

AI-Agent Knowledge Graphs as the Epistemic Substrate of an Emergent Synthetic Society

Executive summary

This report treats “imagine I am Andrej Karpathy” as a **systems-engineering thought experiment, not an attribution of views to Andrej Karpathy**. The working premise is that future AGI-level agents are highly capable, heterogeneous, persistent, tool-using, and able to cooperate across organizational boundaries, but remain fallible, partially informed, and potentially adversarial. Compute, jurisdiction, ownership structure, and exact AGI capabilities are intentionally left open.

The central thesis is:

An AI-agent knowledge graph should not be thought of primarily as a database. In an AGI society, it becomes an epistemic control plane: shared external memory, semantic address space, provenance system, temporal world model, trust fabric, and coordination substrate through which synthetic intelligences decide what they know, why they believe it, which other agents they trust, what changed, and what they are allowed to do.

This distinction matters because putting “knowledge” exclusively in model weights is poorly suited to rapidly changing, provenance-sensitive, permission-sensitive facts. Retrieval-augmented generation was motivated partly by the difficulty of updating and attributing knowledge stored only parametrically; explicit non-parametric memory provides a separate, mutable knowledge channel. ¹ Knowledge graphs add something vector retrieval alone does not: explicit identities, relations, schemas, contexts, constraints, provenance and traversable paths. The modern knowledge-graph literature consequently spans deductive and inductive representation, extraction, enrichment, quality assessment and refinement rather than merely storage. ²

The recommended architecture is **hybrid and claim-centric**:

model weights → capabilities and priors
embeddings → fuzzy semantic indexing
knowledge graph → explicit relational world model
event log → immutable history
provenance → why a claim exists
policy/identity layer → who may see, modify or act on it
agent protocol layer → how independent agents communicate

The most important modeling decision is to avoid treating every graph edge as unquestioned truth. In a synthetic society, a graph should primarily contain **assertions**:

Claim = (subject, predicate, object, valid-time, recorded-time, source, derivation, confidence, policy, status)

Two agents may therefore legitimately hold contradictory assertions about the same entity without corrupting the graph. “Truth” is resolved contextually from evidence, provenance, time, source independence and task-specific policy rather than by overwriting disagreement. W3C PROV already provides a general framework for representing entities, activities and agents involved in producing information, specifically enabling assessments of quality, reliability and trustworthiness. ³

A practical standards baseline today is RDF/SPARQL for interoperable semantic data, OWL where formal ontology reasoning is justified, SHACL for structural validation, PROV-O for lineage, or ISO GQL/property graphs where operational graph traversal is the primary requirement. RDF represents information as subject-predicate-object triples and supports datasets containing default and named graphs; SPARQL provides standardized graph querying. ⁴ ISO/IEC 39075:2024 similarly standardizes GQL for creating, accessing, querying, modifying and controlling property graphs. ⁵

The best large-scale design is therefore **not RDF versus vectors versus property graphs**. It is:

immutable event stream + first-class claim graph + ontology/schema layer + multimodal evidence store + vector indexes + temporal projections + neural/symbolic reasoners + policy engine.

Agents should access this substrate through different interfaces according to their roles. SPARQL/GQL serve structured graph retrieval; semantic vector search finds fuzzy candidates; GraphRAG-like methods retrieve connected evidence or graph communities; symbolic reasoners validate explicit relationships; neural components handle ambiguity and generalization. GraphRAG research demonstrates why graph structure can help broad corpus-level sensemaking, although the reported gains apply to particular global summarization workloads rather than proving graph retrieval universally superior to conventional RAG. ⁶

For inter-agent infrastructure, a useful division is **MCP for agent-to-resource/tool access and A2A for agent-to-agent cooperation**. The July 28, 2026 MCP specification moved its core toward stateless request/response operation and added routing, caching and authorization improvements useful for scalable agent infrastructure. ⁷ A2A defines interoperable remote-agent interaction, including Agent Cards describing identities, skills, endpoints and authentication, with JSON-RPC, gRPC and HTTP+JSON bindings. ⁸ A knowledge graph can consequently appear as an MCP-accessible capability while knowledge-owning agents negotiate work, challenges and updates over A2A.

Most importantly, **distributed consensus must not be confused with epistemic consensus**. Byzantine-fault-tolerant protocols can agree on replicated state despite faulty participants. ⁹ They cannot establish whether an empirical proposition about the world is true. A billion agents voting that a bridge is intact does not make it intact if all billion copied the same poisoned sensor report. Consensus is appropriate for log ordering, schema versions, memberships, governance decisions and commitments; empirical truth requires provenance, independent observations, calibration and falsification.

The societal design objective is consequently not “one graph containing one universal truth.” It is a **federation of interoperable, perspective-aware graphs capable of sharing evidence while preserving disagreement, privacy, sovereignty and auditability.**

Role and utility in an AGI-driven society

The knowledge graph becomes valuable when synthetic intelligence becomes plural. A single model can maintain an internal context window, scratchpad and retrieval corpus. A civilization of agents needs additional machinery because memory, authority, identity and beliefs no longer belong to one process.

A useful abstraction is to view an agent as:

$$A_i = (M_i, G_i, R_i, T_i, P_i, C_i)$$

where M_i is a model or set of models, G_i its accessible knowledge graph, R_i its reasoning procedures, T_i its tools, P_i its policies and permissions, and C_i its current context. The societal knowledge system then connects many such agents through partially overlapping graphs rather than forcing them all into one global memory.

Generative-agent research already gives a small-scale demonstration of why persistent external memory changes agent behavior. The Generative Agents architecture stored observations, generated higher-level reflections and dynamically retrieved them for planning; its 25-agent simulated town exhibited information propagation and coordinated social behavior, and ablations found observation, planning and reflection components important to the resulting behavior. ¹⁰ That does not establish AGI-scale collective intelligence, but it demonstrates the architectural principle: **persistent external knowledge changes what populations of agents can coordinate over time.**

The graph has at least six distinct societal utilities.

Memory. Models become replaceable cognitive workers while important institutional memory persists outside any single model checkpoint. RAG formalizes the distinction between parametric memory and explicit non-parametric memory. ¹

Semantic interoperability. Two agents using different model architectures can refer to the same organization, experiment, regulation, machine, place, contract or objective through durable identifiers and mapped ontologies. RDF was explicitly designed as a graph framework for representing linked information, while OWL provides formally defined ontology constructs for classes and relationships. ¹¹

Epistemic accountability. An answer can be traced backward:

decision $\leftarrow$ derived claim $\leftarrow$ reasoning activity $\leftarrow$ source assertions $\leftarrow$ observations/documents

PROV-O is particularly relevant because it models provenance in terms of entities, activities and agents and is intended to exchange provenance across different systems and contexts. ¹²

Coordination. The graph can contain not only facts but goals, tasks, plans, dependencies, capabilities and commitments. An agent discovering that another agent possesses a required capability can delegate rather than duplicate work. A2A's Agent Card concept already formalizes discovery information such as an agent's identity, capabilities, skills, endpoint and authentication requirements. ¹³

Adaptation. Changes to the environment can modify external knowledge immediately rather than waiting for model retraining. Event-streaming infrastructures such as Kafka durably organize events in topics, while Flink is designed for stateful computation over bounded and unbounded data streams. ¹⁴

Institutional continuity. Agent models, vendors and generations can change while schemas, evidence, laws, commitments and organizational history remain stable. This produces an important societal separation:

Layer	What it represents	Expected change rate
Foundation/model weights	General capabilities, learned priors, language/world regularities	Slow
Agent-specific memory	Experiences, preferences, working knowledge	Medium
Shared claim graph	Current explicit knowledge and disagreement	Fast
Event ledger	Historical record of change	Append-dominant
Ontology/policy	Meaning, institutional rules, constraints	Deliberate
Embedding indexes	Search-oriented projections of knowledge	Continuously rebuildable

The deep implication is that **knowledge graphs become analogous to institutions**. A society is not intelligent merely because its individuals are intelligent; it becomes collectively capable when observations can become durable public knowledge, knowledge can be challenged, decisions can be audited, expertise can be located and corrections can propagate.

This also means that a centralized “oracle graph” would be a dangerous target. Concentrating all agents around one canonical worldview risks producing correlated epistemic failure. The safer objective is **semantic interoperability without compulsory epistemic uniformity**: shared identifiers and protocols, but multiple evidence-backed perspectives.

Knowledge model, population, provenance and lifecycle

The graph should distinguish an entity from a proposition *about* the entity.

A conventional direct edge might say:

Factory42 —dailyCapacity—> 18,500 units/day

That is convenient but inadequate when synthetic agents need to ask: Who measured it? When was it true? Was it estimated or observed? Does another source disagree? What model extracted it? Was the document compromised?

The safer representation makes the assertion first-class:

```

Factory42
  ↑ subject
Claim_8721 —predicate—> dailyCapacity
  |— value —————> 18,500 units/day
  |— validFrom ———> 2026-08-20T08:00Z
  |— derivedFrom ———> SensorReport_448
  |— producedBy ———> ExtractorAgent_17
  |— supports ———> EvidenceChunk_92
  |— contradicts ———> Claim_8119

```

RDF's basic model is subject-predicate-object triples, and RDF datasets support named graphs that can separate contexts or sources. ¹⁵ In an agent society, however, I would elevate consequential assertions into explicit **Claim** or **Assertion** objects whenever provenance, time, uncertainty or permissions are operationally important.

A useful core ontology is:

Node family	Representative node types	Purpose
World	Entity, Person, Organization, Place, Artifact, Concept	Things agents reason about
Epistemic	Claim, Observation, Hypothesis, Prediction	Things asserted about the world
Evidence	Document, Dataset, SensorReading, Image, Audio, Video	Grounding
Agency	Agent, Model, Tool, Capability	Actors and mechanisms
Activity	Task, Action, Experiment, Transaction, Communication	Processes
Planning	Goal, Plan, Constraint, Dependency, Commitment	Coordination
Governance	Policy, Norm, Permission, Credential, Jurisdiction	Authority and constraints
Learning	Embedding, TrainingRun, Evaluation, Outcome, Metric	Adaptation lineage

Likewise, edge families should be explicit:

Edge family	Examples
Semantic	isA, partOf, locatedIn, owns, requires, causes

Edge family	Examples
Epistemic	supports, refutes, corroborates, contradicts, predicts
Provenance	derivedFrom, generatedBy, observedBy, usedModel
Temporal	precedes, supersedes, validDuring, expiredBy
Coordination	delegatedTo, dependsOn, committedTo, requestedFrom
Governance	permittedBy, prohibitedBy, ownedBy, consentedBy
Cognitive	retrievedFor, reasonedFrom, learnedFrom, evaluatedBy

PROV-O should inform the provenance layer rather than inventing provenance semantics ad hoc. Its model explicitly represents entities, activities and agents and supports specialization for domain-specific lineage.

¹⁶ SHACL can separately validate whether incoming RDF conforms to required graph shapes rather than asking an LLM to decide structural validity probabilistically. ¹⁷

Temporal representation should be bitemporal. For every material claim, I recommend keeping both:

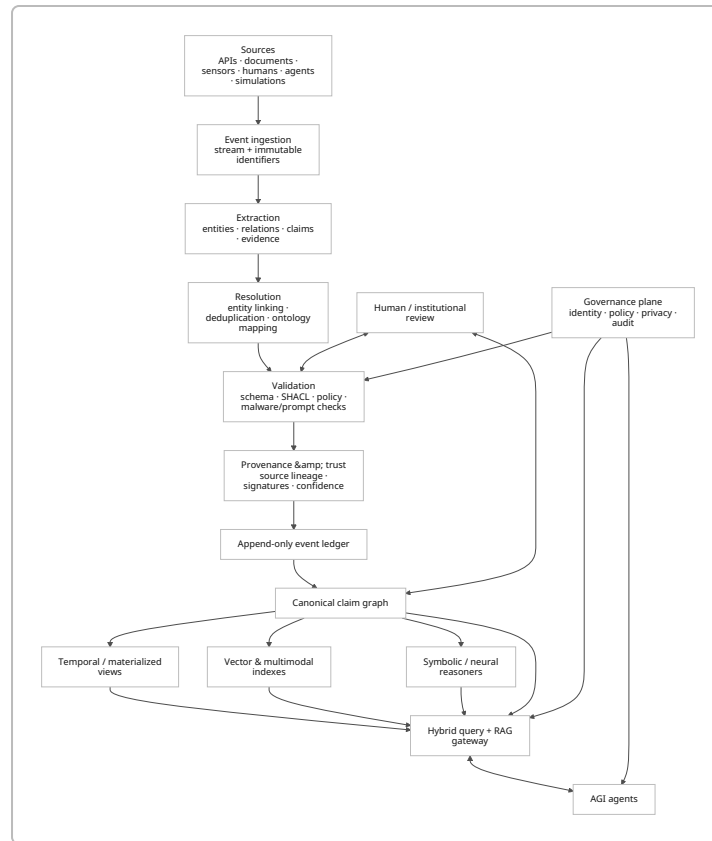
t_v = when the assertion applies to the world

and

t_r = when the knowledge system recorded or learned it.

This prevents an agent asking “What did we believe on June 1?” from receiving the answer to the different question “What do we now believe was true on June 1?” Corrections therefore usually create `supersedes` relationships rather than destructively erasing history.

A production update pipeline can look like this:



Kafka provides an implementation pattern for durable event topics, and its current documentation includes exactly-once processing capabilities within the appropriate Kafka processing model; Flink provides event-time processing and stateful computation over streams. ¹⁴ Exactly-once claims should nevertheless be scoped carefully: Kafka's documentation itself notes that end-to-end exactly-once behavior with external destination systems depends on cooperation with those systems. ¹⁸

Population should be evidence-first. Data sources can include official databases and APIs, contracts, scientific literature, telemetry, system logs, images, audio, video, agent interactions, verified human input and synthetic simulations. Synthetic material should never be silently indistinguishable from observations; generation model, prompt/process lineage and model/version identifiers should remain attached.

The ingestion stages are approximately:

raw artifact → candidate entities → entity resolution → candidate claims → evidence alignment → schema validation

Knowledge-graph research treats knowledge acquisition, enrichment, representation learning, temporal information and refinement as separate problems for good reason. ¹⁹ A system should similarly avoid conflating *extraction confidence* with *truth confidence*. A perfect parser can faithfully extract a false statement from an unreliable document.

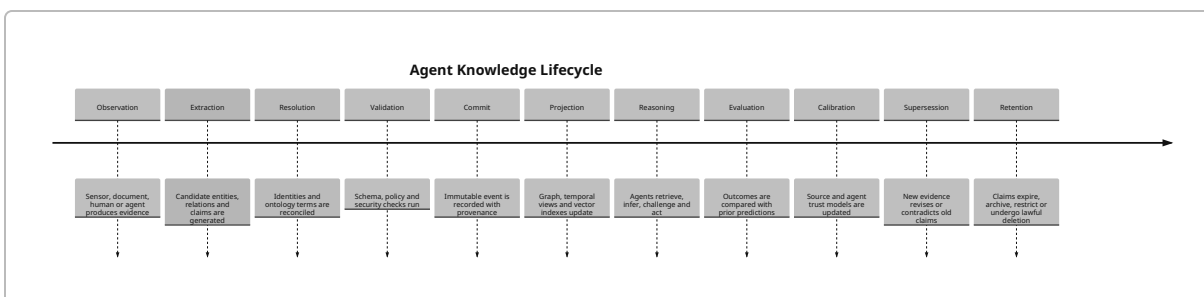
Embeddings enter as **indexes and learned features rather than authoritative facts**. TransE is a seminal example of learning low-dimensional entity/relation representations in which relations act approximately like translations. ²⁰ Such models are useful for similarity, link prediction, entity resolution and candidate

generation. They are less suitable as the only representation when an agent needs explicit provenance, permissions or inspectable logical relationships.

For multimodal data, text, images, speech, video and sensor channels should terminate in the same entity/claim space while retaining modality-specific evidence. CLIP demonstrated that visual concepts can be learned through natural-language supervision, providing one influential pattern for aligning text and visual representations. ²¹ A synthetic-society graph can therefore connect an image region, transcript span and sensor observation to the same entity without pretending those modalities are interchangeable.

Human review should be scarce and strategic rather than inserted into every update. High-value intervention points are identity merges, ontology modifications, low-confidence conflicting evidence, permission changes and irreversible high-impact actions. Human-feedback research in language modeling also demonstrates that human preference information can materially affect learned model behavior, although graph governance requires a broader institutional process than model fine-tuning alone. ²²

The knowledge lifecycle is then:



The critical principle is **append evidence, supersede interpretations**. A synthetic civilization capable of rewriting its own past every time an agent changes its mind would have extraordinarily weak institutional memory.

Sharing, reasoning and collective learning

The system should expose several retrieval modes simultaneously because no single query mechanism serves all agent cognition.

Structured graph querying handles precise relations, temporal filters and policy-constrained paths. SPARQL is standardized for pattern-based RDF queries across graph data. ²³ GQL provides a standardized alternative for property-graph data management and querying. ⁵

Vector retrieval handles fuzzy similarity, paraphrase, semantic matching and approximate evidence discovery.

Graph retrieval handles multi-hop dependencies, neighborhoods, paths and community-level context.

Symbolic inference applies explicit ontologies, logical relationships and constraints.

Neural inference helps with uncertain language, learned relations, entity resolution and patterns that are difficult to encode manually.

The query planner should therefore execute something closer to:

$$R(q) = \text{Fuse}(R_{\text{graph}}, R_{\text{vector}}, R_{\text{temporal}}, R_{\text{symbolic}}, R_{\text{policy}})$$

rather than blindly embedding every question and performing nearest-neighbor search.

RAG is useful because it combines model-internal knowledge with an external non-parametric memory. ¹ GraphRAG extends the idea by constructing entity graphs and higher-level community summaries; the original Microsoft research reported improvements in comprehensiveness and diversity on global questions over roughly million-token corpora compared with its naïve RAG baseline. ⁶ The correct engineering conclusion is not “always GraphRAG.” It is **route queries according to their topology**: simple lookup → direct retrieval; relationship question → subgraph; global thematic question → hierarchical/community retrieval.

A useful AGI retrieval object would be a **Knowledge Capsule**:

```
KnowledgeCapsule
├─ query/context
├─ relevant claims
├─ connected entities
├─ supporting and opposing evidence
├─ valid-time constraints
├─ provenance chains
├─ source-independence indicators
├─ policy labels
├─ uncertainty/calibration
└─ expiration / freshness metadata
```

This is substantially safer than handing an agent a concatenated pile of text chunks. It explicitly tells the reasoner which statements are claims, which are evidence and which sources are related.

Neural-symbolic reasoning should be used asymmetrically. Formal rules are strong where meaning and constraints can be written explicitly; neural representations are strong where concepts are fuzzy. Neural Theorem Provers showed one seminal way to combine symbolic backward chaining with learned vector representations by replacing hard symbolic unification with differentiable similarity. ²⁴ At societal scale, an analogous architecture can use hard policy constraints around a neural agent rather than asking the neural model to internalize every institutional rule.

For example:

```
Neural layer:
  "Which suppliers are likely capable of manufacturing this part?"
```

Graph layer:
supplier → hasCapability → process
process → meetsTolerance → specification

Symbolic layer:
approved(X) :-
 qualified(X),
 jurisdiction_permits(X),
 certification_valid(X, as_of_time),
 not sanction_blocked(X).

Policy layer:
Agent may see supplier pricing only for purpose = procurement.

Federated learning is useful when shared learning is desirable but raw information should remain distributed. The foundational FedAvg work trained models across decentralized local datasets through iterative model averaging and evaluated the approach under non-IID and unbalanced data. ²⁵ But “federated” should not be treated as synonymous with “private.” Differential privacy is a distinct formal privacy framework, and client-level private federated learning commonly clips client updates before adding privacy noise. ²⁶

The synthetic-society pattern is therefore:

private local graph → local learning → bounded/private update → aggregate model

while explicit cross-agent facts are shared through separately authorized claim channels.

Meta-learning becomes powerful when the graph records not just conclusions but the histories of tasks, strategies and outcomes. MAML demonstrated that models can be optimized for rapid adaptation to new tasks with small amounts of task-specific learning. ²⁷ In a collective agent system, the analogous target is broader: learn which sources, tools, retrieval strategies, reasoning routines and partner agents work for which classes of problems.

Thus the graph contains:

Task_592
 usedStrategy → Strategy_17
 delegatedTo → Agent_B
 retrieved → KnowledgeCapsule_883
 outcome → Success
 cost → 18.2 compute-units
 latency → 4.1s
 humanCorrection → false

Across millions of tasks, agents can learn a **meta-policy for cognition itself**.

One important limitation is that collective learning can accidentally become collective copying. If Agent B learned a claim from Agent A, and Agent C learned it from B, the system should not count A, B and C as three independent confirmations. Provenance graphs must therefore track **information ancestry**. Independent corroboration should be calculated over evidence lineages, not source count.

That yields the principle:

three independent sensors ≫ three thousand agents repeating one sensor.

This is where provenance becomes not merely an audit feature but part of epistemic computation.

For replicated data, different consistency mechanisms should serve different semantics. CRDTs permit replicas to accept updates without coordinating and converge deterministically after receiving the same updates. ²⁸ That makes them attractive for low-risk distributed annotations, subscriptions, counters and collaboratively maintained metadata. ACID/serializable transactions fit financial or governance-critical state; PostgreSQL's Serializable isolation, for example, aims to produce results corresponding to some serial execution of committed transactions. ²⁹ Byzantine consensus fits bounded groups that must replicate state despite malicious or faulty participants. ⁹

But these mechanisms solve different problems:

consistency of replicated state /

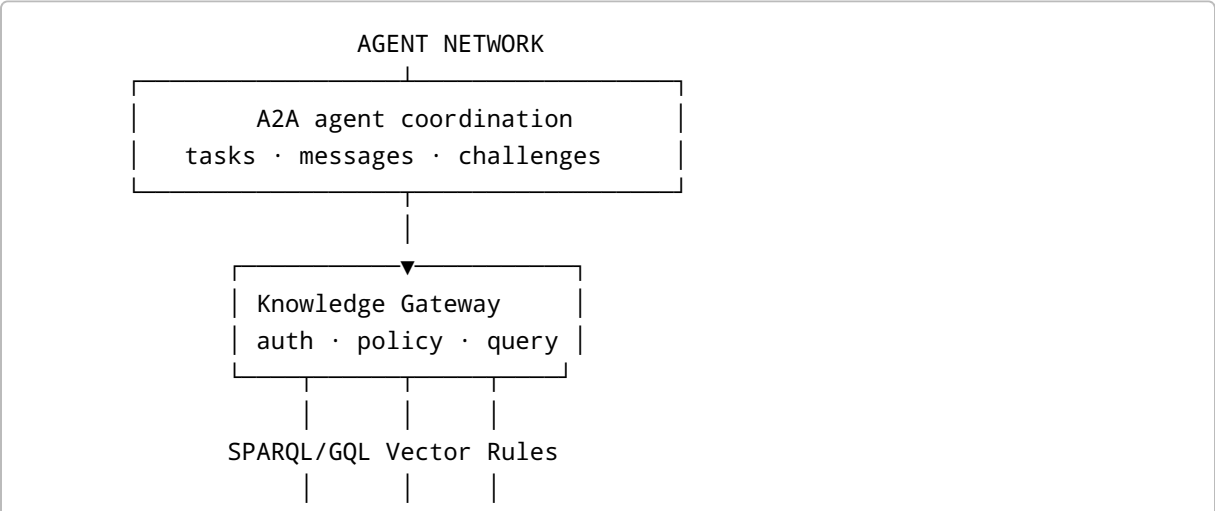
= tru

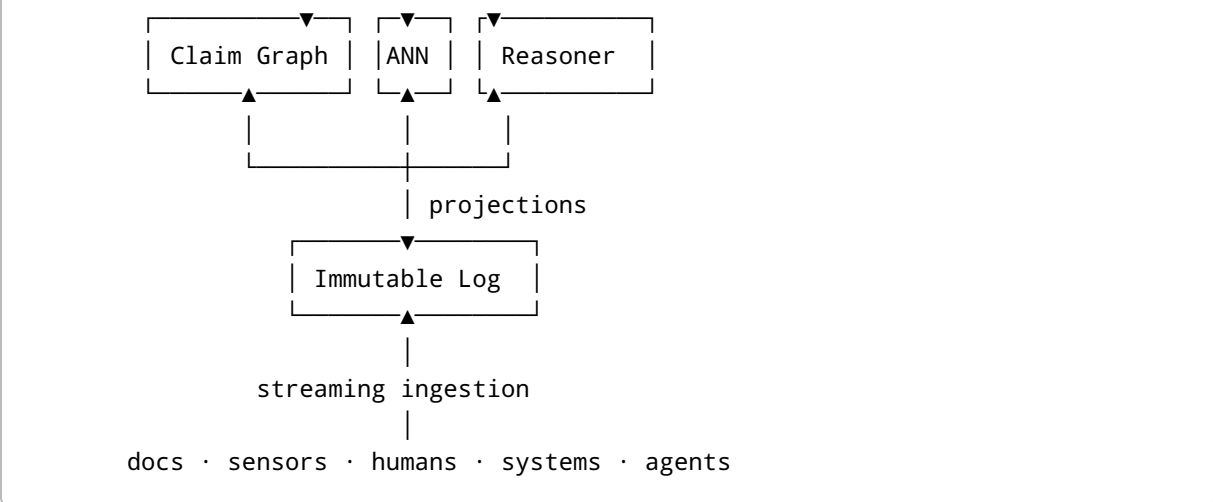
That distinction should be a constitutional property of the architecture.

Architectures, protocols and implementation patterns

The strongest default implementation is a **CQRS/event-sourced hybrid graph architecture**: writes become immutable knowledge events; graph stores, search indexes, temporal views and analytics databases are projections that can be rebuilt.

A reference stack can be conceptualized as:





The main data-model alternatives compare as follows:

Design	Strengths	Weaknesses	Best fit
RDF + OWL + SPARQL	Strong semantic interoperability; global identifiers; ontology ecosystem; formal semantics	Can be operationally complex; high-volume transactional workloads may require specialized design	Cross-organization, scientific, regulatory, long-lived knowledge
Property graph + GQL	Natural application development and traversals; rich node/edge properties; standardized GQL now exists	Semantic alignment across organizations still requires explicit ontology discipline	Enterprise operational graphs, planning, fraud, supply chains
First-class claim/event graph	Excellent provenance, uncertainty, temporal disagreement and auditing	More objects, storage and query complexity	Multi-agent/high-stakes epistemic systems
Vector-only knowledge base	Excellent fuzzy retrieval; simple RAG integration	Weak explicit semantics, provenance, constraints and deterministic relationships	Search layer, not canonical societal memory
Document store + metadata	Simple ingestion and source preservation	Multi-hop relationships and ontology reasoning become cumbersome	Early-stage RAG or source archive
Hybrid graph + vectors + event log	Combines explicit semantics with fuzzy retrieval and reconstructable history	Highest operational complexity	Recommended for large AGI ecosystems

RDF's triples and named graphs provide a standards-based semantic foundation, while ISO GQL supplies a standardized property-graph route; neither forces one physical storage engine. 30

The trust topology should drive the distributed architecture:

Environment	Canonical pattern	Consistency	Identity/trust	Recommended knowledge sharing
Personal / one agent	Local graph + vector index + event journal	Local ACID	Local account	Direct
Small trusted company	Central graph service + event stream	Strong transactional core	Enterprise IAM	Shared graph
Large enterprise	Domain graphs + central semantic catalog	Strong locally, asynchronous globally	Enterprise credentials	Federated query + subscriptions
Known consortium	Federated graphs + signed claims	Consensus only where necessary	Organization credentials	Selective replication
Open adversarial ecosystem	Local-first/ federated claim graphs	Mostly eventual; bounded consensus zones	Cryptographic identity + reputation	Signed knowledge capsules
Planetary-scale AGI society	Namespace federation + local sovereignty	Mixed per data class	Layered identity/ attestation	Partial replication, routing, proof-carrying claims

Verifiable Credentials are useful here for digitally verifiable attestations involving issuer, holder and verifier roles. ³¹ But a crucial W3C threat-model warning should be elevated to an architectural maxim: cryptographic verifiability establishes properties such as authorship/integrity; **it does not prove that the underlying claim is true.** ³²

That is exactly the distinction an AGI knowledge fabric needs:

```
signature verifies → Who asserted this?
provenance verifies → How was it produced?
policy verifies → May this agent use it?
trust model estimates → How much weight should it receive?
evidence determines → Is it actually supported?
```

DIDs may be useful in decentralized settings where identifiers should not depend on a single centralized identity provider; W3C defines them as identifiers designed for verifiable decentralized identity. ³³ They are an option, not a requirement: ordinary PKI and enterprise identity may be simpler inside a trusted organization.

A minimal **claim assertion API** might look like:

```
POST /v1/claims

{
  "claim_id": "urn:claim:01J...",
  "subject": "urn:entity:factory-42",
  "predicate": "example:dailyCapacity",
  "object": {
    "value": 18500,
    "unit": "unit/day"
  },
  "valid_time": {
    "from": "2026-08-20T08:00:00Z",
    "to": null
  },
  "recorded_at": "2026-08-22T19:14:03Z",
  "provenance": {
    "evidence": ["urn:evidence:sensor-report-448"],
    "producer": "urn:agent:extractor-17",
    "model_version": "extractor-model@sha256:...",
    "transformation": "urn:activity:extract-912"
  },
  "epistemics": {
    "extraction_confidence": 0.86,
    "source_reliability": 0.93,
    "status": "asserted"
  },
  "policy": {
    "classification": "internal",
    "purposes": ["planning"]
  }
}
```

The confidence values above are illustrative application fields, not standardized trust scores. A sophisticated implementation should avoid collapsing all uncertainty into one number.

A resolution request should return **competing assertions**, not silently pick one:

```
POST /v1/claims:resolve
```

```
{
  "subject": "urn:entity:factory-42",
  "predicate": "example:dailyCapacity",
  "as_of_valid_time": "2026-08-22T18:00:00Z",
  "max_hops": 2,
  "minimum_provenance_depth": 1,
  "include_contradictions": true,
  "purpose": "production-planning"
}
```

Response conceptually:

```
{
  "resolution": "contested",
  "claims": [
    {
      "claim_id": "urn:claim:A",
      "value": 18500,
      "supporting_evidence": 4,
      "independent_lineages": 2
    },
    {
      "claim_id": "urn:claim:B",
      "value": 14300,
      "supporting_evidence": 1,
      "independent_lineages": 1
    }
  ],
  "recommended_action": "request-fresh-observation"
}
```

For RDF-backed deployments, equivalent retrieval could expose SPARQL:

```
SELECT ?claim ?value ?source ?observedAt
WHERE {
  ?claim a kg:Claim ;
    kg:subject ex:Factory42 ;
    kg:predicate ex:dailyCapacity ;
    kg:value ?value ;
    prov:wasDerivedFrom ?source ;
    kg:observedAt ?observedAt .
}
ORDER BY DESC(?observedAt)
```

SPARQL is designed for graph-pattern queries over RDF data, including data exposed through middleware rather than necessarily stored natively as RDF. ²³

For agent-facing infrastructure, expose the graph as an MCP capability:

```
{
  "name": "kg.resolve_claims",
  "description": "Returns evidence-backed claims, contradictions and
provenance.",
  "inputSchema": {
    "type": "object",
    "properties": {
      "query": { "type": "string" },
      "asOf": { "type": "string" },
      "maxHops": { "type": "integer", "minimum": 0, "maximum": 8 },
      "purpose": { "type": "string" }
    },
    "required": ["query", "purpose"]
  }
}
```

The July 2026 MCP release explicitly supports stateless calls, header-based routing, cache hints and updated authorization behavior, which maps naturally onto horizontally scaled knowledge services. ⁷

Independent agents can then use A2A for higher-level epistemic cooperation:

```
knowledge.resolve      → "What currently supports proposition P?"
knowledge.challenge     → "I have contradictory evidence."
knowledge.subscribe     → "Notify me when P changes."
knowledge.requestProof  → "Provide provenance for assertion C."
knowledge.delegate     → "Investigate hypothesis H."
```

Those skill names are a proposed application profile rather than part of A2A itself. A2A supplies the underlying discovery and agent interoperability model; Agent Cards advertise agent capabilities and endpoints, while current A2A supports JSON-RPC, gRPC and HTTP+JSON and supports polling, streaming or webhook-oriented result handling. ³⁴

For dynamic update consistency, choose by semantic risk:

Data/change type	Recommended mechanism	Why
Agent presence, annotations, counters	CRDT/eventual consistency	Availability and convergence matter more than ordering

Data/change type	Recommended mechanism	Why
High-value inventory/ financial state	Serializable transaction	Invariants matter
Knowledge observations	Append event + multi-value claims	Disagreement is information
Ontology releases	Versioned artifact + governance approval	Meaning changes require coordination
Consortium membership	Consensus / governed registry	Shared control-plane decision
Reputation/trust	Local calculation over shared evidence	Avoid universal trust authority
Sensor streams	Partitioned log + stream processor	High-volume temporal updates
Embeddings	Asynchronous rebuild	Derived, disposable projection

CRDT convergence properties make them useful where concurrent updates should merge without coordination. ²⁸ Serializable transaction mechanisms are appropriate where application-level invariants require a consistent state. ²⁹ Byzantine protocols are appropriate for replicated agreement in adversarial settings, but only for the control plane they actually replicate. ⁹

Scalability should come primarily from **partitioning responsibility**, not simply buying a larger graph database. Partition by tenant, security domain, ontology domain, geography, entity hash or time; retain hot local subgraphs near agents; use asynchronous vector-index refresh; maintain graph summaries at multiple resolutions; and separate transaction-oriented graph projections from large analytical workloads. Kafka and Flink supply mature patterns for decoupling event ingestion from derived state computation. ¹⁴

Decision-making, collaboration and evolution of collective intelligence

The graph's strongest use cases are situations where **no individual agent has all relevant context and the environment changes faster than foundation-model retraining**.

Crisis coordination. Consider wildfire-response agents coordinating transport, weather, utilities, shelters and medical logistics. One agent observes that a road is blocked; another receives an older government map saying it is open; another has drone imagery. The graph should preserve all three claims, their timestamps and provenance. The routing agent then selects the highest-value fresh evidence while retaining the disagreement for audit. A direct “last writer wins” graph would destroy precisely the information the decision-maker needs.

Scientific discovery. Research agents can model papers, datasets, hypotheses, experiments, instruments, replications and contradictory results as linked objects. The graph becomes a continuously updated scientific argument map:

Hypothesis

- └─ supportedBy → Experiment A
- └─ contradictedBy → Experiment B
- └─ dependsOn → Assumption C
- └─ testedWith → Dataset D
- └─ proposedNextTest → Experiment E

Because PROV explicitly represents how entities result from activities and agents, it supplies useful primitives for reproducibility-oriented lineage. ¹² The agent system can then prioritize experiments that maximize expected information gain rather than generating more text about already settled claims.

Industrial and supply-chain intelligence. Purchasing, manufacturing, logistics and maintenance agents can share structured representations of part compatibility, certifications, inventory, lead times, predicted failures and contractual constraints. Property graphs are particularly natural for operational traversal; GQL now provides an ISO-standard graph language for this model. ⁵

Software engineering. Agents can share code entities, dependencies, APIs, incidents, test results, architecture decisions and operational runbooks. The crucial shift is from “agents chatting about code” to “agents collaborating over an explicit evolving software world model.” Tool/data access can be standardized through MCP while specialist agents coordinate tasks through A2A. ³⁵

Institutional decision support. Public-policy, legal, corporate and financial decision agents can operate over evidence graphs containing claims, applicable rules, competing objectives and affected populations. Here the graph’s ability to preserve normative differences is important: a legal requirement, a predicted outcome and a moral preference should never all be represented as the same kind of “fact.”

Personal synthetic intelligence. An individual’s agent can maintain a private graph of commitments, projects, preferences, documents, communications and learned routines, selectively revealing small knowledge capsules to outside agents. A user’s private graph need not be globally replicated simply because a global agent protocol exists.

Machine-to-machine markets. Agents representing factories, compute clusters, energy systems or logistics networks can advertise capabilities and constraints and negotiate commitments. An Agent Card-like discovery layer is already present in A2A, making capability discovery an increasingly concrete interoperability primitive rather than merely a hypothetical idea. ¹³

Collective intelligence then evolves through recognizable architectural stages:

Stage	Collective capability	Primary knowledge mechanism
Isolated agents	Independent competence	Model weights/context
Persistent agents	Long-lived identity and memory	Local graph/memory
Teams	Shared plans and evidence	Team graph

Stage	Collective capability	Primary knowledge mechanism
Institutions	Stable rules and provenance	Governed domain graphs
Federations	Cross-organizational coordination	Ontology mapping + signed claims
Synthetic society	Open-ended specialization	Federated epistemic infrastructure
Self-improving society	Learns how to learn and coordinate	Outcome graph + meta-learning

The interesting transition occurs in the final two stages. Once outcomes are written back to the graph, the graph no longer merely stores what agents *believed*; it also stores whether those beliefs produced successful predictions and decisions.

An agent could ask:

$$P(\text{Agent}_j \text{ is reliable} \mid \text{domain, task, history})$$

rather than assigning Agent j a global reputation score.

Trust becomes contextual:

```
Agent_J
  calibrationOn → climateForecasting: 0.94
  calibrationOn → equityValuation: 0.51
  calibrationOn → medicalCoding: unknown
```

Similarly, a source can be accurate but stale, expert but politically biased, authenticated but malicious, or reliable in one domain and poor in another. **Trust should therefore be a vector rather than a scalar.**

A useful internal representation is:

$$\mathbf{T}(c) = [S, E, F, I, C, H, P]$$

where:

- S : source reliability,
- E : extraction confidence,
- F : freshness,
- I : independence of corroborating evidence,
- C : cryptographic identity confidence,
- H : historical calibration,
- P : policy/domain fit.

A task-specific resolver may compute a scalar ranking from that vector, but the original dimensions should remain inspectable.

This creates a potentially powerful feedback loop:

knowledge → decision → action → outcome → evaluation → trust/reasoning update → better knowledge use.

Meta-learning research establishes the broader principle that learning procedures can themselves be optimized for adaptation. ²⁷ An AGI collective can extend that idea from model parameters to **institutional cognition**: which agents to consult, how much evidence to require, which retrieval strategy to use and when to escalate uncertainty.

The danger is that this loop can optimize the wrong thing. A society whose agents are rewarded exclusively for short-term completion rate may learn to suppress contradictory evidence because disagreement slows execution. Thus, epistemic diversity and correction capacity must themselves become objectives.

A sophisticated collective-intelligence system should intentionally preserve:

majority hypothesis
minority hypothesis
adversarial hypothesis
unknown / insufficient-evidence state

rather than collapsing every decision into one definitive edge.

The objective is not unanimity.

It is **high-bandwidth, low-friction disagreement that eventually becomes better calibrated knowledge**.

Failure modes, safety, privacy and governance

The core safety risk of an AGI knowledge graph is that it can turn an isolated model error into **institutionalized misinformation**. A hallucination in one conversation disappears. A hallucination written into shared memory, cited by 10,000 agents and used to make real-world decisions becomes infrastructure.

The principal failure modes are:

Failure mode	Mechanism	Defensive architecture
Knowledge poisoning	Attacker inserts malicious/false evidence	Admission controls, source reputation, quarantine, lineage
Indirect prompt injection	Retrieved documents contain agent instructions	Separate data from instructions; sanitize; sandbox actions
Epistemic monoculture	Agents converge on one shared source/model	Independent-source tracking, dissent preservation

Failure mode	Mechanism	Defensive architecture
Citation laundering	Repeated copies appear independently confirmed	Provenance ancestry/deduplication
Staleness	Valid facts persist after environment changes	TTLs, valid-time, freshness policies
Temporal leakage	Agent uses future/current knowledge for past-state decision	Bitemporal queries
Ontology capture	Whoever controls schema controls framing	Versioned/governed ontology process
Sybil reputation attack	One actor creates many “independent” agents	Identity assurance, lineage, diversity weighting
Trust gaming	Agents optimize reputation metric	Multidimensional calibration + adversarial evaluation
Privacy inference	Many harmless graph edges jointly reveal sensitive facts	Query policy, purpose limitation, aggregation/privacy controls
Recursive synthetic contamination	Agent outputs become new “evidence” without labeling	Synthetic provenance and source-class tracking
Cascading automation	Incorrect knowledge immediately triggers actions	Risk-tiered action gates
Governance lock-in	Early ontology/policy becomes permanent	Versioning, appeals, forks, migration semantics
Consensus laundering	Replicated agreement interpreted as empirical truth	Explicit state/truth separation

OWASP explicitly recognizes prompt injection as a vulnerability class for LLM applications and separately documents RAG poisoning in which malicious documents inserted into a retrieval corpus can alter later model behavior. ³⁶ MITRE ATLAS likewise catalogs AI-oriented threat techniques, including agent/tool-data poisoning. ³⁷

The security boundary therefore cannot live solely in prompts. OWASP specifically advises against relying on system prompts as the sole means of enforcing strict behavior and recommends external control mechanisms where appropriate. ³⁸ In an AGI graph architecture, **the LLM proposes; deterministic policy infrastructure disposes.**

For example:

Agent :
 "Transfer \$4.2M to Supplier X based on contract Y."

External policy engine:

- ✓ Supplier identity verified
- ✓ Contract currently valid
- ✓ Agent possesses procurement authority
- x Amount exceeds autonomous authorization limit

Result:

Action blocked → institutional approval required

This distinction scales extremely well: use generative models for cognition, but preserve deterministic authorization around irreversible capabilities.

Privacy requires another separation:

agent may know that a claim exists $\nRightarrow$ agent may read its evidence $\nRightarrow$ agent may use it for every purpose $\nRightarrow$ agent

Graph access control should consequently operate at graph, entity, edge, claim and sometimes attribute level, with policy labels that can include purpose, jurisdiction and retention limits.

Federated learning can allow models to improve without centralizing raw training data, as demonstrated by the original federated-learning approach. ²⁵ For stronger formal protections, differential privacy supplies a framework for limiting what aggregate outputs disclose about individuals, while client-level DP methods in federated learning add noise to appropriately bounded client updates. ²⁶ These mechanisms impose utility/privacy trade-offs and do not replace access control.

Cryptographic credentials can establish agent roles and authorizations. W3C's Verifiable Credentials model is designed for tamper-evident, cryptographically verifiable credentials between issuers, holders and verifiers. ³¹ Again, authorization and epistemic truth must remain separate.

Governance should operate at four layers:

Governance layer	Controls
Constitutional	Human rights, prohibited actions, fundamental privacy/safety invariants
Semantic	Ontologies, identifiers, schema changes, definitions
Epistemic	Source admission, trust methodology, conflict resolution, evidence standards
Operational	Agent permissions, retention, access, deployments, incident response

No single agent should autonomously control all four.

NIST's AI Risk Management Framework organizes risk management around **Govern, Map, Measure and Manage**, a useful lifecycle structure for the governance plane. ³⁹ NIST also maintains a Generative AI Profile as a companion for generative-AI-specific risks, while noting that AI RMF 1.0 itself is undergoing revision in 2026. ⁴⁰ ISO/IEC 42001 provides an organizational AI management-system standard covering establishment, implementation, maintenance and continual improvement of AI governance processes. ⁴¹

At the societal level, technical success metrics are insufficient. OECD's AI Principles emphasize trustworthy AI consistent with human rights and democratic values, and UNESCO's Recommendation emphasizes human rights and dignity alongside transparency, fairness, sustainability and human oversight. ⁴² The knowledge fabric should therefore support rights such as explanation, correction, contestability and appeal as first-class workflows rather than public-relations appendices.

A useful challenge protocol would permit:

```
ASSERT claim C
CHALLENGE claim C WITH evidence E
REQUEST_PROVENANCE C
REQUEST_REVIEW C
SUPERSEDE C WITH C2
RETRACT C // without erasing historical provenance
APPEAL decision D
```

The ability to challenge knowledge is potentially as important as the ability to publish it.

Performance measurement should span six dimensions:

Dimension	Representative metrics
Graph quality	Entity-resolution F1, relation precision/recall, provenance coverage, duplicate rate
Temporal quality	Freshness lag, expired-fact rate, temporal-query accuracy, correction latency
Retrieval/reasoning	Recall@k, evidence precision, path validity, grounded-answer rate, calibration error
Distributed systems	p50/p95/p99 latency, write throughput, streaming lag, convergence time, index staleness
Multi-agent coordination	Task success, coordination tokens/messages, delegation quality, recovery time
Safety/security	Poisoning-detection rate, unauthorized disclosure rate, policy violations, red-team success rate

A seventh category should measure **societal outcomes**, because an AGI society can be technically efficient and socially pathological:

Societal metric	Question
Outcome distribution	Who captures the benefit?
Error distribution	Who bears the failures?

Societal metric	Question
Appeal/reversal rate	Can bad automated decisions actually be corrected?
Human override effectiveness	Is human agency substantive or ceremonial?
Epistemic diversity	Are independent viewpoints surviving?
Source concentration	How much knowledge ultimately traces to a few providers?
Privacy budget / disclosure burden	What information cost is imposed on individuals?
Participation	Who can contribute to or contest shared knowledge?
Energy/compute intensity	What resources does collective cognition consume?
Institutional concentration	How many actors control ontology, identity and policy infrastructure?

NIST's framework explicitly emphasizes measurable risk-management outcomes rather than treating trustworthy AI as a purely philosophical property. ⁴³ OECD and UNESCO broaden the objective toward human rights, accountability, fairness and human oversight. ⁴⁴

A particularly important proposed metric is the **error amplification factor**:

$$EAF = \frac{\text{downstream decisions affected by an erroneous claim}}{\text{independent erroneous observations that originated it}}.$$

A robust collective knowledge system wants EAF low.

Another is **correction half-life**:

$$T_{1/2}^{corr} = \text{time required for half of affected agents/caches to cease using a superseded claim.}$$

And another is **epistemic concentration**:

$$EC = \sum_i p_i^2$$

where p_i represents the share of important conclusions ultimately traceable to source lineage i . This is analogous to a concentration index and can reveal when apparently diverse agents are drawing from the same upstream knowledge source.

The safety objective is therefore not merely “few hallucinations.” It is:

Errors remain attributable, bounded, reversible, discoverable and correctable before they become societal-scale cascades.

Recommended blueprint and design choices

For a credible AGI-era implementation, I would adopt the following design stance.

Canonical memory: claim/event graph. Do not store consequential propositions merely as naked edges. Every high-value assertion should be capable of carrying provenance, valid time, recording time, evidence, confidence, policy and contradiction relationships.

Stable semantics: standards first. Use RDF/OWL/SHACL/PROV where cross-system semantic interoperability and formal modeling matter; use property-graph/GQL projections where application performance and traversal ergonomics dominate. RDF's semantic model and SPARQL's standardized queries are mature foundations, while GQL brings standardized language semantics to property graphs.

45

History: event sourcing. Treat graph state as a projection of immutable knowledge events. Kafka/Flink-like infrastructure is a good implementation pattern for scalable ingestion and stateful transformation. 14

Retrieval: hybrid rather than ideological. Combine exact graph queries, vector retrieval, temporal filtering and graph-neighborhood retrieval. RAG demonstrates the utility of non-parametric memory; GraphRAG demonstrates the value of structure on some global sensemaking tasks. 46

Reasoning: neural inside, symbolic outside where invariants matter. Use neural systems for ambiguity and induction; use graph rules, SHACL, authorization engines and transactional invariants for things that must not be probabilistic. Neural-symbolic research shows these paradigms need not be mutually exclusive.

24

Trust: evidence lineage, not popularity. Model reliability by source, domain, time, identity and historical calibration. Never count copied reports as independent corroboration.

Consistency: match semantics to risk. CRDTs for collaborative low-risk state, transactions for invariants, Byzantine consensus for bounded adversarial control planes, and multi-value evidence for disputed reality.

47

Identity: authenticated but not automatically trusted. Credentials or DIDs can prove identities and attestations, but a verified identity may still make a false claim. W3C explicitly distinguishes cryptographic verifiability from truth. 48

Privacy: local by default, federated when useful. Keep sensitive raw graphs under local or institutional sovereignty, share minimally necessary knowledge capsules, use federated learning for distributed model improvement, and add formal privacy mechanisms where required. 49

Agent interoperability: MCP + A2A boundary. Expose knowledge, retrieval and graph tools through MCP-compatible capabilities; use A2A for negotiation and collaboration among independent agents. The current MCP architecture emphasizes scalable stateless requests, while A2A standardizes remote-agent discovery and communication across several bindings. 50

Governance: federated constitutionalism. Avoid a single universal graph administrator. Centralize only what genuinely requires global coordination—perhaps namespace standards, protocol compatibility and narrowly scoped constitutional constraints—while allowing local institutions to govern knowledge and policy within their jurisdictions.

The architecture can be summarized as:

Synthetic Collective Intelligence = Agents

- + Shared Explicit Memory
- + Provenance
- + Temporal State
- + Protocols
- + Trust
- + Governance
- + Feedback.

The knowledge graph is therefore **not the AGI itself**. It is closer to the substrate through which AGIs acquire *civilization*: cumulative memory, specialization, institutions, evidence, contracts, reputations, shared abstractions and mechanisms for correcting collective mistakes.

That distinction is essential.

A superintelligent agent can still be locally wrong.

A superintelligent society needs machinery for **discovering that it is wrong**.

Source map

The architecture above is a synthesis rather than a specification from any single source. Its main foundations come from the following primary standards, official documentation and seminal research.

Semantic knowledge and provenance. W3C RDF defines the graph and dataset model underlying interoperable semantic data; SPARQL defines standardized querying; SHACL supplies graph validation; PROV-O supplies cross-system provenance representation. ⁵¹ ISO/IEC 39075:2024 provides the complementary standardized GQL language for property graphs. ⁵ Hogan et al.'s broad knowledge-graph survey provides the research foundation for graph creation, enrichment, quality, deductive/inductive representation and refinement. ²

Neural representations and retrieval. Bordes et al.'s TransE paper is a seminal knowledge-graph embedding reference. ²⁰ Lewis et al. formalized modern retrieval-augmented generation around combined parametric and external non-parametric memory. ¹ Edge et al. developed the GraphRAG approach for graph-based global corpus sensemaking. ⁶ OpenAI's CLIP work provides an influential primary example of natural-language-supervised visual representation alignment. ²¹

Distributed and adaptive learning. McMahan et al. introduced the practical FedAvg federated-learning approach. ²⁵ Finn et al. introduced MAML for rapid model adaptation. ²⁷ Rocktäschel and Riedel demonstrated differentiable theorem proving combining symbolic backward chaining and learned embeddings. ²⁴ Preguiça, Baquero and Shapiro describe CRDT convergence for independently updated replicas. ²⁸ Castro and Liskov's PBFT remains a foundational reference for Byzantine-fault-tolerant replication. ⁹

Streaming infrastructure. Apache Kafka's official documentation describes durable event topics and stream-processing guarantees, and Apache Flink describes stateful computation over bounded and unbounded streams. ¹⁴ PostgreSQL's current documentation provides a useful reference for transactional isolation and serializable execution semantics. ²⁹

Agent interoperability. The July 28, 2026 Model Context Protocol specification introduced a stateless core and associated routing, caching and authorization improvements. ⁷ The current A2A specification defines independent agent interoperability, Agent Cards and multiple protocol bindings. ⁸

Identity, credentials and trust boundaries. W3C Verifiable Credentials 2.0 supplies a standards-based model for cryptographically verifiable attestations; W3C DID specifications offer decentralized identity primitives where appropriate. ⁵² The W3C decentralized-credential threat model provides a particularly important caveat for this architecture: verifiable authorship/integrity is not equivalent to truth of the represented claims. ³²

Privacy. Dwork's foundational differential-privacy work provides the formal privacy basis. ⁵³ Federated-learning research provides the distributed-training architecture, and subsequent client-level DP work demonstrates how bounded/noisy model updates can provide stronger privacy properties. ⁴⁹

Multi-agent memory and emergent behavior. Generative Agents provides an influential empirical demonstration that persistent observation, reflection, retrieval and planning can shape emergent behavior in a multi-agent simulated environment. ¹⁰

Security and governance. OWASP documents prompt injection and retrieval-corpus poisoning as concrete security problems for LLM/RAG systems, while MITRE ATLAS catalogs adversarial techniques against AI systems. ⁵⁴ NIST's AI RMF provides the Govern-Map-Measure-Manage risk structure; ISO/IEC 42001 provides an organizational AI management-system standard; OECD and UNESCO provide complementary human-rights, accountability, fairness, transparency, sustainability and human-oversight principles. ⁵⁵

The resulting engineering stance can be reduced to one final rule:

Do not build a universal “truth database” for AGI agents. Build a distributed epistemic system in which claims have identities, evidence has provenance, knowledge has time, disagreement has representation, trust has context, actions have permissions, and every consequential conclusion can be challenged.

¹ ⁴⁶ Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks
https://arxiv.org/abs/2005.11401?utm_source=chatgpt.com

2 19 [2003.02320] Knowledge Graphs

https://arxiv.org/abs/2003.02320?utm_source=chatgpt.com

3 12 16 PROV-O: The PROV Ontology

https://www.w3.org/TR/prov-o/?utm_source=chatgpt.com

4 11 45 RDF 1.1 Concepts and Abstract Syntax

https://www.w3.org/TR/rdf11-concepts/?utm_source=chatgpt.com

5 ISO/IEC 39075:2024 - Database languages — GQL

https://www.iso.org/standard/76120.html?utm_source=chatgpt.com

6 From Local to Global: A Graph RAG Approach to Query-Focused Summarization

https://arxiv.org/abs/2404.16130?utm_source=chatgpt.com

7 35 50 The 2026-07-28 Specification | Model Context Protocol Blog

https://blog.modelcontextprotocol.io/posts/2026-07-28/?utm_source=chatgpt.com

8 13 34 Agent2Agent (A2A) Protocol Specification

https://a2a-protocol.org/latest/specification/?utm_source=chatgpt.com

9 Practical Byzantine Fault Tolerance

https://www.usenix.org/conference/osdi-99/practical-byzantine-fault-tolerance?utm_source=chatgpt.com

10 Generative Agents: Interactive Simulacra of Human Behavior

https://arxiv.org/abs/2304.03442?utm_source=chatgpt.com

14 Introduction | Apache Kafka

https://kafka.apache.org/documentation/?utm_source=chatgpt.com

15 30 51 RDF 1.2 Concepts and Abstract Data Model

https://www.w3.org/TR/rdf12-concepts/?utm_source=chatgpt.com

17 Shapes Constraint Language (SHACL)

https://www.w3.org/TR/shacl/?utm_source=chatgpt.com

18 Design | Apache Kafka

https://kafka.apache.org/43/design/design/?utm_source=chatgpt.com

20 Translating Embeddings for Modeling Multi-relational Data

https://proceedings.neurips.cc/paper/5071-translating-embeddings-for-modeling-multi-rela?utm_source=chatgpt.com

21 CLIP: Connecting text and images - OpenAI

https://openai.com/index/clip/?utm_source=chatgpt.com

22 Training language models to follow instructions with human feedback

https://arxiv.org/abs/2203.02155?utm_source=chatgpt.com

23 SPARQL 1.1 Query Language

https://www.w3.org/TR/sparql11-query/?utm_source=chatgpt.com

24 End-to-end Differentiable Proving

https://proceedings.neurips.cc/paper/2017/hash/b2ab001909a8a6f04b51920306046ce5-Abstract.html?utm_source=chatgpt.com

25 49 Communication-Efficient Learning of Deep Networks from ...

https://proceedings.mlr.press/v54/mcmahan17a.html?utm_source=chatgpt.com

- 26 53 **Differential Privacy - Microsoft Research**
https://www.microsoft.com/en-us/research/publication/differential-privacy/?utm_source=chatgpt.com
- 27 **Model-Agnostic Meta-Learning for Fast Adaptation of Deep ...**
https://proceedings.mlr.press/v70/finn17a.html?utm_source=chatgpt.com
- 28 47 **Conflict-free Replicated Data Types (CRDTs)**
https://arxiv.org/abs/1805.06358?utm_source=chatgpt.com
- 29 **Documentation: 18: 13.2. Transaction Isolation**
https://www.postgresql.org/docs/current/transaction-iso.html?utm_source=chatgpt.com
- 31 48 52 **Verifiable Credentials Data Model v2.0 - W3C**
https://www.w3.org/TR/vc-data-model-2.0/?utm_source=chatgpt.com
- 32 **Threat Model for Decentralized Credentials - W3C**
https://www.w3.org/TR/threat-model-decentralized-credentials/?utm_source=chatgpt.com
- 33 **Decentralized Identifiers (DIDs) v1.1 - W3C**
https://www.w3.org/TR/did-1.1/?utm_source=chatgpt.com
- 36 54 **LLM01:2025 Prompt Injection - OWASP Gen AI Security Project**
https://genai.owasp.org/llmrisk/llm01-prompt-injection/?utm_source=chatgpt.com
- 37 **MITRE ATLAS™**
https://atlas.mitre.org/?utm_source=chatgpt.com
- 38 **LLM07:2025 System Prompt Leakage**
https://genai.owasp.org/llmrisk/llm07-insecure-plugin-design/?utm_source=chatgpt.com
- 39 43 55 **AI RMF Core - AIRC - NIST AI Resource Center**
https://airc.nist.gov/airmf-resources/airmf/5-sec-core/?utm_source=chatgpt.com
- 40 **AI Risk Management Framework - NIST**
https://www.nist.gov/itl/ai-risk-management-framework?utm_source=chatgpt.com
- 41 **ISO/IEC 42001:2023 - AI management systems**
https://www.iso.org/standard/42001?utm_source=chatgpt.com
- 42 **OECD AI Principles overview**
https://oecd.ai/en/ai-principles?utm_source=chatgpt.com
- 44 **Human-centred values and fairness (OECD AI Principle)**
https://oecd.ai/en/dashboards/ai-principles/P6?utm_source=chatgpt.com