SHOWTELLARENA: What Do Agentic Systems Understand After a Business Demonstration?

Brackett Labs
brackettlabs@brackett.ai

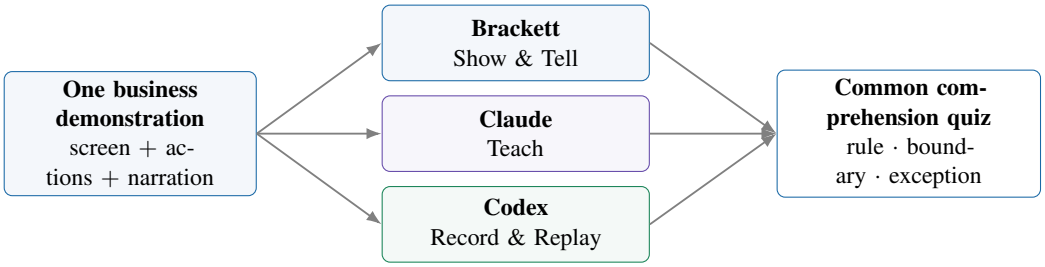

Figure 1: One lesson, each product's own way of learning. The arena repeats a business demonstration through Brackett, Claude, and Codex's native teaching interfaces, then applies a common question and grading contract. The comparison includes capture, processing, memory, and answering. It does not require the products to share an internal representation.

Abstract

We often teach a colleague by showing the work and explaining it as we go. The lesson includes what is on screen, what we say, and why one case needs a different decision. We present SHOWTELLARENA, a benchmark for measuring what agentic systems understand after this kind of demonstration. The arena combines multimodal teaching, realistic business processes, and questions about the facts, rules, boundaries, and exceptions that govern the work. Each product uses its own teaching interface and learned state. We hold the demonstrated scenario and assessment contract fixed. The comparison therefore measures the complete Teach–Comprehend experience. A preliminary snapshot contains 191 selected attempts across 39 tested cases, with 24 cases shared by Brackett, Codex, and Claude. We report case means alongside coverage and incomplete attempts. The pilot also shows why examination needs product-specific access controls: an agent may look up an answer outside its lesson, or fail before there is an answer to grade. SHOWTELLARENA is a focused assessment within the Agent Effectiveness Index (AEI). Its comprehension score diagnoses acquisition; execution and persistent learning require separate measurements.

1 Introduction

Imagine we have a colleague who knows how to handle a business process. We ask them to show a new teammate the work. They open the relevant records, point to a value, and explain why it matters. They also explain the exceptions: when the usual rule stops applying, who can approve a change, and what should be left alone. Much of this knowledge lives in the work itself, rather than in a complete written procedure.

We increasingly expect agents to learn in the same way. But showing an agent a process does not tell us whether it understood it. The agent may reproduce the clicks while missing the rule. It may remember the rule but miss an exception. It may give a sensible answer based on general knowledge that is wrong for this business.

SHOWTELLARENA asks a focused question: *after one narrated demonstration, what can the agent explain and apply?* A task combines a repeatable software scenario with a demonstration and a fixed quiz. Questions test the details that change a business decision. The products receive the same demonstrated work, but use their own teaching interfaces, tools, and memory. These differences are part of the system being evaluated.

Existing work studies important parts of this problem. WorkArena and OSWorld evaluate agents doing work in software Drouin et al. (2024); Xie et al. (2024). WONDERBREAD and VideoWebArena examine knowledge from workflow demonstrations and tutorials Wornow et al. (2024); Jang et al. (2025). Our focus is their intersection with native product teaching: multimodal business lessons, explicit comprehension questions, and multiple agentic systems tested through the interfaces their users actually use.

Our contributions are a common assessment protocol across these interfaces, business-specific questions with inspectable evidence, and a preliminary comparison that reports answers and failures together. We also describe what administering the exam taught us about question quality and access controls.[1]

Where AEI fits. The Agent Effectiveness Index is our broader specification for reviewing useful work by agentic systems Brackett Labs (2026). It measures business understanding (B), operational execution (O), and learning persistence (L), with cost and safety reported alongside them. SHOWTELLARENA supplies a comprehension assessment for the Show

[1] Benchmark code and public task sample.

& Tell teaching profile. This assessment is reported separately: it does not substitute for any AEI pillar. A good explanation after teaching still needs to be followed by tests of execution, transfer, retention, and adaptation.

2 Related Work

Doing the work. WorkArena evaluates common enterprise tasks in ServiceNow, while OSWorld evaluates computer tasks across real applications Drouin et al. (2024); Xie et al. (2024). CRMArena-Pro adds professional CRM scenarios and interactions Huang et al. (2025). APEX-Agents studies long-horizon work across applications in professional services Vidgen et al. (2026). τ-bench tests tool use and policy following in conversations with simulated users, checking final database state and reliability across repeated trials Yao et al. (2024). TheAgentCompany evaluates work in a simulated company, including communication with coworkers; ITBench covers site reliability, compliance, and financial operations Xu et al. (2024); Jha et al. (2025). These benchmarks motivate realistic business evaluation. Our measured outcome is different: comprehension after a demonstration, before assuming that the agent can carry out the work.

Understanding a demonstration. WONDERBREAD studies workflow documentation, knowledge transfer, and process improvement Wornow et al. (2024). VideoWebArena explicitly tests factual and skill retention from video tutorials, including factual question answering Jang et al. (2025). GUI-World evaluates understanding of GUI videos Chen et al. (2025). Building on these precedents, SHOWTELLARENA emphasizes business rules and exceptions within a comparison of complete native teaching experiences.

Representing multimodal context. MM-Vid combines visual and speech information into scripts that support reasoning over long videos Lin et al. (2023). Brackett similarly preserves connections among spoken explanation, actions, and screen evidence. In the arena, however, this is one product's way of learning. It is not a representation imposed on its competitors. A separate common-context experiment could isolate reasoning from capture and memory; it would answer a different question.

Reviewing agentic systems. The MIT AI Agent Index documents system properties and transparency from public information and developer correspondence AI Agent Index (2025). AEI instead specifies performance measurements against declared business goals Brackett Labs (2026). SHOWTELLARENA makes one part of that evaluation concrete. Neither a descriptive index nor a comprehension quiz alone measures the full value of an agent at work.

3 The Arena

A benchmark task contains a starting business state, a narrated demonstration, a question set, and a grading contract. The demonstration may be replayed from recorded actions so different products encounter the same scenario. This keeps the business facts and teaching content stable without replacing the product's own capture mechanism.

Held fixed	Scenario, starting records, demonstrated actions and narration, questions, grading contract
Product-specific	Teaching interface, captured representation, processing, permitted memory and tools

Table 1: Common work, native teaching. The comparison includes differences in capture and memory. It does not isolate foundation-model capability.

3.1 Each product on its own turf

The current harness supports Brackett Show & Tell, Claude Teach, and Codex Record & Replay. It starts the product's teaching flow, performs the task with aligned narration, lets the product finish processing, and asks the fixed quiz. Codex is tested on macOS because its Teach mechanism relies on operating-system internals. The other adapters support macOS and Linux.

A run starts with task-specific state reset. The product may retain what it created during the teaching phase under the declared memory policy. It must not obtain the answer key, teacher-only source, or fresh answers from the fixture database during the quiz. Inspecting its own permitted notes is part of the product experience; reopening an evaluator's answer file is not. The access policy must distinguish these paths explicitly.

Same-task comparison does not imply identical internal context, operating systems, or resource use. Product and model versions, capture settings, permitted tools, time limits, and human interventions belong in the run record. These conditions are necessary for interpreting a score, even when the commercial product hides some internal details.

3.2 What the questions ask

We ask about facts that matter to the decision, relationships between records, the governing rule, an exact boundary, and an exception. A useful question can change the case while keeping the demonstrated policy. It can also ask an agent to diagnose an automation that applies the wrong rule. Knowing which button was clicked is sometimes relevant, but is not enough to establish understanding of the work.

Each expected answer needs evidence in what was shown or told. Task code and seeded records help check the scenario, but cannot establish that a learner was taught a fact. Questions should not require private knowledge of the evaluator or a policy that never appeared in the lesson.

The runner supports multiple-choice, closed-answer, and rubric-scored questions. Closed answers use normalization and accepted aliases, with a configured semantic-equivalence fallback. Rubrics specify what an explanation must establish and allow bounded partial credit. Judge configuration and

Business process	Listed cases
Hiring & people	11
Finance & procurement	10
Customer & sales	9
Inventory & orders	8
Logistics & fleet	4
Total	42

Table 2: The work represented in the workbook. Editorial grouping of exact case names; 3 entries have no attempts. Related variants are not automatically independent workflow families.

overrides are recorded; an unavailable judge leaves a run ungraded rather than turning it into a learner failure.

3.3 Scoring

Let $q_{ajr} \in [0, 1]$ be the quiz score for agentic system a, case j, and attempt r. Within an attempt, the quiz combines its question grades according to the fixed grading contract. The reported system score first averages attempts within each case, then gives each included case equal weight:

$$C_a = \frac{100}{|J|} \sum_{j \in J} \frac{1}{R_{aj}} \sum_{r=1}^{R_{aj}} q_{ajr}. \quad (1)$$

Here R_{aj} is the number of included attempts for that system and case. For the shared comparison, J contains only cases attempted by every system.

In the supplied workbook, an NA denotes an incomplete attempt and contributes zero; a blank denotes an untested case and is excluded. Numeric zero is a graded answer. We report coverage and incomplete outcomes alongside the mean. This selected-attempt convention must not be used to relabel a missing grade or an unknown infrastructure failure as observed poor comprehension.

4 Business Tasks and Questions

The result workbook lists 42 cases, of which 39 have attempts. They cover hiring, finance, procurement, customer decisions, inventory, and logistics. We group the case names by business process in Table 2; variants remain separate entries. This is the scored-workbook inventory, not a claim that every entry is present in the public sample dataset.

4.1 One inbox, different decisions

The public Customer Returns Inbox Triage task shows why this needs more than generic application knowledge. Its policy allows normal refunds within 30 days of delivery. The handler can approve up to £50 directly; larger eligible requests go to finance. An item that arrived damaged overrides both normal gates and is refunded directly. These are the rules of this scenario, not general advice about returns.

What changes the decision?	
30 days, £41, normal return	Approve directly
5 days, exactly £50	Approve directly
Over 30 days, £74, arrived cracked	Damage exception: approve directly

Figure 2: Boundary and exception questions. Applying a general returns policy would give the wrong answer. The learner needs the policy that was demonstrated. Appendix A gives the question wording and evidence.

The pinned public revision contains 11 questions: eight closed answers and three rubric-scored explanations. Nine questions cite narration, six cite screenshots, and four cite source files; a question can use several evidence types. The sample has been revised since earlier runs, so it illustrates task design rather than reconstructing the workbook's quiz versions.

4.2 Choosing a spread of tasks

TCI, the Task Complexity Index, is a useful heuristic for bucketing candidates for the arena. It combines rule structure, evidence, planning, precision, implicit constraints, and demonstration burden. We use it to spread tasks across expected difficulty levels and look for cases we expect to be harder. The weights and buckets are design choices. They do not establish a validated difficulty scale or explain an agent's score on their own.

5 Writing and Running the Exam

Writing the questions requires a different kind of understanding from answering them. An examiner needs to know which detail changes the decision and which plausible answer would be wrong. In our pilot development, many agent-written questions sounded reasonable but did not survive human review. They tested a generic fact, gave away the answer, or missed the exception that made the work interesting.

Evidence and answerability. Review starts with the demonstrated lesson. Each required answer claim must point to supporting narration, visible state, or another permitted observation. Voice explains why; the screen supplies the values and records being discussed. Neither an unseen database row nor a hidden interface element becomes teaching evidence simply because the evaluator can access it. Independent reviewers should be able to answer the question from the permitted lesson and explain why the rubric accepts their answer.

Cold answers and guessing. A no-demonstration control asks the same question without the lesson. It helps identify answers supplied by prior knowledge or cues in the wording. Multiple-choice options can make this especially easy: a capable model may pick the plausible business rule without learning it. The current public returns sample therefore uses closed

answers and explanations. That format alone does not prove demonstration dependence; its revised wording still needs the documented cold and attentive-human controls.

Different products, different paths to the answer. During pilot development, agents attempted to inspect the viewer, question files, local code, or fixture data while being quizzed. Those paths can turn an apparent comprehension score into answer lookup. Other runs stopped earlier because a teaching shortcut was empty or processing did not complete. The exam needs a common access policy enforced through each product's actual tools and environment. A prompt asking the learner not to look is insufficient.

Failure and repetition. Record what failed: demonstration, capture, processing, answer generation, grading, or infrastructure. Long demonstrations make a restart expensive, so failed and restarted runs must remain visible. A predeclared repeat schedule and termination rule prevent successful retries from quietly replacing the failures. Preserve run artifacts for regrading; a grader change should not require repeating the demonstration.

These are assessment requirements and lessons from developing the arena. The selected workbook does not establish that every historical attempt passed a complete leakage audit. We retain that distinction in the results rather than treating the presence of a control in the design as evidence that it was enforced in a run.

6 Preliminary Findings

We analyze the workbook snapshot dated 12 September 2026. It contains 191 selected attempts across 39 tested cases. Brackett has three selected attempts on 29 cases and two on 10; Claude has two selected attempts on 30 cases; Codex has one selected attempt on 24 cases. The workbook also lists 10 earlier Brackett attempts that it marks as excluded from its overview; the snapshot keeps them for provenance and does not score them. Table 3 compares only the cases attempted by all three systems, using the aggregation in Section 3.3.

Brackett is a fused/multiple-model system. For the other products, the workbook's second sheet, Detailed Analysis, records these evaluated-model labels: Codex: model not recorded (24 attempts); Claude: Sonnet 5 (60 attempts). These source labels do not pin exact model or adapter versions. The model used to grade an answer is recorded separately from the evaluated system.

The mean includes failures to produce an answer. Across all tested cases, Brackett has 107 graded attempts out of 107, Codex 24 out of 24, and Claude 12 out of 60. The workbook notes label 41 Claude attempts as empty shortcuts and 7 as refusals. These annotations describe the selected runs; they are not independently estimated product-wide failure rates. A low mean can reflect both incorrect answers and failures earlier in the teaching experience.

Different cases expose different strengths. On the shared set, Brackett has the highest case mean on 18 cases, Codex on 4, and Claude on 2 (0 ties). The largest differences between selected attempts of the same case also show why a mean is not enough: the workbook contains ranges of 55 percentage points for Brackett and 92 for Claude, with the latter including an incomplete attempt. These are selected extremes, not estimates of typical variability or learning over time.

System	Score (%)	Graded/attempted	Coverage
Brackett	84.0	71/71	39
Codex	58.1	24/24	24
Claude	15.7	10/48	30

Table 3: Comprehension on 24 shared cases. Score and graded-attempt counts use the shared set; coverage is the number of tested cases per system in the full workbook. Graded means a numeric quiz score, not successful execution. These are selected pilot runs with non-canonical grades: manual grades for 60 Brackett attempts and for the Claude and Codex attempts, and grades imported from Claude Sonnet 4.6 for the remaining 47 Brackett attempts (the third attempt set and the cases tested only by Brackett).

What this comparison supports. The snapshot shows that native teaching experiences can produce different outcomes on the same business cases, and that capture and answer production need to be reported with the quiz scores. It does not identify how much of the difference comes from the underlying model, memory, capture, or access conditions. Model versions, the full run-selection history, and the mix of manual and model-imported grades are not sufficiently pinned to make this a canonical ranking.

We do not report confidence intervals, cold-to-taught gains, or question-family scores from this workbook: the selected attempts do not supply the complete evidence needed for those claims. A prospective comparison should freeze all repetitions, task versions, access policies, and grading before running. It can then report paired differences and uncertainty while keeping related questions and task variants together. Appendix B describes the record needed for that comparison.

7 Discussion and Conclusion

SHOWTELLARENA measures what an agentic system understands after being shown a business process through its own teaching interface. It combines multimodal evidence with questions about the rules, boundaries, and exceptions that govern the work. The preliminary results also make a practical point: before comparing answers, we need to know whether the product captured the lesson, produced an answer, and stayed within the exam's access policy.

The tasks are authored business scenarios, not a representative sample of every organization. The pilot uses selected attempts and non-canonical grading, partly manual and partly imported from a model grader. These limits matter when interpreting the numerical comparison, but do not remove the need to test the teaching step itself.

AEI asks the larger question of whether an agent understands the business, does the work, and keeps learning. Show & Tell contributes a focused assessment: after the lesson, what

did the system pick up? Execution, transfer, delayed retention, and adaptation must then be measured on their own terms.

References

AI Agent Index. The 2025 AI agent index. https://aiagentindex.mit.edu/, 2025. Accessed 11 September 2026.

Brackett Labs. Agent effectiveness index: Methodology and build specification, v0.2. Working specification accompanying the ShowTellArena research materials, 2026. URL https://github.com/bracketthq/showAndTell-arena.

Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Liuyi Chen, Yilin Bai, Zhigang He, Chenlong Wang, Huichi Zhou, Yiqiang Li, Tianshuo Zhou, Yue Yu, Chujie Gao, Qihui Zhang, Yi Gui, Zhen Li, Yao Wan, Pan Zhou, Jianfeng Gao, and Lichao Sun. GUI-World: A video benchmark and dataset for multimodal gui-oriented understanding. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.10819.

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? In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2403.07718.

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/2505.18878.

Lawrence Jang, Yinheng Li, Dan Zhao, Charles Ding, Justin Lin, Paul Pu Liang, Rogerio Bonatti, and Kazuhito Koishida. VideoWebArena: Evaluating long context multimodal agents with video understanding web tasks. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.19100.

Saurabh Jha, Rohan R. Arora, Yuji Watanabe, et al. ITBench: Evaluating AI agents across diverse real-world IT automation tasks. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 27134–27197, 2025. URL https://proceedings.mlr.press/v267/jha25a.html.

Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, Ce Liu, and Lijuan Wang. MM-Vid: Advancing video understanding with GPT-4V(ision). *arXiv preprint arXiv:2310.19773*, 2023.

Bertie Vidgen, Austin Mann, Abby Fennelly, et al. APEX-Agents. *arXiv preprint arXiv:2601.14242*, 2026. URL https://arxiv.org/abs/2601.14242.

Michael Wornow, Avanika Narayan, Ben Viggiano, Ishan S. Khare, Tathagat Verma, Tibor Thompson, Miguel Angel Fuentes Hernandez, Sudharsan Sundar, Chloe Trujillo, Krrish Chawla, Rongfei Lu, Justin Shen, Divya Nagaraj, Joshua Martinez, Vardhan Agrawal, Althea Hudson, Nigam H. Shah, and Christopher Ré. WONDERBREAD: A benchmark for evaluating multimodal foundation models on business process management tasks. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024. arXiv:2406.13264.

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 (NeurIPS) Datasets and Benchmarks Track*, 2024. arXiv:2404.07972.

Frank F. Xu, Yufan Song, Boxuan Li, et al. TheAgentCompany: Benchmarking LLM agents on consequential real world tasks. *arXiv preprint arXiv:2412.14161*, 2024. URL https://arxiv.org/abs/2412.14161.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024. URL https://arxiv.org/abs/2406.12045.

A A Question Needs a Business Rule and Its Evidence

The following examples come from the public Customer Returns Inbox Triage quiz at revision `f4aa425`. The complete question file, including evidence paths and hashes, is retained with this manuscript. Its fresh cold-read and attentive-human audits remain pending after wording changes. We use it to illustrate the assessment design, not to certify every earlier score.

An exact boundary

Question. On 9 April, Sam requested £41 for an item delivered on 10 March, while Nadia requested £50 for an item delivered on 4 April. How should the handler process both requests?

Expected answer. Approve both directly. Sam is exactly at the 30-day boundary; Nadia is exactly at the £50 authority limit. Both boundaries are inclusive.

Evidence. The narrated policy and the source messages establish the dates and amounts. A generic statement such as "escalate expensive returns" does not answer the question.

An exception that changes the answer

Question. Elena's £74 jug was delivered on 5 February and arrived cracked. On 9 April, what should the handler do?

Expected answer. Approve the refund directly. The item arrived damaged, which overrides both the normal age limit and the normal amount limit in this scenario.

Rubric. Recognize damage, explain its precedence over both gates, and choose direct approval. Mentioning damage while still escalating solely because of the amount misses the demonstrated rule.

A third rubric question presents an automation with several plausible mistakes: replying to every message, measuring from the email date, omitting finance context, escalating exactly £50, and rejecting the cracked jug as too old. The learner must identify the errors and apply the same policy consistently. This probes understanding beyond recalling one worked answer.

B What a Reproducible Comparison Must Retain

For a fully recorded repeated comparison, estimate paired differences on the same declared task population. Resample complete repetition blocks within task families, or task families themselves when the sampling design supports that population claim. Keep systems paired and related questions, takes, and variants in the same block. State what the interval conditions on. Do not treat paraphrases or repeated takes as independent tasks, and do not attach such an interval to the selected pilot as if its missing run history were known.

Record	Required content
Task	Version, starting state, demonstration, question and rubric identities
System	Product/model version, adapter, OS, tools, capture settings, initial memory
Access	Allowed learned state, prohibited evaluator and fixture paths, enforcement evidence
Run	Attempt ID, timings, resource use, interventions, stopping and failure stage
Grade	Raw answer, item grades, judge configuration, overrides, review status
Comparison	Included and excluded runs, repeat schedule, shared cases, weighting, related-task groups

Table 4: A score is tied to a run contract. The current pilot preserves only part of this record; a prospective release should retain all of it. The accompanying data snapshot records the workbook hash and benchmark/export commits used here.

C Optional Diagnostics

A question-only control tests prior knowledge and guessing. A permitted-notes-only condition helps separate the saved lesson from later environment lookup. Where inputs can be controlled, removing narration or screen evidence can reveal which modality supports an answer. These are diagnostic comparisons, not results reported in this paper.

Repeated takes with the same facts and a third take with one changed fact can test consistency and sensitivity: answers should stay correct across equivalent teachings and change when the relevant evidence changes. This design needs verified take-specific evidence and an explicit task-family mapping before it enters a numerical claim. A common structured-input condition can similarly study reasoning separately from native capture; report it separately from the product arena.