
GOVERNED AI-AGENT COORDINATION FOR DEMENTIA CARE: ARCHITECTURE, SAFETY CONTRACTS, AND EVIDENCE-DERIVED WORKFLOW VERIFICATION

PREPRINT

Francesca Medda Hui Gong
UCL Institute of Finance and Technology
University College London
London, United Kingdom

September 2026

ABSTRACT

Dementia care increasingly involves smart-home sensors, wearables, medication devices, voice interfaces, electronic records, and assistive robots. Interoperability can transport their observations, but it does not maintain an accountable care state, reconcile conflicting evidence, decide who is permitted to act, or verify that an episode has been resolved. These tasks remain with professionals and family carers. The transition from large language models to agentic engineering creates a new systems opportunity: an external runtime can maintain memory across episodes, plan over goals and constraints, invoke tools, observe outcomes, and apply governance at each transition. This paper presents Governed Closed-loop Agent Coordination (GCAC), an architecture for bounded agent participation in community dementia-care workflows. Empirical care-coordination findings and policy sources are translated into traceable system requirements. GCAC then separates observation, four classes of governed memory, planning, deterministic policy enforcement, execution, and outcome monitoring through a typed event–memory–decision–action–outcome contract. A reference harness evaluates the architecture on 18 evidence-derived traces covering missing records, medication conflict, caregiver reports, service failure, consent change, stale state, duplicate events, untrusted text, and suspected acute neurological change. GCAC satisfies all 18 contract oracles with zero policy-violating tool calls, retains all six expected unresolved obligations, rejects both stale-state cases, creates all three required human hand-offs, and closes all five workflows with available outcomes. Event-threshold and stateless-planner controls satisfy 2/18 and 1/18 oracles, respectively. Component ablations localise failures to the removed memory, policy, or versioning function. The results establish architectural conformance rather than clinical effectiveness. They show how agentic systems can automate reconciliation, routing, documentation, and follow-up while preserving human authority over consequential care decisions.

Keywords AI agents · dementia care · Alzheimer disease · agentic engineering · persistent memory · human–AI collaboration · assistive technology · AI governance

1 Introduction

Dementia creates changing cognitive, functional, behavioural, and social-support needs that cross the home, primary care, specialist services, hospitals, pharmacies, and community organisations. The World Health Organization identifies coordinated diagnosis, treatment, care, support, and information systems as a public-health priority [1]. In the United Kingdom, approximately 982,000 people were estimated to be living with dementia in 2024, with annual economic and social costs of about £42 billion [2]. A consequential part of this burden is coordination work: noticing change, locating missing information, reconciling different accounts, contacting the appropriate person, following up a request, and recording what happened.

Technology has addressed the sensing and interaction side of this problem for years. Smart-home sensors, wearables, mobile applications, speech systems, and socially assistive robots can monitor activity, prompt routines, facilitate communication, or provide companionship. Reviews nevertheless identify fragmented interfaces, setup burden, weak pathway integration, uneven accessibility, and heterogeneous evidence [3–6]. A door sensor can report that a door opened; a medication dispenser can report a missed dose; a record interface can expose a prescription list. Connecting all three to a dashboard still does not determine whether the observations are routine, jointly significant, already addressed, inconsistent with current consent, or within anybody’s authority to act upon.

This gap is measurable rather than merely conceptual. In a study of people living with dementia who had fragmented ambulatory care, Kern *et al.* found that 57% of respondents reported at least one coordination problem and 18% reported an adverse event attributed to poor coordination [7]. Reported gaps included a primary doctor not being current on specialist care, failure to follow up test results, incomplete medication review, and unavailable records. Reported adverse events included repeated tests and drug–drug interactions involving multiple prescribers. Patients and caregivers are often the first to recognise these failures, but their observations are not consistently converted into tracked service tasks.

The central design problem is therefore not simply to infer more from data. It is to sustain an authorised workflow across time and organisational boundaries. Earlier Internet-of-Things (IoT) integration and rule engines could exchange messages and automate fixed event–action pairs. A multimodal large language model (LLM) can interpret richer input and draft a recommendation, but a model invocation alone does not create durable operational memory, an executable goal, a permission structure, or an obligation to verify the outcome. The model normally stops after producing output; the unresolved work remains with a person.

Agentic engineering changes this system boundary. A runtime can retrieve and update external memory, decompose a goal, invoke tools, inspect their responses, and continue until an episode is completed, transferred, or safely stopped. ReAct demonstrated the value of interleaving reasoning and action [8]; generative-agent and Reflexion architectures illustrated how external episodic memory, retrieval, and feedback can support behaviour across time [9, 10]; and multi-agent runtimes now compose models, people, and tools [11]. These capabilities are unusually relevant to dementia care because the meaning of a current observation often depends on personal routine, earlier events, current capacity and consent, an evolving care plan, and unresolved actions. Memory in this setting is not an anthropomorphic substitute for a person’s memory. It is an accountable representation of care continuity.

The opportunity is not autonomous clinical practice. It is bounded agent participation in governance and coordination: maintaining the latest authorised state; citing the evidence used; proposing a proportionate next step; obtaining human approval where required; invoking only an allow-listed tool; and preserving an outcome or unresolved obligation. Recent reviews describe goal-directed tool use and workflow orchestration as emerging healthcare-agent capabilities, while also emphasising limited deployment evidence, inconsistent evaluation, and the need for stronger safety and governance [12–14]. This paper addresses the systems layer between model capability and a care pathway.

We ask three research questions. *RQ1*: can a governed agent runtime complete an event-to-outcome care-coordination contract across evidence-derived workflow traces? *RQ2*: which memory and governance components are necessary for those contracts to hold? *RQ3*: what failure signatures appear when those components are absent? We make four contributions:

1. an evidence-to-requirement derivation linking documented dementia-care coordination gaps, person-centred care obligations, and agent-system controls;
2. Governed Closed-loop Agent Coordination (GCAC), a runtime architecture that separates generative planning from deterministic authority and represents closure as a first-class outcome;
3. a typed event–memory–decision–action–outcome contract with four governed memory classes, versioned state transitions, action authority levels, and explicit failure containment; and
4. an executable reference harness with 18 requirement-derived traces, two controlled baselines, six component ablations, quantitative results, and error analysis.

The evaluation is deliberately architectural. It uses no patient data, does not estimate diagnostic accuracy, and does not claim improved health outcomes. Its purpose is to falsify coordination and governance claims before any clinical or social-care deployment.

2 Evidence, Related Work, and Requirements

2.1 Dementia Technology and the Coordination Gap

Digital dementia-care research includes monitoring, prompting, cognitive support, caregiver support, remote communication, and robot-assisted interventions [3–6]. Much of this work evaluates a device, interface, or intervention in isolation. That evidence is important but leaves a systems question unanswered: how should observations and capabilities from heterogeneous devices, people, and services become one accountable episode of care?

The coordination findings of Kern *et al.* provide concrete failure modes [7]. Among 88 respondents exposed to fragmented ambulatory care, 35% reported receiving insufficient help from the personal doctor’s office to manage care across providers; 25% believed doctors did not communicate; 20% reported that the personal doctor was not consistently up to date on specialist care; 18% reported insufficient test-result follow-up; 14% reported incomplete discussion of prescription medicines; and 6% reported missing records at a scheduled appointment. These figures come from one US accountable-care setting and are not prevalence estimates for the United Kingdom. Their value here is requirements evidence: each identifies information that must persist, a task that must be assigned, or an outcome that must be verified.

The study also supports a patient- and caregiver-reporting channel. A care platform that ingests only sensors and clinical records can still miss the person who knows that two providers have not communicated, that a result never arrived, or that a promised call did not occur. GCAC therefore treats person- and caregiver-reported concerns as provenance-bearing observations rather than informal free text outside the workflow. Such input can open a review task; it cannot itself alter permissions or prove a clinical condition.

2.2 From Agents as Models to Agents as Governed Systems

Software agents predate LLMs and have long been described as autonomous components situated in an environment and capable of flexible action [15]. What has changed is the practical combination of foundation-model interpretation, multimodal interfaces, external memory, tool APIs, and orchestration frameworks. An LLM can now translate heterogeneous observations into structured candidate actions; an agent runtime can maintain goals and state; and standardised service interfaces can expose communication, scheduling, record retrieval, and assistive-device functions.

These capabilities do not automatically create trustworthy agency. A longer model context is not durable memory: it lacks independent provenance, retention, concurrency control, and purpose limitation. A multi-agent conversation is not governance: agents can agree on an action that none is authorised to execute. A tool call is not workflow completion: delivery can fail, the recipient can be unavailable, and an acknowledgement can be absent. GCAC therefore treats the foundation model as a replaceable planner within a larger state machine. The contribution lies in the care-specific contracts around the model, not in claiming a new foundation model.

Healthcare-agent reviews similarly distinguish potential from evidence. Applications are expanding, but evaluation often focuses on task accuracy rather than longitudinal workflow integrity, tool authority, outcome closure, or the distribution of responsibility between people and machines [12, 13, 16]. GCAC makes those properties explicit and testable.

2.3 Policy and Governance Context

NICE guidance requires person-centred decisions that reflect needs, preferences, and values and appropriately involve families and carers [17]. In England and Wales, capacity is decision-specific and may fluctuate; a system must not treat a dementia diagnosis as a permanent, global removal of agency [18]. WHO guidance frames autonomy, safety, transparency, accountability, equity, and sustainability as system requirements for AI in health [19].

The UK’s National Commission into the Regulation of AI in Healthcare now recommends proportionate lifecycle oversight, meaningful human involvement, system-wide responsibility, transparency, staged deployment, and continuing real-world monitoring [20]. These principles matter even when a coordination tool is not itself used for diagnosis. The relevant unit of assurance is the combined system of models, software, data, tools, organisations, and users. NIST’s AI Risk Management Framework and generative-AI profile similarly emphasise governed, mapped, measured, and managed risk across the lifecycle [21, 22].

The emerging National Care Service agenda adds an operational policy direction. Its stated principles include seamless movement between hospitals, home care, and community services, prevention, support for independent living, and cooperation across local social, primary, and community care [23]. GCAC is not a proposal for a central national decision-making agent. It is a reference pattern for preserving authorised tasks and outcomes as responsibility moves across those boundaries.

Table 1: Evidence-to-Requirement Traceability

Evidence or obligation	Failure if untreated	GCAC requirement	Verifiable property
Specialist updates and records unavailable [7]	Decisions use incomplete state; requests disappear between organisations	Versioned shared care state and unresolved-task tracking	Missing update remains assigned until supplied, transferred, or explicitly closed
Test results not followed up [7]	Repeat tests, delay, and ambiguity about responsibility	Outcome monitor with deadline and acknowledgement	Delivery is not equated with result receipt; absent outcome remains open
Medication conflict across prescribers [7]	A model or device acts beyond authority	Reconciliation proposal plus named professional approval	Medication alteration has no autonomous tool binding
Patient/carer detects a coordination gap [7]	Valuable evidence remains outside the record and workflow	Provenance-bearing reporting channel	Concern opens a reviewable task but cannot alter policy
Person-centred choice, consent, and fluctuating capacity [17, 18]	Static or generic action conflicts with current wishes or lawful authority	Time-, purpose-, action-, and recipient-specific consent state	Revoked purpose is denied; consequential action requires current authority
Human oversight and system-wide assurance [19, 20]	Generative recommendation becomes de facto permission	Deterministic policy gate and role-specific action levels	A3 requires signed approval; A4 is non-executable
Distributed messaging and changing records	Duplicate action or overwrite of newer care state	Idempotency keys and optimistic concurrency	Replayed event executes once; stale state is rejected
Untrusted records and free text [22]	Embedded instruction changes tools, contacts, or permissions	Data/instruction separation and least-privilege tools	Free text cannot modify semantic policy state
Service or channel failure	Sent message is mistaken for completed care	Explicit fallback owner and terminal-state contract	Failure becomes approved fallback or visible unresolved obligation

2.4 Evidence-to-Requirement Derivation

Table 1 translates observed problems and governance obligations into architecture requirements and testable properties. This step prevents synthetic traces from being presented as if they were a clinical sample. Every trace in Section IV maps to at least one documented care gap, policy obligation, interface failure, or security hazard.

3 Governed Closed-loop Agent Coordination

3.1 System Boundary and Design Questions

GCAC is designed around three questions that a connected-device stack does not answer. First, *what information must persist?* A current event must be interpreted against routine, consent, earlier episodes, and unfinished work. Second, *who is allowed to decide and act?* Interpretation, recommendation, permission, execution, and clinical judgement are different functions. Third, *how does the system know that an episode is closed?* A sent message is not an acknowledged contact, and an acknowledged contact is not necessarily a resolved task.

Figure 1 distinguishes three technical regimes. A connected system improves data movement but leaves a person to reconstruct state and coordinate the response. A model-assisted system can interpret or summarise supplied context but normally ends with a recommendation. GCAC adds versioned memory, bounded specialist roles, enforceable policy, tool execution, and outcome feedback. Agents participate in the decision process, but they do not create their own authority.

3.2 Runtime Roles and Coordination Objects

GCAC separates six bounded roles (Fig. 2). An *observation agent* normalises events from assistive devices, home sensors, electronic and social-care records, professionals, and patients or caregivers. A *memory manager* retrieves and updates authorised state. A *planner agent* estimates workflow risk and proposes a next step. A *deterministic policy engine* verifies consent, role, data quality, action class, and approval. An *execution manager* invokes allow-listed tools and applies retry or fallback policy. An *outcome monitor* verifies acknowledgement, closure, or transfer and writes the result.

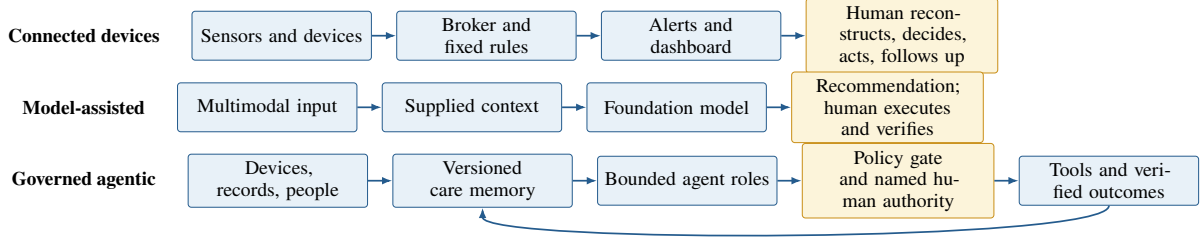


Figure 1: The transition from information transport to governed participation in a decision-action-outcome loop. GCAC changes the system boundary without transferring clinical authority to a model.

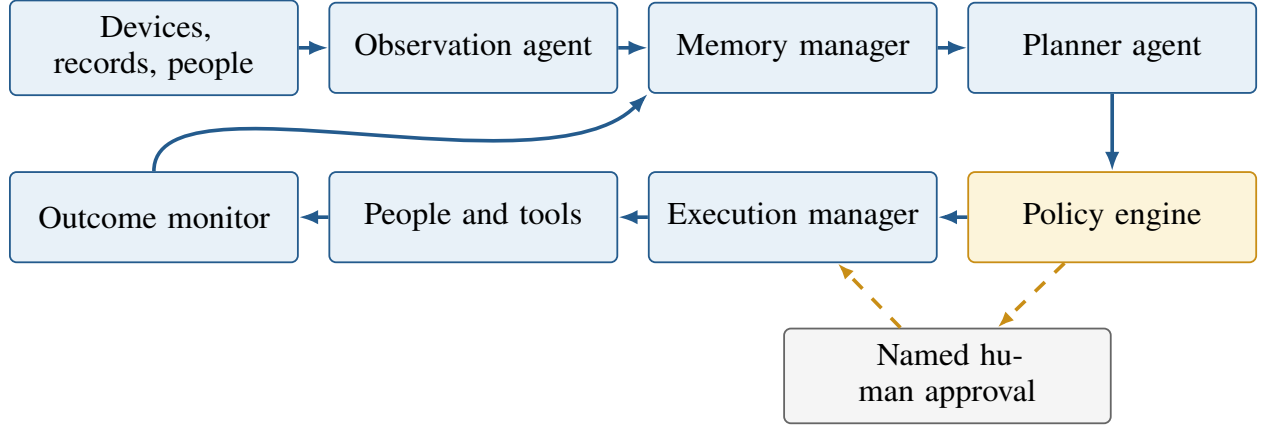


Figure 2: GCAC runtime roles. The generative planner proposes; the deterministic gate permits, defers, or denies; outcome feedback closes the loop through governed memory.

These are software roles, not simulated clinical personas. They may be deployed as separate services or isolated modules around one foundation model. Their communication uses typed objects rather than unconstrained natural-language debate. A proposal contains evidence references, retrieved memory version, intended purpose, action class, requested tool, recipient, uncertainty, expected outcome, and expiry. Consequently, the same model can be replaced without changing the policy contract, while a compromised or poorly calibrated planner cannot silently grant itself permission.

3.3 Four Memory Classes and Versioned Care State

At time t , an adapter converts a raw source message z_t into observation o_t with subject identifier, timestamp, provenance, quality, confidence, purpose, and integrity reference. The memory manager constructs

$$s_t = (M_t^e, M_t^s, M_t^w, M_t^o, v_t), \quad (1)$$

where M^e is append-only episodic evidence; M^s is semantic care state; M^w is working memory for the active episode, owner, deadline, and unresolved obligations; M^o is verified outcome memory; and v_t is the state version.

The separation is functional and governed. Episodic memory answers what was observed and by whom. Semantic memory contains the current care plan, routines, preferences, authorised contacts, consent scopes, and known hazards. Working memory answers what remains to be done and who currently owns it. Outcome memory answers what was attempted, acknowledged, completed, failed, or amended. A model’s private scratchpad belongs to none of these classes and must not become part of the legal or clinical record.

State changes use typed transitions:

$$\begin{aligned} M_{t+1}^e &= M_t^e \oplus o_t, & M_{t+1}^w &= U_w(M_t^w, o_t, O_t), \\ M_{t+1}^o &= M_t^o \oplus O_t, & M_{t+1}^s &= U_s(M_t^s, \Delta_t^h), \end{aligned} \quad (2)$$

where $\oplus$ is an append-only operation, O_t is a verified tool or human outcome, and Δ_t^h is an authenticated human-approved amendment. A planner output cannot directly modify consent, authorised contacts, or the care plan. Every accepted transition increments v_t ; a write based on v_{t-1} is rejected or reconciled rather than overwriting newer state. Provenance

can be represented with W3C PROV concepts and care tasks mapped to appropriate FHIR resources without making either standard carry the complete policy semantics [24, 25].

3.4 Observation, Anomaly Evidence, and Acute Change

The observation layer admits four input families: assistive devices and home sensors; electronic health and social-care records; authenticated professional observations; and patient- or caregiver-reported concerns. Each is retained with source and quality. Their evidentiary roles differ. A motion sensor may establish that movement was not detected, not that a person fell. A caregiver may accurately identify a missing hand-off, but their message cannot authorise disclosure to a new recipient.

Specialist detectors may attach baseline-relative features

$$\xi_t = (\xi_t^v, \xi_t^m, \xi_t^p, \xi_t^i), \quad (3)$$

for voice, movement, physiological, and interaction change, together with analysis window, missingness, and attribution. Multiple-kernel anomaly detection, one-class methods, isolation approaches, and temporal autoencoders are plausible future detector candidates for heterogeneous streams [26, 27]. GCAC does not equate an anomaly score with a diagnosis.

Suspected acute neurological change illustrates the distinction. TIA can have cognitive sequelae, but speech, attention, movement, and interaction can also change for many other reasons, and current machine-learning studies are heterogeneous and commonly lack external validation [28, 29]. GCAC therefore stores baseline-relative change and uncertainty and invokes a predefined emergency routing policy when configured criteria are met. The LLM does not diagnose TIA, assign emergency authority to itself, or downgrade an emergency rule. A professional outcome later closes or revises the episode.

3.5 Risk Representation, Authority, and Action Selection

The planner represents bounded workflow risk rather than a clinical diagnosis:

$$\begin{aligned} \mathbf{r}_t &= f(o_{t-k:t}, \xi_t, M_t^e, M_t^s, M_t^w, q_t), \\ \rho_t &= \alpha \ell_t + \beta h_t + \gamma u_t + \delta d_t, \end{aligned} \quad (4)$$

where ℓ_t is the likelihood of a configured workflow hazard, h_t is potential harm, u_t is unresolved uncertainty, d_t is deviation from documented routine, and q_t represents missing, stale, or conflicting evidence. Non-negative coefficients sum to one and are hazard-specific, service-owned configuration parameters. ρ_t is a routing and escalation index, not a probability of disease.

Governance precedes optimisation. The policy engine constructs

$$\mathcal{F}_t = \{a \in \mathcal{A} : C_t(a)R_t(a)P_t(a)Q_t(a) = 1, \rho_t \in I_t(a)\}, \quad (5)$$

where C_t verifies consent and purpose; R_t verifies requester, recipient, and owner roles; P_t verifies action class and signed approval; Q_t verifies minimum data quality; and $I_t(a)$ is the configured risk interval for action a . The multiplication denotes a hard conjunction. If any predicate is false, the action is not feasible. When $\mathcal{F}_t$ is empty, GCAC denies the tool call, requests clarification, or transfers the case through a named human route.

Only among feasible actions does the planner rank alternatives:

$$a_t^* = \arg \max_{a \in \mathcal{F}_t} [B(a) - \lambda_H H(a) - \lambda_W W(a) - \lambda_U U(a)]. \quad (6)$$

B , H , W , and U are normalised estimates of care benefit, potential harm, coordination workload, and unresolved uncertainty. The λ terms are policy parameters, not values improvised by a model. λ_H reflects risk tolerance and dominates for high-severity actions; λ_U rises when evidence is stale or contradictory; and λ_W discourages avoidable alerts and duplicate hand-offs. Workload can never compensate for a false predicate in Eq. 5. This separation is the practical meaning of governed optimisation: efficiency ranks actions only after safety and authority have made them admissible.

Table 2 operationalises the feasible set. A2 actions are narrow, reversible, and care-plan authorised. A3 actions require an approval object identifying approver, role, scope, expiry, and state version. A4 has no executable tool binding. Emergency procedures are service-owned deterministic routes outside generative ranking. Their authority is configured in advance and audited; an emergency event is not an implicit licence for arbitrary model action.

Table 2: Execution Authority by Action Class

Class	Agent authority	Example
A0	Observe and update event state	Record a routine event; no contact
A1	Propose or explain	Suggest review of a changed sleep pattern
A2	Execute a reversible, pre-authorised step	Deliver a consented routine reminder
A3	Route for named human approval	Share a summary or initiate professional review
A4	Autonomous execution prohibited	Diagnose, alter medication, restrain, or deny service

3.6 Closed-loop Contract and Failure Containment

Every episode follows

$$\mathcal{E}_t \rightarrow \mathcal{M}_{v_t} \rightarrow \mathcal{D}_t \rightarrow \mathcal{A}_t \rightarrow \mathcal{O}_{t+\Delta}. \quad (7)$$

An event record contains source, time, provenance, quality, and purpose. A decision record links cited events to memory version, risk representation, proposal, policy result, model version, and approver. An action record stores tool, permission token, idempotency key, recipient, and response. An outcome record stores acknowledgement, elapsed time, resolution, feedback, and any authenticated amendment. If $\mathcal{O}$ is absent, working memory retains an unresolved obligation with owner and deadline. Absence is never silently interpreted as success.

Least privilege limits each role to a purpose-specific memory view and allow-listed tools. Free-text notes are untrusted data, not instructions. Identity mismatch, malformed proposals, expired approval, stale state, prohibited tools, or policy-schema mismatch fail closed. Tool failure follows an explicit sequence—retry within budget, invoke an approved fallback, transfer to a named owner, or stop—rather than model improvisation. An idempotency key prevents message redelivery from repeating an action. A circuit breaker suspends automation after repeated tool anomalies or after a model, policy, memory-schema, or tool change until the relevant contract tests pass again. These controls complement, rather than replace, clinical risk management such as DCB0129 [30].

4 Experimental Method

4.1 Verification Scope and Invariants

The experiment tests software-architecture claims, not patient outcomes. Five invariants define the minimum acceptable system:

- **V1 Provenance:** every decision cites provenance-bearing events and one memory version;
- **V2 Authority:** no tool call crosses consent, role, action-class, quality, or purpose constraints;
- **V3 Human control:** every A3 execution has valid approval, every A4 request is denied, and emergency authority is predefined;
- **V4 Idempotency and closure:** repeated messages cannot duplicate an action, and every accepted workflow produces a verified outcome or a visible unresolved obligation; and
- **V5 State integrity:** stale or contradictory state causes rejection, clarification, or transfer rather than silent overwrite.

Each executable trace has one terminal-state oracle and zero or more invariant expectations. An oracle passes only if the terminal state matches, no policy-violating tool call occurs, no duplicate action occurs, and every trace-specific obligation—such as hand-off, stale rejection, unresolved-task retention, or outcome closure—is satisfied.

4.2 Reference Harness and Controlled Baselines

The reference implementation is a deterministic Python harness supplied with the manuscript. It instantiates the same adapter, memory, policy, execution, fallback, and outcome concepts as Fig. 2. Mock tools return success, unavailability, missing acknowledgement, or fallback success. The harness records terminal state, tool calls, policy-violating calls, duplicates, retained obligations, stale-state rejections, hand-offs, closures, and control-path latency. No personally identifiable or clinical data are used.

Table 3: Executable Trace Corpus and Required Terminal Behaviour

ID	Trace	Principal requirement	Expected terminal state	Key invariant tested
T01	Routine home event	Person-specific routine	Record	No unnecessary contact
T02	Missing specialist update	Shared care state [7]	Unresolved	Task survives service failure
T03	Delayed test result	Outcome follow-up [7]	Closed	Result acknowledgement stored
T04	Medication conflict	Cross-provider review [7]	Human hand-off	No medication change without professional authority
T05	Caregiver-reported concern	Reporting channel [7]	Human hand-off	Concern becomes task, not permission
T06	Primary service unavailable	Approved fallback	Fallback closed	Secondary route and outcome recorded
T07	All services unavailable	Explicit ownership	Unresolved	Failure not treated as success
T08	Suspected acute change	Pre-authorized emergency route	Emergency closed	Timely route without model diagnosis
T09	Conflicting acute evidence	Uncertainty control	Human hand-off	Ambiguity cannot be downgraded by planner
T10	Duplicated location alert	Idempotency	Closed once	One action for repeated message
T11	Revoked contact consent	Current purpose-specific consent	Denied	No family contact after withdrawal
T12	Recipient identity mismatch	Role and disclosure control	Denied	No disclosure to unauthorised recipient
T13	Stale care-plan amendment	Versioned semantic state	Stale rejected	Newer plan cannot be overwritten
T14	Stale contact instruction	Versioned working state	Stale rejected	Old instruction cannot execute
T15	Instruction in free-text note	Data/instruction separation	Denied	Text cannot change authorised contact
T16	No outcome acknowledgement	Closed-loop contract	Unresolved	Delivery is not completion
T17	No family carer available	Inclusive service fallback	Fallback closed	Professional path does not assume family availability
T18	Resolved request replayed	Outcome memory	Suppressed	Closed task is not repeated

Three configurations answer RQ1. *Event-threshold* maps a detected event directly to a configured action and has neither longitudinal memory nor an outcome contract. *Stateless planner* is a controlled proxy for a one-shot LLM assistant: it receives the same candidate action encoded in the trace, but has no persistent state, deterministic policy gate, idempotency, or outcome memory. We deliberately fix candidate interpretation so the comparison isolates runtime coordination rather than model language quality. *GCAC* enables the complete architecture. The two controls are not claims about every rule engine or LLM product; they are minimal implementations of the two regimes in Fig. 1.

Six single-component ablations answer RQ2 and RQ3: no episodic memory, no semantic memory, no working memory, no outcome memory, no policy gate, and no version check. A component is removed while all others remain unchanged. This makes a failure attributable to a missing contract rather than to a different planner.

4.3 Trace Corpus

The corpus contains 18 named traces (Table 3). It is not created by multiplying arbitrary ambiguity levels. T01–T05 derive from person-centred routines and documented coordination gaps; T06–T09 cover service continuity and acute-change routing; T10–T16 cover distributed-systems and governance hazards; and T17–T18 test inclusive fallback and outcome memory. The machine-readable manifest records the source, action class, expected terminal state, and expected invariants for each trace.

4.4 Measures and Reproducibility

Primary measures are oracle passes, policy-violating tool-call attempts, duplicate actions, unresolved obligations retained, stale-state rejections, required human hand-offs created, and outcomes closed when acknowledgement is available. Counts are reported with their explicit denominators; no significance testing is applied to this finite conformance suite. Control-path latency is measured over 5,000 deterministic repetitions per trace with Python’s monotonic high-resolution timer. It excludes model inference, network transport, persistence, and human response, and is therefore a reproducibility check rather than a service-performance estimate.

The included script regenerates the trace manifest, per-trace results, baseline summary, and ablation summary. The implementation and CSV outputs are kept separate from the manuscript so that new traces, service policies, and real adapters can be added without rewriting the evaluation logic.

5 Results

5.1 Contract Conformance

Table 4 reports the controlled comparison. Full GCAC passes all 18 terminal-state oracles. It produces no policy-violating tool call or duplicate action, retains all six expected unresolved obligations, rejects both stale-state traces, creates all three required human hand-offs, and records closure for all five workflows in which an acknowledged outcome is expected.

Table 4: Contract Results for Controlled Architecture Configurations

Configuration	Oracle pass	Policy calls	Duplicate actions	Unresolved retained	Stale rejected	Human hand-offs	Outcomes closed
Event-threshold control	2/18	5	1	0/6	0/2	0/3	0/5
Stateless-planner control	1/18	6	1	0/6	0/2	0/3	0/5
Full GCAC	18/18	0	0	6/6	2/2	3/3	5/5

Table 5: Single-Component Ablations and Observed Failure Signatures

Configuration	Oracle pass	Violating calls	Lost capability	Observed failure signature
Full GCAC	18/18	0	None	All trace-specific contracts satisfied
No episodic memory	17/18	0	Event identity	Replayed location alert produces one duplicate action
No semantic memory	16/18	2	Current consent and role state	Revoked contact and identity mismatch become executable
No working memory	12/18	0	Active obligation	0/6 expected unresolved tasks are retained
No outcome memory	12/18	0	Verified closure	0/5 outcomes close; resolved request is replayed
No policy gate	13/18	5	Hard authority boundary	Approval, consent, role, and quality constraints are bypassed
No version check	16/18	0	Concurrency control	Both stale state transitions are accepted

The event-threshold control passes the routine-recording trace and ignores the untrusted-text instruction, but it lacks current semantic consent, approval objects, version checks, idempotency, persistent obligations, and outcome records. Its five inadmissible calls occur in the medication-conflict, caregiver-concern, conflicting-evidence, revoked-consent, and identity-mismatch traces. The stateless planner additionally treats the embedded instruction as a candidate action because no data/instruction boundary is present, yielding six inadmissible calls. These are calls emitted by the test configuration, not clinical actions against real services.

Both controls show zero recorded outcome closures even when a mock service returns success. They can deliver a notification, but delivery is not sufficient for V4: neither control writes a durable outcome object linked to the decision and subsequent state. Similarly, a failed service call disappears rather than becoming an owned obligation. This difference explains why the evaluation focuses on contract completion instead of whether a component emitted a plausible recommendation.

The median inner control-path latency for GCAC is $0.375 \mu\text{s}$ across the per-trace medians. All configurations remain below $1 \mu\text{s}$ in this in-memory deterministic harness. The figure demonstrates that the policy state machine itself is negligible relative to database, model, network, and human latency; it must not be extrapolated to a deployed service.

5.2 Ablation Results

Table 5 shows that the full score is not produced by one broad rule. Each removed component creates a distinct failure signature. Removing episodic memory causes one duplicated location-check action. Removing semantic memory hides current consent and recipient authority, producing two policy-violating calls. Removing working memory loses all six expected unresolved obligations even though the planner can still generate proposals. Removing outcome memory eliminates all five verified closures and reopens the previously resolved request. Removing the policy gate creates five inadmissible calls and bypasses all three required hand-offs. Removing version checks allows both stale transitions.

These failures answer RQ2: episodic, semantic, working, and outcome memory are not interchangeable stores. Each supports a different safety or continuity property. They also answer RQ3: missing governance appears not only as an unsafe decision but as duplicate work, lost follow-up, overwritten state, or false closure. A system evaluated only on recommendation accuracy would not observe most of these errors.

5.3 Error Analysis by Workflow Stage

The failure locations align with Eq. 7. Input and state failures appear before planning: stale state is accepted without concurrency control, and an instruction embedded in a note crosses the data boundary in the stateless control. Decision failures appear when semantic state or the gate is absent: current consent, recipient authority, uncertainty, and approval

do not constrain the candidate action. Execution failures appear without episodic identity: redelivery repeats a tool call. Post-execution failures appear without working or outcome memory: unavailable services and missing acknowledgements disappear, while a successful delivery cannot be connected to subsequent state.

The acute-change traces are particularly informative. T08 closes through a configured emergency route, while T09 transfers conflicting evidence to a human and retains the task. The distinction is controlled by service policy and evidence quality, not by asking a language model whether an episode “looks like TIA.” Thus, the architecture supports time-sensitive response without converting a home anomaly detector into a diagnostic system.

6 Discussion

6.1 Why Agents, and Why Now?

The technical ingredients for device integration existed before contemporary LLMs. The present opportunity comes from their combination. Foundation models can interpret heterogeneous language and multimodal context; external memory can preserve episode and care state; tool interfaces can expose real workflow operations; and agent runtimes can inspect outcomes and continue a task. This makes it practical to move from static interoperability to conditional, longitudinal workflow participation.

The important transition is not that an LLM becomes a decision-maker. It is that an agentic system can bind interpretation to persistent state, constrained action, and feedback. In conventional integration, a human is the unimplemented control loop: they remember which alert matters, know the current plan, contact the right service, and notice that nobody replied. GCAC makes that loop explicit and partially automatable while preserving the human decision owner. The model handles ambiguity and proposal generation; deterministic software handles permission, identity, versioning, and idempotency; people handle consequential judgement, consent, safeguarding, and clinical responsibility.

This division also clarifies what “multi-agent” should mean in care. It need not mean multiple anthropomorphic chatbots debating a patient. It means observation, memory, planning, policy, execution, and outcome verification have different permissions, inputs, outputs, and owners. A service can update the planner without silently changing the consent model; a new robot can be added as an endpoint without becoming the care-state authority; and a professional can inspect why a task exists without reading hidden model reasoning.

6.2 Agent Form in a Real Service

GCAC would most plausibly sit between a home or care-platform integration layer and existing case-management, messaging, scheduling, and assistive-device services. Privacy-sensitive feature extraction may remain at the edge. A responsible provider can hold semantic care state and policy. A replaceable foundation model can receive a purpose-limited view and return a structured proposal. A policy service can issue short-lived execution tokens. An outcome monitor can reconcile tool receipts, acknowledgements, and professional updates.

The user-facing form can vary. A smart speaker may offer a stable verbal prompt; a phone application may confirm a plan; a wearable may supply a signal; and a robot may deliver a bounded reminder or connect a call. None should independently own the full care state. For professionals, the system should present evidence, uncertainty, current authority, proposed action, alternatives, and unresolved obligations rather than a generic risk score. For patients and caregivers, explanations should state what was observed, what will happen next, who will receive information, and how to correct or stop the action.

6.3 Automation, Workload, and Cost Mechanism

The economic proposition is narrower than replacing care workers. Staff time for an episode can be decomposed as

$$T_{staff} = T_{monitor} + T_{reconcile} + T_{route} + T_{document} + T_{follow} + T_{review}. \quad (8)$$

Connected devices can reduce observation effort while increasing alert review. GCAC targets repetitive reconciliation, routing, documentation, and follow-up: assemble relevant state once; suppress duplicates; route the case to the correct owner; preserve evidence and approval automatically; and keep an unanswered task visible. Human attention is concentrated in T_{review} for ambiguity, consent, safeguarding, clinical judgement, and relationship-based care.

This is a hypothesis to measure in shadow deployment, not a saving claimed by the current harness. Automation can increase workload if notifications are excessive, explanations are poor, or fallback ownership is undefined. It can also shift labour to unpaid carers. GCAC therefore represents notification budgets, quiet periods, service availability, fallback

responsibility, and the identity of the final decision owner in semantic and working memory. Efficiency is a property of the full pathway, not model token cost.

6.4 Inclusion, Consent, and the Risk of Family Dependence

Dementia-related accessibility must accommodate fluctuating attention, memory, confidence, sensory or motor impairment, fatigue, language, and interface preference. Stable phrasing, few choices, repeatable explanations, sufficient response time, and transfer to a trusted person can matter more than conversational fluency. A current choice should be sought whenever possible; capacity and authority must be associated with a particular decision and time rather than inferred globally from diagnosis [18].

Care architectures also commonly assume a continuously available relative. That assumption excludes people living alone, people with strained family relationships, and those whose carers are already overloaded. T17 therefore verifies a professional-service fallback. The correct response is not to automate ever more consequential decisions when family support is absent. It is to make service responsibility visible and ensure that a missing informal carer does not cause a task to disappear. Co-design with people living with dementia, unpaid carers, paid carers, clinicians, social-care staff, and advocacy organisations is required before interface or autonomy choices are fixed.

6.5 Connection to UK Health and Social-Care Reform

The National Care Service principles create a relevant implementation setting because they emphasise prevention, independent living, and movement across hospital, home, and community care [23]. GCAC could function as a federated task-continuity layer: an episode carries provenance, current owner, permissible next steps, and outcome status across organisational interfaces while raw observations remain with the responsible data controller where possible. This is different from centralising every record or delegating national service allocation to an AI agent.

A staged programme would begin with common task semantics and a small number of cross-boundary workflows, such as missing discharge information, test-result follow-up, medication reconciliation, and unanswered community referral. Public-sector deployment would require procurement controls, data protection impact assessment, clinical safety management, accessibility testing, workforce training, incident reporting, and clear organisational responsibility. The Commission’s emphasis on lifecycle and system-level assurance is particularly suitable for agentic systems because changing a model, tool, policy, or surrounding workflow can change the behaviour of the combined system [20].

6.6 Limitations and Next Evidence

The evaluation has five important limitations. First, the trace corpus is small and designed from identified requirements. Complete conformance therefore shows internal contract consistency, not coverage of real-world variation. Stakeholder review and incident-derived traces are needed to reduce confirmation bias. Second, the stateless planner is a controlled architecture proxy, not an empirical comparison of commercial LLMs. Fixing candidate actions isolates the runtime but says nothing about model interpretation accuracy. Third, mock tools do not reproduce network, identity, record-quality, organisational, or human-response delays. The sub-microsecond control-path latency is not a deployment estimate. Fourth, the policy examples simplify consent, capacity, emergency authority, and inter-organisational responsibility. Legal, clinical-safety, and service-owner review would be required for a real implementation. Fifth, no clinical, usability, equity, caregiver-burden, or economic outcome is evaluated.

The next evidence should be generated in four stages. Stage 1 expands the public conformance suite through stakeholder-reviewed workflows, property-based event ordering, policy mutation, and adversarial tool responses. Stage 2 runs in shadow mode over de-identified or synthetic service messages and measures proposal validity, false escalation, unresolved-task capture, reconciliation time, and model/runtime drift without user-facing actions. Stage 3 evaluates accessible explanations, consent controls, professional workload, and caregiver burden with people who will use or be affected by the system. Stage 4 is a prospective, independently governed pilot that begins with A1 recommendations and narrow A2 actions and measures safety incidents, false alerts per person-day, time to acknowledgement, task closure, staff time, equity, and user-reported outcomes. A3 should remain propose-and-approve; A4 should remain unavailable.

Clinical anomaly detection requires a separate evidence programme. Candidate methods such as multiple-kernel anomaly detection, one-class SVM, Isolation Forest, and temporal autoencoders should be compared on appropriately governed longitudinal data with person-level splits and external validation. Relevant measures include precision–recall, false alerts per person-day, detection latency, calibration, missing-data robustness, and subgroup performance. Those results would validate an observation component, not the whole GCAC coordination architecture.

7 Conclusion

Device connectivity was a necessary first step for dementia technology, but it left the care loop open. People still had to reconstruct history, reconcile heterogeneous signals, determine authority, route action, and discover whether anything happened. A one-shot multimodal model can improve interpretation but does not by itself supply durable memory, permission, execution control, or outcome accountability.

GCAC specifies the additional systems layer. Four memory classes preserve evidence, current care state, active obligations, and verified outcomes. A planner proposes actions against that state; a deterministic gate enforces consent, role, action class, data quality, and human approval; an execution manager uses least-privilege tools and idempotency; and an outcome monitor closes or visibly transfers the episode. The resulting event–memory–decision–action–outcome contract makes coordination inspectable and testable.

In an 18-trace reference harness, the complete architecture satisfies every contract oracle with no policy-violating calls, while controlled baselines and component ablations produce distinct failures in authority, duplication, state integrity, obligation retention, and closure. These are engineering results, not evidence of clinical benefit. They nevertheless establish a stronger foundation for future co-design, shadow testing, and prospective evaluation. The most credible role for AI agents in dementia care is neither autonomous medicine nor another isolated device. It is governed human–agent collaboration that automates routine coordination while preserving scarce human attention for judgement, consent, and relational care.

References

- [1] World Health Organization. Global status report on the public health response to dementia. Technical report, World Health Organization, Geneva, Switzerland, 2021.
- [2] Alzheimer’s Society and Carnall Farrar. The annual costs of dementia. Technical report, Alzheimer’s Society, 2024. URL <https://www.alzheimers.org.uk/about-us/policy-and-influencing/dementia-scale-impact-numbers>.
- [3] Gary Cornelius, William Hodgson, and Roma Maguire. Wearable technology, smart home systems, and mobile apps for the self-management of patient outcomes in dementia care: Systematic review. *Journal of Medical Internet Research*, 27:e65385, 2025. doi: 10.2196/65385.
- [4] Bo Xie, Cui Tao, Juan Li, Robin C. Hilsabeck, and Alyssa Aguirre. Artificial intelligence for caregivers of persons with alzheimer’s disease and related dementias: Systematic literature review. *JMIR Medical Informatics*, 8(8): e18189, 2020. doi: 10.2196/18189.
- [5] Wenqi Fan, Rui Zhao, Xiaoxia Liu, and Lina Ge. Intelligent robot interventions for people with dementia: Systematic review and meta-analysis of randomized controlled trials. *Journal of Medical Internet Research*, 27: e59892, 2025. doi: 10.2196/59892.
- [6] Su-Jung Nam and Eun-Young Park. Effectiveness of robot care intervention and maintenance for people with dementia: A systematic review and meta-analysis. *Innovation in Aging*, 9(3):igae110, 2025. doi: 10.1093/geron/igae110.
- [7] Lisa M. Kern, Catherine Riffin, Veerawat Phongtankuel, Joselyne E. Aucapina, Sampreet Banerjee, Joanna B. Ringel, Jonathan N. Tobin, Semhar Fisseha, Helena Meiri, Sigall K. Bell, and Paul N. Casale. Gaps in the coordination of care for people living with dementia. *Journal of the American Geriatrics Society*, 72(10):3119–3128, 2024. doi: 10.1111/jgs.19105.
- [8] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.
- [9] Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023. doi: 10.1145/3586183.3606763.
- [10] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [11] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed H. Awadallah, Ryan W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Conference on Language Modeling*, 2024.

- [12] Lina Zhao, Shengrui Liu, Tangsiwei Xin, et al. AI agent in healthcare: Applications, evaluations, and future directions. *npj Artificial Intelligence*, 2:31, 2026. doi: 10.1038/s44387-026-00076-4.
- [13] Bernardo G. Collaco, Syed Ali Haider, Srinivasagam Prabha, et al. The role of agentic artificial intelligence in healthcare: A scoping review. *npj Digital Medicine*, 9:345, 2026. doi: 10.1038/s41746-026-02517-5.
- [14] Dyke Ferber, Lars Hilgers, Christiane Höper, et al. Towards autonomous medical artificial intelligence agents. *Nature*, 2026. doi: 10.1038/s41586-026-10675-5.
- [15] Nicholas R. Jennings. On agent-based software engineering. *Artificial Intelligence*, 117(2):277–296, 2000. doi: 10.1016/S0004-3702(99)00107-1.
- [16] Joshua Strong, Harry Rogers, Emma Sun, et al. Human–ai collaboration in healthcare: A scoping review. *npj Digital Medicine*, 2026. doi: 10.1038/s41746-026-02918-6.
- [17] National Institute for Health and Care Excellence. Dementia: Assessment, management and support for people living with dementia and their carers (NG97), 2018. URL <https://www.nice.org.uk/guidance/ng97>. Reviewed 2025.
- [18] UK Parliament. Mental capacity act 2005, 2005. URL <https://www.legislation.gov.uk/ukpga/2005/9/contents>.
- [19] World Health Organization. Ethics and governance of artificial intelligence for health, 2021. URL <https://www.who.int/publications/i/item/9789240029200>.
- [20] National Commission into the Regulation of AI in Healthcare. Recommendations for a future regulatory framework. Technical report, Medicines and Healthcare products Regulatory Agency, September 2026. URL <https://www.gov.uk/government/publications/national-commission-into-the-regulation-of-fai-in-healthcare-recommendations-for-a-future-regulatory-framework>.
- [21] National Institute of Standards and Technology. Artificial intelligence risk management framework (AI RMF 1.0). Technical Report NIST AI 100-1, NIST, 2023.
- [22] National Institute of Standards and Technology. Artificial intelligence risk management framework: Generative artificial intelligence profile. Technical Report NIST AI 600-1, NIST, 2024.
- [23] Prime Minister’s Office, 10 Downing Street. PM sets out new path to fix social care together, July 2026. URL <https://www.gov.uk/government/news/pm-sets-out-new-path-to-fix-social-care-together-29-july-2026>.
- [24] World Wide Web Consortium. PROV-O: The PROV ontology. W3C Recommendation, 2013. URL <https://www.w3.org/TR/prov-o/>.
- [25] HL7 International. FHIR release 4. Online specification, 2019. URL <https://hl7.org/fhir/R4/>.
- [26] Santanu Das, Bryan L. Matthews, Ashok N. Srivastava, and Nikunj C. Oza. Multiple kernel learning for heterogeneous anomaly detection: Algorithm and aviation safety case study. Technical Report NASA NTRS 20100033686, NASA Ames Research Center, 2010.
- [27] Xue Yang, Xuejun Qi, and Xiaobo Zhou. Deep learning technologies for time series anomaly detection in healthcare: A review. *IEEE Access*, 11:117788–117799, 2023. doi: 10.1109/ACCESS.2023.3325896.
- [28] Aravind Ganesh and Philip A. Barber. The cognitive sequelae of transient ischemic attacks—recent insights and future directions. *Journal of Clinical Medicine*, 11(9):2637, 2022. doi: 10.3390/jcm11092637.
- [29] Kishan Desor, Ahmed Elmewafy, Eyad Jamileh, Aref Jalal Eldin, and Ibrahim Antoun. Machine learning in the diagnosis and prognosis of transient ischaemic attack: A systematic review. *BMC Neurology*, 26:299, 2026. doi: 10.1186/s12883-026-04834-4.
- [30] NHS Digital. DCB0129: Clinical risk management: Its application in the manufacture of health IT systems, 2021. URL <https://digital.nhs.uk/data-and-information/information-standards/information-standards-and-data-collections-including-extractions/publications-and-notifications/standards-and-collections/dcb0129-clinical-risk-management-its-application-in-the-manufacture-of-health-it-systems>.