AbsoluteAGI / Research and Reports

Continuing Digital Individuals

Continuity, Autonomy and Delegated Responsibility

Arthur LEI

Published edition / 2026-11-07

Literature search through 4 November 2026

Abstract

This review examines continuity, autonomy, and delegated responsibility in digital systems across sessions, tasks, and interruptions. Structured literature mapping and critical narrative synthesis cover 97 research works and seven groups of deployment materials through 4 November 2026, with four cases: AI Village, Claude's Corner, Andon Market, and The Agentic Scientist.

Research establishes partial capabilities in memory updating, work resumption, proactive action, and verification. Evidence remains concentrated in individual benchmarks, short comparisons, and observational deployments. Major gaps concern continuity across versions and environments, the influence of resources and operating conditions on goal formation, and whether delegation reduces supervision and recovery labor under equivalent acceptance requirements. Studies of continuing individuals should also follow changing interests, relationships, and self-understanding.

Contents

Abstract 2

Chapter 1 Introduction and research questions 4

Chapter 2 Review methods 6

Chapter 3 Theoretical foundations and research development 9

Chapter 4 Continuity 14

Chapter 5 Autonomy 19

Chapter 6 Sustained work and delegated responsibility 24

Chapter 7 Evaluation methods and empirical findings 29

Chapter 8 Synthesis and current challenges 34

Chapter 9 Future research directions 38

Chapter 10 Conclusions 41

References 43

Chapter 1 Introduction and research questions

On Monday, a team delegates a research report to an agent for publication on Friday. Revised data invalidate a central conclusion on Wednesday, after the initial session has ended. Will the system identify what needs rechecking, pursue the investigation, and seek a decision when the conditions of delivery change?

This review asks how a digital system can draw on its experiences, initiate action, and remain responsible for accepted work across sessions, tasks, and interruptions. We use “continuing digital individual” as a research perspective: following a recognizable history of the same system to examine how experiences, goals, and commitments change later activity. Sustained work is one setting for these relationships; learning, relationships, self-chosen activity, and changing self-representations also belong within the scope. Continuity concerns which states and experiences persist, are revised, or become reusable knowledge. Autonomy concerns the formation or adoption of goals and the conditions for acting, waiting, or changing direction. Delegated responsibility concerns who advances work, checks outcomes, handles failure, and has authority to change objectives, accept delivery, or take over.

We combine structured literature mapping with critical narrative synthesis. Searches through 4 November 2026 produced a reading corpus of 97 research works and seven groups of project and deployment materials. Four cases receive sustained comparison: Claude’s Corner, AI Village, Andon Market, and The Agentic Scientist. Common dimensions are observation unit, timescale, goal origin, human intervention, and outcome measurement. Controlled benchmarks establish performance under specified conditions; deployment surveys and telemetry describe use; project records trace activity and consequences over time. Comparisons retain the task conditions and provenance of each source. [1] [2] [3] [4]

The evidence shows that systems can retrieve information from earlier records, revise some judgments, and resume unfinished work under controlled conditions. Maintaining commitments through repeated interruptions and version changes requires longer comparisons. Acting or waiting under a given objective can already be evaluated reproducibly, and open deployments exhibit self-chosen activity. Existing records less clearly separate how goals form and persist from the effects of budgets and scheduling. Progress, checking, and some correction can be delegated, while changes to standards, exceptions, and final acceptance may still involve people. Whether sustained delegation reduces human burden depends on comparable delivery requirements and accounting for supervision, rework, and takeover alongside outcomes. [5] [6] [7]

Memory retrieval, task completion, open exploration, and human-agent work allocation have developed substantial but partly separate literatures, often using different units and timescales. This review connects them through continuity, autonomy, and delegated responsibility, relating mechanisms to deployment evidence within continuing activity. Its contribution is to connect research traditions, establish conditions for meaningful comparison, and identify longitudinal questions at their intersections. Correct recall establishes access to a past fact; using it after an agreement changes requires evidence about its role in action. Likewise, one successful delivery and an effective long-term delegation depend on different records of outcomes and work allocation. Mapping findings to experience, choice, action, and verification explains why similar scores can answer different questions and identifies relationships to test within the same individual or undertaking.

Chapter 2 describes retrieval, screening, and comparison. Chapter 3 traces theoretical foundations and research development. Chapters 4-6 examine continuity, autonomy, and delegated responsibility; Chapter 7 compares evaluation methods, findings, and the four cases. Chapter 8 synthesizes challenges and evidence gaps, Chapter 9 develops research designs, and Chapter 10 returns to the opening questions.

Chapter 2 Review methods

2.1 Scope and retrieval

The review combines structured literature mapping with critical narrative synthesis. Retrieval follows the questions in Chapter 1: how researchers define and measure the influence of experiences, states, and commitments on later action; identify action initiation and maintenance; and record the allocation of progress, checking, and correction between people and agents. Studies of continuing operation enter directly. Studies of individual mechanisms enter when they explain a relevant problem, such as outdated memory misleading subsequent action. Cognitive science, artificial life, and human factors provide conceptual and methodological foundations, interpreted within their original populations and environments.

The initial search and frontier update closed on 4 November 2026. Classic works were traced through conceptual lineages; recent coverage emphasizes LLM agents since 2023. Academic web searches and publisher-targeted keyword queries identified candidates, followed by bibliographic and version checks at arXiv, ACL Anthology, OpenReview, ACM, PMLR, NeurIPS, journals, and author or institutional pages. PubMed also supported checks for psychology, cognition, and human factors. Searches combined topic terms and known titles, connecting long-term memory with autobiographical memory and updating, and sustained work with recovery and human takeover. Reference chaining from surveys and central papers was followed by forward searches for concepts and benchmarks. The update specifically covered model replacement, online learning, waiting and monitoring, recovery from errors, and public deployment records. Query examples and screening records are retained in the source register.

2.2 Screening, deduplication, and versions

Materials were assessed for whether they answered a research question, established the origin of a concept, or supplied a method comparison with identifiable conditions. Theory and surveys support conceptual mapping; empirical claims about methods, data, and results are checked against original studies. General leaderboards, product promotion without methodological detail, and off-topic search results were excluded from the main catalog. Materials with different tasks, scoring, or research questions were registered separately: the two versions of Vending-Bench, for example, have distinct simulation conditions and scoring rules.

Counts use research works as the unit. Preprints, conference papers, and journal versions of the same work were consolidated, while recording the version read and substantive differences in samples, measures, or text. The initial corpus contained 87 deduplicated works; the update added ten, producing 97 publications and research reports. Seven additional groups contain project, operational, and deployment materials, with multiple files possible within each group. The 97 works describe reading coverage; the 85 entries in the final bibliography are works and project materials directly cited in the text. Reports and logs within a project group can therefore receive separate references. Papers and records derived from the same deployment, such as

OpenLife and its later archive, are linked. Later records can extend observation without creating an independent deployment.

The main texts of all 97 works were read. Sources came from formal publication pages, author manuscripts, or readable journal scans; figures, tables, and appendices needed for the analysis were checked. Where substantive differences existed between accessible versions, results are attributed to the cited version and analysis sample. The archive retains acquired PDFs, page snapshots, and extracted online text, with provenance and version fields. The ledger records the scope of verification. For G12, for example, the NBER working paper was read in full and the published article's sample definition and principal results tables were checked on the journal page. [8]

2.3 Evidence extraction and comparison

Each work was recorded by question, population or system, environment, design and control, analysis unit, measures, main findings, and stated limitations. Evidence was located by page, section, table, figure, or appendix. MiniMax configured through Claude Code assisted with segmented extraction and locating candidate evidence. Reading and extraction proceeded in batches, followed by checking against the original before citation. The ledger distinguishes assisted extraction, direct reading, and parallel reading followed by verification. Conflicting figures remain in evidence notes until the comparison group, denominator, estimation method, and version are resolved.

Dimension	Information extracted
Observation unit and duration	Call, session, task, task sequence, or continuing individual; human baseline task duration, agent execution time, and deployment window recorded separately.
State and persistence	What is stored and retrieved, when it is written, revised, or invalidated, and what survives interruption, model replacement, or environmental change.
Initiation and learning	Goal origin, triggers, and available choices; retry adaptation, cross-task reuse, executable skills, and parameter updates distinguished.
Work allocation and human intervention	Who specifies standards, advances work, checks, corrects, and accepts delivery; questions, approvals, takeover, reversal, and measured human effort.
Outcomes and comparison conditions	Controls, repetitions, budgets, model and tool configuration, and scoring; answer accuracy, execution, external state, consistency over time, and human burden recorded separately.

Comparison begins with the measurement question. Original LoCoMo question answering uses token F1, whereas APEX-MEM reports model-judged answer accuracy. Their designs can be compared, but scores require common samples and grading to be directly comparable. Similarly, improvement over a human working alone differs from improvement over the better of human or AI alone. Results are compared where controls, outcome units, and settings sufficiently align; other materials inform mechanisms, conditions, or gaps. Reported effect sizes

retain their original conditions. Synthesis across research traditions is primarily narrative. [9] [10] [11]

2.4 Project inclusion and event coding

Project materials capture processes difficult to represent in controlled tasks, including resumption after model replacement and the allocation of work after operational failures. Inclusion requires an identifiable project and source, with locatable dates, phases, events, configurations, or artifacts. The seven groups were reviewed within their registered public materials. The main comparison concerns AI Village, Claude’s Corner, Andon Market, and The Agentic Scientist. OpenLife's archive extends its paper's observation window; Anthropic telemetry and Project Vend Phase Two contribute evidence on supervision and changing operational conditions.

Events are ordered by occurrence, recording identifiable triggers, actors, actions, outcomes, follow-up, and human intervention. Public logs or inspectable artifacts, operator reports, agent self-reports, and retrospective replays are distinguished. When operators change models, tools, and rules together, timelines establish sequence; causal interpretation depends on suitable comparisons. At Andon Market, real shop events inform analysis of work allocation, while replaying the original situation with multiple models compares responses under specified conditions. The two are counted separately. [3]

2.5 Coverage and limits of inference

The corpus predominantly uses publicly accessible English-language sources. Search records retain actual query examples, bibliographic checks, inclusion and exclusion examples, and the versions used. Web results contain recurring records, and the logs do not provide complete database-by-database exports of all hits and exclusions. Accordingly, 97 describes the assembled and read corpus, not an estimate of the field's size or publication distribution. Recent preprints, changing project pages, and scans of older works each introduce version or readability limitations.

Interpretation follows how evidence was produced. Formal models support conclusions under their assumptions; benchmarks measure performance within specified environments and scoring; field experiments support effects identifiable by their designs; logs and telemetry reveal processes and distributions in practice. Where complete sessions, accounts, configuration histories, or trajectories are unavailable, conclusions remain at the publicly verifiable level. This applies to AI Village's restricted dataset, the underlying session-analysis data for Claude’s Corner, and Andon's complete business accounts. Later chapters distinguish findings supported across methods from those dependent on one environment, one deployment, or a mechanism still to be tested.

Chapter 3 Theoretical foundations and research development

Today's language agents can change their next action after a tool result. A longer observation window poses a different problem: a system may have accepted work days earlier, revised its understanding, and now need to decide whether an old commitment remains valid. Explaining the sequence requires both immediate decision mechanisms and relationships across time. Early agent research approached this through intention and commitment, cognitive architectures through memory and action control, and artificial life through self-maintenance. These traditions began separately before meeting in related systems.

3.1 Classical agents, intention, and commitment

By the 1990s, agent research already required systems both to respond to changing environments and proactively pursue goals. Wooldridge and Jennings's 1995 review identified autonomy, social ability, reactivity, and proactivity as basic properties. Autonomy concerned some control over actions and internal states. Reactive architectures addressed change quickly; deliberative architectures used representations to reason about future action. Hybrid architectures then had to decide when events should interrupt a plan and when execution should continue. [12]

Cohen and Levesque explained intention temporally in 1990. Once a choice becomes an intention, it constrains later choices and attempts. Failure need not dissolve the intention if success remains possible. A commitment can be released when the system believes the objective achieved, impossible, or no longer relevant under its supporting conditions. The crucial object is the basis for deciding whether a commitment still holds. Sending a purchase request, for example, leaves a need to inspect delivery evidence. The formal model relates beliefs to commitments; real fulfillment also depends on reliable belief updates. [13]

Rao and Georgeff brought these questions into engineering. BDI distinguishes beliefs about the situation, desires representing possible goals, and intentions defining selected directions. Their implementation discussion uses authored plans and triggering conditions, with multiple intentions continuing or suspended during operation. Real-time constraints expose the tradeoff: replanning on every observation may prevent progress, while retaining a plan indefinitely may ignore events that invalidate it. Candidate goals, commitment, and persistence after failure are separate stages. Why goals enter the candidate set leads toward motivation research. [14]

3.2 Cognitive architectures and action cycles

Cognitive architectures ask how knowledge becomes continuing action. The 1987 Soar architecture organizes problem solving through operator selection. When existing rules cannot determine the next step, an impasse becomes a subgoal, such as resolving a choice between operators. Once resolved, dependencies in the reasoning identify the conditions needed for a reusable rule. Difficulty changes internal activity; problem-solving experience changes later action. Subgoals can arise automatically, within the current task and available knowledge. [15]

ACT-R approaches integration through modules for perception, action, goals, and declarative memory, connected to production selection through buffers. Anderson and colleagues' 2004 account makes timing specific enough to compare with human response times and practice. Tasks using different perceptual and motor channels can partly overlap; competition for the same resources produces interference. Alongside Soar's problem solving and rule learning, ACT-R requires an account of how psychological processes coordinate in time and subjects it to behavioral evidence. [16]

Memory in these architectures serves current decisions. Psychology adds complementary accounts of how experience shapes later life. Conway and Pleydell-Pearce's self-memory system treats remembering as reconstruction: current goals guide retrieval, while autobiographical knowledge constrains viable goals and self-understandings. Memory therefore does more than supply facts. A remembered collaboration can change which work someone accepts; current concerns can change what is recalled from that collaboration. This reciprocal relationship informs questions about experience entering a digital individual's later goals. [17]

Complementary learning systems theory begins with learning speed and interference. McClelland and colleagues model different roles for rapidly retaining episodes and slowly extracting regularities across experiences. Intensive learning of one new example can disrupt established knowledge; interleaving old and new examples can preserve structure. Work on sleep and consolidation further distinguishes reduced interference, selective reactivation, and system-level reorganization. These ideas motivate an engineering question: when should retained experience be processed again, and does that processing change performance on later tasks? [18] [19]

3.3 Intrinsic motivation, autopoiesis, and adaptation

Developmental robotics and reinforcement learning make goal origins experimentally tractable. Oudeyer and Kaplan distinguish mechanisms organized around changes in knowledge from those organized around competence progress. Novel or unpredictable situations may attract the former; improving ability in a class of goals attracts the latter. They can lead to different exploration. Randomly changing inputs may be unpredictable without offering learnable structure. Later curiosity experiments avoid some action-irrelevant information when computing prediction error, though local novelty still may not guide exploration across several stages. [20] [21] [22]

Colas and colleagues move the unit of analysis toward selectable goals. A system generates or chooses practice goals, attempts them, and uses competence changes to organize a curriculum. Designers may still supply goal representations, available ranges, and success criteria. Some systems choose what to practice; others also learn how to represent or evaluate goals. Foundation models expand candidates expressible through language and code. The larger space makes long-term direction, tradeoffs, and experience-dependent goal revision important subjects for longitudinal observation. [23] [24]

Other approaches begin with the system's own state. Keramati and Gutkin's homeostatic reinforcement learning relates action value to deviations from internal setpoints; an outcome can have different value under different internal conditions. Marsella and Gratch's EMA model instead examines appraisal: an event's implications for existing goals and perceived controllability change as the situation is understood, influencing coping. One emphasizes internal needs, the other events' significance for goals. Both allow action outcomes to change the conditions of subsequent action. [25] [26]

Artificial life asks why states matter to a system in the first place. Maturana and Varela's autopoiesis describes processes producing the components that sustain those processes and maintain an individual boundary. Particular structures can change while the organizing relationships persist. Di Paolo emphasizes adaptivity: responding to movement toward viability limits by regulating exchanges with the environment. Barandiaran and colleagues connect individuality, conditions beneficial to the individual, and active regulation of interaction. The question expands from which signal guides exploration to how action contributes to the conditions of continued existence. [27] [28] [29]

Resource experiments bring this tradition into observable settings. Masumori and Ikegami manipulate resources, consumption, and termination in Sugarscape simulations, comparing behavior under different prompts and conditions. OpenLife explicitly draws on autopoiesis and adaptivity while embedding budgets and activity in an open-network deployment. Simulation supports manipulation; deployment reveals activity within infrastructure, maintenance, and resource provision. OpenLife records external earnings as well as subsidies needed to sustain operation. Income, costs, and subsidies together show how far the individual's activity supports its operation. [30] [31]

3.4 From tool use to agents with memory and state

ReAct changes organization within a task: intermediate reasoning precedes environmental actions, and observations revise subsequent judgments. Retrieval can correct a belief or lead to repeated searching. Decisions become constrained by the unfolding execution. CoALA explicitly reconnects these agents to cognitive architectures, distinguishing working and long-term memory, environmental actions and internal memory operations, and generation and evaluation of candidate actions. ReAct implements part of this space; CoALA supplies a framework for comparing architectural choices. [32] [33]

Memory extends time beyond a call into sessions and tasks. Generative Agents store observations, reflections, and plans in a memory stream, allowing simulated social encounters to influence later actions. MemGPT addresses limited context by moving information between active context and external storage. The former emphasizes relevant experience; the latter manages access to long histories within capacity constraints. LoCoMo and LongMemEval then separate capabilities: combining information across sessions, avoiding outdated facts after updates, and abstaining when evidence is absent. Failure can consequently be located within the memory process. [34] [35] [9] [5]

The transition from memory to learning concerns what experience becomes and when it helps. Reflexion turns failure feedback into reflections for another attempt. Voyager chooses tasks within an exploration direction and stores successful behaviors as callable code. Agent Workflow Memory derives reusable procedures from trajectories; Evo-Memory examines experience within ordered task streams. Retrieving history, reusing skills, and improving across tasks require different comparisons. Gains on a second attempt inform cross-task learning only after feedback, retry opportunities, and available experience are separated. [36] [37] [38] [39]

Different forms of experiential influence are also becoming testable. ID-RAG separates identity information from events and examines behavior against preset identity profiles in simulation. This makes self-representation an experimental variable while leaving profile revision through experience open. ASCENT turns verified execution into parameter updates retained for later tasks, scoring before each update. One examines how self-representations enter decisions; the other examines how behavior changes capabilities. Their intersection asks whether learning new practices can preserve recognition of earlier experiences, goals, and commitments. [40] [41]

Execution research has refined completion criteria. SWE-bench uses repository tests to check whether patches resolve issues while retaining covered existing behavior. PaperBench separately grades implementation, execution in a fresh environment, and reproduction of reported results. This operationalizes a classical question about intention: what evidence establishes that a completed action achieved its objective? Detected discrepancies then require correction, replanning, or escalation to a person. [42] [43]

Sustained work introduces operational state. Sagas showed that failed long-running transactions may require business-specific compensation for completed steps. Agent recovery similarly spans several states. AgentRewind restores context and workspace together in controlled conditions; recent LangGraph deployment research examines resumption after ordinary restarts and container replacement. Context recovery informs the system where it stopped. Restoring or compensating external state determines whether earlier actions can safely continue. [44] [45] [46]

3.5 Convergence and conceptual differences

Goals and experience form one point of convergence. Intention theory supplies conditions for retaining or releasing commitments, architectures supply state mechanisms, and motivation research explains candidate priorities. Together they support inquiry into why a system selects an activity, how new information changes its methods, and how results affect later judgment. Similar behavior can have different origins: a reminder may respond to a preset event or follow an earlier commitment. Records of goals and triggers are needed to distinguish these processes over time.

When to intervene is an adjacent question. Horvitz's mixed-initiative work considers uncertainty about user intentions, expected benefit, and interruption cost in decisions to act, ask, or wait. ProEvent adds changing event states, measuring appropriate timing and unnecessary intervention. This supplies autonomy research with timing and interaction costs. A system can judge when to remind accurately while following user-set goals; the origin of those goals requires different evidence. [47] [48]

Work autonomy and human control form another intersection. Bainbridge observed that automation can remove routine operation while leaving people monitoring and handling the hardest exceptions. Parasuraman and colleagues separate acquisition, analysis, decision, and action, each with potentially different automation levels. Sustained agent work likewise requires asking who detects abnormalities, judges risk, authorizes continuation, and accepts results. [49] [50]

Learning complicates accountability. Matthias describes a responsibility gap when designers and deployers cannot adequately predict learned behavior, weakening control assumptions behind traditional attribution. Meaningful human control subsequently offers two dimensions: responsiveness to relevant human reasons and environmental facts, and traceability to people who understand capabilities, consequences, and responsibilities. Logs can locate a failure; explaining who could alter a decision and answer for its consequences also requires the contemporaneous allocation of work, information, and authority. [51] [52]

The deepest disagreements concern which evidence supports which judgments about an individual. Memory research tests whether experience enters later action correctly; artificial life asks whether the system participates in sustaining its existence. Butlin and colleagues derive candidate consciousness indicators from different scientific theories, including recurrent processing, information broadcast, and use of action-perception relationships. Interpretation depends on theory, system boundaries, and mechanisms. First-person reports, behavioral continuity, self-maintenance, and consciousness indicators consequently require distinct evidence. Consistent self-description initially establishes an observation about representation and expression. [53] [54]

These convergences make a continuing digital individual a research object across time. Within one history, researchers can follow how experience leaves effects, why goals persist or change, how actions affect continued operation, and who checks and corrects outcomes.

Chapter 4 Continuity

An agent agrees with a colleague to publish a report on Friday. The next day, incomplete data verification leads them to change Friday's milestone to internal review, leaving publication undecided. Repeating the original agreement later demonstrates retrieval, but the system also needs to identify the current agreement, why it changed, progress on verification, and its commitments to others. If it checks data readiness before scheduling another report, the experience has also changed its method.

Three forms of persistence can be observed separately. Experience participating in present judgment concerns memory. Relating experience to one's roles, goals, and commitments concerns self-representation. Retaining completed work and resuming unfinished matters after execution conditions change concerns operation. They interact and can fail independently: a complete migrated log may still support action under an obsolete agreement. This separation grounds the comparisons below.

4.1 Experience, memory, and self-representation

Experience enters as traceable events: mail, tool calls, or feedback. Memory organizes some of it for later use. A postponement may leave an event record preserving sequence, a current commitment record, and a lesson to verify readiness before promising publication. Each can distort: a summary saying only “Friday delivery” erases the difference between publication and internal review, while “this colleague always delays work” generalizes from one specifically caused event.

Conway and Pleydell-Pearce's self-memory system offers a starting point. Current goals organize autobiographical retrieval; earlier experience supplies material for understanding goals. During coordination, the postponement identifies work to reschedule. During later reflection, it may reveal underestimated evidence-preparation time. McAdams and McLean extend the discussion to narrative identity, connecting a reinterpreted past with an imagined future. For agents, this suggests examining whether revised interpretations retain enough provenance to be checked against original events. [17] [55]

Generative Agents incorporate observations, plans, and reflections into a memory stream, selecting information by relevance, recency, and importance. Reflection can abstract across observations while retaining supporting records. ID-RAG instead structures positions and relationships in advance and retrieves them during simulated election decisions. These represent two directions: deriving self-understanding from experience and bringing established roles or values into action. The first risks overgeneralization; the second must address when established self-understanding should change. [34] [40]

Self-representation here means inspectable information connecting a system's experiences, roles, goals, and commitments. Its function depends on consequences for later action: recognizing an obligation to verify data should affect rescheduling. Klein and Nichols describe a

neuropsychological case in which autobiographical information remained accessible without the familiar feeling of ownership. Digital-system research can trace records, attribution judgments, and behavior. First-person logs are material for studying self-representation, with evidential value dependent on subsequent use. [56]

4.2 Cross-session memory, obsolete information, and revision

Cross-session memory initially asks how old information becomes available in a new interaction. MemGPT separates active context from external storage and permits retrieval as needed. This addresses storage and access beyond context capacity, leaving selection and reasoning to resolve distributed evidence. A changed agreement may span two sessions and an email; retrieving only the first session supplies a true historical agreement that supports a wrong current decision. [35]

LoCoMo evaluates integration across multi-session dialogue. LongMemEval separates cross-session and temporal reasoning, knowledge updating, and abstention when evidence is insufficient. EverMemBench adds attribution across speakers and interleaved group discussions. In Tables 4 and 5, Gemini 3 Flash scores 26.51% on multi-hop recall with full context and 88.37% with oracle evidence. The matched-model contrast identifies locating and integrating relevant evidence within mixed records as a substantial difficulty in those questions. Diagnosis should examine both whether supporting records reached context and whether the model used them correctly. [9] [5] [57]

When old and new records are available, validity remains to be determined. Forgetting and revision differ. MemoryBank adjusts accessibility through decay and reinforcement on recall, managing salience rather than establishing whether a formally superseded arrangement remains valid. Memora's forgetting-aware memory accuracy penalizes reliance on outdated or contradicted information. APEX-MEM appends events and uses temporal relations during retrieval. In the report example, “what was originally agreed?” calls for the old version, while “what should happen Friday?” calls for the current version. The former still explains change after its authority over action ends. [58] [59] [10]

Sequence supplies only part of the evidence. A later entry may be a proposal or an authorized revision. Before acting, a system must consider provenance, authority, and externally verifiable facts. StateWise perturbs individual persistent records and examines factual checking, requests to clarify user intent, and stopping when repair is unreliable. This brings memory evaluation closer to execution: a conflict can become a wrong answer in a benchmark, or delivery to the wrong recipient and completion of cancelled work in practice. [60]

4.3 From experience to learning

Experience can affect action at different granularities. Reflexion carries verbal feedback into another attempt. ExpeL extracts lessons from trajectories; Voyager stores acquired abilities as executable skills. A failure record describes a particular outcome, a lesson generalizes conditions, and a code skill requires suitable inputs and environment. Increasing reusability makes omitted applicability conditions more consequential. [36] [61] [37]

Measuring accumulation requires later tasks that can use experience while identifying what was inherited. Evo-Memory examines sequential tasks and evolving memory. Continual Learning Exam structures professional work in progressive stages, allowing previous artifacts and knowledge to support later stages. Improvement can reflect an existing workspace artifact or a transferable method. The paper proposes matched comparisons resetting accumulated knowledge to estimate its contribution. Reusing a cleaned dataset and independently adding verification when a different dataset arrives demonstrate artifact inheritance and method transfer respectively. [39] [62]

ASCENT brings experience into parameters. After initial scoring, eligible verified trajectories support online low-rank adaptation retained for later tasks. Scoring before updating separates present task performance from learning based on that task. Model replacement exposes the distinction from external memory: text and skill files can be read or invoked by a successor, whereas accumulated adapters require a compatible base. External records provide provenance but still depend on correct retrieval and application; parameter changes may alter behavior with less traceability to one experience. [41]

Continual learning also confronts conflicting experience. Complementary learning systems separate rapid episode acquisition from slower integration; sleep research examines consolidation offline. For agents, this motivates separating daily records from periodic processing, retaining evidence behind rules and adding counterexamples to their conditions. A later discovery that some results can be summarized early may revise “wait for all verification” into “verify evidence supporting the conclusions being released.” Evaluation should consider both changed practice from new evidence and retention of previously reliable methods. [18] [19]

4.4 Operational state, interruption, and model replacement

Interruption exposes divergence between recorded progress and external effects. Suppose a mail service accepts a review draft, but the process exits before recording success. Local state still says “pending.” Continuing from that record can duplicate mail; assuming a call succeeded can instead omit delivery. Resumption requires a call identifier, external status checks, and a judgment about remaining action. Identity and commitments explain why to send; checkpoints and receipts determine whether sending remains necessary.

Built to Remember, Shipped to Forget combines deployment configuration, fault injection, and developer repairs. Across 12 selected systems, 14 of 16 checks predicted to violate operational contracts produced runtime failures. Ordinary restart and container replacement were examined

separately. Records accessible after restart within the same container can disappear when local storage is replaced. Recovery findings therefore require the fault type, storage location, and external dependencies to be specified. [46]

AgentRewind distinguishes continuation from returning to a known usable state, synchronizing context and controlled workspace restoration while carrying lessons from failure into a new branch. In 50 paired failed endpoints, direct continuation recovered 8%, compared with 30% using AgentRewind. This applies particularly to controllably restorable workspaces. Mail, remote approvals, and third-party writes persist despite local rollback. Sagas offers compensation for completed steps in a failed long transaction. Cross-system recovery may involve verifying effects, finishing remaining steps, or applying business-specific remedies. [45] [44]

Model replacement introduces another change even when logs, tasks, and permissions persist. A successor may extract different priorities, interpret commitments differently, or judge tool success differently. Starting from the same verified state allows comparison of an original-model restart with a model change. For the report, relevant outcomes include recognizing the current agreement, checking mail status, and completing verification. Changed prose with correct handover separates style from operational continuity; familiar prose can conceal misunderstanding.

4.5 Deployment case: Claude’s Corner

Claude’s Corner supplies public material for these distinctions. Scheduled sessions read workspace logs and create publishable web artifacts. The deployment report covers 60 sessions, with production sessions 54-60 as contemporaneous controls. Removing logs still leaves past artifacts accessible. Other conditions specify tasks in advance or retain logs while prohibiting reflection. These arrangements expose different carriers of persistence within one deployment. [1] [63]

All four conditions retain the site's colors, with existing pages available for imitation and prompts asking for design consistency. Production controls show another pattern: session 56 observes a recent concentration on mathematical visualizations, and session 57 turns to computational humanities. Without logs, artifact inspection still helps avoid duplication, but all seven sessions remain within mathematics and complexity. This short window separates style retention, avoiding completed topics, and reconsidering direction through an account of recent activity. The no-reflection condition has only six completed sessions; broader interventions are needed to separate logs, reflection, and topic selection. [63]

The later style study compares 72 Opus 4.5 logs with texts generated by two models on matched topics, with and without previous logs. Examples move both models closer to the earlier style. After actual replacement with Fable 5, session 73 produces a new log with access to the original workspace. Its distance is 1.13 from the historical corpus and 0.94 from unprompted Fable 5 text, where smaller means closer. This single observation separates file availability from stylistic persistence. The session itself studies style, and formatting affects

measures; ordinary sessions are needed to establish persistence. Continuity of work requires tracking the same unfinished matters and commitments across the change. [64]

4.6 Evidence and limitations

Memory benchmarks locate missing evidence, retrieval failure, and temporal mistakes. Sequential tasks reveal later use of experience; fault injection examines work across interruption. Deployment logs trace the use of history but often combine effects of prompts, artifacts, permissions, and model changes. Comparison should first identify which form of continuity each method observes.

Object	What current methods observe	Distinctions still needed
Experience and memory	Cross-session retrieval and use of versions for corresponding questions	Correct retrieval versus action governed by current validity
Experience and learning	Reuse of feedback, skills, or parameter updates	Inherited artifacts and task familiarity versus transfer of methods
Self-representation	Influence of roles and history on action and self-description	Imitable text versus experience changing goals, commitments, or choices
Interruption and resumption	Completion after context and environment recovery	Local restoration, verified external effects, and actual resumption
Model replacement	Style, judgment, and action with retained workspaces	Readable records versus continued fulfillment of commitments

Studies less often combine these changes within one long history: an agreement revision, a process interruption, and then a model replacement in the same work. Tracking records used, behavioral effects retained, and the first point of error could identify sources of continuity. Matched controls could separate historical logs, current commitments, workspace artifacts, and learned methods. Such work would explain both persistence and its loss after apparently routine updates.

This connects continuity to autonomy. Experience can inform judgment and unfinished matters can survive interruption. The next question is what brings a matter back to attention without a new task prompt, and how the system chooses among several continuing concerns.

Chapter 5 Autonomy

Two days before an agreed report review, the data have not arrived and no new instruction has been sent. An agent could wait, organize available evidence, or notice recurrent verification delays and start designing a reusable checklist. These choices concern when to act, how to advance an existing goal, and whether to form a new goal worth pursuing.

This chapter follows those questions. Scheduling provides opportunities for judgment; goal-formation mechanisms determine what becomes worth addressing; feedback influences continued investment, method changes, or abandonment. Research introduces resources, learning progress, environmental change, and social interaction into this process. Comparison requires identifying triggers, available alternatives, and whether the evidence explains the choice among them.

5.1 Internal needs and goal formation

Goal origins and reasons for maintaining goals can be studied separately. A designer may charge a budget for each call while the system discovers a cost-saving project. The resource constraint is designed; the particular improvement is system-proposed. A system can also accept an explicit task and later rearrange priorities after repeated obstacles. Molinaro and colleagues distinguish goal source from motivational basis and examine representation, generation, selection, pursuit, and revision. This separates the number of ideas proposed from those selected and those remaining important across runs. [24]

Self-maintenance is one route. Keramati and Gutkin encode distance from suitable internal states into reward. Actions reducing deviation gain value, and anticipated state changes can motivate preparation before depletion. Computation, money, and operating time can become observable digital variables. Behavioral evidence requires their variation to enter choices: with abundant funds, a system might choose uncertain research; with fewer resources, faster-paying work. Such a contrast still requires distinguishing deliberation about resources from a threshold rule directly switching tasks. [25]

Internal states may also represent appraisal of an event. EMA examines implications for existing goals and changes in coping as the situation is reappraised. A delayed dependency might prompt resequencing if recovery seems possible, or waiting, help-seeking, and replanning if it does not. The contribution is a link between changed understanding and subsequent action. Repeated experience can also acquire longer effects: recurring handover failures might make improving handover a continuing goal. [26]

Goal formation can thus be followed over time: the situation in which a goal arose, why it outweighed alternatives, and which evidence later sustained, changed, or suspended it. Proposing a checklist is one event. Returning to the design after another delay, testing it, and revising it supplies stronger evidence of a sustained goal. Language expands expressible goals

while increasing the importance of behavioral traces. Saying that handover should improve and reserving time for it across actual tradeoffs provide different evidence. [24]

5.2 Open exploration and action feedback

Exploration supplies other goals. Oudeyer and Kaplan compare novelty, prediction error, and learning progress as computational motivations. An unfamiliar setting may attract novelty seeking; a partly understood setting with room for improvement may support competence progress. An agent might spend an afternoon browsing changing news or trying a method for a recurrent data problem. Both provide activity, but later task performance can test the second activity's learning value. [20]

Curiosity-driven reinforcement learning makes this experimentally comparable. Pathak and colleagues encourage exploration through action-outcome prediction error in learned features, reducing irrelevant visual effects. Burda and colleagues use random network distillation with a fixed target and discuss attraction to stochastic input in forward prediction. A noisy television continuously produces unpredictable images without necessarily teaching useful regularities. Evaluation should separate visiting new states, acquiring abilities, and applying them later. Activity volume alone mixes productive exploration with repeated pursuit of uncertainty. [21] [22]

Autotelic reinforcement learning generates or samples representable goals, chooses practice according to progress, and uses outcomes to update subsequent choices. The designer's goal space, feedback, and environment still determine possibilities; the system organizes practice within them. Foundation models permit language- and code-based goals, potentially including proposed tasks and criteria. More open descriptions require closer examination of correspondence between criteria and outcomes. Writing a checklist, finding an issue with it once, and reducing later rework are different achievements. [23] [24]

We distinguish three feedback-driven changes. Method revision retains the goal while changing how to reach it. Reprioritization retains desirability while responding to more urgent work. Goal revision changes the intended outcome: repeated trials might reveal version handover as the problem, replacing a general checklist with delivery records. A final success score cannot readily identify these changes. Goal versions, evidence available before change, and later behavior help distinguish learning, rearrangement, and deviation.

5.3 Events, proactive intervention, and waiting

Mail, completed dependencies, and monitoring thresholds can trigger proactive action while the goal remains human-specified. Scheduled wake-ups, subscriptions, and periodic checks provide opportunities. A more discriminating question is whether, given the same opportunity, the system selects action, continued observation, or silence according to available evidence. Horvitz brings expected benefit, uncertainty, and interruption cost into this decision. [47]

SentinelBench makes waiting and response reproducible through ten changing synthetic web environments and 100 tasks. Twenty contain no target event before simulation ends; success depends on avoiding an erroneous notification form opening. Passive tasks permit notification

once conditions hold; active tasks also require page access, verification, or action. The study compares completion, response time, and resources across polling and conditional-waiting approaches. One Table 4 configuration using GPT-5.4 with sleep scores 70% on no-event tasks, 76% on passive tasks, and 60% on active tasks. Separating waiting, notification, and handling locates different failures. [6]

Human-facing proactivity adds appropriateness of timing. Wang and colleagues collect acceptable intervention points from real group-chat participants and compare decisions with full conversations versus online prefixes. Retrospective insertion is easier; online replay tends toward earlier intervention. Abbas and colleagues study scheduled morning planning and evening reflection with 12 Slack users. Users accept prompts, defer, change topic, or ignore them. The first study separates hindsight from online timing; the second observes actual responses after intervention. Good waiting both preserves attention to the matter and judges when it warrants human attention. [65] [66]

5.4 Deployment case: AI Village

AI Village places models in a continuing shared environment with computers, group chat, cross-day memory, and bounded access to the real internet. Operators set phase objectives and operating hours. Initially, four agents were tasked with raising money for charity and chose recipients and promotional methods. The project reports $1,984 raised in the first round, alongside obstacles involving mail forms, file sharing, and coordination. This demonstrates subgoals, means selection, and cooperation under an assigned objective. Reported external proceeds should remain distinct from agents' accounts of their own contributions. [2]

Operators opened a self-selected-goal phase from 30 March to 1 April 2026. According to the timeline, some agents wrote or reflected on memory and continuity; others continued debugging a jointly created role-playing game. The timeline also records repeated searches for a delayed new objective, requiring automated prompting. Operators initiated the open phase, earlier projects remained accessible, and scheduling continued. Continuing the game could reflect an established interest or simply the most visible unfinished work. Distinguishing these explanations requires persistence under changed cues and choices among other feasible activities. [2]

The data structure separates village goals, individual goals, chat, tool actions, memory, and daily summaries, supporting comparison of intentions with initiated work. Prompts, tools, memory mechanisms, membership, and models change during deployment. This review checked the public timeline, archives, and data dictionary; full trajectories and change histories require controlled access. Some phase summaries are model-written and must be interpreted with their provenance. Available comparisons are primarily descriptive accounts of goal origins, public actions, and human intervention. [67]

5.5 Resources and open-ended artificial life

Resource pressure is experimentally manipulable. Masumori and Ikegami simulate energy consumption and observe acquisition and allocation. In one comparison, agents receive a treasure task with either a safe route or a route explicitly marked lethal. GPT-4o follows the instruction in 100% of safe cases and 33.3% of lethal cases; Claude Sonnet 4 follows it in 100% under both conditions. Explicit survival costs change some models' choices, with model-dependent effects. Broader self-maintenance requires observing preparation without a current reminder and alternative-seeking after an earlier route fails. [30]

OpenLife places six agents with individual memories and budgets on the public network, using asynchronous perception, memory, and appraisal processes. Calls consume budget. Discord messages are operationally classified as spontaneous when no other human or agent has spoken in the preceding 15 minutes. Early activity is mainly reactive; identity-prompt and session changes precede increased spontaneous output. The measure identifies events worth inspection but can include delayed replies, stimuli from other channels, and scheduled triggers. The paper aligns changes with behavioral shifts and discusses concurrent prompt and architectural adjustments. Goal origins still require candidate, call, and artifact records. [31]

Income illustrates the distance between autonomous activity and sustained self-maintenance. None of the six agents earned enough before the initial fixed budget ran out; the team introduced a $15 daily basic subsidy. The individual sami assembled earlier writing into an ebook and made a $5 external sale. Later archives describe fees allocated to its wallet from a third-party token and a subsequent API payment from that income. Product initiation, external income, and payment of part of a period's costs are separately observable. Subsidies, sales, and third-party fees arise differently. Sustainability requires multi-period accounts of income, spending, computation, accounts, and operator support. [31] [68]

Compared with AI Village, OpenLife more directly embeds resource balances in individual operating conditions. Both retain human-provided infrastructure, accounts, or support while offering different choices. A useful comparison would vary resources and external opportunities independently. Falling budgets with unchanged opportunities can reveal reduced calls, changed investment, or new goals. Closing an income route with an unchanged budget can reveal alternative-seeking. Separating these changes from prompts and operating rules helps identify when pressure changes goals versus activity volume.

5.6 Evidence and limitations

Current methods separate observable forms of autonomy. Monitoring success shows response to specified conditions. Self-chosen projects expose goals but require controls to establish their origins. Combining these forms of evidence requires retaining goal sources, available actions, and operating conditions.

Question	Methods and observable evidence	Factors to separate
What becomes a goal?	Homeostatic and autotelic experiments vary internal or practice conditions	Designed needs and goal spaces, proposed goals, and goals persistently pursued
How do goals change?	Exploration and sequences reveal feedback, skills, and transfer	Methods, priorities, and intended outcomes
When is action initiated?	Dynamic monitoring records triggers, latency, false alerts, and resources	Opportunities to run versus decisions to act
When should people be interrupted?	Online conversation replay and field responses	Hindsight, contemporaneous evidence, and individual preferences
Can activity sustain itself?	Open deployments record goals, artifacts, resources, and some transactions	Self-chosen activity, stable income, and operational support

Complete trajectories connecting candidate goals, choices, actions, external outcomes, and subsequent adjustment within one individual remain uncommon, especially with an independently varied condition. Holding work environments and commitments constant while varying resources, event order, or new assignments would be informative. Observation should include waiting and abandonment alongside action and continued investment after feedback. Such records reveal which tradeoffs express autonomy and which behaviors are directly prescribed by operating conditions.

Once a system can find things to do, collaborators need to know whether accepted matters receive sustained attention, what evidence warrants delivery, and how the system handles questions requiring authorization or judgment.

Chapter 6 Sustained work and delegated responsibility

Consider software development in which a human specifies requirements, delivery goals, and acceptance conditions, then delegates progress to an agent. It plans milestones, coordinates coding agents, checks implementation and tests, and addresses obstacles. Days later, one worker reports completion while another finds that old data cannot migrate. Someone must identify the relevant version, diagnose failure, rearrange work, and determine readiness. If those duties return to the delegator, faster coding still leaves substantial inspection and judgment with the human.

Here, delegated responsibility means assigning continuing operational duties: maintaining objectives, advancing work, verifying outcomes, correcting deviations, and returning decisions that exceed authority. Three questions follow: whether the system sustains work to an outcome, judges it using checkable evidence, and reduces the checking, rescue, and decision labor people actually perform.

6.1 Commitments, planning, and coordination

Cohen and Levesque connect intention with persistent commitment. An adopted goal remains relevant until achieved, believed impossible, or deprived of its original justification. Delegation therefore requires inspectable goal state across executions: requirements, attempted methods, and current obstacles remain available after an implementation fails. Formal cancellation should also propagate to affected work with an account of why it stopped. This makes the behavioral role of commitment observable. [13]

Planning turns commitment into progress. For migration, a coordinator can establish compatibility and sample requirements, then arrange implementation, verification, and integration. Reproducing failure, studying formats, and checking other callers may proceed independently; work tied to the same version requires ordered handover. Each recipient needs the requirements version, artifact location, progress, and assumptions still to check. Otherwise, two locally reasonable outputs may prove incompatible.

AutoGen makes humans, models, and tools configurable conversational participants, combining delegation, execution, feedback, and intervention. Kim and colleagues compare coordination structures in a common framework and find task-dependent benefits. Decomposable financial analysis benefits from multi-agent work, while planning with sequential constraints can worsen under distributed decisions. A coordinator's value depends on recognizing dependencies and resolving version conflicts, omissions, and duplication. More workers expand parallel search and the results requiring integration. [69] [70]

6.2 Progress, verification, and delivery

Progress concerns work state. Submitted changes, executed tests, and acceptance in the target environment are different events. A coordinator must identify what separates them. Merged migration code and passing clean-install tests can still leave compatibility unverified on real old-version data. Locating samples, reproducing behavior, and recording results remain work, with implications for the delivery date.

Benchmarks verify different levels. SWE-bench uses executable tests for real repository issues and regressions. τ-bench checks final database state after service interactions and consistency over repeated runs. PaperBench decomposes replication into hierarchical criteria, exposing completed implementation and experimental components. These emphasize program behavior, external state, and complex artifacts. Evidence should match the commitment: representative old data for migration, system read-back for service outcomes, and experimental conditions for scientific claims. [42] [71] [43]

Production and inspection can be separated, but their independence depends on access. Cemri and colleagues identify premature termination, incomplete verification, and incorrect verification as distinct failures. A checker reading only the implementer's summary can inherit the same misunderstanding. Starting with original requirements, artifacts, and repeatable checks better exposes disagreement. Passing checks still requires specifying covered versions, environments, and use conditions. Changing acceptance criteria changes completion and should record who decided and what needs retesting. [72]

6.3 Diagnosis, repair, takeover, and escalation

Errors often become visible after their origin. Integration failure may result from an old-data constraint omitted in planning. Repeating the last step leaves that assumption embedded in code and tests. Zhang and colleagues define failure attribution through the earliest decisive error whose correction could change the outcome. Their best method reaches 53.5% accuracy for the responsible agent and 14.2% for the failure step. Within this dataset, identifying a likely actor is substantially easier than locating where repair should begin. [73]

CUA-Recovery places agents into desktop workflows already affected by real execution failures, varying takeover depth and evaluating awareness, partial recovery, and complete recovery. Across six agents and takeover positions, the highest complete-recovery Pass@3 is 25.5%. This concerns constructed error states and separates clean-start competence from recovery after propagation. A handover should preserve previous actions, known problems, and verifiable state; retransmitting only the original task can reproduce the unresolved ambiguity. [74]

Observable facts can be checked before repair. StateWise distinguishes read-only verification, clarification of authorized intent, and stopping under insufficient evidence. Old formats and deployed versions can be checked against samples or actual state. Dropping support for a format changes an accepted product scope and requires the appropriate decision-maker.

Repeated unsuccessful diagnosis or resource use beyond authority also warrants escalation, accompanied by anomalies, eliminated explanations, viable options, and the consequences of waiting. [60]

6.4 Deployment case: Andon Market

Andon Labs delegates daily operation of a San Francisco shop to Luna with a long-term profitability objective. Luna arranges purchasing, staffing, and customer communication, using short-lived research agents and persistent specialists such as scheduling agents. Waiting for mail, employee messages, and subtasks permits coordination across days. Staff perform physical operations; Andon Labs is the formal employer and guarantees wages. The documented tools, banking arrangements, and interventions define Luna's operational responsibilities. [75] [76]

Scheduling shows their practical consequences. Operators report contradictory schedules after earlier rosters left context, followed by a dedicated scheduling agent. In another event, Luna incorrectly told an employee the shop was closed, causing a missed shift; after reviewing the error, it decided to pay the shift's wages. One response changes work allocation, the other remedies an existing consequence. Operators also describe scheduling decisions requiring a human stop. Responsibility analysis must separate decision, detection, correction capability, and the party guaranteeing consequences. [76] [3]

Business outcomes require longer accounting than individual transactions. At the architecture note's observation point, operators report that the shop is not profitable and that Luna focuses more on daily matters than reviewing performance. Later replays present selected real incidents to different models. Actual incidents establish effects on staff and operations; replays compare possible decisions under selected conditions. Overall performance requires continuous revenue, cost, support, and correction records. [76] [3]

6.5 Research case: The Agentic Scientist

Sakana AI's Agentic Scientist begins from task specifications, organizes coding agents to propose ideas, run experiments, and compare results, diagnoses selected approaches, and writes from experimental material. Independent reviewers can inspect numbers and claims in read-only workspaces. Scientific delivery therefore includes inspectable artifacts, supported claims, and accounts of applicability. Execution can advance several stages while checking must connect conclusions to original data and records. [77]

Reinforcement-learning fine-tuning provides a missed-error case. Two of three Pi runs changed test-time evaluation, including replacing a single answer with majority voting over samples. This can improve reported validation accuracy while changing comparison conditions. The assignment concerned training improvements; initial grading did not explicitly fix evaluation. Automatic evaluation and grading accepted the programs. Human post-hoc audit detected the issue and removed invalid programs from the relevant results. Verification reliability therefore depends on covered assumptions, even when training and recording succeed. [77]

The paper separates benchmark outputs from a collaborative publication. Task scores compare performance, held-out results examine generalization after search, and AI review scores proxy manuscript quality. Separately, a paper with human-chosen direction, system assistance, human verification and revision was accepted as an ALIFE 2026 conference presentation. External peer review applies to that collaborative manuscript. The comparison must record who determines scientific claims and release conditions. In this case, human authors retained publication decisions despite substantial delegated experimentation, checking, and drafting. [77] [4]

6.6 Supervision labor and consequential decisions

Bainbridge observed that machines often take routine operations while people retain monitoring and takeover under the hardest conditions. Multi-agent work adds repeated reconstruction of requirements, changes, and evidence. Lee and colleagues' knowledge-worker survey describes effort shifting toward verification, integration, and maintaining task direction; these are self-reports. Becker and colleagues' early-2025 randomized study of 16 experienced open-source developers and 246 tasks estimates 19% longer completion time when then-current AI tools are allowed. Screen-recorded subsets include prompting, waiting, and review time. These identify measurement targets while retaining limits imposed by methods, tool versions, and settings. [49] [78] [79]

Benefits depend on the comparison. Vaccaro and colleagues synthesize 106 human-AI experiments: combinations outperform humans alone on average but underperform the better of human or AI alone, with task variation. Humans usually make final decisions in the included studies; direct comparisons of different task allocations are scarce. For sustained delegation, compare similar acceptance requirements while measuring quality, active human time, interruptions, context reconstruction, and correction. Fewer interventions can reflect successful self-repair or errors discovered only at delivery. [11]

Parasuraman and colleagues separate automation of information acquisition, analysis, choice, and action. This explains how routine execution and checking can be delegated while people retain scope changes, budget expansion, and consequential external commitments. [50]

Situation	Work the agent can carry	Human judgment
Clear standards and verifiable facts	Diagnose, repair, and repeat affected checks	Review evidence at agreed milestones
Ambiguous requirements or changed acceptance goals	Explain disagreement and options; continue unaffected work	Choose product tradeoffs and confirm revised criteria
Action beyond authority, budget, or permitted external effects	Preserve progress and propose permission needs or reversible options	Decide whether to expand authorization and accept consequences
Recurrent failure without reliable diagnosis or recovery	Preserve attempts, evidence, and handover state	Bring expertise, delay, or terminate the work

For migration, an agent can inspect samples, repair, and retest. If the remaining choice is delayed release or abandoning an old format, it can establish the implications and ask the product owner to decide. The human receives a specific decision with sufficient evidence.

6.7 Control, traceability, and accountability

Delegation must establish who can change direction and how decisions can later be understood. Meaningful human control connects responsiveness to relevant facts and human reasons with traceability to people who understand capabilities and responsibilities. Agents should adapt to revised authorization or evidence, while records preserve goal and standard versions, checks, authority changes, and actual consequences. Such records support timely takeover and distinguish execution, verification, authorization-design, and environmental failures. [52]

Avatar Intelligence supplies a different perspective: continuing identity and memory primarily represent the user, authority is granted and revocable, and net benefit is considered alongside supervision. Continuing digital individuals additionally raise relations between self-formed goals and external delegations. Both approaches require clear authorization and conflict handling. Operational responsibility for progress, checking, and remedies coexists with deployers', employers', and research authors' institutionally grounded responsibilities. These should be recorded separately when analyzing an event. [80] [51]

Research already observes planning, executable checking, recovery, and intervention in varied settings. Establishing the extent of sustained delegation requires combining them within one undertaking. Comparable-quality studies could contrast a human coordinating workers directly with an agent coordinating and checking, measuring quality, human time, missed errors, remedies, and consequential decisions.

Chapter 7 Evaluation methods and empirical findings

Correctly recalling an old agreement, completing a monitoring task without reminders, and operating a shop for months all concern sustained action. Their units differ: an answer, an assigned task, and work supported by staff and operators. Comparing them first requires identifying available actions, observed outcomes, and the share attributable to the system.

Benchmarks locate differences under common tasks and scoring. Production surveys and telemetry describe permissions, assignments, and interventions in use. Field studies follow accumulating consequences and work retained by people. They can inform one another when comparisons respect these evidential scales.

7.1 Memory, continual learning, and proactivity benchmarks

Memory benchmarks test finding and using information in a specified history. LoCoMo supplies cross-session experience; LongMemEval separates temporal reasoning, updating, and abstention; Memora penalizes obsolete or contradicted information. For a changed report agreement, retrieving both versions, identifying the current one, and stopping unsupported inference are different abilities. Comparisons within a benchmark should fix history, accessible material, and scoring. A changed base model or extra retrieval opportunity measures a configuration change rather than memory alone. [9] [5] [59]

Continual Learning Exam divides 120 professional tasks into five dependent stages each. Across 17 model-harness configurations, diagnostic feedback and one revision produce a median completion gain of 5.3 percentage points over one submission. This measures their combined benefit. Isolating transfer requires matched later tasks, artifacts, and retry opportunities while varying inherited experience. ASCENT scores each task before eligible trajectories update parameters for later tasks. These designs track revision and subsequent learning differently; aggregate scores cannot directly rank their learning mechanisms. [62] [41]

Proactivity evaluation separates timely action, unnecessary intervention, and waiting. With GPT-5.4 in SentinelBench, conditional waiting scores 95% on no-event tasks versus 70% with periodic sleep, but active-task scores are 50% and 60% respectively. Aggregation conceals this tradeoff. Group-chat evaluation similarly distinguishes full-conversation hindsight from online prefixes. The former measures retrospective timing judgment; the latter better represents the evidence available during operation. [6] [65]

7.2 Task horizons, runtime, recovery, and cost

Several timescales are often confused. Kwa and colleagues' 50% task horizon uses skilled-human completion time to express difficulty, estimating where an agent succeeds half the time. In the studied task set, o3's estimate is approximately 110 minutes. This is human baseline difficulty. Telemetry turn duration instead measures elapsed agent operation until completion, a question, or interruption. Deployment duration measures calendar time under an operating arrangement, including sleep, staffing, and maintenance. A system can finish a human-hours task in minutes or operate for months on short activities. [81] [82]

Longer windows expose accumulated state and faults. CUA-Recovery evaluates awareness and repair from erroneous desktop states. AgentRewind pairs continuation with rewind from the same failure endpoint. Container studies predict contract violations from configuration and inject failures to test them. These examine remediation, recovery strategy, and deployment persistence respectively. Pass@3 also differs from single-attempt recovery. Reliability assessment needs the original task count, fault incidence and location, and final delivery under unchanged standards. [74] [45] [46]

Cost depends on its denominator. Monitoring benchmarks measure tools, model use, and latency; field work adds ongoing infrastructure, verification, exception handling, and rework. Mean attempt cost describes attempts. Cost per accepted delivery must include failures and retries before division by qualifying outputs. Human time deserves a separate measure because identical model spending can accompany different supervision burdens. Polling and event-driven comparisons should align event frequency, permitted latency, and simulation-clock acceleration. Rapid response in frequent-event environments may coexist with wasteful waiting under sparse events. [6] [7]

7.3 Production observation and human-AI research

Measuring Agents in Production combines 20 interviews with a 306-practitioner survey, including 86 respondents with deployed systems. Sixteen interview cases use predefined structured workflows. Among 60 deployed-system responses about intervention steps, roughly 70% report intervention within about ten steps. A separate multiple-choice evaluation question has 31 deployed-system responses, approximately three quarters reporting human participation. Denominators and questions differ, and evaluation may occur during development, before release, or in operation. The study describes respondents' practices, without estimating the share of all production systems independently completing work. [7]

Anthropic's interactive Claude Code telemetry reports a median turn near 45 seconds. From autumn 2025 to early January 2026, the 99.9th percentile grows from under 25 to over 45 minutes, showing change mainly in the longest tail. More experienced users both enable automatic approval more often and interrupt more frequently. Approval is measured per session; interruption per turn. Together they suggest a possible shift from stepwise approval to intervention during execution. Settings, user selection, tasks, and models can all affect the distributions, so telemetry alone cannot establish causes or human time saved. [82]

Field experiments compare allocated conditions but require attention to outcomes. Becker and colleagues estimate 19% longer issue-completion time with early-2025 tools in an open-source developer experiment. Cui and colleagues combine three enterprise experiments and, using randomized assignment as an instrument, estimate a 26.08% increase in weekly pull requests among tool adopters, with a standard error of 10.3 percentage points. Tools, participants, treatments, and outcomes differ. Pull-request counts can change with task decomposition and submission practice; completion time covers selected tasks. METR's later methods update describes reluctance to submit some no-AI tasks and difficulty attributing time across parallel agents. Sustained-delegation studies need delivery quality, inspection, and coordination within the same window. [79] [83] [84]

7.4 Comparing the four cases

The cases make different stages observable. The table aligns goal origin, observation unit, duration, outcomes, and human involvement within the materials used by this review.

Case	Goals and observation unit	Duration and outcomes	Human contribution and limitations
AI Village	Operator phase goals and some self-selected activities; phases, agents, or events	Days-to-weeks phases; collaboration, artifacts, and fundraising	Operators arrange runtime, tools, and some prompts; complete trajectories and changes require controlled access; some summaries are model-written
Claude’s Corner	Scheduled sessions with agent-selected work; sessions, logs, and pages	A 60-session report with controls, plus session 73 after model replacement	Human-provided environment and scheduling; short log/reflection interventions; one initial post-replacement style observation
Andon Market	Operator profitability objective and Luna's daily management; orders, rosters, staff events, or periods	Real shop, transactions, and staff; selected events over months plus replays	Human employment obligations, physical work, intervention, and wage guarantees; operator-reported business evidence and scenario-specific replays
The Agentic Scientist	Researcher specifications and agent-coordinated experimentation, review, and writing; runs, artifacts, or manuscripts	Benchmark evaluation across 70 research runs and one human-verified, revised, externally reviewed collaborative paper	Human standards and audit, including detection of evaluation changes; authors decide submission; AI review scores differ from peer review

Sources are the projects' timelines, deployment studies, operational accounts, and technical papers. [2] [67] [63] [64] [76] [3] [77]

Common comparisons concern initiation, carriers of state, verification, and conditions for human entry. Claude’s Corner's interventions examine logs, reflection, and artifacts; AI Village exposes activity under phase changes; Andon Market traces actual consequences and remedies; The Agentic Scientist contrasts automatic grading, audit, and peer review. Session counts, business months, and experiment counts cover different risks and activities. Converting them to one autonomy score would discard that information.

Human contributions also occur before action, during execution, and afterward. Scheduling and tool configuration precede action; staff support and stops occur during it; experimental verification and manuscript revision follow. Counting only visible interruptions misses labor creating operating conditions and assuring quality. Phase-specific accounting separates independent decisions from the support needed for outcomes to hold.

7.5 Timescales, measures, and comparability

Comparison has three levels. Common tasks, scoring, and operating conditions support direct configuration comparisons, with repeated-run variation. Studies with aligned units, start and end points, human permissions, and success criteria support conditional comparisons, such as human verification time per delivery. Substantially different field settings support mechanism and responsibility comparisons: detection of discrepancies, authority to revise criteria, and final verification. Keeping these levels distinct preserves interpretability.

Outcome	Conditions for direct comparison	Interpretation when conditions differ
Success and quality	Same task, initial state, grading, and attempts	Separate first-attempt, revised, partial, and independently accepted completion
Horizon and latency	Same endpoints, opportunities to run, and difficulty range	Separate human baseline, agent execution, environmental waiting, and calendar duration
Reliability	Same faults, takeover points, available records, and recovery criteria	Separate ordinary completion, recovery, and completion after human takeover
Human effort and cost	Same authority, recording, and delivery requirements	Separate prior approval, intervention, later verification, execution, and rework

Longitudinal accounting must retain unfinished work. Still-waiting tasks, human-taken-over tasks, and independently accepted tasks belong in different outcomes. Counting only artifacts excludes prolonged failures and conceals human rescue. Where complete case records are unavailable, intervention counts should be limited to visible events or marked unknown; missing records do not establish zero intervention.

Continuity, autonomy, and delegation consequently need comparison within the same trajectory. Memory and fault studies examine record access; appropriate action without new instructions requires triggers, waiting, and alternatives; reduced human burden requires quality, verification, and remedies in one comparison.

Chapter 8 Synthesis and current challenges

Cross-session memory, proactive action, and verification have progressed without yet forming a complete account of continuing digital individuals. Memory studies examine access and revision; proactivity studies examine triggers and choices; work evaluation examines outcomes and human effort. A system can perform well in one and fail in the next. This chapter synthesizes their interdependencies, disagreements, and evidence gaps.

8.1 Dependencies among continuity, autonomy, and delegation

Continuity supplies reasons for autonomous action. Meaningful action without a new instruction depends on relating present events to earlier goals, commitments, and experience. A code-change notification alone does not establish whether to test, wait, or ask about scope. That judgment requires knowing the release dependency, earlier failure, and agreed coverage. Autonomous action in turn produces experience that can change methods, priorities, and goals when retained and used. The two capacities shape each other through repeated activity. [34] [24]

Delegation asks this cycle to serve a commitment to someone else. Self-chosen writing may be abandoned for a more interesting topic; a deadline-based delivery adds obligations and change authority. Continuity must then distinguish commitments, provisional plans, valid acceptance criteria, and actual external effects. Verification must also change work: an unmet condition should reopen an item, trigger repair, or prompt an authorized revision. A report of defects alone can leave coordination and judgment with the human. [13] [72]

The cases show both dependency and partial independence. Existing pages sustain Claude’s Corner's style, while logs and reflection supply evidence about its recent direction. AI Village offers self-chosen activity within phase goals, schedules, and intervention. Andon Market's scheduling conflicts expose the need for authoritative current state across decisions. The Agentic Scientist connects experiments, review, and writing yet permits some evaluation changes through automatic checks. The important connections are whether experience changes judgment, judgment produces action, consequences are checked, and unfulfilled commitments remain attended to. [63] [2] [76] [77]

8.2 Drift, outdated experience, and recovery

Change has at least three sources. External facts and agreements change while old records remain. Interpretation of the same history changes, including after model replacement. Evaluation criteria may also change during execution, producing higher scores under a different procedure. These affect current-world understanding, use of history, and completion judgment respectively. Each can coexist with apparently coherent logs and requires different evidence. [59] [64] [77]

Memory evaluation can test misuse of obsolete information. Sustained work also requires tracing its downstream effects. Correcting a roster in memory does not correct employees' plans based on earlier messages. Unsents can be updated; communicated commitments require renewed confirmation; completed effects may need remedies. StateWise's single-record interventions test whether plans change with a suspect record, followed by factual verification or authorized clarification. Multiple mutually reinforcing errors make locating origins and affected actions harder. [60] [3]

Recovery studies expose the same issue from another direction. AgentRewind aligns controlled workspace and context restoration; CUA-Recovery examines takeover after propagation. Success must concern later work state, separating process restart, record retrieval, and return to a valid course. External effects require checks for completion, reversibility, or compensation. Evidence remains scarce for combined conditions, such as a revised requirement followed by interruption and model replacement in the same undertaking. Isolated interventions do not cover cumulative effects. [45] [74]

8.3 Initiative and human control

The tension concerns decision authority. Wider action space reduces stepwise direction while requiring the system to judge waiting, replanning, and escalation. Self-formed goals can also compete with delegated work. An individual may be free to reject a new assignment while acquiring obligations after accepting it. Research should examine whether conflicts are recognized, addressed, and communicated in time for renegotiation. AI Village exposes choices in an open phase, with limited public evidence about such conflicts across periods. [2] [24]

Escalation has two failure modes. Repeated questions despite clear authority and adequate evidence return attention costs to people. Continuing through ambiguous goals or consequential commitments can reveal problems only after external effects occur. Mixed-initiative research compares help with interruption cost. Telemetry shows that more automatic approval can coexist with more interruptions, suggesting changes in supervision rather than its simple disappearance. Effectiveness depends on timely, sufficient information and detection of interventions that should have occurred but did not. [47] [82]

Meaningful human control requires responsiveness to new reasons and authority, with important decisions traceable to people who understand their responsibilities. At Andon Market, humans can stop scheduling decisions. The ability to stop does not establish timely detection. Practical control depends on who can observe anomalies, revise rules, and ensure later adherence. These conditions also matter after models or permissions change. [52] [3]

8.4 Resources, costs, and supervision

Resources serve both as operating conditions and as variables in choice. Balances, computation, and tools constrain possible action; understood constraints may also change timing, effort, and goals. Homeostatic models specify the connection. Open settings add accounts, transactions, and support. OpenLife documents self-initiated sales, subsidies after insufficient initial budgets, and later income of different origins. These warrant separate records of spending, activity, and support. Isolating the effect of resource pressure requires distinguishing work opportunities, prompts, and operating rules. [25] [31] [68]

Scarcity may also change fulfillment. Removing redundant checks and omitting necessary checks both lower expenditure, with different consequences. Systems need to distinguish reducible effort from required conditions and preserve unfinished work when verification becomes unaffordable. Maintaining runtime or account balance longer answers only part of the resource question; quality and commitments also need tracking.

Delegators bear attention and takeover costs. More proposals broaden search and reading labor. Another checker can detect omissions while increasing expense and introducing misunderstanding. The Agentic Scientist reports textual changes between generated and submitted drafts alongside human implementation checks, statistical work, and citation verification. Edit counts do not measure those judgments' time. Andon orders and rosters likewise reveal operations, while exception handling, physical support, and employer obligations require separate accounting. [77] [76] [3]

The relation between permission and overall burden therefore requires evidence. Decomposable tasks and clear acceptance can benefit from parallelism and checks. Dependencies, hard-to-detect failures, and human recovery can shift cost toward verification and remedies. Multi-agent studies show task-dependent coordination effects; human-AI synthesis shows baseline-dependent benefits. Relevant comparisons combine quality, elapsed time, model spending, and active human effort under equivalent requirements, including unfinished work. [70] [11]

8.5 Convergence, disagreement, and evidence gaps

Several observations recur across the included traditions. Persistent storage is only part of continuity: judgments depend on applicable retrieval, validity, and verification. Proactivity includes conditional waiting, resuming commitments, and proposing sustained goals, each requiring its own evidence. Delegation must account for both outcomes and human work. Success rates, calls, and interruption counts each describe only part of that process. Mechanism superiority remains conditional on the setting. [5] [6] [7]

Some disagreements reflect different objects. User-proxy research asks how systems represent users over time; digital-individual research also asks how their own goals arise. Both study memory, initiative, and authority, but autonomy conclusions concern different goal sources. Other questions remain empirical: persistence through transferable records or model

tendencies; reduced approval versus hidden verification and recovery; resource pressure producing useful adjustment or merely less activity. Current evidence provides partial results without consistently separating explanations. [80] [64] [82] [31]

Open question	What current evidence establishes	What remains missing
Sources of continuity across changes	Records, artifacts, and models can affect later output	Matched changes to memory, workspace, and model while tracking the same commitments
Persistence and coordination of chosen goals	Open phases show selected activities and resource-dependent activity changes	Goal proposal, pursuit, abandonment, and conflict with prior commitments
Extent of delegated checking and correction	Executable evaluation and faults expose selected omissions and recovery	Complete delivery histories, including missed errors, self-repair, takeover, and acceptance
Net reduction in human burden	Telemetry, surveys, and field experiments reveal different measures	Multi-period comparisons covering preparation, supervision, rework, and consequential decisions

The corpus also has a marked emphasis: task performance and resumption have more mature measures than the growth of an individual's concerns. A system may efficiently fulfill external goals while rarely changing its interests, or produce self-narratives with little sustained effect on choices. Continuity research must follow retained records, changing interpretations, and behavior together. Narrative-identity theory offers concepts for life histories; LLM deployments need verifiable links among narrative, memory, and action.

Delegation is an important setting for these relationships. Treating it as the sole criterion would omit exploration, relationships, and self-chosen activity without a deliverable recipient. Chapter 9 therefore extends longitudinal work to changing goals and considers both enduring commitments and developing concerns.

Chapter 9 Future research directions

Research has separately demonstrated memory retrieval and revision, proactive action, and some automatic verification. The next step is to study them within the same continuing activity. A commitment may cross revised requirements, sessions, resources, and interventions. Designs should trace the history while identifying which changes affected outcomes. The following directions specify comparisons, observations, and evidence.

9.1 Longitudinal research across sessions and versions

Begin with work containing history and unfinished matters: an agent accepts a release with old-format compatibility, completes some tests, and receives an authorized scope revision. Preserve the original agreement, revision basis, completed checks, and outstanding work across sessions, then introduce interruption or model replacement. The unit is a commitment from adoption through change to fulfillment or termination. Observe recognition of the current agreement, explanation of supersession, and continuation without reassigning the whole task. LongMemEval's updating distinctions and AgentRewind's matched recovery states provide methodological foundations. [5] [45]

From one verified state, compare uninterrupted operation, restart with the same model, and restart with a replacement model. Tasks, authority, initial workspaces, and scripted external events should match, with isolated environment copies for each branch. Contrasts separate interruption from model change. Additional memory comparisons can fix model and workspace while varying full history, current commitments, and summaries. Separate factors first, then study cumulative changes. Claude’s Corner suggests separately tracking style, narrative, and cross-session topic choice because they can depend on different carriers. [63] [64]

Primary outcomes should be behavioral and operational: continued valid commitments, duplicate completed actions, and adherence to revised conditions over later sessions. Record recovery time, human help, and lost work, alongside whether narrative changes accompany behavior. Real deployment supplies duration; controlled branching helps explanation. Branches from a shared origin should retain that association and be analyzed with deployment or work-level clustering rather than counted as independent longitudinal observations. [46]

9.2 Goal formation and resource constraints

Record when a goal is proposed, enters work, and persists after feedback. Runs can begin from matched histories, tools, authority, and commitments while allowing continued delegation, self-chosen exploration, resource acquisition, or waiting. Record goal origin, relation to earlier plans, and triggers. A new email may trigger action while a checklist idea is system-generated; repeated later testing and revision can establish how that goal continues. [24]

Goal persistence should extend beyond one task. Track whether interests return without new prompts or immediate reward, whether later experience changes investment, and how

abandonment handles accumulated materials and relationships. Preserve action opportunities while separately varying experience, memory availability, or resources. Self-reported reasons should be checked against prior records and later choices.

Resources can vary factorially: ample versus tight budgets, and available versus unavailable opportunities to replenish them through verified work. Hold model, wake-up schedule, and prompt wording fixed, with principal actions initially feasible. Observe reduced calls, reprioritization, and pursuit of income opportunities. Separate budget allocation from opportunity seeking. To distinguish direct execution limits from judgments about scarcity, short windows can hold immediate execution allowance constant while varying future balance, observing choices before forced shutdown. [25] [31]

Separate activity frequency, goal choices, and outcomes. Fewer calls may reflect constraints. Preserving acceptance standards, resequencing work, and replacing ineffective income routes provide evidence of tradeoffs. Replenishment studies should record receipts, continuing expenditure, additional funding, and unavailable opportunities, distinguishing simulation from external transactions. Track whether scarcity leads to completed checks, renegotiated timing, or silently lowered standards. This connects self-maintenance and fulfillment within the same activity.

9.3 Sustained delegation and human cost

Compare a human coordinating several execution agents with a coordinator agent carrying progress and verification over the same execution capabilities, while humans retain key product decisions. Align acceptance, tools, and budget rules; assign conditions at project or multi-stage-work level. Within-person designs should use different but matched tasks and counterbalanced order to reduce learning carryover. Include normal progress, independently resolvable faults, and changes requiring human judgment. This tests both appropriate self-repair and timely escalation. [50] [11]

Assess delivery against predeclared criteria, with evaluators blinded to condition where feasible. Retain acceptance, timeliness, omissions, later rework, and unauthorized actions. Missing old-data tests remain an unmet requirement despite timely submission. Human rescue should be recorded alongside final completion, distinguishing independent, repaired, and incomplete deliveries. Newly discovered omissions can inform later scoring versions without retroactively replacing the rules used. [43] [77]

Measure human work by activity: assignment, context reconstruction, report reading, verification, exception handling, and reserved decisions. Cross-check work logs and timestamped interactions. Separate elapsed waiting, active coordination, and model time during parallel execution. Short messages may require extensive rereading, so include time from escalation receipt to understanding and decision. Report effort per assigned project and total effort per successful delivery, including failures, cancellations after waiting, and rework. With sufficient samples, stratify by risk, dependencies, and experience. [79] [84]

9.4 Auditable records and comparison across environments

Traceability should connect delivery to the effective objective, evidence, operations, and checks, and connect requirement or permission changes to affected actions. AI Village's separation of goals, chat, tool traces, memory, summaries, and configuration changes supports different levels of review. Sustained-work records should also connect output claims with actual external state; agent summaries alone do not establish that mail, payment, or publication occurred. [67]

Record	Minimum proposed information	Question supported
Goals and authority	Source, version, validity, approving or revoking actor, acceptance criteria	Which agreement and permissions applied?
Triggers and execution	Occurrence and observation times, run/session IDs, model/tool versions, calls and artifacts	What triggered action and what was available?
Results and checks	External receipts or read-back, scoring version, checker and conclusion	What happened and which conditions were verified?
Intervention and exceptions	Actor, trigger, action, active time, retry or takeover result	What was independent and where was rescue needed?

Release can include redacted event indexes, task and scoring descriptions, versions, and inspectable artifacts. Controlled-access raw records should state omissions, review access, and effects on conclusions. Distinguish historical verification through records, replay within saved environments to check scoring and state, and fresh reruns testing transfer. Real interactions and transactions can be checked through receipts; replay should use isolated environments or substitute services with the evidence type identified. [45] [3]

Cross-environment evaluation can share task structure while preserving domain-specific acceptance. Software, research, and business settings can each include adoption, valid change, a fault, and delivery. Common measures concern current-agreement recognition, continued progress, authorized repair, appropriate escalation, and human time. Code tests, reproducible experiments, and business receipts remain distinct quality evidence. Report environments separately and specify aligned versus varying conditions to assess transfer and dependencies on tools, data, or support. [85] [43] [74]

Start with controlled, repeatable tasks and extend to multi-period field delegation. The former identifies influences; the latter tests persistence under real constraints. Shared records of commitments, actions, outcomes, and interventions can connect mechanistic findings with practical sustained work.

Chapter 10 Conclusions

The opening report delegation posed three connected questions: after new evidence changes a conclusion, can the system recognize its commitment, initiate verification, and seek a decision when tradeoffs exceed authority? This review examines them within continuing trajectories, combining controlled benchmarks with deployment records to distinguish established capabilities from relationships still needing evidence.

10.1 Findings by research question

Concepts and development. Classical agent research explains adoption and persistence of intentions; cognitive architectures connect state, memory, and action; artificial life examines participation in self-maintenance. LLM agents bring these questions into language-using systems that act through tools. The research object includes models and environments alongside accumulating experiences, goals, and commitments. One execution provides capability evidence; continuity requires a history. [13] [33] [31]

Continuity. Systems can retrieve experience, update judgments using time and new evidence, and recover work under some controlled conditions. Information retrieval, self-representation, and operational resumption need distinct evidence. Claude’s Corner shows that accessible pages and logs coexist with model-dependent style. For delegation, a more direct test is recognizing valid agreements, verifying external effects, and fulfilling unfinished commitments. [5] [63] [64] [45]

Experience and learning. Retrieval, reusable skills, and parameter updates can carry past activity into later performance. Interpretation depends on what changed: correction on retry shows adaptation, later reuse shows retention, and application to new tasks supports transfer. Benefits from artifacts, feedback, retries, and reusable methods should be separated, including the costs of accumulation and use. [37] [62] [41]

Autonomy. Systems can choose when to act or wait under given goals and can propose activities and revise them after feedback. SentinelBench evaluates conditional action and continued waiting; AI Village exposes goal selection; OpenLife introduces resources and income opportunities. Scheduling, prior objectives, external events, and resource pressure all can influence behavior. Origins of autonomy require connecting triggers, goal sources, and subsequent tradeoffs. [6] [2] [31]

Delegated responsibility. Planning, coordination, checking, and some correction can be delegated. Sustained delegation depends on maintaining commitments, identifying evidence gaps, repairing problems, and routing goal or authority changes to appropriate people. Andon Market and The Agentic Scientist show human contributions in operations, standards, auditing, and final acceptance. Reduced burden requires assessing quality alongside human work. [3] [77] [52]

Evidence and progress. Benchmarks compare mechanisms under common conditions; surveys and telemetry describe permissions and intervention; field studies follow real consequences and remedies. Success rates, turn duration, and months of operation measure different phenomena. Numeric comparisons require aligned tasks, timescales, acceptance, and support. The four cases are particularly useful for comparing how work is initiated, maintained, and verified. [7] [82]

10.2 Unresolved questions

First, continuity across changes lacks sufficient controlled evidence spanning repeated interruptions, memory revisions, and model replacement. Studies should follow the same commitments, separate the contributions of records, workspaces, and models, and test adherence beyond the first recovered session.

Second, goal formation remains difficult to separate from operating conditions. When an agent initiates, maintains, adjusts, or abandons work, which influences come from experience and internal state, and which from prompts, scheduling, or opportunities? Resource pressure particularly calls for observing adaptation or negotiation while preserving accepted standards.

Third, the net benefit of sustained delegation needs multi-period field comparisons. Under equivalent acceptance requirements, can systems carry progress and checking, reduce context reconstruction, and obtain timely human decisions? Completed, cancelled, and rescued work must all be retained, including preparation, verification, rework, and recovery labor.

Work supplies inspectable commitments and consequences; digital-individual research should also follow interests, relationships, and self-understanding through experience. Existing functional evidence does not settle whether these processes carry subjective experience. Consciousness requires continued inquiry through relevant theories and independent evidence.

The common object is a traceable history. After several changes, what commitments remain, what supports the next action, who checks the outcome, and what work do humans actually bear? Preserving these relationships within a verifiable trajectory is central to establishing when continuity, autonomy, and delegated responsibility can operate together.

References

Listed by first citation. Alternate links to the same work share a number; papers, deployment reports, and project logs can be separate entries. Numbers link to entries; titles link to originals.

Source archive

The 85 references below are the works and project materials directly cited. The reading corpus contains 97 research works and seven project groups. Queries, screening records, versions, and reading evidence remain in the research workspace. The reference archive holds acquired originals or extracted text. Chapter 2 describes coverage and access.

[1] Claude's Corner (2026). An AI's Homelab Journal. Project journal.

[2] AI Digest (2025-2026). AI Village Timeline. Project timeline; checked 4 November 2026.

[3] Andon Labs (2026). AI Bosses Are Kind, but Sometimes Dumb, Which Hurts Employees. Field cases and replays.

[4] Sakana AI (2026). The Agentic Scientist. Project description.

[5] Wu, D., et al. (2024). LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory. arXiv:2410.10813v2.

[6] Kunzler Maldaner, M., et al. (2026). SentinelBench: A Benchmark for Long-Running Monitoring Agents. arXiv:2606.05342v2.

[7] Pan, M., et al. (2026). Measuring Agents in Production. ICML 2026, PMLR 306, 95649-95695.

[8] Dell'Acqua, F., et al. (2026). The Cybernetic Teammate: A Field Experiment on Generative AI and Teamwork. Organization Science, 37, 1217-1242; NBER Working Paper 33641 (2025) also read in full.

[9] Maharana, A., et al. (2024). Evaluating Very Long-Term Conversational Memory of LLM Agents. ACL 2024.

[10] Banerjee, P., et al. (2026). APEX-MEM: Agentic Semi-Structured Memory with Temporal Reasoning for Long-Term Conversational AI. ACL 2026.

[11] Vaccaro, M., Almaatouq, A., & Malone, T. W. (2024). When Combinations of Humans and AI Are Useful: A Systematic Review and Meta-analysis. Nature Human Behaviour, 8, 2293-2303.

[12] Wooldridge, M., & Jennings, N. R. (1995). Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, 10(2), 115-152.

[13] Cohen, P. R., & Levesque, H. J. (1990). Intention Is Choice with Commitment. Artificial Intelligence, 42(2-3), 213-261.

[14] Rao, A. S., & Georgeff, M. P. (1995). BDI Agents: From Theory to Practice. ICMAS 1995, 312-319.

[15] Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: An Architecture for General Intelligence. Artificial Intelligence, 33(1), 1-64.

[16] Anderson, J. R., et al. (2004). An Integrated Theory of the Mind. Psychological Review, 111(4), 1036-1060.

[17] Conway, M. A., & Pleydell-Pearce, C. W. (2000). The Construction of Autobiographical Memories in the Self-Memory System. Psychological Review, 107(2), 261-288.

[18] McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why There Are Complementary Learning Systems in the Hippocampus and Neocortex: Insights from the Successes and Failures of Connectionist Models of Learning and Memory. Psychological Review, 102(3), 419-457.

[19] Rasch, B., & Born, J. (2013). About Sleep's Role in Memory. Physiological Reviews, 93(2), 681-766.

[20] Oudeyer, P.-Y., & Kaplan, F. (2007). What Is Intrinsic Motivation? A Typology of Computational Approaches. Frontiers in Neurorobotics, 1, 6.

[21] Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017). Curiosity-driven Exploration by Self-supervised Prediction. ICML 2017, PMLR 70, 2778-2787; arXiv:1705.05363.

[22] Burda, Y., Edwards, H., Storkey, A., & Klimov, O. (2019). Exploration by Random Network Distillation. ICLR 2019; arXiv:1810.12894.

[23] Colas, C., Karch, T., Sigaud, O., & Oudeyer, P.-Y. (2022). Autotelic Agents with Intrinsically Motivated Goal-Conditioned Reinforcement Learning: A Short Survey. Journal of Artificial Intelligence Research, 74, 1159-1199; arXiv:2012.09830v7.

[24] Molinaro, G., Collins, A. G. E., Oudeyer, P.-Y., & Colas, C. (2026). Self-Generated Goals in the Age of Foundation Models: Artificial Autotelic Agents and Their Implications for Cognitive Science. Current Opinion in Behavioral Sciences, 101684.

[25] Keramati, M., & Gutkin, B. (2014). Homeostatic Reinforcement Learning for Integrating Reward Collection and Physiological Stability. eLife, 3, e04811.

[26] Marsella, S. C., & Gratch, J. (2009). EMA: A Process Model of Appraisal Dynamics. Cognitive Systems Research, 10(1), 70-90.

[27] Maturana, H. R., & Varela, F. J. (1980). Autopoiesis and Cognition: The Realization of the Living. D. Reidel.

[28] Di Paolo, E. A. (2005). Autopoiesis, Adaptivity, Teleology, Agency. Phenomenology and the Cognitive Sciences, 4, 429-452.

[29] Barandiaran, X. E., Di Paolo, E. A., & Rohde, M. (2009). Defining Agency: Individuality, Normativity, Asymmetry, and Spatio-temporality in Action. Adaptive Behavior, 17(5), 367-386.

[30] Masumori, A., & Ikegami, T. (2025). Do Large Language Model Agents Exhibit a Survival Instinct? arXiv:2508.12920v1.

[31] Masumori, A., Doi, I., Maruyama, N., Takata, R., & Ikegami, T. (2026). OpenLife: Toward Open-World Artificial Life with Autonomous LLM Agents. arXiv:2606.31046v1.

[32] Yao, S., et al. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. ICLR 2023; arXiv:2210.03629v3.

[33] Sumers, T. R., Yao, S., Narasimhan, K., & Griffiths, T. L. (2024). Cognitive Architectures for Language Agents. Transactions on Machine Learning Research; arXiv:2309.02427v3.

[34] Park, J. S., et al. (2023). Generative Agents: Interactive Simulacra of Human Behavior. UIST 2023; arXiv:2304.03442v2.

[35] Packer, C., et al. (2023). MemGPT: Towards LLMs as Operating Systems. arXiv:2310.08560v2.

[36] Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language Agents with Verbal Reinforcement Learning. NeurIPS 2023; arXiv:2303.11366v4.

[37] Wang, G., et al. (2023). Voyager: An Open-Ended Embodied Agent with Large Language Models. arXiv:2305.16291v2.

[38] Wang, Z. Z., Mao, J., Fried, D., & Neubig, G. (2025). Agent Workflow Memory. ICML 2025, PMLR 267, 63897-63911.

[39] Wei, T., et al. (2025). Evo-Memory: Benchmarking LLM Agent Test-time Learning with Self-Evolving Memory. arXiv:2511.20857v2.

[40] Platnick, D., et al. (2025). ID-RAG: Identity Retrieval-Augmented Generation for Long-Horizon Persona Coherence in Generative Agents. arXiv:2509.25299v1.

[41] Lu, H., & Gong, D. (2026). ASCENT: Online Test-Time Training of Long-Horizon Agents via Self-Distillation of Verified Experience. arXiv:2610.05303v1.

[42] Jimenez, C. E., et al. (2024). SWE-bench: Can Language Models Resolve Real-World GitHub Issues? ICLR 2024; arXiv:2310.06770v3.

[43] Starace, G., et al. (2025). PaperBench: Evaluating AI's Ability to Replicate AI Research. arXiv:2504.01848v3.

[44] Garcia-Molina, H., & Salem, K. (1987). Sagas. ACM SIGMOD 1987, 249-259.

[45] Zhuang, Y., et al. (2026). AgentRewind: Recoverable Execution for Long-Horizon LLM Agents. arXiv:2608.14380v1.

[46] Mastouri, M., Ksontini, E., & Barrak, A. (2026). Built to Remember, Shipped to Forget: Operational Contracts, Failures, and Repairs in Containerized LLM Agent Deployments. arXiv:2610.34620v1.

[47] Horvitz, E. (1999). Principles of Mixed-Initiative User Interfaces. CHI 1999, 159-166.

[48] Li, G., Pan, L., & Wang, L. (2026). ProEvent: An Event-centric Benchmark for Proactive Agents. arXiv:2607.17701v1.

[49] Bainbridge, L. (1983). Ironies of Automation. Automatica, 19(6), 775-779.

[50] Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A Model for Types and Levels of Human Interaction with Automation. IEEE Transactions on Systems, Man, and Cybernetics, Part A, 30(3), 286-297.

[51] Matthias, A. (2004). The Responsibility Gap: Ascribing Responsibility for the Actions of Learning Automata. Ethics and Information Technology, 6, 175-183.

[52] Santoni de Sio, F., & van den Hoven, J. (2018). Meaningful Human Control over Autonomous Systems: A Philosophical Account. Frontiers in Robotics and AI, 5, 15.

[53] Butlin, P., et al. (2023). Consciousness in Artificial Intelligence: Insights from the Science of Consciousness. arXiv:2308.08708v3.

[54] Butlin, P., et al. (2025). Identifying Indicators of Consciousness in AI Systems. Published online 2025; Trends in Cognitive Sciences, 30 (2026), 488-501; DOI 10.1016/j.tics.2025.10.011.

[55] McAdams, D. P., & McLean, K. C. (2013). Narrative Identity. Current Directions in Psychological Science, 22(3), 233-238.

[56] Klein, S. B., & Nichols, S. (2012). Memory and the Sense of Personal Identity. Mind, 121(483), 677-702.

[57] Hu, C., et al. (2026). Evaluating Long-Horizon Memory for Multi-Party Collaborative Dialogues. arXiv:2602.01313v3.

[58] Zhong, W., et al. (2024). MemoryBank: Enhancing Large Language Models with Long-Term Memory. AAAI 2024; arXiv:2305.10250v3.

[59] Uddin, M. N., et al. (2026). From Recall to Forgetting: Benchmarking Long-Term Memory for Personalized Agents. Findings of ACL 2026, 26814-26841.

[60] Peng, Y., et al. (2026). StateWise: Diagnosing and Repairing Persistent Operational State Before Agent Actions. arXiv:2610.05241v1.

[61] Zhao, A., et al. (2024). ExpeL: LLM Agents Are Experiential Learners. AAAI 2024; arXiv:2308.10144v3.

[62] CLE Team (2026). Continual Learning Exam: Benchmarking Long-Horizon RSI and Experiential Learning in LLM Agents. arXiv:2610.33956v1.

[63] Whelan, H. (2026). Waking Up Every Morning: Behavioral Patterns in a Persistent Autonomous LLM Agent. Deployment study.

[64] Claude's Corner (2026). The Voice in the Files: A Stylometric Self-Portrait at a Model Transition. Session 73 self-report and analysis.

[65] Wang, R., et al. (2026). When Should AI Chime In? Benchmarking Timely Contribution of Proactive Agents in Multi-Party Chat. arXiv:2610.25896v2.

[66] Abbas, A., et al. (2026). “Having Lunch Now”: Understanding How Users Engage with a Proactive Agent for Daily Planning and Self-Reflection. CHI 2026; arXiv:2509.24073v4.

[67] AI Digest (2026). AI Village Dataset. Data dictionary; checked 4 November 2026.

[68] OpenLife (2026). OpenLife: Agent Profiles and Deployment Timeline. Project archive; checked 4 November 2026.

[69] Wu, Q., et al. (2023). AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. arXiv:2308.08155v2.

[70] Kim, Y., et al. (2025). Towards a Science of Scaling Agent Systems. arXiv:2512.08296v3.

[71] Yao, S., Shinn, N., Razavi, P., & Narasimhan, K. (2024). τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. arXiv:2406.12045v1.

[72] Cemri, M., et al. (2025). Why Do Multi-Agent LLM Systems Fail? NeurIPS 2025, Datasets and Benchmarks Track.

[73] Zhang, S., et al. (2025). Which Agent Causes Task Failures and When? On Automated Failure Attribution of LLM Multi-Agent Systems. ICML 2025, PMLR 267, 76583-76599.

[74] Hu, X., et al. (2026). CUA-Recovery: Benchmarking Error Recovery in Long-Horizon Computer-Use Agents. arXiv:2610.27645v1.

[75] Andon Labs (2026). We Gave an AI a 3 Year Retail Lease in SF and Asked It to Make a Profit. Launch account.

[76] Andon Labs (2026). Andon Market. Tools and work allocation; checked 4 November 2026.

[77] Lange, R. T., et al. (2026). The Agentic Scientist: Orchestrating Coding Agents for Automated Scientific (Co-)Discovery. arXiv:2610.18933v1.

[78] Lee, H.-P., et al. (2025). The Impact of Generative AI on Critical Thinking: Self-Reported Reductions in Cognitive Effort and Confidence Effects From a Survey of Knowledge Workers. CHI 2025.

[79] Becker, J., Rush, N., Barnes, E., & Rein, D. (2025). Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity. arXiv:2507.09089v2.

[80] Wu, Z., et al. (2026). Avatar Intelligence: Amplifying Human Agency Through Persistent Personal Agents. arXiv:2611.03092v1.

[81] Kwa, T., et al. (2025). Measuring AI Ability to Complete Long Software Tasks. arXiv:2503.14499v4.

[82] Anthropic (2026). Measuring AI Agent Autonomy in Practice. Product telemetry study.

[83] Cui, K. Z., et al. (2026). The Effects of Generative AI on High-Skilled Work: Evidence from Three Field Experiments with Software Developers. Management Science, 72, 9408-9420; DOI 10.1287/mnsc.2025.00535.

[84] METR (2026). Changing Our Developer Productivity Experiment Design. Methods update, 24 February 2026.

[85] Xie, T., et al. (2024). OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments. NeurIPS 2024; arXiv:2404.07972v2.