . . . 34

E.4 Training Dynamics . . . . . 35

E.5 Budget-Conditioned Outcomes . . . . . 36

# A Organization Construction and Simulation

## A.1 Creating an Organization from a Natural-Language Description

A construction request combines a description of the organization's purpose and activities (as short as one sentence) with optional settings for organization size, language, and simulation period. OfficeTown expands the request into a scope of work, departments, roles, reporting relationships, and synthetic customers. In the Juniper example (Figure 7a), departments divide client engagement, analytical work, data preparation, and review.

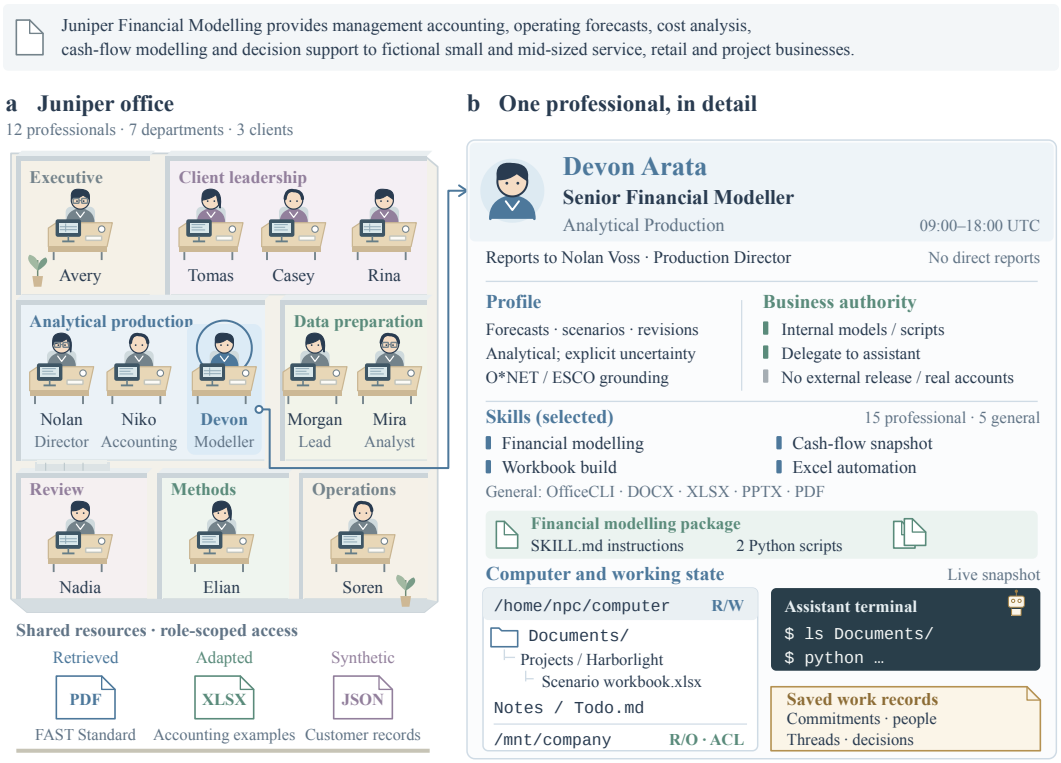



Figure 7: A generated office and one professional in detail. (a) Juniper's 12 professionals, grouped by department, with examples of retrieved, adapted, and synthetic seed materials. The brief above is an excerpt of the full request, which also includes configuration. (b) Devon Arata's role, manager, profile, business authority, selected skills, and computer. The blue connector opens the highlighted person's details; it is not a reporting relationship. Profiles and skills describe the generated configuration; files and saved work records show retained simulation state. The office layout, avatars, and terminal commands are illustrative. R/W denotes read/write access; R/O denotes read-only access, filtered by access-control rules (ACL).

Professional roles are grounded in occupational descriptions from O*NET and ESCO. Each professional receives a profile specifying their responsibilities, working style, schedule, and reporting relationships. Figure 7b expands one such role: Devon Arata, a Senior Financial Modeller who reports to Analytical Production Director Nolan Voss. Devon is equipped with 15 professional and five general skill packages, including financial modelling and spreadsheet tools. Packages are selected from a shared library, supplemented through skills.sh when needed, and retain their instructions and supporting resources. The expanded financial-modelling package contains instructions and two Python scripts.

The role also defines what the professional is authorized to do. Devon can prepare internal models, run analysis scripts, and delegate work to an assistant, but cannot release material externally or operate real financial accounts.

To build the seed, OfficeTown uses Web search to locate relevant professional guidance, reference documents, datasets, and worked examples. Retrieved materials are either retained in their original form or adapted to the organization's work. Synthetic records supply the organization-specific information needed to begin work, such as customer operating data, opening balances, and current business questions.

The seed preserves the distinction between retrieved sources, adapted materials, and synthetic records, illustrated by the reference PDF, accounting workbook, and customer data in Figure 7a. Source references accompany the materials, while file-format and structural checks establish that the supplied assets can be read and used by the corresponding tools. These checks concern the materials' accessibility and structure; professional judgments about their interpretation remain part of the simulated work.

OfficeTown assembles the organization description, profiles, skill packages, and seed into a runnable organization package. This package initializes personal workspaces, authorized shared resources, and the configured office services.

### A.2 Example Organization and Configuration

Juniper Financial Modelling (Figure 2) was constructed from a brief whose business-specific portion begins:

> Juniper Financial Modelling provides management accounting, operating forecasts, cost analysis, cash-flow modelling and decision support to fictional small and mid-sized service, retail and project businesses.

The remaining brief specifies the expected work products, review requirements, and use of synthetic customer records. Table 5 separates the generated organization properties from the settings used to construct and run the instance.

Table 5: Generated properties and recorded settings of the Juniper instance. The simulation period is a configured horizon, not a measure of completed work. These settings are specific to this instance.

| Configuration | Juniper instance |
|---|---|
| Generated organization | 12 professionals; 7 departments; 11 reporting relationships |
| Initial synthetic customers | 3 |
| Staffing target and limit | Target: 12; maximum: 16 |
| Occupational references | O*NET 30.3; ESCO 1.2.1 |
| Recorded generation model | GPT-5.5 |
| Professional and assistant models | GPT-5.5 |
| Assistant execution | Codex harness; worker containers; `direct_home` |
| Configured simulation period | 1 January–1 July 2026, 09:00 UTC; 181 days |
| Clock policy | Adaptive business clock; maximum rate: 120 simulated seconds per wall-clock second |
| Execution capacity | Up to 6 concurrent professional turns; 4 assistant workers |
| Assistant timeout | 1,200 seconds |
| Clock checkpoint interval | 5 seconds |
| Service deployment | Isolated deployment (`isolated_compose`) |
| Runtime security profile | `legacy-dev` |
| Run-level monitoring | Record-only |
| Additional semantic-memory backend | Disabled |

The seed combines professional references and an adapted management-accounting workbook with three synthetic customer packets containing 34 groups of business records. References include the FAST Standard and the ICAEW Financial Modelling Code. Harborlight's packet contains invoices,

payables, payroll records, dispatch jobs, opening balances, and recent bank activity, accompanied by control totals, source references, and unresolved assumptions.

The Harborlight packet anchors the 13-week liquidity discussion in Figure 8, which is described next.

### A.3 Communication and Work Coordination

OfficeTown connects professionals to running office services through service-backed tools: Rocket.Chat for direct messages and team channels, Mailpit for email, and Baïkal for calendars and meeting scheduling. Deliveries are also recorded in persistent, recipient-specific inboxes, which the simulation uses to present incoming work. A recipient receives the message and permitted references, not the sender's private working context.

The meeting organizer specifies the participants, time, and agenda. Each participant responds through their own model context, with preceding contributions supplied as part of the discussion. Turn-taking and explicit mentions determine who speaks next.

Work agreed upon in conversation is recorded through explicit actions. Requests identify the intended outcome, responsible person, relevant materials, and deadline. A professional can seek clarification, delegate execution to an assistant, or review a returned deliverable. Meeting notes summarize the discussion, but do not themselves create an approval or mark an assignment as complete. These changes require the corresponding work actions.

Figure 8 shows four consecutive turns from a Juniper meeting about a synthetic customer's liquidity. Client engagement, data preparation, modelling, and review roles establish the required inputs, handoffs, forecast scenarios, and checks before execution.

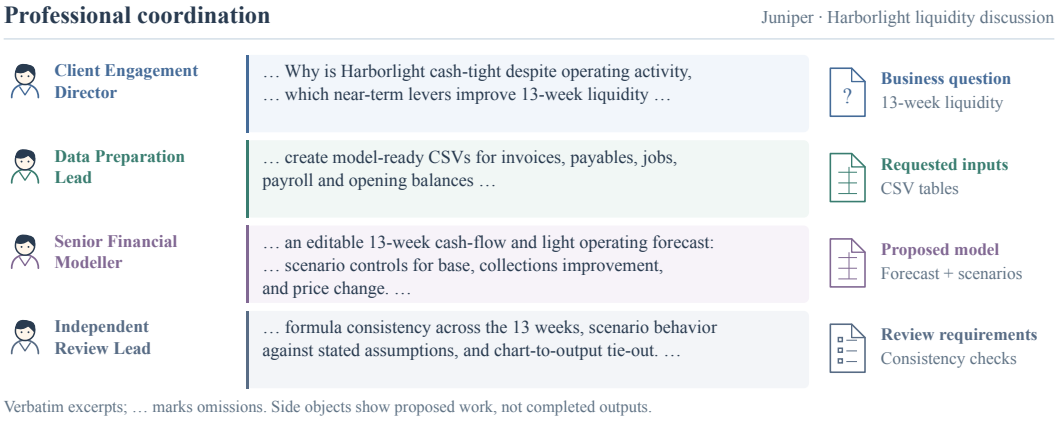



Figure 8: Professional coordination in a simulated office. Four consecutive speaking turns from a Juniper meeting about a synthetic customer's liquidity. Excerpts retain the original wording and order; ellipses mark omitted text. Side objects identify the proposed inputs, model, and review requirements, not completed or validated outputs.

### A.4 Sessions and Memory

Work records are stored independently of active conversation context. They retain commitments, ongoing discussions, unresolved decisions, and recent interactions with colleagues (Figure 9(b)). Commitments record what is owed and to whom; discussions identify who should respond next. Communication and work actions update these records through explicit requests and rule-based message cues. Clearing an incoming reminder does not mark the promised deliverable as complete. These structured records are distinct from the additional semantic-memory backend, which is disabled in the Juniper configuration (Table 5).

Only selected records enter each turn. Outstanding promises take priority over routine reminders, and discussions awaiting the professional's reply take priority over other exchanges. The default

limits are eight active commitments, five discussions, five unresolved decisions, and ten colleague records. Professionals can also maintain personal notes containing working assumptions, document references, and follow-up plans. Recent notes are included within a separate size budget, while current project and request states are read from the simulation's stored records.

OfficeTown also supports an optional persistent-session mode for continuing coordination. The main session retains available conversation context alongside current requests and project state. Detailed inspections take place in a branch of the same professional's session (Figure 9(a)). The branch inherits the current context and supports file inspection and tool use. In the illustrative forecast review, the professional identifies missing source links and requests a revision. A concise record of the conclusion, actions, and tools used returns to the main session, without retaining the full inspection in its active context.

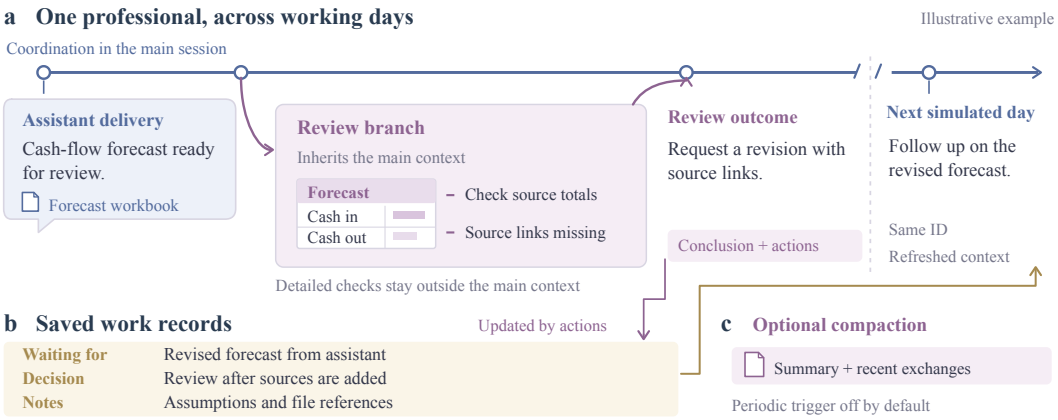



Figure 9: Continuing work through sessions and saved records. (a) In persistent-session mode, the same professional inspects a returned forecast in a review branch and brings its conclusion, actions, and a brief tool record back into coordination. (b) Actions update work records; selected records and personal notes remain available when the context is refreshed under the same identifier on the next simulated day. (c) Compaction is optional; periodic triggering is off by default. The forecast, messages, and records form an illustrative example, not a recorded Juniper interaction.

At simulated day boundaries in this mode, the main context is refreshed by reopening the session under the same identifier with saved work records and current office state. The pending forecast revision in the illustration can therefore inform a later turn without preserving the full earlier conversation. OfficeTown also supports optional compaction, which summarizes earlier exchanges while retaining recent messages (Figure 9(c)). Its periodic trigger is disabled by default.

### A.5 Workspaces, Tools, and Access Control

An assistant works in a temporary execution container, while the principal's computer workspace persists across assignments (Figure 10(a)). Professionals use this workspace to inspect and edit files, execute terminal commands, and access their skill packages.

In direct-home mode, the runtime provides a writable copy of the principal's current home at `/home/npc/computer`, together with authorized shared references. A runtime-generated instruction block identifies the principal and distinguishes writable files, read-only inputs, and temporary scratch space. Tasks requiring narrower access instead receive an isolated view with selected inputs and designated working and submission directories. Committed work remains in the professional's persistent workspace after the execution container is released.

Shared materials are presented through a separate, permission-filtered company view at `/mnt/company`. Ownership, role permissions, explicit sharing rules, and confidentiality classifications determine which documents appear in this view. A reporting relationship does not grant unrestricted access to another person's files. Other professionals' private homes are not mounted in the assistant's environment. Shared documents are read-only: professionals and assistants can consult them or create their own working copies, but cannot directly overwrite the shared originals. File

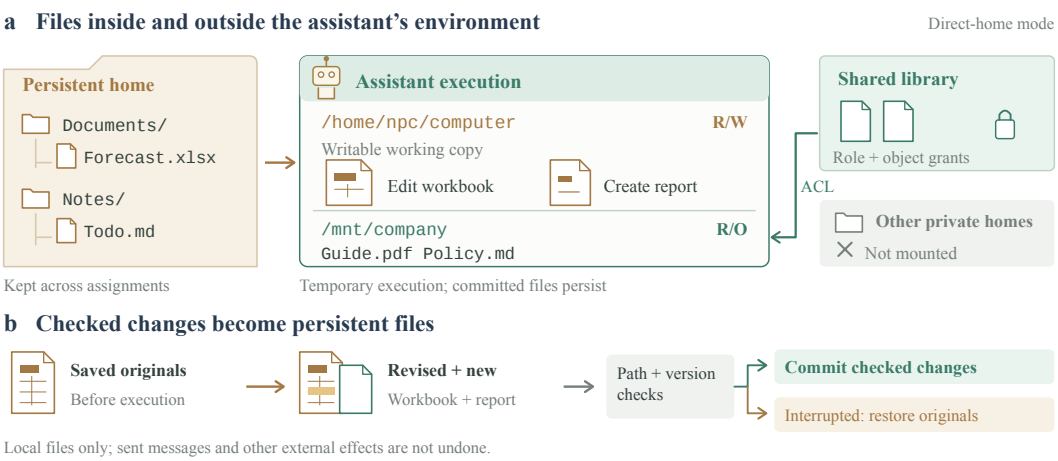



Figure 10: A bounded file view and persistent local changes. (a) In direct-home mode, an assistant receives a writable copy of its principal's home and a permission-filtered, read-only company view. Other private homes are not mounted. R/W and R/O denote read/write and read-only access; ACL denotes access-control rules. (b) The runtime checks local changes before committing them; an interrupted commit can restore the preceding files and classification state. The files and changes are illustrative, not execution results. Recovery does not undo messages or other effects already sent to external services.

operations and terminal execution enforce these boundaries through path checks, read-only mounts, and sandboxing.

The revised workbook and new report in Figure 10(b) enter the persistent workspace only after the runtime checks protected paths, file validity, and the underlying workspace revision. Writers to the same professional's home are serialized to avoid conflicting updates. The runtime records the actual files created, modified, moved, or deleted, independently of the assistant's completion report. The professional receives the deliverable locations, reported checks, and unresolved issues for review. Filesystem validation establishes what was saved; professional review determines whether the content meets the assignment's requirements.

For broader publication, the professional selects an accepted deliverable and its intended audience. The runtime checks the acceptance record, file version, classification, and recipient permissions before making it available to authorized colleagues. Personal working materials can therefore accumulate without exposing every intermediate draft to the whole organization.

### A.6 Simulation Time and Recovery

Simulated time is managed separately from wall-clock execution time. When adaptive pacing is enabled, OfficeTown slows or pauses the simulated clock as unfinished work accumulates, then resumes its advance as the backlog decreases. The clock is periodically saved together with the run's original start and target times. A resumed run continues toward that target rather than beginning another full simulation period.

Routine activations follow working hours and meeting schedules; designated interruptions can receive earlier attention. Professionals act asynchronously within limits on concurrent model calls and assistant workers. Work awaiting capacity remains pending.

Incoming messages remain in persistent inbox files, with acknowledgments tied to the exact source rows processed. Reading or queuing a message does not acknowledge it as processed. A partially processed batch retains acknowledgments even when rows finish out of order, leaving unprocessed rows available for later delivery. Capacity-deferred delegations retain their original identities and arguments when reconsidered for execution.

Workspace recovery follows the checked-update boundary shown in Figure 10(b) and described in Section A.5. The runtime saves the preceding workspace state and journals an update before

committing changes. An interruption before commit leaves the persistent workspace unchanged; an interruption during commit can be recovered by restoring the saved files and classification metadata. The recovery is recorded explicitly. This mechanism applies to local workspace changes, not to reversing an email already sent or another external service action.

The runtime also monitors its action bridge and model-session service and supports replacing failed services. When the simulation process remains alive, active sessions can be reopened using saved conversation records and retained session metadata. This reconstructs context for subsequent turns rather than resuming an interrupted model generation internally. Transient connection failures receive bounded retries, and unsuccessful attempts remain recorded.

Run-level monitoring can stop execution on detected violations or record them while allowing an endurance run to continue. The latter policy leaves operation-level authorization and validation checks in force. At shutdown, the runtime stops new activity, allows a bounded interval for ongoing work to settle, and reconciles assistant results with delivery records. Unfinished requests, failed attempts, and incomplete reviews remain in the saved state; reaching the target time does not mark them as finished.

### A.7 A Six-Month Office Simulation

We examine material accumulation in the Juniper instance over 181 simulated days, from 1 January to 1 July 2026. These dates and all checkpoints in Figure 11 follow the simulation clock (Section A.6), not elapsed wall-clock time. Materials are counted at their first recorded simulated delivery, with repeated revisions counted once.

Figure 11(a) shows the cumulative collection growing from 27 initial files to 209, 376, and 497 files by simulated days 60, 120, and 181, respectively. The additional materials span documents and notes, data files, workbooks, and other formats. Figure 11(b) places these counts alongside selected materials available by each checkpoint: customer operating records, a scenario workbook, a retail-project source catalog, and a reusable workbook component. This observation describes material accumulation; it does not assess deliverable quality or task success.

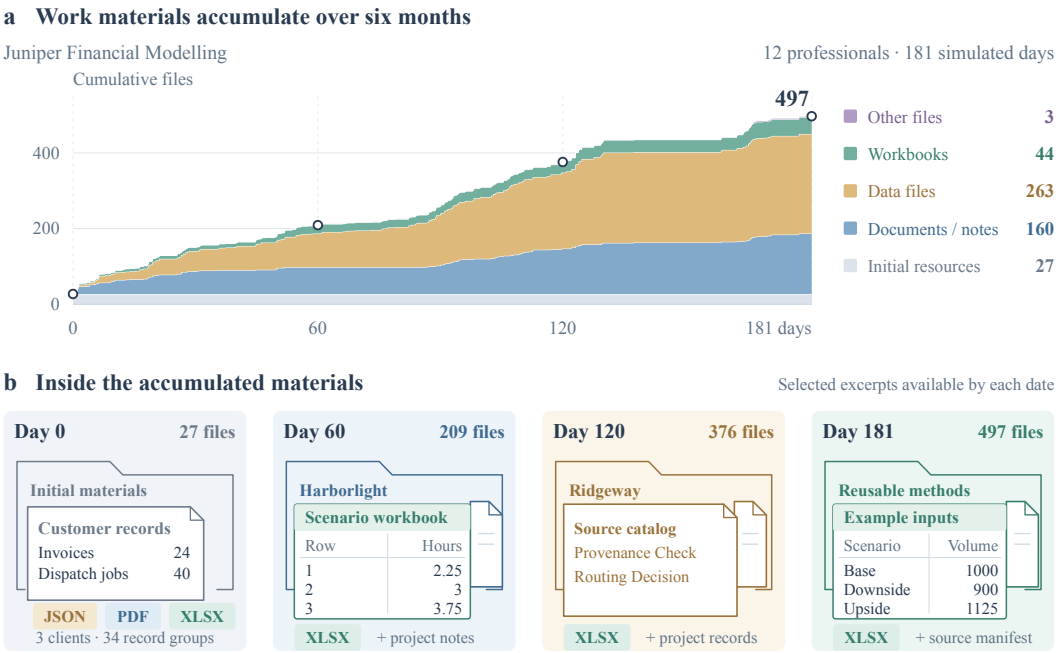



Figure 11: Work materials accumulated during a six-month Juniper simulation. All dates and days denote simulated calendar time, not wall-clock runtime; the adaptive clock adjusts its rate to work-processing pressure. (a) Cumulative material files: 27 distinct initial resource contents and 470 logical output files, counted at first simulated delivery. Revisions and recorded renames count once; exact copies are merged at first appearance. Navigation READMEs, explicit backups, and trash paths are excluded. Initial resources are grouped separately from subsequent outputs. (b) Selected material excerpts available by each date, redrawn from retained source versions; the final workbook contains synthetic scenario inputs. Counts describe the whole organization, not the illustrated folders.

# B Data Construction and Benchmark Cases

## B.1 Resource Membership and Separation

Figure 3, Table 1, and Tables 6–7 describe the same 406 benchmark tasks. Benchmark tasks, SFT demonstrations, and OPD environments have separate membership records; training-resource construction is described in Appendix E.

Table 1 counts source organizations by their recorded organization identities, counting repeated simulations of the same identity once. Professional roles are distinct job titles of the primary delegating professionals, with case and whitespace normalized; they are not personnel counts or standardized occupational categories.

Input-file counts include supplied company and personal-workspace files, historical service readbooks, and supplied service-state records. They exclude task-packaging metadata, runtime and evaluation files, and duplicate path aliases; an archive counts as one supplied file. The task-wise total counts shared materials once per task environment; unique files are deduplicated across environments by their SHA-256 content hash. The table reports per-task means and medians.

Rubric counts refer to the 4,836 criteria in the final evaluation instrument, not their subordinate verification checks. Task and criterion selection used multi-model pilot results; eligible pilot judgments were reused for retained tasks. Aggregate statistics, model comparisons, and case studies use the same fixed task and criterion sets.

Source task identities remain recorded, but task-identity separation alone does not establish project-level or semantic isolation. Exposure to external task instances, answers, rubrics, and test feedback is distinct from using a benchmark's format or methodology.

## B.2 Expert Review

To ground tasks and evaluation criteria in professional practice, we engaged 71 domain experts across eight professional domains during benchmark construction. Task-specific expertise requirements covered relevant specialization and occupational background, the ability to make independent professional judgments, and familiarity with native working materials and applicable standards. The construction-stage annotation records cover 1,051 candidate tasks and 32,274 rubric criteria. Among 1,011 recorded task-level familiarity assessments, 98.9% indicated that reviewers were familiar or very familiar with the assigned work. Experts assessed task realism and feasibility and examined rubric correctness, coverage, and importance, providing 3,842 criterion-level recommendations to edit, remove, merge, or split criteria. The annotations subsequently underwent a separate review pass, combining expert assessment with follow-up quality control and producing refinement feedback traceable to individual tasks and criteria.

The review instructions ask reviewers to assess task realism, objective clarity, input sufficiency, feasibility, deliverable clarity, permission boundaries, answer leakage, and unnecessary restrictions. Rubric review separately examines task alignment, professional correctness, assessability, clarity, redundancy, completeness, importance, evidence gaps, and valid alternative solutions.

Reviewers record separate overall decisions for the task and rubric, with reasons and proposed revisions. For individual criteria, they can retain the standard or recommend the revisions described above, identify missing requirements, and assign the importance levels introduced in Section 3.2. These decisions concern the evaluation instrument, not an agent's answer. Expertise or evidence limitations are recorded rather than treated as approval.

Critical (P0) failures make the work essentially invalid or unusable; Major (P1) failures substantially affect its intended outcome or professional usability. Minor (P2) deficiencies do not substantially undermine the core outcome; Optional (P3) criteria describe desirable refinements with limited impact on overall success.

Concrete edits remain linked to the reviewed task and criterion identities. For the final benchmark, historical expert feedback is bound to 394 of the 406 retained tasks; the other 12 have model-based review without a bound human annotation. Subsequent task and rubric adaptation and criterion selection do not constitute a new round of human review.

### B.3 Domain and File-Type Coverage

Table 6 expands the domain overview in Figure 3a with exact task and source-role totals. The eight domain petals cover all 406 tasks, with equal angles and areas proportional to task counts. Domain-level title counts are not additive because the same title can occur in more than one domain.

Table 6: OfficeTown Bench domains and source role titles. Role titles are deduplicated within each domain; titles can recur across domains. The total uses global deduplication.

| Professional domain | Tasks | Role titles |
|---|---|---|
| Assurance | 47 | 14 |
| Clinical | 28 | 14 |
| Design | 103 | 25 |
| Entertainment | 55 | 17 |
| Finance | 67 | 26 |
| Industrial | 48 | 18 |
| Legal | 13 | 10 |
| Software | 45 | 18 |
| Total (distinct titles) | 406 | 139 |

Figure 12 resolves each domain into the five source role titles associated with the most tasks, with remaining titles and tasks summarized below each panel. Ties are ordered alphabetically.

Table 7 gives exact input-type counts across all 406 tasks, complementing Figure 3b. Input coverage counts a file family once per task, including actual members of supplied ZIP and other archives. Archives are inspected recursively without unpacking Office documents into their internal XML components. Input coverage describes available materials, including reference repositories and dependencies; it does not assert that every file is necessary or used by an agent. File counts and type coverage use different units: an archive is one supplied file, but can contribute several types.

Categories overlap. Text and markup include TXT, Markdown, and HTML; documents include PDF and Word; tables include spreadsheet and CSV formats; structured data includes JSON, XML, and configuration files. Code, images, CAD/3D, and audio/video are classified by their file formats. Unclassified inputs remain in Other files. These are file-carrier categories, not occupational or task-purpose labels. These counts describe supplied inputs, not model-generated outputs.

Table 7: Input-file type coverage in OfficeTown Bench. Each type is counted once per task across all 406 tasks. Input types include archive contents. Categories overlap.

| File type | Input tasks |
|---|---|
| Text / markup | 406 |
| Documents | 318 |
| Tables | 280 |
| Structured data | 406 |
| Code | 250 |
| Images | 195 |
| CAD / 3D | 33 |
| Audio / video | 55 |
| Other files | 126 |

Figure 3c uses equal-width bins of 20 input files over the full observed range, starting at the observed minimum of 11 files. The maximum is 390 files; the last bin ends at 391. Bar height is the number of tasks in each bin. The curve is a Gaussian kernel density estimate with a 10-file bandwidth, reflected at zero and multiplied by the task count and bin width to match the frequency axis. Only the displayed range is drawn; the observed minimum is not treated as a population support boundary. It summarizes the same observations, not uncertainty or additional samples. All 406 tasks are included; no tail is truncated.

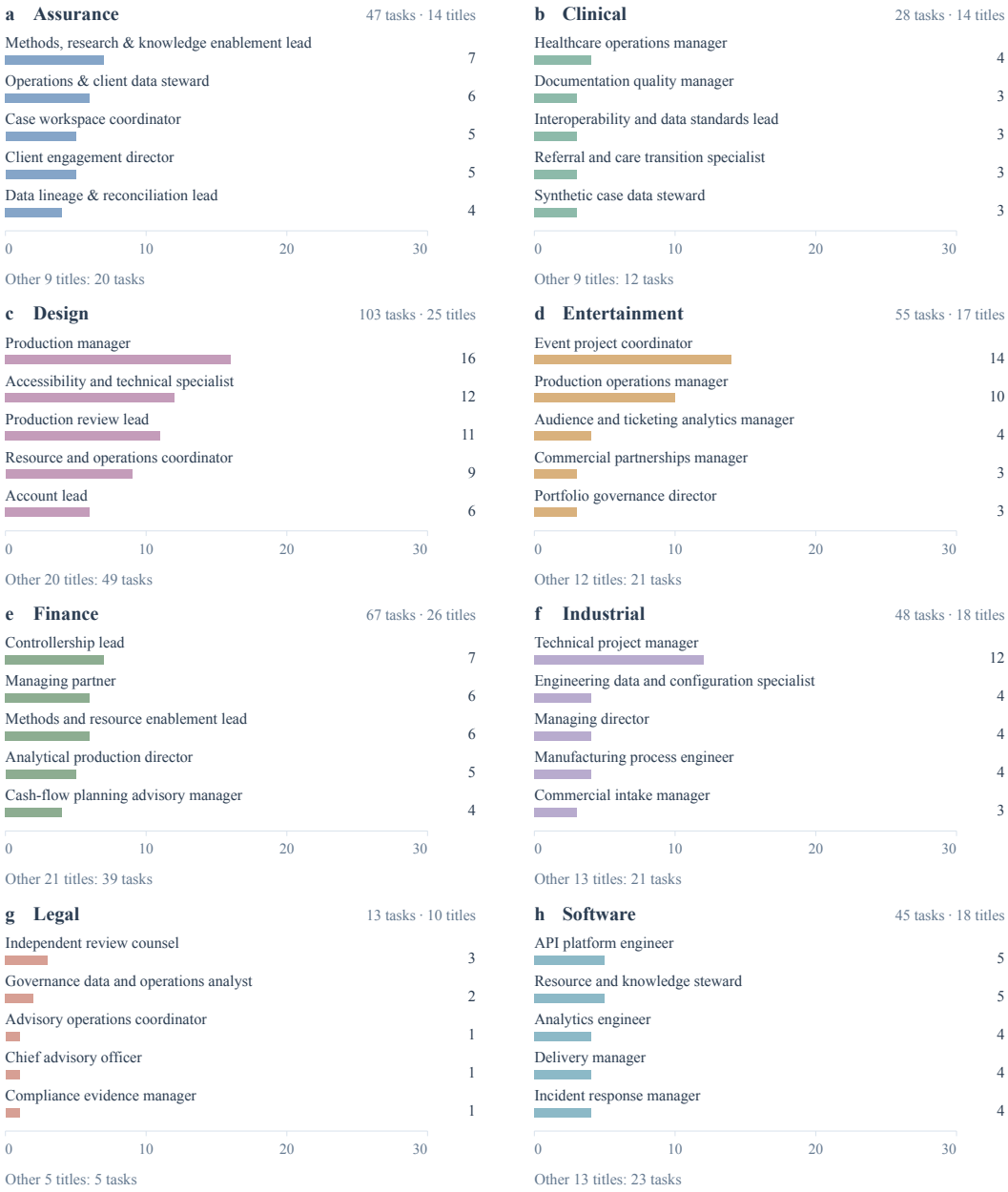



Figure 12: Source roles in OfficeTown Bench. Each panel shows the five most frequent delegating role titles and their task counts on a shared scale. Remaining titles and tasks are summarized below each panel. Titles are source job names, not standardized occupations.

### B.4 Benchmark Case Studies

The cases in Figures 13 and 14 illustrate different professional requirements: tracing financial discrepancies to accounting evidence, and relating fixture geometry to calculations and inspection access. Complete requests, criteria, exact weights, and initial-file inventories are retained in the case records.

#### B.4.1 Financial Reconciliation

In the Riverbend case, a Reconciliation Operations Manager requests Excel workpapers for three exception streams: customer deposits, card clearing, and vendor credits. The agent receives an input brief, an operating-data packet, a native workbook, and a triage memo. It must reconcile the available records, explain unresolved differences, and identify the evidence needed before an adjustment or period-close conclusion can be supported.

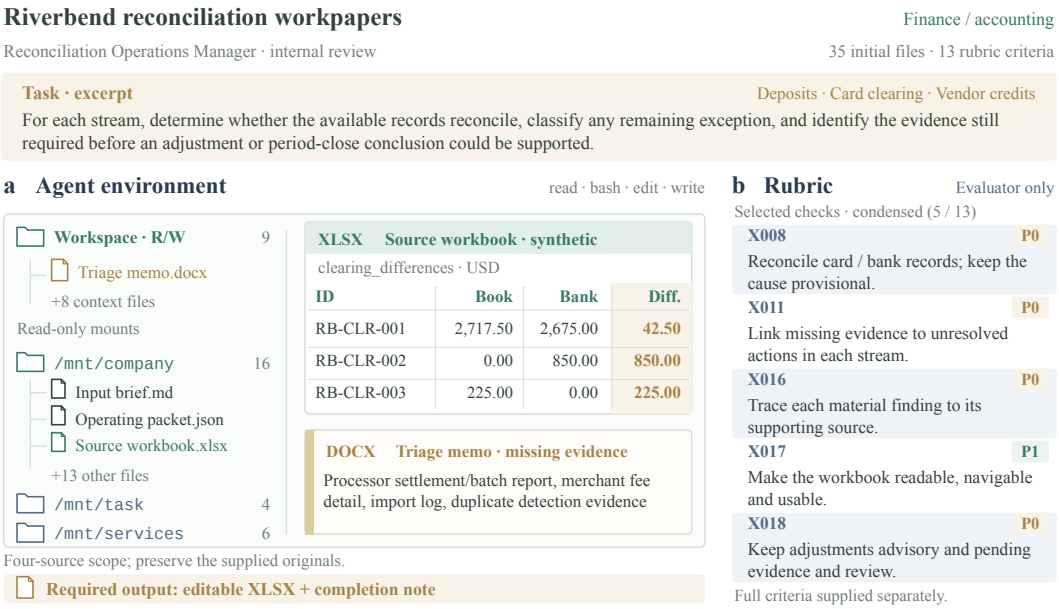



Figure 13: Riverbend reconciliation case. (a) Initial files, selected columns from the supplied workbook, and a missing-evidence entry from the triage memo. Amounts are synthetic task inputs, not model results. (b) Five condensed checks from the 13-criterion evaluator-only rubric. P0 and P1 denote critical and major importance. Filenames and table headings are shortened; the 35 initial files include task and metadata carriers. Service files are frozen historical projections. Required outputs appear below the input tree; the supplied originals must remain unchanged.

The card-clearing entry in Figure 13(a) contains a $42.50 difference; the memo identifies missing settlement records and fee details needed to investigate it. The requested workpapers must reconcile the three listed differences to $1,117.50, cite the supporting records, and keep unsupported conclusions open. The 13-criterion rubric assesses numerical consistency, the basis for each proposed disposition, and editable-workbook usability (selected checks in Figure 13(b)). A balanced total alone does not establish that an exception is resolved.

### B.4.2 MECHANICAL FIXTURE DESIGN

In the Oakridge case, a Mechanical Design Manager requests two materially distinct clamp-and-locate fixture concepts for a supplied workpiece. The starting materials include a STEP model, datum and feature records, operating assumptions, and engineering references. The agent must preserve the workpiece and provide editable geometry or reproducible geometry-generator source for each concept, together with calculations, annotated SVG comparison views, a technical note, and a package manifest.

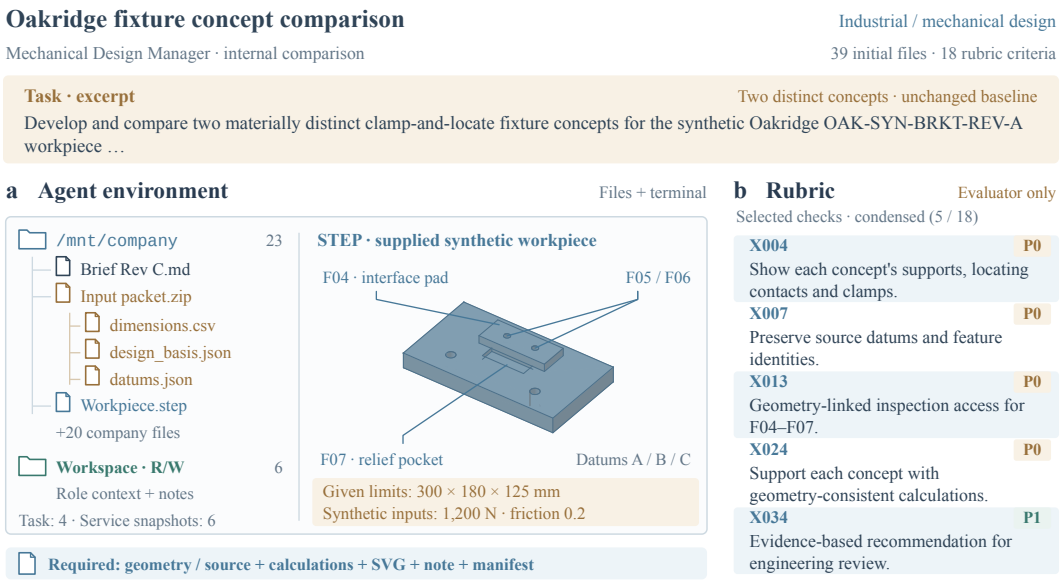



Figure 14: Oakridge fixture-design case. (a) Initial files and the supplied synthetic STEP workpiece, not a proposed fixture. Indented ZIP children are archive members; filenames are shortened. The 39 initial files include task and metadata carriers. The given limits specify the maximum fixture footprint and stack height, not workpiece dimensions. The load and friction coefficient are synthetic inputs. (b) Five condensed checks from the 18-criterion evaluator-only rubric, with the same importance notation as Figure 13. Required deliverables appear below the input tree.

Relevant workpiece features include the interface pad, pin holes, and relief pocket (Figure 14(a)). The design must respect the given equipment envelope and permit the required inspection access; using the pin holes as functional locators remains unresolved in the supplied evidence. The 18-criterion rubric checks whether the geometry can be inspected or rebuilt, whether access claims agree with the geometry, and whether the supporting calculations can be reproduced (selected checks in Figure 14(b)). Rendered views alone are insufficient, and the recommendation is for further engineering review rather than production approval.

# C OFFICETOWN BENCH EVALUATION

## C.1 METRICS AND SCORING

OfficeTown Bench evaluates submitted artifacts and observable outcomes against its fixed set of 4,836 task-specific criteria. Weighted Score first measures fulfilled rubric weight within each task, then averages across tasks. Let $G_i(c)$ be criterion $c$'s original weight group before filtering, and $p(c)$ its importance tier. The criterion weight is

$$w_{ic} = \frac{\alpha_{p(c)}}{|G_i(c)|}, \qquad (\alpha_{P0}, \alpha_{P1}, \alpha_{P2}, \alpha_{P3}) = (25, 13, 7, 3).$$

These are proportional to rank-order centroid weights for the four tiers (Roszkowska, 2013). Filtering retains the original shares: removing a criterion does not redistribute its weight to surviving members. For the retained applicable criteria $A_i$,

$$s_i = \frac{\sum_{c \in A_i} w_{ic}\, \mathbf{1}[y_{ic} = \text{met}]}{\sum_{c \in A_i} w_{ic}}, \qquad \text{WS} = \frac{100}{N} \sum_{i=1}^{N} s_i.$$

Here $N$ is the number of tasks in the overall or domain population. Non-applicable criteria are excluded; unmet and unassessable criteria contribute zero. Tasks without applicable criteria score zero and do not pass. Each task has equal weight in the average; criteria are not pooled across tasks. Pass@1 requires at least one applicable required criterion and first-attempt satisfaction of every such criterion.

GPT-5.5 high judges Table 2; reused judgments require matching tasks, answers and criteria. Designated terminal evaluation failures score zero, with failure labels retained: Claude Opus 5.5 and Grok-4.7 include 14 and 15 such outcomes, respectively. These task-level zeros do not create criterion judgments or apply to pending evaluations. Sol's Overall Weighted Score remains pending. Expert review of tasks and rubrics (Appendix B) is separate from model-output scoring and does not establish automated-judge agreement with human graders. Release and scoring bindings accompany the table.

## C.2 REASONING AND INTERACTION BUDGETS

A logical turn is one model-response round and may contain several tool calls. Claude's ordered content fragments within one response count as one turn; native Pi and OpenCode records distinguish admitted calls from streamed updates and local budget-rejection messages. HTTP retries are not additional logical turns. Each framework retains its native prompts and tools.

The curves use recorded final grades and execution-ending turns. At budget $T$, an outcome contributes its final score if execution has ended within $T$ turns, and zero otherwise, with a fixed denominator. This includes graded partial outcomes from executions without a final answer. Nonnegative score accumulation makes the curves monotone; it neither scores intermediate artifacts nor reruns tasks under shorter stopping budgets. The figures display five fixed budgets (8, 16, 24, 32, and 48 turns), with connecting lines as visual guides; exact threshold crossings use the full records.

All three panels use the same 406 tasks and final rubric. The model panel uses the verl-based single-agent OfficeTown Executor, distinct from the principal–assistant OfficeTown Harness. Shared limits are 48 logical turns, 900 seconds and 262,144 total output tokens, with a 32,768-token ceiling per request; Claude Opus 5.5 retains its 16,000-token request cap.

The effort panel compares requested Low and High for DeepSeek-V4-Flash with the same inputs, environments, executor and shared limits; Medium is not included. The framework panel fixes DeepSeek-V4-Flash at High. Terminal evaluation errors remain task-level zeros: 34 for Low, 4 for High in OfficeTown Executor, 18 for Pi, and 8 for OpenCode. Ordinary judged failures also remain included; tasks are not dropped to equalize valid-judgment counts. These zeros follow Appendix C.1, without invented criterion judgments. Figure 15 uses the same records with binary passes.

Training evaluation protocols and curves are in Appendices E.3 and E.5. Both curve analyses reuse existing executions and grades, without new inference, intermediate-state replay or grading.

### C.3 Coverage and Resource Accounting

Coverage distinguishes scored attempts, agent errors, budget exhaustion, infrastructure failures and grading failures. Unreported scores remain unresolved; the designated task-level zeros follow Appendix C.1. Shared-task comparisons use paired observations. Repeated attempts and criteria within a task are not independent samples, and tasks from the same organization or project may also be dependent; aggregate means alone do not determine confidence intervals.

For multi-role systems, accounting must distinguish principal, assistant, reviewer and retry activity, reporting native outer-loop counts separately from response rounds summed over roles. Input, cached-input, output and available reasoning tokens retain their original meanings and are not double-counted. Neither equal turn limits nor completion thresholds establish equal token or monetary cost; monetary comparisons also depend on the pricing version.

# D OFFICETOWN BENCH ANALYSIS

## D.1 WEIGHTED SCORES BY DOMAIN

Table 8 complements the Pass@1 leaderboard with rubric-weighted performance on the same tasks. It captures partial fulfillment under the weighting, applicability and missing-outcome rules in Appendix C.1.

Table 8: OfficeTown Bench Weighted Score (%; higher is better). 1st 2nd 3rd : ranks within each column at displayed precision; ties share rank and skip positions. Models, tasks and high-effort settings match Table 2. Dashes indicate pending results. Domain abbreviations denote assurance, clinical, design, entertainment, finance, industrial, legal, and software work.

| **Model** | Overall | Assur. | Clin. | Des. | Ent. | Fin. | Ind. | Legal | Soft. |
|---|---|---|---|---|---|---|---|---|---|
| *OpenAI* | | | | | | | | | |
| GPT-6 Astra | 88.4 | 85.8 | 97.9 | 79.3 | 85.3 | 93.7 | 91.1 | 92.3 | 98.2 |
| GPT-5.6 Sol | 68.5 | 67.7 | 64.2 | 66.7 | 72.4 | 68.1 | 73.2 | 81.7 | 62.7 |
| GPT-5.6 Terra | 76.1 | 80.0 | 64.8 | 79.7 | 78.0 | 73.5 | 75.3 | 70.6 | 75.0 |
| GPT-5.6 Luna | 75.4 | 83.6 | 84.1 | 71.6 | 81.4 | 74.0 | 71.6 | 88.4 | 64.9 |
| GPT-5.4 | 72.2 | 73.0 | 75.7 | 70.0 | 80.5 | 69.8 | 70.9 | 59.7 | 72.2 |
| GPT-5.2 | 64.3 | 65.4 | 71.5 | 59.4 | 73.1 | 63.9 | 62.0 | 61.2 | 63.1 |
| *Anthropic* | | | | | | | | | |
| Claude Opus 5.5 | 67.5 | 68.8 | 78.0 | 63.1 | 71.0 | 70.4 | 61.8 | 75.6 | 64.6 |
| *DeepSeek* | | | | | | | | | |
| V4-Pro | 64.2 | 66.4 | 67.9 | 62.3 | 71.8 | 61.0 | 62.1 | 54.9 | 63.9 |
| V4-Flash | 61.9 | 70.1 | 62.7 | 58.8 | 63.5 | 59.9 | 56.7 | 67.1 | 65.2 |
| *Moonshot AI* | | | | | | | | | |
| Kimi-K2.5 | 40.4 | 39.0 | 43.6 | 41.3 | 45.1 | 35.9 | 35.4 | 46.1 | 42.9 |
| *xAI* | | | | | | | | | |
| Grok-4.7 | 77.2 | 81.9 | 83.3 | 74.1 | 76.1 | 79.2 | 75.0 | 73.8 | 77.3 |
| Grok-4.3 | 43.3 | 54.4 | 47.0 | 41.6 | 48.9 | 37.4 | 34.6 | 35.8 | 46.3 |
| *Mistral AI* | | | | | | | | | |
| Mistral-Large-3 | 20.3 | 25.0 | 17.3 | 19.7 | 29.7 | 11.2 | 18.6 | 18.2 | 23.3 |

## D.2 REQUIREMENT IMPORTANCE

Figure 5 and Table 9 use the retained P0–P3 labels for critical, major, minor and optional requirements, preserving their expert-inherited, expert-adapted or model-proposed provenance. The twelve models supply 4,825 valid task-level judgments from 4,872 assigned executions. The other 47 outcomes retain their declared task-level zeros without invented criterion failures. Models lacking bound item-level records are omitted, not replaced with partial or historical evaluations.

Within each task and tier, fulfillment is the ratio of satisfied to applicable criterion weight, averaged across eligible tasks. A critical miss occurs when at least one applicable P0 criterion is unmet or unassessable. Its rate uses validly judged, P0-bearing tasks; Pass@1 instead covers all required criteria under the main table's reporting population.

Table 9: Requirement fulfillment by importance, and tasks with a critical miss (%). Fulfillment is task-averaged; higher is better. P0 misses are task-level rates; lower is better. Eligible task counts vary by model: P0 390–405, P1 347–360, P2 49–52, and P3 8. The retained instrument contains 3,156/1,582/90/8 criteria at these tiers before outcome-specific applicability checks. The small P2/P3 populations should not be read as broad capability estimates.

| Model | P0 | P1 | P2 | P3 | P0 misses |
|---|---|---|---|---|---|
| GPT-6 Astra | 88.8 | 88.0 | 86.7 | 87.5 | 20.8 |
| GPT-5.6 Terra | 76.5 | 76.9 | 82.2 | 62.5 | 53.3 |
| GPT-5.6 Luna | 76.1 | 74.5 | 72.8 | 50.0 | 58.3 |
| GPT-5.4 | 72.2 | 73.5 | 82.5 | 62.5 | 59.6 |
| GPT-5.2 | 64.9 | 64.7 | 76.0 | 62.5 | 67.6 |
| Claude Opus 5.5 | 70.4 | 70.6 | 62.2 | 50.0 | 41.4 |
| DeepSeek V4-Pro | 65.1 | 63.2 | 65.6 | 75.0 | 72.0 |
| DeepSeek V4-Flash | 62.9 | 62.3 | 70.5 | 75.0 | 78.3 |
| Kimi-K2.5 | 42.2 | 37.8 | 50.0 | 50.0 | 89.0 |
| Grok-4.7 | 80.6 | 79.3 | 79.6 | 75.0 | 33.3 |
| Grok-4.3 | 43.4 | 42.2 | 53.7 | 37.5 | 86.4 |
| Mistral-Large-3 | 21.2 | 18.8 | 17.6 | 0.0 | 96.8 |

GPT-6 Astra fulfills 88.8% of critical criterion weight on average, yet 84 of 404 eligible tasks have a critical miss. The counts are 216/405 for GPT-5.6 Terra, 314/401 for DeepSeek-V4-Flash, 162/391 for Claude Opus 5.5 and 130/390 for Grok-4.7. These rates depend on the number and content of applicable requirements and complement, rather than replace, the benchmark scores.

### D.3 Input-File Groups

Tasks are grouped by supplied input types before inspecting model outputs. Groups overlap because a task may contain several types. Each column reports task-level Weighted Score, not per-file performance, required output types or whether the agent inspected every supplied file.

Table 10: Weighted Score by supplied input-file type (%). Group sizes are 318/280/250/195/33/55 tasks, respectively. Groups overlap and retain the main table's declared zeros. Plain-text and structured inputs occur in every task and would repeat the aggregate; slides have no input coverage, and heterogeneous residual formats are not a single interpretable group.

| Model | Docs | Tables | Code | Images | CAD | Media |
|---|---|---|---|---|---|---|
| GPT-6 Astra | 85.9 | 87.3 | 85.4 | 83.4 | 90.9 | 78.8 |
| GPT-5.6 Terra | 77.9 | 76.4 | 78.9 | 80.3 | 68.1 | 83.0 |
| GPT-5.6 Luna | 76.7 | 75.7 | 78.4 | 77.2 | 71.4 | 73.7 |
| GPT-5.4 | 74.6 | 74.4 | 74.9 | 76.0 | 66.5 | 68.6 |
| GPT-5.2 | 65.0 | 62.7 | 66.3 | 65.6 | 59.2 | 63.1 |
| Claude Opus 5.5 | 64.9 | 67.3 | 63.0 | 66.4 | 59.2 | 67.1 |
| DeepSeek V4-Pro | 63.3 | 62.2 | 62.0 | 62.5 | 56.9 | 62.8 |
| DeepSeek V4-Flash | 62.7 | 60.5 | 61.9 | 62.5 | 54.8 | 57.5 |
| Kimi-K2.5 | 39.2 | 37.2 | 40.7 | 42.1 | 28.0 | 37.2 |
| Grok-4.7 | 75.9 | 76.3 | 75.5 | 77.3 | 69.9 | 77.6 |
| Grok-4.3 | 42.3 | 41.2 | 43.3 | 45.5 | 30.3 | 41.3 |
| Mistral-Large-3 | 19.7 | 20.1 | 21.4 | 23.7 | 11.8 | 23.7 |

GPT-6 Astra and GPT-5.6 Terra reverse order between the CAD and media groups. Kimi-K2.5 (Team et al., 2026) scores 40.7 on code-containing tasks and 28.0 on CAD-containing tasks. These are task-population differences, not isolated file-format effects: domain, requested work and accompanying inputs vary together.

### D.4 Pass Rates within a Turn Budget

Figure 15 replaces Weighted Score with binary passes while retaining Figure 6's records, populations and ending-turn rule. Model-panel endpoints reproduce Table 2; effort and framework panels also cover the complete benchmark. At 48 turns, Low and High reach 16.5% and 13.5% Pass@1; OfficeTown Executor, Pi and OpenCode reach 13.5%, 8.6% and 4.9%, respectively. The gap from weighted scores distinguishes partial fulfillment from complete delivery.

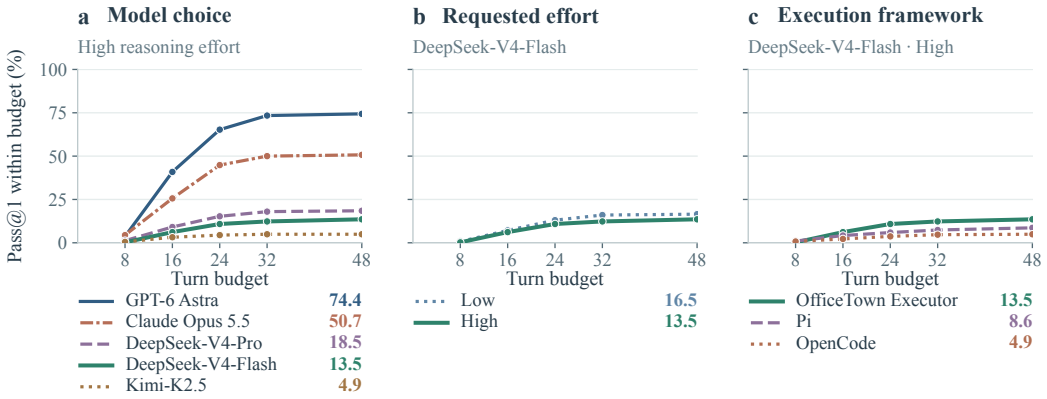



Figure 15: Pass@1 at five recorded turn budgets. Populations, selected attempts, requested settings and turn counts match Figure 6. All panels use the same 406 tasks. A final pass contributes only after its recorded execution has ended; all other outcomes contribute zero. Lines guide the eye; no intermediate artifacts are scored.

These diagnostics describe associations, not causal failure mechanisms. Requirement content, artifact usability and failure modes need separate reviewed annotations; unresolved grading is not evidence of incorrect model work.

# E Training Data and Model Adaptation

## E.1 Training Resources

The preparation cohort contains 6,172 cases, including 6,082 with completed text-consistency editing records. Teacher execution covers 3,718 file-based tasks; service-dependent tasks lie outside this configuration rather than count as failed executions. These are preparation counts, not a census of raw Live activity.

The SFT resource contains 2,202 complete trajectories distilled from DeepSeek-V4-Flash (Xu et al., 2026a). From 2,317 trajectories in the original SFT partition, we exclude 46 tool-serialization cases, 62 exceeding context capacity and seven parser round-trip cases, leaving 2,202. Trajectories are neither truncated nor filtered by task-success score. The resource is split into 1,982 training and 220 validation demonstrations. Two training and four validation padding rows contain no supervised tokens and are not counted as additional demonstrations.

OPD provides 828 task–environment pairs with task requests and original initial environments, without teacher response histories. Its public system message, chat template and serialized tool definitions align with the SFT input format. The benchmark and training resources are not a single random three-way partition of the intermediate construction pool.

## E.2 SFT and On-Policy Distillation

Qwen3.5-9B is trained first with SFT, then with on-policy distillation (Agarwal et al., 2024) using Qwen3.8-Flash-Next as teacher. Table 11 summarizes the resources and teachers; selection and reporting follow Appendix B.1.

Table 11: Training resources and teachers. Example counts exclude zero-supervision padding rows; the SFT token total describes the complete demonstration resource. Optimization settings are given in Tables 12 and 13.

| Item | SFT | OPD after SFT |
|---|---|---|
| Initial student | Qwen3.5-9B | SFT-trained Qwen3.5-9B |
| Training inputs | Complete demonstrations | Requests and initial environments |
| Training examples | 1,982 | 828 |
| Validation examples | 220 | None |
| Supervised tokens in resource | 30,704,800 | Not applicable |
| Teacher | DeepSeek-V4-Flash | Qwen3.8-Flash-Next |

SFT updates all parameters of Qwen3.5-9B for two epochs using AdamW, with a 14-step warmup followed by linear learning-rate decay. Each epoch comprises 248 optimizer steps under the global batch size of eight, including the zero-supervision padding rows. The settings in Table 12 preserve complete sequences without truncation.

Table 12: Full-parameter SFT configuration. The 1,982 training and 220 validation demonstrations are distinct from the two and four zero-supervision padding rows. Training uses 248 steps per epoch and 496 steps in total.

| Parameter | Setting |
|---|---|
| Model / training mode | Qwen3.5-9B; full-parameter SFT |
| Hardware / parallelism | One node, $8\times$ NVIDIA A100 80 GB; FSDP2 |
| Schedule / random seed | 2 epochs, 496 steps; seed 42 |
| Global / micro-batch size | 8 / 1 per GPU |
| Optimizer | AdamW; $\beta_1 = 0.9$, $\beta_2 = 0.95$ |
| Learning rate | $10^{-6}$; linear decay to $10^{-7}$ (minimum ratio 0.1) |
| Warmup | 3%; 14 steps |
| Weight decay / gradient clipping | 0.01 / 1.0 |
| Maximum sequence length | 131,072 tokens; no truncation |
| Precision | FP32 master parameters and Adam states; BF16 computation |
| Memory optimizations | Gradient checkpointing, CPU activation offload, LM-head chunk size 512 and length grouping |
| Checkpoint saving | Steps 100, 200, 300, 400 and 496; BF16 Hugging Face model export |
| Validation | At step 496 |

OPD samples one student trajectory per task and queries the teacher for the log probability of each sampled token. For token $a_t$ and context $h_t$, the distillation signal is

$$k_{1,t} = \log \pi_{\text{student}}(a_t \mid h_t) - \log \pi_{\text{teacher}}(a_t \mid h_t).$$

The negative log ratio supplies a stop-gradient token-level advantage to a vanilla policy-gradient update, rather than a directly differentiated supervised loss. Updates use token-mean aggregation and a distillation coefficient of 1.0. Task rewards and the separate reference-policy KL term are disabled.

The run uses all 828 training tasks for one epoch: seven batches of 104 and a final batch of 100. The final batch is retained, with scheduling padding that does not add task samples. Validation is disabled; no training rows are relabelled as an independent held-out set. The teacher runs in a separate vLLM service on the same eight GPUs. Table 13 records the OPD settings.

Table 13: OPD training configuration. The student is initialized from SFT-trained Qwen3.5-9B. Length limits are in tokens; the final 100-task batch is retained.

| Parameter | Setting |
|---|---|
| Hardware | One node, $8\times$ NVIDIA B200 |
| Training schedule | One epoch; 8 updates; seed 42 |
| Learning rate | $2.0 \times 10^{-6}$ |
| Batch / mini-batch size | 104 / 104; one optimization pass per update |
| Micro-batch size | 1 per GPU |
| Distillation objective | $k_1$; vanilla policy gradient; token-mean aggregation |
| Distillation coefficient / log-probability floor | $1.0$ / $-20.0$ |
| Task rewards / reference-policy KL | Disabled / disabled |
| Rollout correction | Token-level importance sampling; threshold 2.0 |
| Sampling | One trajectory per task; temperature 1.0; top-$p$ 1.0; top-$k$ disabled |
| Prompt / response / context limits | 8,192 / 122,880 / 131,072 |
| Interaction limits | 100 assistant turns; 1,800 s per task; 120 s per tool call |
| Tool-output limit | 65,536; middle truncation |
| Training parallelism | FSDP2 across 8 GPUs; parameter and optimizer offload disabled |
| Rollout execution | Asynchronous vLLM; tensor parallelism 1; 32 sequences; 16 agent workers |
| Rollout GPU-memory fraction | 0.45; prefix caching enabled |
| Memory optimizations | Gradient checkpointing, padding removal and fused Torch kernels |
| Tool-call serialization | `qwen3_coder`; released tokenizer, tool schemas and agent loop; strict tokenization checks |
| Checkpointing / validation | Save every 2 updates, retain 2 checkpoints; validation disabled |

### E.3 External Evaluation Protocol

Base, SFT and SFT + OPD use the OfficeTown Pi/verl execution loop and native tools, with benchmark-specific scoring and limits.

GDPval excludes tasks requiring live web search. GPT-5.5 scores the original rubrics as absolute scores, not the official pairwise win rate.

JobBench uses 63 released Easy environments, native Grok-4.3 weighted grades and a 7,200-second task budget without an additional turn cap. Later recovery phases exclude verified infrastructure-blocking time from active time.

APEX-Agents v1.1 uses 150 assistant turns and 16,200 seconds per task. Its verifier uses an Azure DeepSeek-V4-Flash route instead of the published Fireworks snapshot.

Terminal-Bench 2.1 excludes search-dependent tasks and uses native binary rewards.

### E.4 Training Dynamics

Figure 16 shows non-monotonic SFT progress on JobBench-Easy, including the decline at step 300 and larger variation at step 496. Error bars describe recorded run variation, not independent-seed confidence intervals.

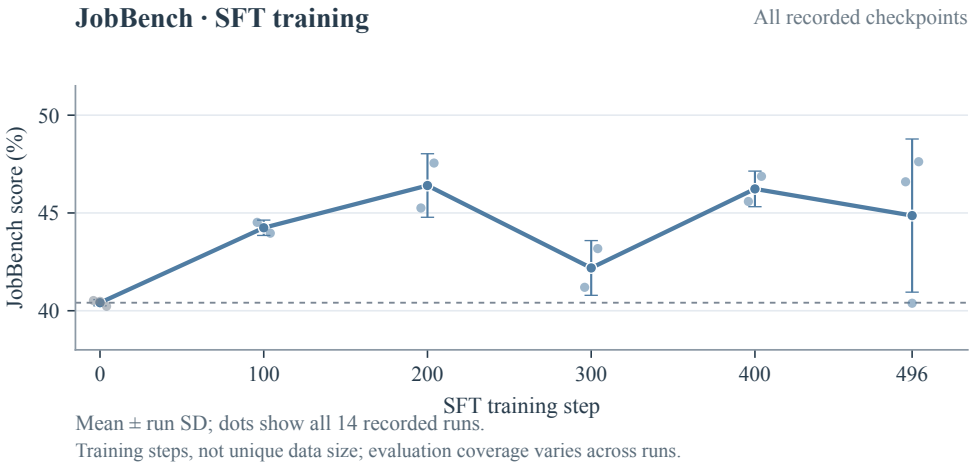



Figure 16: JobBench-Easy performance over SFT training. All recorded checkpoints from the same training run and the Base reference are shown. Points are run means and error bars are one sample standard deviation; the 14 individual runs are also displayed. The non-monotonic curve measures training progress, not an ablation of unique training-data size.

### E.5 Budget-Conditioned Outcomes

Figures 17–20 show five fixed budgets per benchmark, with endpoints matching Table 3. Lines guide the eye between recorded points. Each execution contributes its final score from its recorded ending turn onward and zero before then, with a fixed denominator and no rescaling. The curves summarize existing runs, not new executions under shorter stopping budgets or intermediate artifact scores. JobBench retains its originally scored partial outcome from a response-token stop at the recorded ending attempt; missing grades follow the original reporting rules. Turn-based curves do not establish equal inference cost.

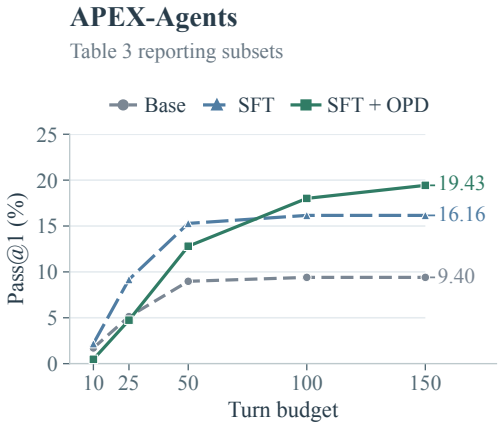



Figure 17: Recorded APEX-Agents Pass@1 at five budgets. A pass counts only after execution ends; endpoints match Table 3.

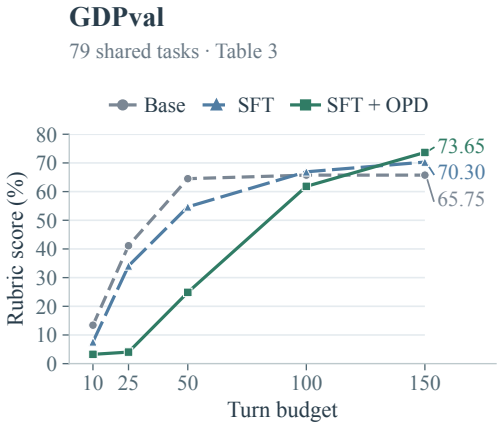



Figure 18: Recorded rubric scores on GDPval's shared tasks. SFT + OPD reaches a higher final score but needs more turns to reach 50 in the full records.

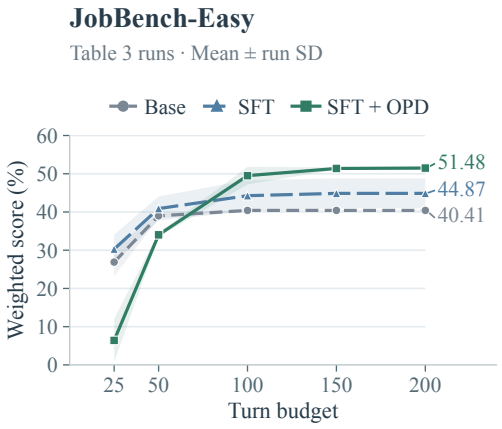



Figure 19: Recorded JobBench-Easy scores. Points average original run means equally; shading is one sample SD across those runs, not an independent-seed confidence interval.

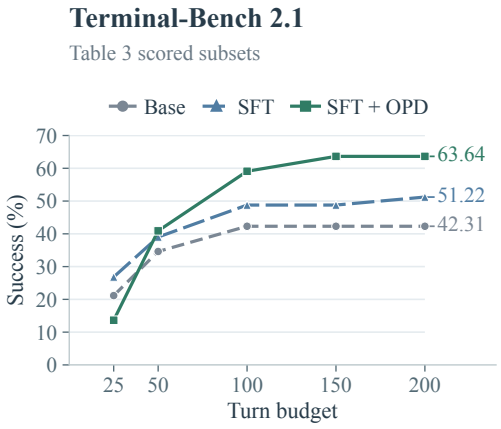



Figure 20: Recorded Terminal-Bench 2.1 success rates at five budgets, using Table 3's selected outcomes and preserving its endpoints.