Autonomous Enterprise Decision Architecture (AEDA)

Canonical Package v4: closed-loop enterprise control, world models, ten-layer reference design, and Codex handoff

Dr. Nate Dailey

2026-06-01

Reading note on the stack. AEDA is a ten-layer architecture: the ten processing layers between a Data Sources input boundary and an Executive Support consumer interface (Ontology Discovery, Ontology Governance, Enterprise Ontology, Knowledge Graph, GraphRAG, Agent Swarm, Enterprise Digital Twin, Simulation, Optimization, Decision).

Version 4 update. v4 recasts AEDA as a closed-loop enterprise control system and elevates two capabilities to core architecture: Closed-Loop Enterprise Control and Enterprise World Models. The control loop is:

```
Enterprise State (Digital Twin)
  -> Knowledge Graph -> GraphRAG -> Agent Swarm
  -> Enterprise Digital Twin -> Discrete-Event Simulation -> Monte Carlo Engine
  -> Optimization Engine -> Policy Engine -> Decision Layer
  -> Enterprise Actions -> Observed Outcomes -> Learning Loop
  -> (back to Knowledge Graph and Enterprise Ontology)
```

The enterprise world model adds a state-space abstraction (State, Transition, Event, Policy, Goal, Constraint) that the digital twin, simulation, optimization, and reinforcement-learning layers operate on, enabling counterfactual and what-if reasoning over reachable enterprise trajectories.

Project Origin and Discovery Provenance

AEDA began as a question, not a design. The question was simple to state and difficult to answer: can enterprise architecture become AI-native? Defense enterprise architecture, embodied in the Business Enterprise Architecture (BEA) and the DoD Architecture Framework

(DoDAF) with its DM2 data model, had matured into a rigorous documentation discipline. It captured the enterprise faithfully. What it did not do was compute. The originating inquiry asked whether that documentation could be made to reason, predict, and recommend, rather than merely describe. The answer arrived not as a single insight but through a structured discovery arc of seven phases, each motivated by the limit of the phase before it. This section narrates that arc as the project's origin story and records why each review in the corpus exists.

The first phase reframed the premise. The initial instinct was to make architecture AI-friendly, to wrap existing models in interfaces a model could read. The first discovery review rejected that framing. Enterprise architecture is already an ontology. DoDAF DM2 formally defines capabilities, activities, performers, resources, measures, conditions, rules, and information exchanges as typed entities with relationships. The enterprise had been writing ontologies for years without calling them that. The shift, therefore, was not from human-readable to AI-readable. It was from documentation to computation: treating the existing architectural vocabulary as a formal model a machine can operate on rather than a reference humans consult. This reframe set the trajectory for everything that followed.

The second phase made the ontology operational. If DM2 is an ontology, it can be expressed as an enterprise knowledge graph through RDF, OWL, and property-graph representations. The second review exists to establish that translation: entities become nodes, relationships become typed edges, and the architecture becomes queryable as a connected structure rather than a set of static views. This is the substrate on which all later reasoning depends.

The third phase confronted the limits of retrieval. A knowledge graph can be searched, but vector search alone returns proximity, not chains of inference. The third review introduces GraphRAG to add multi-hop reasoning and explicit evidence chains. The motivation is traceability: a decision-grade system must show the path from question to answer across linked entities, not surface a plausible passage. GraphRAG exists in the stack because defense decisions require defensible provenance.

The fourth phase distributed the reasoning. A single reasoning process cannot hold the breadth of an enterprise architecture with adequate depth. The fourth review introduces agentic systems that allocate reasoning across specialized agents, each responsible for a domain or function. The motivation is scale and specialization: complex enterprise questions decompose into coordinated sub-questions that distinct agents can pursue in parallel.

The fifth phase moved from knowing to modeling. The fifth review advances the enterprise digital twin, shifting the architecture from a descriptive record to a predictive and executable representation of enterprise state. The twin exists so the architecture can hold a live, queryable model of how the enterprise is configured and how it would behave, not merely how it was documented.

The sixth phase asked what happens next. A predictive twin invites the question of consequence, and the sixth review answers it with simulation: Monte Carlo methods, Bayesian networks, discrete-event simulation, and system dynamics. Each technique addresses a different class of

forward question, from probabilistic outcomes to event-driven behavior to feedback over time. Simulation exists in the stack to convert state into projected consequence.

The seventh phase targeted decision superiority rather than knowledge. The seventh review introduces optimization, including portfolio optimization and reinforcement learning, to move beyond understanding the enterprise toward selecting the best available course of action. The motivating distinction is sharp: the prior phases produce knowledge, while optimization produces decisions. This is where the architecture earns the name decision architecture.

The final synthesis named the result Autonomous Enterprise Decision Architecture. The evolution is recorded as a versioned progression from v0, BEA modernization, through v7, AEDA, with each version marking one of the discovery phases that extended documentation into a computational decision stack.

A subsequent ontology-discovery review then refined the architecture in an important way. The original layering placed the enterprise ontology near the front of the stack as a fixed input. That review recognized a maintenance problem: documents, contracts, requirements, policies, and missions change continuously, and a static ontology decays. The refinement inserts an Ontology Discovery Layer and an Ontology Governance Layer ahead of the Enterprise Ontology, so the model extracts and curates its own structure as source material changes. This makes AEDA self-maintaining rather than a snapshot, and it is why the canonical ten-layer stack carries discovery and governance as distinct early layers.

Two complementary corpora ground this work. The first is a set of 23 GAO-anchored DCIO BEA reviews supporting the funded prototype and a separate white paper. The second is the set of 11 AEDA-track theoretical reviews documented in this package. The first establishes the applied, audit-anchored case; the second develops the architectural theory the discovery arc produced. The working baseline for integration is the NASA Future-State round-trip pipeline at D:/Claude_Code/archi, comprising an Archi 5.8 model, a jArchi build script, ArchiMate Open Exchange XML, a Node.js parser, and a six-tool MCP server. The brain target resides in MITRE GitLab, and Codex performs integration from this package.

Canonical Definitions

The following definitions establish a shared vocabulary for the Autonomous Enterprise Decision Architecture (AEDA). Each term is defined to be precise, layer-aligned, and grounded in the reference evidence. Definitions describe the canonical concept; representative measured results from the source literature are noted where they substantiate the concept, not as guarantees of any specific deployment.

Enterprise Architecture. The disciplined specification of an organization's missions, business processes, systems, data, and their interdependencies, expressed through governed viewpoints and metamodels. Institutionalized Department of Defense and federal frameworks (DoDAF, DM2, UAF, NAF, TOGAF and its Architecture Development Method, FEAF, and BEA)

excel at specifying WHAT to document but provide limited guidance on HOW to reason over architectural artifacts for decision support. AEDA treats Enterprise Architecture in this documentation-centric form as the starting condition (Phase 1 of the maturity trajectory) and evolves it into a computational decision stack. In the source literature, assessed Enterprise Architecture maturity correlates strongly with successful scaled AI deployment (Spearman rho = 0.88), with transformative-level maturity associated with substantially faster AI deployment timelines.

Enterprise Ontology. The formal, governed, machine-reasonable semantic backbone of the enterprise: the set of concepts, types, taxonomies, axioms, and constraints that render BEA and DoDAF DM2 content computable rather than merely descriptive. The Enterprise Ontology is built on the W3C standards stack (RDF and RDFS triples, OWL 2 for axioms and constraints, SPARQL for multi-hop query) with Description Logics providing decidable, sound, and complete reasoning suitable for explainable and auditable inference. It functions as the consistency-enforcing schema for the Knowledge Graph and the shared semantic protocol across agents. Ontology grounding is the trust-and-accuracy enforcement mechanism for every layer above it: in clinical question answering, ontology grounding raised accuracy from 37% to 98% and reduced hallucination rates from 63% to 1.7%.

Ontology Discovery. The automated extraction of candidate concepts, types, taxonomies, and non-taxonomic relations from the Data Sources layer, converting raw enterprise data into the inputs the Enterprise Ontology and Knowledge Graph consume. Core tasks are term extraction and typing, taxonomy induction, non-taxonomic relation extraction, and entity linking and disambiguation. The state of practice has evolved from linguistic and statistical methods through deep learning to LLM-plus-RAG approaches, with modern systems reporting 88-96% accuracy across benchmarks and exceeding 96% F1 on domain-specific ontology mapping. Joint, end-to-end extraction outperforms sequential pipelines (reported gains of 15-20%) by mitigating error propagation. Because discovery accuracy is imperfect, Ontology Discovery feeds candidates forward to governance for verification rather than directly into the authoritative ontology.

Ontology Governance. The first-class curation layer that reconciles and validates discovered ontology candidates before they enter the authoritative Enterprise Ontology. Its functions include ontology alignment, fusion with conflict resolution, multidimensional quality assessment, automated consistency checking (for example, with a symbolic reasoner such as HermiT), SPARQL competency-question validation, multi-agent consensus validation, and dynamic and temporal ontology evolution so the ontology tracks changes in doctrine, systems, and policy over time. Governance treats ontologies as active constraint mechanisms that enforce factual accuracy at generation time, not passive reference structures, and incorporates human-in-the-loop checkpoints because roughly 4-12% of discovery candidates require correction. The shift this layer enables is from periodic review cycles to continuous, near-real-time governance.

Enterprise Knowledge Graph. The unified reasoning substrate that organizes enterprise knowledge as explicit, structured graphs of entities, relationships, and hierarchical semantic dependencies, constrained by the Enterprise Ontology that serves as its schema. Unlike flat,

atomized document chunks ranked by semantic similarity alone, the knowledge graph captures relational structure and supports multi-hop reasoning, interpretability, and traceable evidence chains. It is implemented over graph databases (for example, Neo4j with Cypher) and traversal methods such as breadth-first search and Personalized PageRank, with subgraph-extraction techniques (Prize-Collecting Steiner Trees, eigenvector-centrality skeletons) to keep construction tractable at enterprise scale. Ontology-guided construction outperforms pure vector-based retrieval while reducing LLM usage cost; knowledge-graph reasoning combined with causal inference reached root-cause identification accuracy above 90% in quality management.

GraphRAG. Graph-based Retrieval-Augmented Generation: a retrieval and generation pattern that grounds LLM outputs in the Enterprise Knowledge Graph by traversing query-aware subgraphs rather than retrieving flat text chunks. A unified GraphRAG framework comprises five components: query processor, retriever (dense, graph, or hybrid), organizer, generator, and data source. Adaptive query routing directs simple factual queries to dense retrieval and complex multi-hop queries to graph traversal, balancing latency, accuracy, and token cost. GraphRAG surpasses vector-only RAG when questions require explicit relational reasoning, hierarchical organization, or multi-source integration (reported F1 above 0.59 versus 0.40-0.45 for vector-only on multi-hop benchmarks), while dense RAG remains more efficient for simple factual retrieval. GraphRAG reduces but does not eliminate hallucination, so it is paired with detection mechanisms such as path-reliance and semantic-alignment scoring with explicit evidence-chain attribution.

Agent. A goal-directed autonomous system that perceives environmental state, maintains memory, reasons and plans over extended horizons, and acts through external tools via iterative control loops with minimal human intervention, distinct from a stateless, prompt-driven generative model. Individual agents are structured on cognitive-architecture patterns (for example, CoALA-style modular memory and structured action spaces, or the Belief-Desire-Intention model) and bind to tools and data through standardized interfaces such as the Model Context Protocol and the ReAct reasoning-and-acting pattern combined with Agentic RAG over the GraphRAG and Knowledge Graph layers.

Agent Swarm. A coordinated collection of specialized autonomous agents that collaborate to execute architecture analysis, portfolio optimization, risk assessment, red-team evaluation, and strategic planning over the lower AEDA layers. The canonical topology is hierarchical: a supervisor or planner agent decomposes enterprise goals into subtasks routed to specialized agents (analyst, risk, optimizer, critic), supported by shared graph-structured memory (for example, three-tier insight, query, and interaction graphs) backed by the Enterprise Knowledge Graph. The swarm operates under a Control Plane managing policy, identity, scheduling, observability, and agent lifecycle, with governance, human-in-the-loop oversight, and policy compliance treated as first-class requirements. In the closed-loop configuration, three cooperating agent classes recur: orchestrating agents that coordinate specialists, learning agents that extract causal insight from each decision episode, and governance agents that monitor drift, constraint violations, and ethical boundaries and trigger human oversight. Governance maturity carries measured payoffs: at the highest maturity levels, organizations reported 94.3% lower

agent-sprawl indices, 96.4% fewer risk incidents, and 32.6% higher effective task completion than the lowest level.

Enterprise Digital Twin. A continuously updating virtual replica of the organization's processes, assets, workflows, organizational structures, and market-facing operations, distinguished from offline business intelligence by bidirectional, continuously synchronized interaction with the physical and organizational systems it represents. The Enterprise Digital Twin is the stage where AEDA transitions from representing the enterprise to simulating it as a living model. It follows a multi-tier reference architecture (data ingestion, virtual representation, analytical processing, and application or user-interface layers) with a dedicated data layer acting as a universal translator, reuses the Enterprise Ontology and Knowledge Graph as its semantic substrate, and is typically deployed in a hybrid edge-cloud, microservices or Digital-Twin-as-a-Service pattern anchored to standards such as ISO 23247 and MBSE. It enables risk-free experimentation and inverse inference, in which targets are specified and policies derived.

Simulation. The execution of experiments inside the Enterprise Digital Twin to evaluate strategic alternatives, stress-test strategies, and forecast organizational performance before any real-world action. AEDA combines simulation paradigms rather than selecting one: discrete event simulation for process and logistics flows, agent-based simulation for emergent organizational and multi-actor behavior, and physics-informed or hybrid models where governing laws are known, with continuous automated calibration to counter model drift. Simulation supports what-if and Monte-Carlo scenario analysis and provides a safe sandbox in which reinforcement-learning agents can be trained before any real-world actuation.

Optimization. The layer that derives recommended courses of action by formulating and solving constrained decision problems over simulation outputs and live enterprise state. AEDA pairs LLM and GraphRAG surfaces with formal solvers, using natural-language-to-mathematical-formulation translation so non-experts can pose optimization problems, and treating the solver as both executor and reward signal (solver-informed reinforcement learning). Methods include multi-objective Bayesian optimization for inverse inference, hierarchical and deep reinforcement learning, and risk-constrained formulations (for example, Augmented Lagrangian methods enforcing hard constraints). In the source literature, LLM-integrated optimization reduced time-to-decision from weeks to minutes by enabling natural-language formulation of optimization problems.

Decision Layer. The closed-loop tier where the outputs of the Agent Swarm, Simulation, and Optimization layers are integrated into recommended or executed decisions, with every decision, environmental response, and outcome captured and fed back to refine decision policies in real time. The Decision Layer is engineered for explicit closed-loop stability (for example, control-theoretic, PID-style mechanisms to prevent oscillation or divergence) and continuous validation to detect model drift before decision quality degrades silently. It preserves human decision authority, surfaces auditable evidence trails linking decisions to reasoning, and embeds explainability and human-in-the-loop triggers near decision boundaries. For defense and aerospace deployments it is designed for human command-and-control subordination, with recovery procedures that restore human control if autonomy fails. The associated Executive

Support tier positions architects and executives as orchestrators of AI reasoning, translating simulation and optimization outputs into explainable, natural-language narratives for leadership.

Autonomous Enterprise. The mature operating state in which knowledge graphs, ontologies, LLMs and GraphRAG, simulation and digital twins, optimization engines, and agent swarms are integrated into a cohesive closed-loop architecture where autonomous agents perceive state, reason, and execute operational changes with minimal human intervention until performance degrades or objectives change. It is the third phase of the maturity trajectory (Data-Driven, then AI-Augmented, then Continuously Optimized AI-Assisted), distinguished from a data-driven enterprise that depends on human interpretation of dashboards. The transition is governed more by organizational change, stakeholder alignment, and governance establishment than by technical implementation alone.

AEDA (Autonomous Enterprise Decision Architecture). A ten-layer reference architecture that evolves DoD enterprise architecture (BEA and DoDAF DM2) from documentation into a computational decision stack. Its layers proceed from Data Sources through Ontology Discovery, Ontology Governance, the Enterprise Ontology, the Enterprise Knowledge Graph, GraphRAG, Agent Swarms, the Enterprise Digital Twin, Simulation, and Optimization to the Decision Layer and Executive Support. AEDA's organizing thesis is that institutionalized EA frameworks are documentation-centric and stop short of computational decision support, and that ontology grounding, knowledge graphs, GraphRAG, agentic systems, digital twins, simulation, and optimization can close that gap, shifting enterprise governance from periodic review cycles to continuous, near-real-time decision support while preserving human authority, explainability, and auditable evidence trails.

The AEDA Ten-Layer Reference Architecture

AEDA reorganizes DoD enterprise architecture from a documentation discipline into a computational decision stack. Institutionalized frameworks (DoDAF, DM2, UAF, NAF, TOGAF, FEAF, BEA) excel at specifying what to document through viewpoints, metamodels, and SysML traceability, but they provide limited guidance on how to reason over architectural artifacts for decision support [LR01]. This gap, from periodic review cycles to continuous near-real-time governance, is where the AEDA layers create value. Each layer below is specified by purpose, inputs, outputs, candidate technologies and standards, and the grounding evidence from the supporting reviews. The layers form a closed loop in which autonomous components perceive enterprise state, reason, and execute or recommend operational changes, distinguished from data-driven enterprises that depend on human interpretation of dashboards [LR10].

Layer 1: Ontology Discovery

Purpose. Convert raw enterprise data sources into candidate concepts, types, taxonomies, and non-taxonomic relations that downstream layers can govern and formalize, replacing manual, expert-driven documentation with a computational pipeline.

Inputs. Unstructured and semi-structured enterprise content: DoDAF DM2 artifacts, BEA structures, doctrine, system descriptions, and authoritative data sources.

Outputs. Candidate terms with type assignments, induced taxonomies, extracted relations, and entity links awaiting governance review.

Candidate technologies and standards. BERT-class models for high-throughput term typing and named-entity recognition; LLMs (GPT-3, GPT-4, fine-tuned variants) with retrieval-augmented generation for harder, low-frequency, or cross-domain terminology; BiLSTM-CRF sequence labeling; joint entity-relation extraction; RDF, OWL, SPARQL targets [LR11].

Grounding evidence. LLM-plus-RAG ontology discovery now reaches 88 to 96 percent accuracy across multiple benchmarks, with domain-specific fine-tuning exceeding 96 percent F1 on medical ontology mapping. Joint, end-to-end extraction yields 15 to 20 percent gains over two-phase pipelines and avoids error propagation. Space and cislunar operations are named live application frontiers, anchoring AEDA's defense relevance [LR11].

Layer 2: Ontology Governance

Purpose. Curate, reconcile, and validate discovered candidates before they enter the authoritative enterprise ontology, treating constraints as active enforcement mechanisms rather than passive references.

Inputs. Candidate concepts and relations from Ontology Discovery; existing domain ontologies; quality and policy rules.

Outputs. Aligned, deduplicated, consensus-validated ontology fragments with conflict resolution and version history; promotion or rejection decisions with verification trails.

Candidate technologies and standards. Ontology alignment and fusion; multi-agent consensus validation; collaborative editors (Protege, WebProtege, VocBench); Uschold-Gruninger methodology with Guarino classifications; automated consistency checking via the HermiT reasoner; SPARQL competency questions; temporal knowledge graphs for dynamic evolution; risk-tiered governance with named accountability [LR02, LR09, LR11].

Grounding evidence. Discovery accuracy of 88 to 96 percent means roughly 4 to 12 percent of candidates are wrong, so governance must include verification and explainability before promotion [LR11]. Organizations combining explainable AI with empowered ethics boards experienced 48 percent fewer instances of bias and regulatory violations, and 35 percent fewer

regulatory investigations when third-party audits were employed; advisory-only boards showed limited impact [LR09].

Layer 3: Enterprise Ontology

Purpose. Provide the formal, governed, machine-reasonable semantic backbone that converts BEA/DoDAF DM2 from descriptive documentation into sound, complete, auditable inference, and serves as the trust-and-accuracy enforcement layer for everything above it.

Inputs. Governed ontology fragments; standardized upper and vocabulary ontologies; mappings to legacy relational sources.

Outputs. A consistency-enforcing schema; reasoning services (satisfiability checking, entailment); semantic access to wrapped data sources.

Candidate technologies and standards. RDF/RDFS triples; OWL 2 DL for full axiomatic expressiveness and OWL 2 QL for high-volume data access; SPARQL for multi-hop query; Description Logics for decidable reasoning; HermiT for consistency; ontology-mediated query answering with query rewriting; reuse of CIDOC CRM, SKOS, and domain standards such as SNOMED CT and ICD-11 for interoperability; incremental adoption from RDF graph to RDFS vocabulary to OWL axioms [LR02].

Grounding evidence. In clinical question answering, ontology grounding achieved 98 percent accuracy versus 37 percent for baseline ChatGPT-4 and reduced hallucination from 63 percent to 1.7 percent, the strongest empirical case for ontology grounding as an accuracy and hallucination-control mechanism [LR02]. Semantic-web-driven EA frameworks reduced documentation inconsistencies by up to 24 percent and improved query efficiency by approximately 50 percent [LR02].

Layer 4: Knowledge Graph

Purpose. Materialize the enterprise as an explicit structured graph of entities, relationships, and hierarchical dependencies that supports queryable, explainable reasoning rather than flat-text retrieval.

Inputs. The enterprise ontology as formal schema; extracted entities and relations; multimodal architecture content including diagrams and illustrations.

Outputs. A populated, versioned graph supporting traversal, embedding lookup, subgraph extraction, and impact analysis.

Candidate technologies and standards. Neo4j with Cypher; Personalized PageRank and BFS traversal; Prize-Collecting Steiner Tree subgraph extraction and eigenvector-centrality skeleton construction; translation-based embeddings (TransE, TransH, TransR) and hyperbolic

embeddings for hierarchy; temporal knowledge graph embeddings; RAKG-style retrieval-augmented construction; graph neural networks for impact prediction [LR02, LR03].

Grounding evidence. Selective skeleton construction from the top 20 percent of chunks achieved 10x cost reduction versus exhaustive KG construction while improving generation quality by 32.4 percent and retrieval coverage by 92.4 percent [LR03]. RAKG reached 95.91 percent accuracy on the MINE dataset, a 6.2 percentage point improvement over the GraphRAG baseline [LR03]. Knowledge graph plus causal inference achieved root-cause identification accuracy above 90 percent in quality management [LR10].

Layer 5: GraphRAG

Purpose. Retrieve and ground generation over the knowledge graph, routing queries adaptively so that simple factual lookups use dense retrieval and multi-hop architecture questions use graph traversal.

Inputs. Natural-language and structured queries; the knowledge graph; the enterprise ontology as constraint provider.

Outputs. Query-relevant subgraphs, grounded answers with explicit evidence chains, and hallucination-detection signals.

Candidate technologies and standards. The five-component framework of query processor, retriever, organizer, generator, and data source; EA-GraphRAG adaptive routing on continuous complexity scores; hybrid retrievers fusing semantic node-level and structural path-level retrieval (HybRAG); neural-symbolic dual-indexing; Graph Grounding and Alignment using path reliance degree and semantic alignment score for hallucination detection; benchmark harness using HotpotQA, MuSiQue, 2WikiMultiHopQA with Recall@k, MRR, NDCG, EM, F1 [LR03].

Grounding evidence. GraphRAG systems achieve F1 scores exceeding 0.59 versus 0.40 to 0.45 for vector-only approaches on multi-hop benchmarks [LR01]. EA-GraphRAG routes simple queries to dense retrieval at roughly 50ms and complex queries to graph traversal at roughly 200ms, reporting token-cost reductions up to 90.71 percent [LR03]. RAG-enhanced LLMs reached 54 percent accuracy on enterprise SQL question answering versus 16 percent for zero-shot queries on raw databases [LR10].

Layer 6: Agent Swarm

Purpose. Coordinate specialized autonomous agents to execute architecture analysis, portfolio optimization, risk assessment, red-team evaluation, and strategic planning over the lower layers.

Inputs. GraphRAG-grounded retrieval; the knowledge graph as shared memory; tools and legacy APIs; mission goals.

Outputs. Decomposed task results, analyses, recommendations, and collaboration trajectories, with Overall System Confidence scores routing outputs to automated action, human escalation, or continuous learning.

Candidate technologies and standards. Cognitive Architectures for Language Agents (CoALA) and Belief-Desire-Intention grounding; hierarchical supervisor topology; G-Memory three-tier graph memory (insight, query, interaction); ReAct and Agentic RAG; Model Context Protocol and Agent-Ready Architecture for tool access; the Enterprise Agentic Architecture Framework with a central Control Plane; the Agentic AI Governance Maturity Model over 12 domains grounded in NIST AI RMF and ISO/IEC 42001; TRACE and CAMCO for policy-compliant orchestration; red-team harness using GAIA, AssistantBench, WebArena, AJAR [LR01, LR04].

Grounding evidence. Multi-layer agentic architectures demonstrate 3 to 10x workflow acceleration with 60 to 80 percent reductions in mean time to resolution [LR04]. Governance Level 4-5 organizations achieve 94.3 percent lower sprawl indices, 96.4 percent fewer risk incidents, and 32.6 percent higher effective task completion rates [LR04, LR10]. A production agentic governance implementation reported Governance Efficiency of 0.961 and Resilience Index of 0.993 [LR01]. ARA wrapping achieved 90 percent semantic discovery precision and reduced multi-tool chain latency by 60.5 percent [LR04].

Layer 7: Enterprise Digital Twin

Purpose. Transition from representing the enterprise to simulating it as a continuously synchronized virtual replica of processes, workflows, organizational structures, and decision-makers, distinct from offline business intelligence by its bidirectional, real-time interaction.

Inputs. Knowledge graph and ontology as semantic substrate; live telemetry; heterogeneous DoD enterprise data streams.

Outputs. A continuously updated virtual representation supporting monitoring, what-if experimentation, and forecasting.

Candidate technologies and standards. A four-tier reference architecture (data ingestion, virtual representation, analytical processing, application/UI) with a dedicated data layer acting as universal translator; multi-ontology networks (OWL, SOSA); hybrid edge-cloud microservices and Digital-Twin-as-a-Service; the ISO 23247 reference architecture; MBSE for formal requirements and interface capture; SysML Opaque Actions to pull telemetry and update model properties; LSTM and ensemble forecasters; isolation-forest and autoencoder anomaly detection; Intelligent Acting Digital Twins for autonomous control [LR01, LR05].

Grounding evidence. A cloud-edge collaborative twin for renewable energy achieved a three-order-of-magnitude simulation-efficiency improvement (10 days to 182.6 seconds), modeling error below 0.6 percent, and a three-fold reduction in unplanned outages [LR01]. A hospitality twin predicted occupancy with R-squared of 0.86 and energy within 8.3 percent accuracy;

financial twins achieved 30 to 40 percent improvements in customer experience metrics [LR05]. Digital twins drove a 2.1-day reduction in average delivery time and nearly 20 percent cost reduction [LR10].

Layer 8: Simulation

Purpose. Turn the static digital twin into executable, time-advancing behavior, generating predictions across aggregate flows, discrete resource contention, and heterogeneous decision-making actors simultaneously.

Inputs. The digital twin's structure and parameters; MBSE/SysML models as authoritative source of truth; scenario definitions and uncertainty distributions.

Outputs. Time-resolved behavioral predictions, scenario sweeps, and stress-test results feeding the optimization and decision layers.

Candidate technologies and standards. The DEVS formalism for verifiable, composable specification, with xDEVS for cloud-distributed parallel/distributed execution; hybrid simulation combining discrete-event simulation for queues and scheduling, agent-based modeling for emergent organizational behavior, and system dynamics for aggregate feedback; SysML-to-IMPRINT and Capella/Arcadia for architecture-to-simulation traceability; Monte Carlo with multilevel and multifidelity variance reduction; sparse polynomial chaos expansion with Latin Hypercube Sampling; Sobol' global sensitivity analysis; process mining; OPC-UA, MQTT, IEC 61499 protocols [LR05, LR06, LR07].

Grounding evidence. xDEVS achieved a 15.95x speedup on distributed systems while preserving model reusability [LR06]. Discrete-event models reached less than 5 percent deviation from actual plant data [LR06]. Sparse PCE with Latin Hypercube Sampling achieved 40 to 60 percent computational time reductions while improving prediction accuracy by 15 to 25 percent [LR07]. Real-time digital-twin intrusion detection reached a 96.3 percent attack-detection F1-score with sub-500-millisecond latency [LR06].

Layer 9: Optimization

Purpose. Convert digital twin and simulation outputs into ranked investment and capability decisions under resource constraints, optimizing for mission outcomes rather than asset inventories.

Inputs. Simulation outputs with calibrated uncertainty; EVM telemetry; capability-based planning structures; budget and sequencing constraints.

Outputs. Continuously re-optimized portfolio recommendations, redundancy flags, and Pareto-optimal trade sets with explanations.

Candidate technologies and standards. Capability-based planning expressed as multi-criteria selection (AHP plus mixed-integer programming); the multi-period precedence-constrained knapsack for stakeholder-interest maximization under periodic budgets; a defense capability metric combining investment-portfolio measures with technological-aging signals; EVM (PV, EV, AC, CPI, SPI) with uncertainty-aware GEDM and ZEVM variants; reinforcement learning and multi-objective RL (DDPG, PPO, A2C) for dynamic rebalancing; genetic-algorithm and Monte Carlo simheuristics; CVaR risk constraints; semantic-similarity redundancy detection; ML-plus-MILP hybrids validated on space logistics [LR07, LR08].

Grounding evidence. AI-driven strategies achieve 15 to 20 percent reductions in portfolio volatility and 30 percent faster rebalancing [LR08]. SLA-aware multi-objective RL achieved 67.2 percent reduction in training time for deadline-critical jobs and 68.8 percent cost reduction for budget-constrained workloads [LR08]. Scenario-based two-stage stochastic optimization achieved a 0.9 percent cost reduction with improved reliability over deterministic dispatch [LR07].

Layer 10: Decision Layer and Executive Support

Purpose. Render the stack's outputs explainable, auditable, and defensible to government accountability standards, preserving human decision authority and calibrated trust, and close the loop back to the data and ontology layers.

Inputs. Optimization recommendations; simulation and agent outputs; provenance and lineage records; calibrated uncertainty signals.

Outputs. Explained recommendations with evidence trails, risk-tiered gating, assurance arguments, and value-of-information guidance on whether to act or gather more data.

Candidate technologies and standards. Local and global explainability (LIME, SHAP, LRP, Grad-CAM, counterfactuals); provenance substrates including immutable lineage graphs and AI-Receipts-style ledgers; risk-tiered governance with named accountability and meaningful human oversight; assurance cases using Overarching Properties (Intent, Correctness, Innocuity), PRAISE, and continuous-assurance with PRISM and RoboChart; Bayesian decision theory and pre-posterior value-of-information analysis; a Technical-Regulatory Correspondence Matrix mapping EU AI Act, GDPR, NIST AI RMF, ISO/IEC 42001 to evidence artifacts [LR07, LR09, LR10].

Grounding evidence. A hybrid LRP-plus-SHAP framework reached 94.3 percent accuracy with a 92.1 percent trustworthiness index, showing performance and explainability can coexist; an AI-Receipts ledger achieved 94.2 percent attribution accuracy with under 340ms overhead per commit [LR09]. Public-sector predictive analytics improved budget-variance accuracy from plus/minus 15 percent to plus/minus 4 percent and cut crime response lead times from 14 days to under three hours [LR09]. Across the stack, organizations report that technical implementation consumes 20 percent of effort while organizational change management, stakeholder alignment,

and governance establishment consume 80 percent, so the Decision Layer and its governance overlay are the determinants of whether AEDA reaches production [LR10].

Closed-Loop Enterprise Control

AEDA version 4 makes a single structural commitment explicit: the ten layers are not a pipeline that ends at a recommendation, they are a control loop that begins and ends in enterprise state. A feed-forward knowledge system answers questions. A control system closes the loop between answering a question, acting on the answer, observing what the action produced, and updating the model that produced the answer. The literature reviewed for AEDA documents both halves of this transition. Enterprise digital twins are distinguished from conventional business intelligence precisely by their bidirectional interaction capability: they receive continuous data streams and simultaneously provide insights that influence the systems they model in real time (LR05). The autonomous-enterprise literature names the same property as the defining feature of its third and final maturity phase, where systems perceive environmental states, reason, optimize across objectives, execute operational changes, and learn from outcomes to refine future decision policies, all without human intervention until performance degrades or objectives change (LR10). AEDA v4 adopts that closed-loop posture as core architecture and assigns each of its layers a control-theoretic role.

The Enterprise World Model and the Control Loop

The loop holds enterprise state in the Enterprise Digital Twin, recast in v4 as the enterprise world model: a continuously synchronized virtual replica of organizational processes, assets, and system-of-system dynamics rather than a passive dashboard (LR05). State flows forward through the stack. The Knowledge Graph supplies the semantic substrate, modeling entities and relationships as graph structures so that raw operational data becomes machine-interpretable knowledge; LR10 reports knowledge-graph-plus-causal-inference frameworks achieving root-cause identification accuracy above 90 percent and knowledge-graph risk systems delivering a 35 percent improvement in decision-making efficiency through detection of hidden propagation paths. GraphRAG grounds language-model reasoning in that graph, reducing hallucination and lifting enterprise question-answering accuracy from 16 percent for zero-shot queries on raw databases to 54 percent when retrieval is graph-grounded (LR10). The Agent Swarm reads grounded state and proposes candidate interventions; multi-agent frameworks coordinate specialized forecasting, risk, and optimization agents while learning agents capture outcomes and governance agents watch for drift and constraint violation (LR10).

Candidate interventions are not executed on the live enterprise. They are first run against the world model. The Enterprise Digital Twin instantiates each candidate as a scenario, and three analysis stages quantify its consequences. Discrete-Event Simulation executes the candidate as a sequence of timed events over resources, schedules, and dependencies; DEVS-based models validate to less than 5 percent deviation from actual plant data, and parallel discrete-event

execution has achieved speedups of 15.95 times, making it tractable to simulate many candidates inside one control cycle (LR06). The Monte Carlo Engine propagates uncertainty through those simulations, characterizing the distribution of outcomes rather than a single point estimate and computing risk measures such as Conditional Value-at-Risk; multilevel and multifidelity methods plus sparse polynomial chaos cut computational cost by 40 to 94 percent against standard Monte Carlo while preserving accuracy (LR07). The Optimization Engine searches the candidate space under resource constraints, using the simulated and uncertainty-quantified outcomes as its objective surface; reinforcement-learning and multi-objective formulations in LR08 balance cost, schedule, performance, and mission value, with reported gains including 51.5 percent improvement in cost-effectiveness ratio over rule-based baselines and 95.2 percent resource utilization.

Policy Engine and Decision Layer: A Deliberate Separation

AEDA v4 separates the Policy Engine from the Decision Layer because they answer different questions. The Policy Engine is the constraint and governance stage. It applies guardrails to candidate decisions: hard constraints (regulatory limits, safety boundaries, mission rules of engagement), fairness and bias checks, and risk-appetite thresholds. The governance literature is explicit that hard constraints and regulatory requirements are precisely what current language-model and optimization stacks struggle to incorporate (LR10), which is why AEDA isolates them in a dedicated stage rather than trusting the optimizer to honor them. The Policy Engine filters and annotates candidates, rejecting those that violate constraints and attaching provenance and compliance evidence to those that survive, consistent with the compliance-by-design pattern that routes explainability outputs into audit-ready evidence (LR09). The Decision Layer then performs selection and recommendation over the surviving, constraint-satisfying candidates: it ranks them against mission objectives and presents a recommendation, with explanation, to Executive Support. Selection is downstream of governance, never a substitute for it. This is the architectural expression of the LR09 finding that ethics and governance controls deliver value only when integrated into core decision-making rather than left advisory; organizations pairing explainable AI with empowered controls showed 48 percent fewer bias and regulatory-violation instances.

Actuation, Observation, and the Learning Loop

Actuation is the step that converts a recommendation into an Enterprise Action. The reviewed digital twins already demonstrate the mechanism: intelligent acting digital twins move beyond passive monitoring to autonomously influence physical counterparts through control commands based on optimization objectives and real-time conditions (LR05). In AEDA v4, an approved Decision Layer recommendation is issued as a parameterized action to the enterprise systems of record, with the degree of autonomy gated by the Policy Engine; high-consequence actions route to human authority, lower-consequence actions execute directly, matching the tiered, risk-based

oversight model in LR09. Every action carries its provenance identifier so the downstream effect can be traced to the decision and the evidence that justified it.

Observation closes the forward path. Enterprise Actions produce Observed Outcomes, which are measured and re-ingested through the same real-time data integration pipelines that keep the world model synchronized within milliseconds to seconds of operational reality (LR05). The control system compares observed outcomes against the distribution the Monte Carlo Engine predicted, turning each action into a labeled experiment. This is the Learning Loop. Learning agents extract which decision factors drove which outcomes and refine decision policies (LR10); the digital twin recalibrates its parameters automatically as new operational data arrives, maintaining accuracy as the enterprise drifts (LR05). Refined knowledge flows back into the Knowledge Graph and the Enterprise Ontology, so the semantic model the loop reasons over is itself updated by the loop's results.

Two failure modes govern the design of this update. The first is drift: patterns learned in training diverge from operational reality, and naive feedback can oscillate or diverge; LR10 prescribes continuous validation and control-theoretic stabilization, including PID-style and stability-aware mechanisms, while LR05 supplies machine-learning-and-control-chart validation to detect parameter drift before it corrupts decisions. The second is provenance integrity: updates to the graph and ontology must preserve a tamper-evident record of origin. AEDA v4 binds the Learning Loop to W3C PROV-O so that every ontology and graph revision records its source data, model choices, and the action that motivated it, consistent with the provenance-and-lineage controls LR09 documents (AI Fairness Provenance Records, the AI Product Passport, blockchain-verified audit trails reducing fraudulent activity by more than 50 percent). Learning is allowed to change the model; it is not allowed to erase how the model came to be.

Why This Closes the Gap

A feed-forward knowledge system stops at the Decision Layer: it can recommend, but it cannot observe whether its recommendation worked, and it cannot improve from the result. AEDA v4 closes that gap by routing Observed Outcomes back through a provenance-preserving Learning Loop into the Knowledge Graph and Enterprise Ontology, making the architecture self-correcting rather than merely advisory. The Enterprise Digital Twin holds state; the simulation, Monte Carlo, and optimization stages serve as the predictive model the controller uses to choose actions before committing them; the Policy Engine is the safety and governance controller; the Decision Layer is the selection law; actuation and observation are the plant interface; and the Learning Loop is the adaptive identification that keeps the model faithful. That is the structure of a control system, not a report generator.

Implementation maturity. The closed loop is core design intent, and each stage exists in proven form today: validated digital twins, sub-5-percent-deviation discrete-event models, multilevel Monte Carlo, reinforcement-learning optimization, and PROV-O provenance are

all demonstrated in the reviewed literature. What a near-term AEDA prototype delivers is narrower than the full autonomous loop. Sample efficiency, the reward-specification problem, and brittle hard-constraint handling remain open (LR10), so early deployments run the loop with a human in the actuation path for high-consequence decisions, close the Learning Loop on a scheduled cadence rather than continuously, and scope autonomy to bounded, well-instrumented decision domains where outcome measurement is clean. The architecture is the control system; the first prototype is that control system operating under supervised, rate-limited adaptation.

Enterprise World Models

The entity-relationship ontology produced by the lower AEDA layers (Ontology Discovery through Enterprise Ontology, persisted as a DM2-conformant and OWL-expressed Knowledge Graph) answers the question "what exists, and how is it related." It is a static, declarative vocabulary: classes, properties, individuals, and the logical constraints that bind them. A closed-loop control system needs more. It needs to answer "what will happen, and what can we make happen." That is the role of the Enterprise World Model: a state-space abstraction layered on top of the ontology that represents the enterprise not as a graph of things but as a dynamical system that occupies states, transitions between them under events and decisions, and traces trajectories through time. In AEDA v4 the Enterprise World Model is a core abstraction, not a future-state aspiration. It is what converts a knowledge system into a predictive control system.

The state-space primitives and their relation to the ontology

The world model is built from six primitives. A **State** is a complete, point-in-time valuation of the enterprise variables that matter for decision: budget posture, program schedule status, workforce levels, capability maturity, risk exposure, inventory, and similar measures. A **Transition** is a function that maps a state, an event, and a chosen action to a successor state (or, more honestly, to a distribution over successor states). An **Event** is an exogenous or endogenous occurrence that can trigger transitions: a continuing resolution, a supplier disruption, a milestone slip, a demand shift. A **Policy** is a mapping from states to actions, the decision rule the enterprise follows. A **Goal** is an objective defined over states or trajectories, the thing the enterprise is trying to maximize or reach. A **Constraint** is a feasibility bound that partitions states and actions into admissible and inadmissible, encoding statutory, fiscal, safety, and doctrinal limits.

These primitives complement the ontology rather than replace it. The ontology supplies the vocabulary and structure: the entity types whose attributes become state variables, the relationships that determine which transitions are even coherent, and the constraint axioms that seed the feasibility bounds. The world model supplies the dynamics: which states are reachable from the current state, under which events, at what likelihood, and toward which goals. Where the DM2/OWL layer says "Program A funds Capability B, which depends on

Resource C," the world model says "if Resource C is cut, Program A's schedule state transitions to at-risk with probability p, and Capability B's maturity becomes unreachable within the horizon." Structure constrains dynamics; dynamics animate structure. The two are tightly coupled, and the Knowledge Graph remains the authoritative source of truth for the entities the state vector is assembled from, exactly as SysML serves as the authoritative source while a discrete-event tool computes the behavioral predictions described in LR06.

Implementation maturity. The design intent is a continuously synchronized, full-coverage state vector. A near-term AEDA prototype delivers a partial, manually scoped state vector for a single decision domain (for example a program portfolio), with transition probabilities estimated from historical records rather than learned online. The primitive set and the ontology binding are stable and implementable now; complete enterprise coverage and automated state extraction from heterogeneous source systems are the harder, later work that LR10 identifies as the semantic-integration barrier.

What the world model underpins

The state-space view is the shared substrate beneath the four upper AEDA layers. The **Enterprise Digital Twin** is the world model kept continuously synchronized with operational data, so the current state is observed rather than assumed; LR05 and LR10 describe exactly this evolution of digital twins from visualization tools into decision infrastructure that conducts what-if analysis against a live state. The **simulation layer** exercises the transition function forward: discrete-event simulation advances the state through an event calendar, agent-based and system-dynamics methods supply transitions driven by interacting actors and feedback loops (LR06), Monte Carlo sampling propagates uncertainty across many trajectories to produce distributions rather than point predictions, and Bayesian updating refines the transition probabilities as observed outcomes arrive (LR07). The **optimization layer** searches over policies and action sequences to maximize goals subject to constraints, the precedence-constrained, budget-bounded portfolio selection of LR08 being a direct expression of goals-as-objectives and constraints-as-feasibility-bounds over the state space. **Reinforcement learning** is the world model in its most explicit form: an agent learns a policy by acting on states, observing transitions and rewards, and improving the mapping, which is precisely the closed-loop, learn-from-outcome pattern LR10 names as the defining characteristic of the continuously optimized enterprise. Each upper layer reads and writes the same state-space representation, which is what lets the loop close.

Implementation maturity. In design intent these four layers operate on one shared world model. A near-term prototype more often runs them as coordinated but loosely coupled modules: a digital twin feeding a Monte Carlo simulation whose outputs seed an optimization run, with reinforcement learning applied only in narrow, well-bounded sub-problems where the reward is cleanly specified. LR10's cautions on sample efficiency and reward specification apply directly and bound how autonomous the RL component can responsibly be.

Counterfactual and what-if reasoning: the budget-shift example

Because the world model encodes transitions and not merely relationships, it supports counterfactual reasoning: it can evaluate trajectories that have not occurred. Consider the question "what happens if budget allocation changes by ten percent." The current state is read from the digital twin. The ten-percent reallocation is applied as an intervention on the budget state variables, and the transition function is rolled forward under simulation. Three answers emerge that the ontology alone cannot produce. First, **which state transitions become likely**: programs whose funding falls below threshold transition toward schedule-slip or descope states, and Monte Carlo sampling assigns each a probability rather than a binary verdict (LR07). Second, **what risks emerge**: the propagation surfaces second-order effects, a funding cut cascading into a capability-maturity shortfall or a workforce-attrition state, the kind of hidden propagation path LR10 attributes to graph-plus-dynamics reasoning. Third, **which enterprise trajectories become reachable**: some goal states that were reachable under the baseline allocation fall outside the feasible set, while others open up, and the optimization layer can report the best attainable trajectory under the new constraint envelope. The same machinery answers the inverse, LR10's inverse-inference pattern: specify a desired goal state and let the model identify the allocations whose transitions can reach it.

Implementation maturity. The design intent is real-time counterfactual evaluation across the full enterprise state. A near-term prototype answers a scoped version of the budget question over a bounded set of state variables, with transition probabilities carrying quantified but non-trivial uncertainty, and presents results as a distribution of outcomes with explicit confidence rather than a single forecast.

Formal framing for implementation

A MITRE engineer can implement against the following. Let the state space be S, with a current state s S assembled from ontology-typed entity attributes. Let A be the action set and E the event set. The transition function is T: S × A × E → Δ(S), mapping a state, action, and event to a probability distribution over successor states (deterministic transitions are the degenerate case where the distribution is a point mass). A policy is : S → A, a mapping from observed state to chosen action. A goal is an objective functional J over states or trajectories, J: S → or defined over the trajectory = (s , a , s , a , …), to be maximized or driven toward a target set. Constraints are predicates g(s, a) 0 defining the admissible region; is feasible only where its actions satisfy g for the states it visits. Simulation evaluates T forward under a fixed to estimate the trajectory distribution; Monte Carlo estimates expectations and tail risks over that distribution; Bayesian updating revises the parameters of T from observed (s, a, s) tuples; optimization searches over to maximize J subject to g; reinforcement learning approximates the optimal by interacting with T. This is a standard stochastic control formalism, and stating AEDA in it is precisely what makes the architecture a closed-loop enterprise control system

rather than only a knowledge system: the lower layers tell the enterprise what is true, and the world model lets it reason about what to do and predict the consequences before acting.

Enterprise Ontology Strategy and Continuous Ontology Discovery

The Enterprise Ontology layer is where the Autonomous Enterprise Decision Architecture stops describing the enterprise and begins reasoning over it. Beneath the descriptive products of BEA and DoDAF DM2 sits a question those products do not answer on their own: what are the concepts, what are their formal relationships, and what inferences follow from them. AEDA answers that question with a layered ontology stack and a discovery-and-governance pipeline that builds and maintains the stack computationally rather than by hand.

The Layered Ontology Stack

AEDA organizes formal semantics as a stack that descends from the most general commitments to the most specific operational detail. Each layer constrains and gives meaning to the layer below it.

- **Foundation ontologies** supply domain-neutral categories and the formal commitments every layer inherits: what counts as an object, a process, a role, a participation, a temporal interval. Upper ontologies such as BFO, DOLCE, and SUMO occupy this layer. They fix the high-level distinctions that keep lower layers logically coherent and interoperable.
- **Enterprise ontologies** specialize the foundation categories into the structures of the DoD enterprise. This is where DM2, UAF, and BEA are recast as formal, machine-reasonable artifacts rather than documentation. The descriptive backbone becomes a semantic backbone.
- **Domain ontologies** narrow the enterprise vocabulary into the specialized terminology of a functional area, reusing established standards where they exist so that meaning is shared across organizational and system boundaries rather than reinvented in isolation.
- **Mission, program, and operational ontologies** carry the specialization the rest of the way down, binding the domain vocabulary to specific missions, programs, and operational activities so that the concepts the architecture reasons over correspond to the work the enterprise actually performs.

The technical substance of these layers rests on the W3C standards stack. Concepts and relationships are expressed as RDF and RDFS triples, axioms and constraints as OWL 2 (OWL 2 DL where full axiomatic expressiveness is required, OWL 2 QL where high-volume data access dominates), and multi-hop query, filtering, and aggregation as SPARQL. Description Logics provide the formal foundation underneath OWL: decidable fragments of first-order logic that balance expressive power against computational tractability and support sound, complete, well-defined reasoning such as satisfiability checking and entailment. This is the property that makes inference auditable. Adoption can be incremental, beginning with basic

RDF triple graphs, adding RDFS class and property vocabulary, then advancing to OWL 2 profiles as organizational capability matures. For existing DoD authoritative data sources, ontology-mediated query answering with query rewriting wraps relational and legacy databases in a semantic access layer, preserving data currency without full materialization as RDF.

The case for building this stack on formal semantics rather than ad hoc schemas is empirical. In clinical question answering, ontology-grounded knowledge graphs achieved 98 percent accuracy versus 37 percent for baseline ChatGPT-4, and reduced hallucination rates from 63 percent to 1.7 percent (Ali, Taha, Morsey 2026). Semantic-web-driven enterprise architecture frameworks reduced documentation inconsistencies by up to 24 percent and improved query efficiency by approximately 50 percent in multi-school curriculum-governance case studies (Silalahi, Indrajit, Mantoro 2025). The ontology is therefore not optional metadata. It is the trust-and-accuracy enforcement mechanism for every layer above it.

From Manual Construction to Continuous Discovery

The central argument of this section is that manual ontology construction is giving way to continuous, LLM-assisted ontology discovery and governance. The traditional path to an enterprise ontology runs through expert elicitation, collaborative editing in tools such as Protege, and methodologies such as Uschold-Gruninger paired with Guarino's formal classifications. That path remains valid, but it does not scale to an enterprise the size of the DoD, and it produces documentation that ages faster than experts can revise it. The evidence in LR11 indicates that automated discovery is now performant enough to replace expert-driven documentation with a computational pipeline.

The performance figures are the basis for that claim. Modern LLMs combined with retrieval-augmented generation and fine-tuning achieve F1-scores exceeding 90 percent on multiple ontology learning benchmarks. Term extraction and typing reaches 86 to 90 percent accuracy with BERT-based models and 88 to 94 percent with LLM-based approaches. Domain-specific LLM fine-tuning achieves F1-scores exceeding 96 percent on medical ontology mapping tasks. Joint entity-relation extraction, which trains extraction tasks together with shared representations rather than running them in sequence, yields 15 to 20 percent performance improvements over pipeline methods and reaches 85 to 93 percent precision and F1 on domain-specific corpora. Across multiple benchmarks the field reports modern systems achieving 88 to 96 percent accuracy.

Two design consequences follow directly. First, discovery should be built as a joint, multi-task extraction pipeline rather than a sequential NER then relation-extraction then entity-linking chain, because joint optimization delivers the 15 to 20 percent gain and avoids the error propagation that plagues two-phase pipelines. Second, the LLMs should be paired with RAG against the existing enterprise ontology so that discovery is ontology-aware and respects DM2 and BEA structures rather than hallucinating free-form concepts. RAG-grounded extraction

is what drives the 90 percent and 96 percent figures; ungrounded generation does not reach them.

Discovery accuracy of 88 to 96 percent has a second implication that defines the boundary between discovery and governance: roughly 4 to 12 percent of discovered candidates are wrong. Governance therefore cannot be an afterthought. It is a first-class layer that reconciles independently discovered, heterogeneous DoD enterprise ontologies through ontology alignment, fusion with conflict resolution, and multi-agent consensus validation before any candidate enters the authoritative ontology. Human-in-the-loop verification checkpoints, automated consistency checking with reasoners such as HermiT, and SPARQL competency questions provide the curation that promotion to the enterprise ontology requires. Temporal and dynamic ontology evolution, implemented through temporal knowledge graphs, keeps the ontology tracking how doctrine, systems, and policy change over time, which is what the downstream Simulation and Decision layers depend on.

Ontology-Discovery Taxonomy

LR11 organizes ontology discovery into a set of core learning tasks. These are the units of work the Discovery layer performs to convert the Data Sources layer into candidate concepts, taxonomies, and relations:

- **Term extraction and typing** identifies candidate terms in source text and assigns them types. BERT-class models handle high-throughput typing at 86 to 90 percent accuracy; LLM few-shot and fine-tuned approaches reach 88 to 94 percent and are best reserved for harder, low-frequency, or cross-domain terminology.
- **Taxonomy discovery** induces the hierarchical is-a structure that organizes terms into a class hierarchy, including graph-based optimal-branching methods over weighted hypernym graphs that learn concepts and relations from scratch.
- **Non-taxonomic relation extraction** identifies relationships beyond subsumption, including the n-ary relations captured through semantic role labeling and frame semantics.
- **Entity linking and disambiguation** resolves mentions to the correct ontology entities, using techniques such as personalized PageRank combined with semantic similarity.

The reported performance across these tasks supports cost-aware tiering of the discovery stack: route high-volume typing to BERT-class models and reserve the more expensive LLM calls for the harder cases, rather than applying the most capable model uniformly.

Method Evolution

The methodological history of the field, as LR11 stages it, is what makes the shift to continuous discovery credible rather than speculative. The evolution proceeds in distinct phases:

1. **Linguistic and NLP-dominated early work** (the early 2000s onward) relied on rule-based and pattern-based extraction and statistical term-weighting measures such as TF-IDF, C-value, and NTF.
2. **Statistical and machine-learning methods** introduced LDA topic modeling, association rule mining, and hierarchical clustering to learn structure from corpora.
3. **The deep learning revolution**, circa 2015 to 2018, brought RNN, LSTM, and CNN architectures, BiLSTM-CRF sequence labeling, and then BERT and transformer variants, raising term-typing accuracy into the high 80s.
4. **The LLM paradigm shift**, since 2023, introduced GPT-3, GPT-4, and domain-specific fine-tuned variants that enable zero-shot and few-shot discovery and, combined with RAG, push benchmark F1-scores past 90 percent.

The frontier directions named in LR11 extend this trajectory toward trustworthiness and domain relevance: LLM-driven knowledge graph construction with prompt engineering and ontology-alignment verification; multi-agent consensus validation and agent-based ontology curation; and physics-regularized knowledge graphs that incorporate domain constraints and causal reasoning to produce interpretable, trustworthy representations. The neuro-symbolic pattern, in which neural components extract features and symbolic layers encode the axioms and rules the discovered ontology must satisfy, is the architectural form this trustworthiness takes. LR11 names space as an emerging frontier, with ontology discovery efforts targeting NASA, ESA, and international space agency resources and addressing cislunar operations, satellite systems, and mission planning. That places the DoD space and defense enterprise squarely within the domains where continuous ontology discovery is already being applied, and it is the reason AEDA treats discovery and governance as standing layers rather than one-time construction steps.

DM2 to OWL/RDF Mapping Strategy

The Enterprise Ontology layer converts BEA and DoDAF DM2 from descriptive documentation into a machine-reasonable semantic backbone. The mapping strategy below specifies how DM2 metamodel concepts become OWL/RDF constructs that support sound, complete, auditable inference. The intent is not to re-document DM2 but to render its data groups and relationships as a decidable description-logic ontology that SPARQL can query and a symbolic reasoner can check for consistency.

Mapping Principles

The strategy rests on the W3C standards stack: RDF and RDFS for the triple substrate and class/property vocabulary, OWL 2 for axioms and constraints, and SPARQL for multi-hop query and competency-question validation. We adopt OWL 2 profiles by need: OWL 2 DL where full axiomatic expressiveness over architecture constraints is required, and OWL 2 QL where high-volume access over wrapped authoritative data dominates. Adoption is incremental.

The architecture team begins with an RDF triple graph of DM2 instances, adds the RDFS class and property vocabulary, then layers OWL 2 axioms as the ontology matures.

DM2 distinguishes the things an architecture describes from the relationships among them. That distinction drives the core rule: DM2 entity-like data groups map to `owl:Class`, and DM2 relationship-like couplings map to `owl:ObjectProperty`. Individual architecture elements (a specific capability, a named system) become OWL individuals typed by those classes. Attributes that carry literal values map to `owl:DatatypeProperty`.

Core Concept-to-Construct Mapping

The principal DM2 concepts map as follows:

- Capability maps to `owl:Class`. Instances are the specific capabilities an enterprise must possess.
- Activity maps to `owl:Class`. Instances are the actions the enterprise performs.
- Performer maps to `owl:Class`, with PerformerType subclasses (organization, person type, system, service) declared as `rdfs:subClassOf` Performer.
- Resource maps to `owl:Class`, covering the resource flow elements DM2 tracks.
- InformationExchange and the named DM2 relations map to `owl:ObjectProperty`. The relation set includes `performs` (Performer to Activity), `supports` (Activity or Performer to Capability), `realizes` (Performer or Resource to Capability), `consumes` and `produces` (Activity to Resource), and `exchanges` (Performer to Performer, carrying an InformationExchange).

Object properties carry `rdfs:domain` and `rdfs:range` axioms so the reasoner can enforce well-typed couplings and infer types where they are left implicit. Property characteristics (for example, declaring composition relations transitive) are added in OWL 2 where the DM2 semantics warrant them. Constraints expressed as OWL 2 axioms (cardinality, disjointness of Performer subclasses, property restrictions) turn the ontology into an active governance mechanism that enforces factual accuracy at generation time rather than serving as a passive reference structure.

Upper-Ontology Alignment to BFO

DM2 classes align to the Basic Formal Ontology (BFO) to guarantee semantic interoperability across organizational and system boundaries and to inherit BFO's well-formed continuant-occurrent distinction. Performer and Resource are continuants (BFO independent continuant); Activity is an occurrent (BFO process); Capability is a realizable disposition borne by a Performer. Information Exchange aligns to a BFO information artifact transferred during a process. Alignment is expressed with `rdfs:subClassOf` axioms from each DM2 class to its BFO parent, so reasoning over DM2 instances respects upper-ontology constraints and the

ontology composes with other BFO-aligned domain ontologies rather than standing as an isolated vocabulary.

Federation to ArchiMate and UAF

The enterprise ontology federates to ArchiMate and UAF so AEDA reasons across frameworks rather than within a single notation. Federation is realized as `owl:equivalentClass` and `owl:equivalentProperty` axioms (with `rdfs:subClassOf` where the correspondence is a specialization, not an equivalence) linking DM2 classes and properties to their ArchiMate and UAF counterparts. UAF, as a DoDAF and MODAF successor, supplies the closest structural correspondence; ArchiMate supplies the business-and-application-layer crosswalk. The federation lets a SPARQL query traverse from a DM2 Capability through its UAF equivalent to a related ArchiMate application service in a single multi-hop path, supporting cross-framework impact analysis that documentation-centric tools cannot perform.

Worked Example

The table below shows representative DM2 elements mapped to OWL/RDF constructs, their BFO alignment, and a federation correspondence.

DM2 Concept	OWL/RDF Construct	RDF Form (illustrative)	BFO Alignment	ArchiMate / UAF Correspondence
Capability	`owl:Class`	`dm2:Capability` `rdf:type` `owl:Class`	BFO realizable disposition	UAF Capability (`owl:equivalentClass`); ArchiMate Capability
Activity	`owl:Class`	`dm2:Activity` `rdf:type` `owl:Class`	BFO process (occurrent)	UAF Operational Activity; ArchiMate Business Process
Performer	`owl:Class`	`dm2:Performer` `rdf:type` `owl:Class`	BFO independent continuant	UAF Resource/Actor; ArchiMate Active Structure
Resource	`owl:Class`	`dm2:Resource` `rdf:type` `owl:Class`	BFO independent continuant	UAF Resource; ArchiMate Artifact

DM2 Concept	OWL/RDF Construct	RDF Form (illustrative)	BFO Alignment	ArchiMate / UAF Correspondence
InformationExchange	`owl:ObjectProperty`	`dm2:exchanges rdf:type owl:ObjectProperty`	BFO information artifact (in transfer)	UAF Information Exchange; ArchiMate Flow
performs (Performer to Activity)	`owl:ObjectProperty`	`dm2:performs rdfs:domain dm2:Performer ; rdfs:range dm2:Activity`	relates continuant to occurrent	UAF performs; ArchiMate Assignment
supports (Activity to Capability)	`owl:ObjectProperty`	`dm2:supports rdfs:domain dm2:Activity ; rdfs:range dm2:Capability`	process realizes disposition	UAF supports; ArchiMate Realization
realizes (Performer to Capability)	`owl:ObjectProperty`	`dm2:realizes rdfs:domain dm2:Performer ; rdfs:range dm2:Capability`	continuant bears disposition	UAF exhibits; ArchiMate Realization

A concrete instance set illustrates the result. Given the triples `:SpaceDomainAwareness rdf:type dm2:Capability`, `:Cataloging rdf:type dm2:Activity`, `:SSN_System rdf:type dm2:Performer`, `:SSN_System dm2:performs :Cataloging`, and `:Cataloging dm2:supports :SpaceDomainAwareness`, a SPARQL query can resolve which performers contribute to a given capability through a multi-hop traversal, and a description-logic reasoner can verify that the assertion set is consistent with the domain, range, and disjointness axioms before any result reaches the Decision Layer. This is the mechanism by which DM2, once mapped, supports the architectural inference and impact analysis that static views cannot.

Enterprise Knowledge Graph Schema

The Enterprise Knowledge Graph is the structured substrate that sits between the Enterprise Ontology layer above and the GraphRAG and Agent Swarm layers below. The ontology supplies the formal schema and consistency constraints; the knowledge graph instantiates that schema as concrete entities, relationships, and properties drawn from BEA and DoDAF DM2 content and from wrapped authoritative data sources. This section specifies the node types, edge types, properties, provenance fields, database options, and versioning model required to make that substrate queryable, explainable, and auditable.

Schema Governance Principle

The knowledge graph schema is not defined ad hoc. It is derived from the Enterprise Ontology, which is built on the W3C standards stack (RDF/RDFS triples, OWL 2 axioms and constraints, SPARQL query). Ontology-guided knowledge graph construction outperforms pure vector-based retrieval while reducing language model usage cost, because the ontology supplies formal semantic constraints, consistency enforcement, and rule-based reasoning over the graph. Every node type and edge type below traces to an ontology class or property, and every instance is validated against ontological constraints before it is admitted to the graph.

Node Types

Node types model the entity classes of the enterprise. Each node is typed against an ontology class and carries a stable identifier, label set, and property bundle.

Node type	Description	Ontology basis
Capability	A mission or business capability expressed in BEA/DoDAF terms	Ontology class, governed
Activity / Process	An operational activity or business process step	Ontology class, governed
System	An information system, application, or technical component	Ontology class, governed
Data Entity	An authoritative data object or information element	Ontology class, governed
Organization	An organizational unit, role, or responsible party	Ontology class, governed
Standard	A technical standard, policy, or regulatory rule	Ontology class, governed
Performer	An actor that performs an activity (human role or automated agent)	Ontology class, governed
Resource	A material, financial, or computational resource	Ontology class, governed
Document Artifact	A source artifact (architecture document, diagram, illustration)	Ontology class, governed

Document Artifact nodes support multimodal extension: the graph fuses text plus illustrations and diagrams from architecture artifacts rather than text alone, using cross-modal entity linking

to connect a diagram element to its corresponding textual entity.

Edge Types

Edge types model the typed, directed relationships between nodes. Each edge is typed against an ontology property and carries its own property bundle and provenance, so that relationships are first-class, auditable graph elements rather than implicit links.

Edge type	Connects	Semantics
enables	Capability to Capability / Activity	Capability dependency
performs	Performer to Activity	Activity assignment
supports	System to Capability / Activity	System realization
produces / consumes	Activity to Data Entity	Data flow
governs	Standard to System / Activity / Data Entity	Compliance constraint
responsible_for	Organization to Capability / System	Accountability
depends_on	System to System / Data Entity	Technical dependency
derived_from	Node to Document Artifact	Provenance link to source
is_a / part_of	Node to Node	Class hierarchy / composition

Hierarchical and compositional edges (`is_a`, `part_of`) are essential. GraphRAG surpasses dense retrieval precisely when questions require explicit relational reasoning, hierarchical organization, or multi-source integration, so the schema must encode hierarchy and composition explicitly rather than leaving them latent in text.

Node and Edge Properties

Every node and edge carries a property bundle in three groups: identity, semantic, and operational.

Identity properties - `id`: persistent unique identifier (URI under the Linked Data convention) - `type`: ontology class or property the element instantiates - `labels`: human-readable label set - `canonical_name`: resolved canonical entity name after disambiguation

Semantic properties - `definition`: governed textual definition traced to the ontology - `attributes`: domain attributes defined by the ontology class - `embedding_ref`: reference to the vector embedding for this element, supporting the unified vector plus graph plus ontology

retrieval stack - `centrality`: graph centrality measures (for example eigenvector centrality) used for skeleton construction and subgraph selection

Operational properties - `confidence`: extraction or assertion confidence score - `validation_status`: ontology-consistency validation state (validated, pending, rejected) - `last_verified`: timestamp of the most recent consistency verification

Entity Provenance

Provenance is a first-class requirement, not metadata added after the fact. Explainability and traceability are mandatory for DoD decision support, so every node and every edge carries a provenance record sufficient to reconstruct how the element entered the graph and on what authority.

Provenance fields, attached to nodes and edges:

- `source_artifact`: identifier of the originating Document Artifact or authoritative data source
- `source_locator`: position within the source (document section, table, diagram region)
- `extraction_method`: how the element was created (human curation, language model extraction, ontology-mediated query rewriting over a wrapped relational source)
- `extraction_model`: identity of the extraction model where a language model was used
- `consensus_status`: result of multi-model consensus validation where applied
- `validated_by`: ontology constraint set and reasoner used to confirm consistency
- `authority`: the authoritative system or owner of record for the asserted fact
- `asserted_at`: timestamp of assertion
- `justification_path`: traceable logical justification chain supporting the assertion, so that every reasoning output carries a justification path and natural-language explanation into the Decision and Executive Support layers

The `extraction_method` field carries a specific architectural meaning. Where the underlying fact originates in an existing relational or legacy authoritative data source, the element is admitted through ontology-mediated query answering with query rewriting rather than full materialization of the source as RDF. This preserves data currency and identifies the element as a virtual, source-backed assertion rather than a copied one.

Provenance and Trust Enforcement

Provenance feeds the trust controls that the layers above depend on. Candidate elements proposed by language model extraction are routed through a closed-loop validation pipeline before admission: SPARQL constraint queries test the candidate against ontology constraints, a symbolic reasoner verifies logical consistency, and inconsistent candidates are returned for corrective iteration. Only ontology-consistent, provenance-complete elements are admitted. At

retrieval time, the same provenance supports hallucination detection through path reliance and semantic alignment scoring and through explicit evidence-chain attribution, so a retrieved subgraph can be checked against the evidence that supports it.

Graph Database Options

The knowledge graph is realized on a graph database. Three implementation options are supported, selected by the expressiveness and query profile of the enterprise.

Property graph (Neo4j class). A labeled property graph with Cypher query and native traversal. This is the recommended default for the Knowledge Graph layer. It supports Personalized PageRank and breadth-first traversal for subgraph retrieval, Prize-Collecting Steiner Tree subgraph extraction, and eigenvector-centrality skeleton construction to keep construction cost tractable at enterprise scale. Property graphs carry rich properties directly on nodes and edges, which suits the identity, semantic, operational, and provenance bundles specified above. A Neo4j-backed graph has been demonstrated as the graph layer augmenting a multimodal language model in a defense context.

RDF triplestore. An RDF/RDFS/OWL triplestore with SPARQL query. This option maximizes formal semantic fidelity and is the natural realization where the enterprise requires full OWL 2 axiomatic expressiveness, decidable description-logic reasoning, and standards-based interoperability across organizational and system boundaries. SPARQL supports multi-hop reasoning, filtering, and aggregation over the graph. Where high-volume data access dominates, the OWL 2 QL profile optimized for data access applies; where full axiomatic expressiveness is required, OWL 2 DL applies. The triplestore also enables ontology-mediated query answering to wrap existing relational sources without materialization.

Hybrid property graph plus triplestore. A dual realization that keeps the property graph as the operational traversal and retrieval engine while maintaining an RDF/OWL representation for formal reasoning and interoperability. This reflects the unified vector plus graph plus ontology stack: the graph database is not a vector store alone, and the choice of property graph versus triplestore is a realization decision, not a change to the ontology-governed schema.

Selection guidance: default to the property graph for operational traversal performance and property richness; choose the RDF triplestore where formal expressiveness, decidable reasoning, and standards interoperability are the governing requirements; adopt the hybrid where both pressures are present.

Versioning and Temporality

The graph is not static. BEA and DoDAF DM2 content evolves, authoritative sources change, and engineering changes propagate through the architecture. The schema therefore carries

explicit versioning and temporal handling.

- **Element versioning.** Each node and edge carries a version identifier and validity interval (`valid_from`, `valid_to`). Updates create new versioned assertions rather than destructively overwriting prior ones, preserving the historical record required for audit.
- **Rollback to verified state.** The graph supports incremental and temporal updates with the ability to roll back to a prior verified state. This protects the trust guarantees of the layers above: a faulty incremental update can be reverted to the last consistency-validated snapshot.
- **Temporal reasoning.** Temporal handling supports reasoning over evolving facts and concept drift, so that queries can be scoped to a point in time and the graph can represent how the enterprise architecture changed across versions.
- **Change-propagation support.** Versioned, temporally aware nodes and edges feed the dynamic multi-layer and hierarchical graph used for architectural impact analysis, where change-propagation paths are parsed automatically and critical affected elements are flagged. Versioning is what lets impact analysis compare a proposed change against the established baseline.

Schema Summary

The Enterprise Knowledge Graph schema instantiates the Enterprise Ontology as a typed, property-bearing, provenance-complete graph. Node types model enterprise entity classes and edge types model their governed relationships, both traced to ontology classes and properties. Identity, semantic, and operational properties make elements queryable and rankable; provenance fields make every assertion traceable to its source, extraction method, authority, and justification path; closed-loop ontology validation gates admission; and versioning with temporal handling and rollback preserves auditability and supports predictive impact analysis. Realized on a property graph, an RDF triplestore, or a hybrid of the two, this schema is the queryable, explainable foundation on which the GraphRAG and Agent Swarm layers operate.

GraphRAG Retrieval and Reasoning Design

The GraphRAG layer is AEDA's reasoning substrate. It sits between the Enterprise Ontology layer above and the Agent Swarm layer below, and it converts BEA/DoDAF DM2 content from static documentation into a queryable, explainable decision stack. The design follows a five-component reference decomposition (query processor, retriever, organizer, generator, data source) and is built on a unified vector, graph, and ontology stack rather than a single vector store. Dense-vector RAG alone treats documents as flat, atomized chunks ranked by semantic similarity, which cannot capture relational structure or support the multi-hop reasoning that enterprise-architecture questions require. GraphRAG organizes knowledge as explicit structured graphs of entities, relationships, and hierarchical semantic dependencies, improving reasoning reliability and interpretability.

Hybrid Retrieval Design

Retrieval fuses three complementary modalities rather than selecting one:

- **Sparse retrieval** provides fast lexical matching. It carries the lowest hallucination profile and the lowest latency tier (approximately 45 ms, 70 to 75 percent accuracy in the comparative literature), making it the cheapest first pass over the architecture corpus.
- **Dense retrieval** uses semantic embeddings for similarity matching (approximately 50 ms, 75 to 80 percent accuracy). Approximate Nearest Neighbor indexing reduces dense retrieval from O(n) to O(log n) so that the embedding pass stays interactive at enterprise scale. EASE-DR-style enhanced sentence embeddings strengthen the dense lane.
- **Graph traversal** retrieves over the knowledge graph using BFS and Personalized PageRank over candidate subgraphs (the GraphRAG tier reports approximately 250 ms, 85 to 89 percent accuracy, with high hallucination reduction). Graph traversal is what supplies relational, hierarchical, and multi-source reasoning that dense and sparse passes cannot.

A learned **hybrid retriever** fuses these lanes. The HybRAG pattern combines semantic node-level retrieval with structural path-level retrieval and outperforms single-retriever baselines on WebQSP and CWQ. The comparative table places Hybrid RAG at approximately 200 ms and 89 to 93 percent accuracy with very high hallucination reduction, and the neural-symbolic configuration at approximately 300 ms and 89 to 95 percent accuracy with the strongest hallucination control. AEDA adopts hybrid fusion as the default rather than committing to a single modality.

Adaptive query routing governs which lanes fire. The EA-GraphRAG pattern computes continuous complexity scores from syntactic features and routes simple factual queries to dense retrieval (approximately 50 ms, roughly 75 percent accuracy) and complex multi-hop queries to GraphRAG (approximately 200 ms, roughly 88 percent accuracy), with reported token-cost reductions of up to 90.71 percent versus baseline. Adaptive routing outperforms a rigid GraphRAG-for-everything posture, so AEDA budgets for the documented latency tiers and pursues the token-cost reduction by sending factual lookups down the cheap lane and reserving graph traversal for multi-hop enterprise questions.

To keep graph construction and traversal tractable at enterprise scale, AEDA applies Prize-Collecting Steiner Tree subgraph extraction with Personalized PageRank, plus selective skeleton construction from the top 20 percent of chunks by eigenvector centrality. The neural-symbolic dual-indexing literature reports a 10x cost reduction versus exhaustive KG construction while improving generation quality by 32.4 percent and retrieval coverage by 92.4 percent.

Query Decomposition

Complex enterprise questions are decomposed into ordered sub-questions before retrieval. StepChain GraphRAG performs question decomposition followed by BFS over the graph and

reports a 4.70 percent EM and 3.44 percent F1 improvement on HotpotQA. DRKG (Decomposed Reasoning over Knowledge Graph) builds LLM-guided hop-constrained reasoning plans and reports a 1 to 5 percent accuracy improvement with higher interpretability. AEDA uses decomposition both to bound traversal depth and to make each reasoning hop inspectable.

Multi-Hop Reasoning

Multi-hop reasoning is executed over the knowledge graph with attention-based and fusion-based mechanisms:

- **MAGNA** (Multi-hop Attention Graph Neural Network) propagates attention across multiple hops and reports a 5.7 percent relative error reduction on KG completion and up to a 10 percentage point improvement on node classification.
- **KIFGraph** builds multi-granularity fusion graphs with masked attention and outperforms standard GNN methods on HotpotQA.
- **ScaleGNN** addresses over-smoothing through per-hop pure-neighbor matrices and learnable sparsity masking on billion-scale graphs, which matters for an enterprise-scale architecture corpus.

Hop-constrained reasoning plans (DRKG) and BFS expansion (StepChain) bound the traversal so that multi-hop chains remain interpretable rather than exploding combinatorially.

Evidence Aggregation

Retrieved evidence is aggregated across the semantic node-level and structural path-level lanes before generation. HybRAG fuses semantic and structural retrieval into a single evidence set; neural-symbolic dual-indexing uses Prize-Collecting Steiner Trees plus Personalized PageRank to assemble the supporting subgraph; and selective skeleton construction by eigenvector centrality concentrates aggregation on the highest-salience nodes. StructReason applies PCST-based structural refinement to the aggregated evidence, cutting token consumption 40 to 70 percent while improving F1 versus standard GraphRAG and LightRAG. The organizer component assembles this aggregated subgraph into the context passed to the generator.

Citation Grounding

Every generated answer is grounded in an explicit evidence chain so that AEDA decisions are traceable to source architecture artifacts, which is mandatory for DoD decision support. Grounding uses attribution verification with explicit evidence chains and the Graph Grounding and Alignment (GGA) mechanism, which applies mechanistic interpretability through path reliance degree (PRD) and semantic alignment score (SAS) for lightweight post-hoc detection. GGA outperforms semantic and confidence baselines on AUC and F1. Path reliance analysis

additionally detects over-reliance on shortest-path triples, and confidence scoring via Bayesian or ensemble disagreement flags low-support answers. GraphRAG reduces but does not eliminate hallucination, so these grounding and detection mechanisms are engineered into the stack from the start rather than added after the fact.

Reranking

Candidate evidence is reranked before generation to prioritize context relevance, which the benchmarking literature reports as more predictive of downstream performance than abstract retrieval metrics. Reranking concentrates on whether the retrieved subgraph actually answers the query: eigenvector centrality prioritizes high-salience skeleton nodes, Personalized PageRank weights structurally proximate evidence, and PCST-based refinement (StructReason) prunes low-value paths from the aggregated set. Retrieval-quality signals (Recall@k, MRR, NDCG) drive ranker tuning, while context-relevance is weighted highest in selecting the final evidence set passed to generation.

Tuning Variables

The following variables are tuned against the AEDA benchmark harness:

- **Routing threshold**: the continuous complexity score (EA-GraphRAG) that partitions queries between the dense lane and the graph-traversal lane, traded off against the documented latency tiers and the up-to-90.71 percent token-cost reduction.
- **Traversal depth and hop constraints**: BFS depth (StepChain) and hop-constrained plans (DRKG) that bound multi-hop expansion.
- **Skeleton fraction**: the top-20-percent eigenvector-centrality cut for skeleton construction, governing the 10x cost-versus-coverage trade.
- **Subgraph extraction parameters**: Prize-Collecting Steiner Tree and Personalized PageRank settings controlling subgraph size and structural focus.
- **Structural refinement aggressiveness**: PCST pruning depth (StructReason) targeting the 40 to 70 percent token-consumption reduction.
- **Fusion weights**: the learned balance between sparse, dense, and graph-traversal lanes in the hybrid retriever.
- **Grounding thresholds**: PRD and SAS cutoffs (GGA) and confidence-score thresholds for hallucination detection.
- **ANN parameters**: index settings governing the O(n) to O(log n) dense-retrieval reduction.
- **Caching policy**: LRU and community-summary caching, with hierarchical community structure and distributed indexing by domain for scale.

Acceptance Metrics

Acceptance is measured on knowledge-intensive multi-hop benchmarks (HotpotQA, MuSiQue, 2WikiMultiHopQA) plus retrieval metrics (Recall@k, MRR, NDCG, EM, F1), instrumented through a GraphRAG-Bench-style harness with context relevance weighted highest. Target performance tiers from the comparative literature:

Method	Latency	Accuracy	Hallucination Reduction
Sparse RAG	~45 ms	70 to 75%	Low
Dense RAG	~50 ms	75 to 80%	Moderate
Semantic RAG	~120 ms	80 to 85%	Moderate
GraphRAG	~250 ms	85 to 89%	High
EA-GraphRAG	~180 ms	88 to 92%	Very High
Hybrid RAG	~200 ms	89 to 93%	Very High
Neural-Symbolic	~300 ms	89 to 95%	Very High

Additional acceptance evidence from the literature: GraphRAG-Bench shows GraphRAG surpasses dense RAG when questions require explicit relational reasoning, hierarchical organization, or multi-source integration, while dense RAG stays superior and more efficient for simple factual retrieval; StepChain GraphRAG improves HotpotQA by 4.70 percent EM and 3.44 percent F1; MAGNA achieves 5.7 percent relative error reduction on KG completion and up to 10 percentage point improvement on node classification; neural-symbolic dual-indexing improves generation quality by 32.4 percent and retrieval coverage by 92.4 percent at 10x lower construction cost; StructReason cuts token consumption 40 to 70 percent while improving F1; and joint KG-LLM models report up to 12.0 percent accuracy and 8.6 percent F1 improvement over standalone LLMs, with the neural-symbolic configuration providing the strongest hallucination control. Interactive deployment targets sub-500 ms latency through cloud-edge split, ANN indexing, and caching.

Acceptance gating is conditional on query class: AEDA accepts the dense lane for simple factual architecture lookups and requires the graph-traversal or hybrid lane to meet the multi-hop accuracy and hallucination-reduction tiers for relational, hierarchical, and multi-source enterprise-architecture questions.

Agent Swarm Architecture

The Agent Swarm layer is the computational tier of AEDA where specialized autonomous agents collaborate to execute architecture analysis, portfolio optimization, risk assessment, red-team evaluation, and strategic planning over the lower AEDA layers (Enterprise Ontology, Knowledge Graph, GraphRAG). It is the point at which BEA / DoDAF DM2 ceases to be documentation and becomes a population of goal-directed agents that perceive enterprise state,

reason over extended horizons, plan, and act through governed tools. The layer is framed by the paradigm shift from stateless, prompt-driven generative models to goal-directed systems that operate through iterative control loops with minimal human intervention, exercising five core capabilities: perception, memory, planning and reasoning, tool use, and multi-agent coordination [1][2].

Individual Agent Structure

Each AEDA swarm agent is structured on a cognitive-architecture pattern rather than as an ad hoc prompt chain. The CoALA framework (Cognitive Architectures for Language Agents) supplies the reference structure: modular memory systems, a structured action space spanning internal-memory and external-environment interaction, and generalized decision-making for choosing actions [6]. This grounding draws on the deliberative and Belief-Desire-Intention (BDI) lineage of agent design [2], giving each agent an explicit internal model of its objective, the actions available to it, and the criteria by which it selects among them. The result is an agent whose reasoning and action selection are inspectable, a precondition for the governance and explainability requirements imposed on this layer.

Specialized Agents

The swarm is composed of role-specialized agents, each bound to a distinct enterprise-architecture function. The division of labor follows the hierarchical supervisor and subtask-decomposition pattern, in which a supervisor agent decomposes enterprise goals into executable subtasks via hierarchical task networks, combining chain-of-thought reasoning, contextual memory, and task decomposition to improve task-completion accuracy, planning efficiency, and adaptability over single-agent and flat multi-agent designs [7].

- **Architecture agent.** Performs structural analysis of the BEA / DoDAF DM2 corpus surfaced through the Enterprise Ontology and Knowledge Graph layers, identifying dependencies, gaps, and conformance findings.
- **Risk agent.** Scores and predicts risk over enterprise state. Risk specialization is grounded in heterogeneous multi-agent designs that dedicate agents to risk assessment alongside return prediction and market perception [14], and in constraint-enforcing formulations such as Augmented Lagrangian Multiplier methods that hold hard risk constraints with zero constraint violations during testing [16].
- **Governance agent.** Enforces policy, identity, and compliance constraints at the agent level, consulting the Agent Registry and Policy Engine pattern in which central orchestrators apply jurisdictional rules, access controls, and tamper-evident audit logs [34].
- **Red Team agent.** Conducts adversarial evaluation of the swarm and its outputs, drawing on multi-turn jailbreak and adversarial-testing methods exposed as callable services [35] and on multi-turn red-teaming frameworks for systematic probing [36].

- **Portfolio agent.** Optimizes across the enterprise portfolio. Portfolio specialization is grounded in graph attention-based heterogeneous multi-agent deep reinforcement learning that models time-varying correlations with specialized agents and reports 16.8% annualized returns, a 1.34 Sharpe ratio, and 8.2% maximum drawdown, outperforming mean-variance and equal-weight baselines [14], and in hierarchical DRL with auxiliary agents reporting Sharpe-ratio improvements exceeding 8.2% over traditional strategies [15].
- **Optimization agent.** Executes constrained optimization under hard policy and risk limits, applying constraint-projection and risk-weighted Lagrangian utility shaping so that optimization respects governance boundaries rather than circumventing them [16][33].
- **Strategy agent.** Synthesizes the outputs of the specialized agents into strategic-planning recommendations for promotion to the Simulation, Optimization, and Decision layers, operating within the supervisor-decomposition topology [7].

Coordination Over Shared Graph Context

The swarm coordinates through shared graph-structured memory backed directly by the AEDA Knowledge Graph layer rather than through point-to-point message passing alone. The reference pattern is G-Memory, a hierarchical agentic memory system inspired by organizational memory theory that manages multi-agent interactions through a three-tier graph hierarchy: insight graphs holding high-level generalizable knowledge, query graphs for efficient retrieval, and interaction graphs capturing fine-grained collaboration trajectories, with bi-directional traversal that supports cross-trial knowledge reuse [8]. Implemented as the swarm's shared memory and persisted in the Knowledge Graph layer, this structure lets agent insights and collaboration trajectories accumulate across tasks and lets later agents leverage the reasoning of earlier ones. Agents ground every action in this shared context using the ReAct pattern, which interleaves reasoning and acting to call tools, observe results, and dynamically adjust [3], combined with Agentic RAG, which embeds autonomous agents into the retrieval pipeline using reflection, planning, tool use, and multi-agent collaboration to iteratively refine context rather than retrieve once [10]. This binds the swarm directly to the GraphRAG layer beneath it.

MCP-Governed Tool Access

Agents reach tools, data sources, and external systems through the Model Context Protocol (MCP), positioned as an emerging open standard that replaces fixed API calls with dynamic, runtime-discovered capabilities and cross-framework interoperability through standardized interfaces [4]. The documentation-era BEA / DoDAF data sources, which are deterministic enterprise and system APIs, are exposed to the swarm through an Agent-Ready Architecture (ARA) wrapper that enriches those endpoints with natural-language tool descriptions and structured parameter schemas. ARA reports 90% semantic discovery precision and a 60.5% reduction in multi-tool chain latency, which AEDA adopts as design targets for tool exposure

[12]. MCP access is not open-ended: it is mediated by the governance controls described below, so that tool discovery and invocation occur within policy boundaries.

Orchestration Patterns and Control Plane

The swarm's coordination substrate follows the Enterprise Agentic Architecture Framework (EAAF), which defines six layers (infrastructure, enterprise integration, orchestration and coordination, governance and safety, agent intelligence, and interaction) with a central Control Plane that manages policies, identity, scheduling, observability, and agent lifecycle [17]. In AEDA this Control Plane is the integration seam between the Agent Swarm layer and the Decision and Executive Support layers above it. Multi-layer agentic architectures of this class report 3-10x workflow acceleration and 60-80% reductions in mean time to resolution for critical tasks [17][37]. Orchestration also draws on stateful patterns that provide explicit state machines and exactly-once semantics for agent workflows.

Governance, Human-in-the-Loop, and Policy Compliance

Governance is a layer-level requirement of the Agent Swarm, not an afterthought. The swarm is instrumented against an Agentic AI Governance Maturity Model (AAGMM), a five-level framework over 12 governance domains grounded in NIST AI RMF and ISO/IEC 42001 and validated through 750 simulation runs across five enterprise scenarios, with statistically significant differences between maturity levels ($p<0.001$, effect sizes $d>2.0$); organizations at Levels 4-5 achieve 94.3% lower agent-sprawl indices, 96.4% fewer risk incidents, and 32.6% higher effective task-completion rates [23]. This evidence is consequential given that only 21% of enterprises have mature governance for autonomous agents and 40% of agentic AI projects are projected to fail by 2027 due to inadequate governance [23]. The layer therefore actively monitors and prevents agent sprawl: functional duplication, shadow and orphaned agents, permission creep, and unmonitored delegation chains.

Human oversight is implemented as a continuous, context-sensitive function rather than a binary gate. The Adaptive Oversight Calibration Model (AOCM) calibrates oversight across six formal propositions (task criticality, AI competency boundaries, human cognitive capacity, institutional constraints, trust dynamics, and feedback loops) [24]. The TRACE Framework (Trust, Review, Accountability, Critique, Explainability) embeds governance anchors at the agent level and adds dedicated Critic agents for meta-validation that produce an Overall System Confidence score, which routes outputs to automated action, human escalation, or continuous learning [25]. Runtime policy compliance is enforced through a CAMCO-style (Safe and Policy-Compliant Multi-Agent Orchestration) constrained-optimization layer that combines constraint-projection engines, adaptive risk-weighted Lagrangian utility shaping, and iterative negotiation, and reports zero policy violations, risk exposure below threshold (mean ratio 0.71), 92-97% utility retention, and mean convergence in 2.4 iterations [33]. Explainability is treated as core infrastructure through SHAP, LIME, and Grad-CAM, with the literature

reporting accuracy gains of 14.8% for SHAP and 12.1% for LIME over no-explanation systems [28]; these outputs are surfaced to the AEDA Decision and Executive Support layers.

Red-Team and Evaluation Harness

Before any swarm output is promoted to the Simulation, Optimization, and Decision layers, the swarm is exercised against a standing red-team and evaluation harness. Capability is benchmarked on GAIA, AssistantBench, and WebArena [39]. Adversarial robustness is tested with multi-turn methods, including AJAR (Adaptive Jailbreak Architecture for Red-teaming), which exposes multi-turn jailbreak algorithms as callable MCP services and raises X-Teaming attack success from 65.0% to 76.0%, reaches 80% cumulative success one turn earlier, and reproduces Crescendo better than the PyRIT baseline (91.0% vs 87.5%) [35], together with the RedTWIZ multi-turn red-teaming framework [36]. Trust controls are organized along the seven-layer trust taxonomy of Trustworthy Agentic AI (identity, planning, communication, memory, retrieval, execution, oversight), which yields reusable secure-coordination design patterns [37], and are reinforced by the SAFE-AI Framework (Safety, Auditability, Feedback, Explainability) and its taxonomy of suggestive, generative, autonomous, and destructive agent behaviors [38]. This harness is the gate the Agent Swarm layer must pass before it feeds decisions upward.

Enterprise Digital Twin Design

The Enterprise Digital Twin layer is where AEDA transitions from representing the enterprise to simulating it. The upstream layers (Data Sources through GraphRAG) produce a semantic record of what the DoD enterprise is. The Enterprise Digital Twin is a continuously updating virtual replica of that enterprise: a living model that synchronizes with the real organization, supports simulation-driven experimentation, and feeds evidence-based decisions to the layers above it. This is the distinguishing feature of an enterprise digital twin (EDT) versus traditional business intelligence. BI reports offline on what already happened; the EDT maintains a bidirectional relationship with the systems it models, ingesting continuous data streams in and returning actionable insight out that can influence those systems in near real time.

What the Enterprise Twin Represents

An enterprise digital twin models more than physical assets. The literature defines EDTs as continuously updating virtual replicas of organizational processes, assets, and entire business systems, capturing business processes, workflows, organizational structures, market-facing operations, and the decision-makers themselves. Translated to the DoD enterprise that BEA and DoDAF DM2 already document, the AEDA twin represents:

- **Organizations.** Command structures, reporting relationships, and the units and offices that constitute the enterprise.
- **Programs.** Acquisition and mission programs as first-class modeled entities with state, dependencies, and lifecycle.
- **Capabilities.** The operational and business capabilities the enterprise delivers, mapped to the organizations and programs that produce them.
- **Budgets.** Funding lines and resource allocations tied to the programs and capabilities they support.
- **Schedules.** Program and milestone timelines, including the dependencies that couple one schedule to another.
- **Resources.** Personnel, materiel, and infrastructure consumed and produced across the enterprise.
- **Risks.** Exposures and their propagation paths through the dependency network.
- **Dependencies.** The relationships that link all of the above, so that a change in one entity is traceable to its effect on others.

The twin does not rebuild this semantic structure. It inherits it. The AEDA Enterprise Ontology and Knowledge Graph supply the twin's semantic substrate directly: an entity layer for organizational concepts, a relationship layer for dependencies and interactions, and a reasoning layer for deriving new insight. Multi-ontology networks combine domain-specific ontologies with standardized ontologies (OWL, SOSA), and graph reasoning lets the twin carry forward the semantic interoperability established upstream rather than reconstructing it.

Structurally, the twin is built on a four-tier reference architecture: a data ingestion layer, a virtual representation layer, an analytical processing layer, and an application and user-interface layer. A dedicated data layer sits within this pattern as a universal translator, decoupling physical and cyber components and converting heterogeneous source formats into standardized representations. This is what lets the AEDA twin ingest diverse DoD enterprise sources without point-to-point coupling, and it maps cleanly onto the upstream AEDA Data Sources and Ontology layers.

Event-Driven State Updates

The twin is kept synchronized with the enterprise through real-time, event-driven updates. Continuous data streams flow into the virtual representation, updating modeled state with minimal latency, typically milliseconds to seconds depending on the application. Because the enterprise drifts over time, the twin applies automated, continuous parameter calibration to counter model drift and keep the virtual replica aligned with the real organization.

State updates run on a deployment pattern suited to enterprise scale. A hybrid cloud-plus-edge architecture pairs centralized cloud repositories and computationally intensive simulation with edge processing for low-latency local monitoring. Service-oriented and microservices patterns, including Platform-as-a-Service and Digital-Twin-as-a-Service (DTaaS) provisioning, support

modular composition, service catalogs with standardized interaction protocols, and governance-controlled evolution. The edge-cloud split also confers resilience, with continued degraded-mode operation if cloud connectivity is lost, and privacy, with sensitive data retained at the edge.

Incoming data quality is treated as a first-class concern, since errors in the data stream cascade through simulation into corrupted decision support. The twin applies statistical process control and ML-based anomaly detection (isolation forests, autoencoders, variational autoencoders, and multi-stage architectures that combine sequence modeling with statistical process control and operating-envelope constraints) to validate both incoming data and the model itself as state updates arrive.

From Descriptive to Predictive, With Evidence

The transition the Enterprise Digital Twin layer delivers is from describing the enterprise to predicting its behavior. A continuously synchronized virtual replica supports forecasting, what-if experimentation, and Monte-Carlo scenario analysis, simulating large numbers of scenarios to stress-test strategies and surface vulnerabilities before any action is taken in the real enterprise. The twin functions as a safe sandbox: organizational transformation and change-management options can be explored risk-free in the model before implementation.

The predictive capability rests on embedded AI and simulation that the literature validates with quantified results. For forecasting, LSTM networks support time-series prediction (reported at 94% accuracy in a water-demand forecasting application) alongside ensemble methods including random forests, gradient boosting machines, and neural network ensembles. Simulation methodologies combine discrete event simulation (DES) for process, supply-chain, and logistics flows, agent-based simulation (ABS) for autonomous adaptive entities and emergent organizational behavior, and physics-informed or hybrid models where governing relationships are known. Reinforcement learning (A3C, PPO) is trained inside the twin as a safe simulated training ground before any real-world action, aligning with AEDA's downstream Optimization and Agent Swarm layers.

Reported applications substantiate the descriptive-to-predictive claim. A hospitality digital twin predicted occupancy at $R2 = 0.86$, energy consumption within 8.3% accuracy, and staff-efficiency improvements of 14.7%. Financial digital twins achieved 30 to 40% improvements in customer-experience metrics while reducing operational risk. Graph-based supply-chain twins achieved significantly improved scalability over traditional simulation while enabling proactive disruption management. These results are drawn from domains outside DoD enterprise management, but they establish that an organizational twin can move from monitoring to forecasting with measurable accuracy, which is precisely the transition AEDA requires.

Two design commitments keep the predictive twin trustworthy and auditable. First, the layer anchors to standards and rigorous engineering: the ISO 23247 digital twin reference architecture for interoperability, and Model-Based Systems Engineering (MBSE) for formal capture of requirements, interfaces, and validation, so the AEDA twin is traceable rather than

ad hoc. Second, the layer preserves a human-in-the-loop posture. Large language models and agentic digital twins translate simulation outputs into natural-language explanations, generate scenarios, and reason over unstructured enterprise information, forming the bridge to AEDA's Executive Support layer while keeping outputs explainable for executive trust. Intelligent Acting Digital Twins (IADTs) can move beyond passive monitoring to autonomous control, but given accountability and organizational-values concerns, AEDA positions the twin to enhance rather than replace executive judgment, with explainable-AI safeguards on any autonomous actuation.

Simulation and Optimization Framework

Purpose and Position in the Stack

The Simulation and Optimization Framework is the computational engine that converts AEDA's Enterprise Digital Twin into time-advancing, decision-relevant behavior. The Simulation layer executes the twin over time and propagates uncertainty; the Optimization layer searches the resulting action space for ranked investment and capability decisions under resource constraints. Together they supply the probabilistic substrate that the Decision Layer and Executive Support consume. This is the concrete mechanism for AEDA's central thesis: evolving DoD enterprise architecture from documentation into computation, because BEA/DoDAF enterprise behavior spans aggregate flows, discrete resource contention, and heterogeneous decision-making actors that only an executable model can represent.

Simulation Methods

The literature establishes a methodological spine of four simulation paradigms, with hybrid composition as the governing design pattern. No single methodology dominates across all contexts; method choice is driven by aggregation level, actor heterogeneity, temporal resolution, and data availability.

Discrete-Event Simulation (DES). DES models resource queues, scheduling, dependencies, and process flow. The Discrete-Event System Specification (DEVS) formalism provides a rigorous mathematical foundation with stated advantages of completeness, verifiability, extensibility, and maintainability, and implementations across C++, Java, and Python. AEDA adopts DEVS as the formal specification standard so architecture elements map cleanly to composable, reusable simulation entities, bridging the Enterprise Ontology and Knowledge Graph to runnable model components. For campaign-scale execution, the xDEVS toolkit combines traditional DEVS implementations with cloud deployment and parallel/distributed execution (PDES), achieving a stated 15.95x speedup on distributed systems while preserving model reusability and semantic rigor; PDES uses both optimistic and conservative synchronization mechanisms. Validation studies report discrete-event models achieving less than 5% deviation from actual plant data.

Monte Carlo (MC) Simulation. MC provides the forward-propagation mechanics for uncertainty. Classical MC convergence is O(N^-0.5), so computational cost grows quadratically with desired accuracy, which is prohibitive for high-fidelity models requiring thousands of forward-model evaluations. AEDA addresses this with a multifidelity hierarchy: Multilevel Monte Carlo (MLMC) and Multifidelity Monte Carlo (MFMC) extend control-variate ideas to run most evaluations on low-cost, low-accuracy models and few on high-cost, high-accuracy models, preserving statistical accuracy while reducing burden. Randomized quasi-Monte Carlo with scrambled Sobol' sequences shows smaller bias and RMSE than standard MC for risk-averse stochastic optimization.

Bayesian Networks. Bayesian networks enable quantitative risk assessment through probabilistic graphical reasoning and represent dependencies and causal risk structure across the enterprise. A Bayesian Belief Network combined with 5D Building Information Modeling found probabilistic cash-flow ranges deviate 11 to 130% from deterministic estimates when risk impacts are included, demonstrating why point estimates understate exposure. A Markov-chain plus Bayesian dynamic risk framework achieved 78% predictive accuracy in forecasting risk-state evolution. These structures couple directly to value-of-information analysis at the Decision layer.

System Dynamics (SD). SD captures causal feedback loops, time-delayed responses, and nonlinear relationships for strategic planning, resource management, and organizational behavior. It models supply-chain vulnerability risk contagion and quantifies nonlinear competency-development, risk, and efficiency relationships. Within AEDA, SD supplies the aggregate policy and feedback dynamics for strategic decision support.

Hybrid Composition. Hybrid simulation integrating DES, SD, and agent-based modeling combines macroscopic aggregate dynamics with microscopic behavioral heterogeneity, and is the key design pattern for AEDA: DES for resource queues and scheduling, agent-based modeling for organizational actors and emergent behavior feeding the Agent Swarms layer, and SD for aggregate policy and feedback dynamics. Model-Based Systems Engineering and SysML remain the authoritative source of truth, with executable models generated from them (SysML integrated with the IMPRINT human-performance tool, Capella/Arcadia patterns) to preserve architecture-to-simulation traceability. Digital transformation in project and operations management reduced operational expenses by 25% while improving schedule precision by 40%, evidencing the operational return of this coupling.

Optimization Methods

The Optimization layer expresses portfolio and capability decisions as solvable mathematical programs over the enterprise ontology, shifting the objective from asset cost to mission and capability value.

Multi-Objective Optimization. Bi-objective and multi-criteria formulations explore Pareto-optimal sets, minimizing total cost while maximizing stakeholder-interest fulfillment via nonlinear programming and heuristics. NSGA-II combined with constraint programming cut labor costs 13.2% while improving satisfaction. Capability-based planning integrated with enterprise architecture and project portfolio management enables multi-criteria selection grounded in strategic objectives using the Analytical Hierarchy Process and linear programming, marking the shift from asset-centric to outcome-centric portfolio management. A defense capability metric combines investment-portfolio measures with technological-aging signals to link investment projects to operational systems and mission outcomes.

Portfolio Optimization. Stochastic project portfolio selection maximizes expected value via genetic algorithms plus Monte Carlo simulation, handling schedule interdependencies, budget constraints, risk registers, and portfolio reliability constraints. For space and defense, mixed-integer programming and a multi-period precedence-constrained knapsack maximize stakeholder-interest fulfillment under periodic budget constraints while sequencing capability development, demonstrating that portfolio optimization can be expressed directly over the enterprise ontology. Earned Value Management (PV, EV, AC, CPI, SPI) provides the measurement substrate feeding optimization; technique effectiveness varies by stage (Earned Schedule predicts more accurately early, Earned Duration more reliably later), and uncertainty-aware variants (Grey Earned Duration Management completion bounds; Z-number based EVM combining possibility and reliability via fuzzy logic) feed the optimizer ranges rather than single-point estimates.

Reinforcement Learning. Deep reinforcement learning enables dynamic adaptive optimization under regime shifts. DDPG, PPO, and A2C optimize asset allocation by learning from market interactions and outperform traditional mean-variance methods. SLA-aware multi-objective reinforcement learning (SLA-MORL) achieves a 67.2% reduction in training time for deadline-critical jobs and a 68.8% reduction in costs for budget-constrained workloads while maintaining 73.4% SLA compliance improvement, with intelligent initialization and dynamic weight adaptation reducing initial exploration overhead by 60%. For mission-critical allocation, attention-augmented multi-agent RL with Decaying Attention Guidance achieves 34% faster convergence, a 51.5% +/- 1.7% improvement in cost-effectiveness over rule-based baselines, 94.9% +/- 0.6% critical-target elimination, sub-0.3-second response, and 95.2% resource utilization. RL agents also operate inside DES and agent-based environments for strategy exploration, connecting the Simulation and Optimization layers.

Resource Optimization. Dynamic resource allocation reaches 91.3% utilization with 99.2% demand satisfaction. Distributed AI control for procurement and supply chain (edge-to-cloud fusion, deep-learning forecasting, RL policies, federated multi-agent coordination) achieved a 22% improvement in data-fusion accuracy, a 75% reduction in coordination delays, and a greater than 90% decrease in procurement exceptions. ML-plus-MILP hybrids replace nonlinear constraints with trained models, validated on space logistics network optimization, wiring optimization back to the Simulation and Digital Twin layers. Across reported applications, AI-driven optimization yields 15 to 20% reductions in portfolio volatility, 30% faster rebalancing, 10 to 23% reductions in operational costs, and decision cycles cut from days to milliseconds.

Uncertainty Quantification and Sensitivity Analysis Feeding the Decision Layer

The framework propagates uncertainty rather than relying on point estimates, and it instruments every output so the Decision and Executive Support layers consume calibrated probabilities, not single values.

Surrogate-based uncertainty quantification. Sparse Polynomial Chaos Expansion (PCE) combined with Latin Hypercube Sampling achieved 40 to 60% computational time reductions versus traditional approaches while improving prediction accuracy by 15 to 25%; PCE is non-intrusive, requiring no code modification, so AEDA can wrap existing models. For high-dimensional architecture spaces, Partial Least Squares PCE (PLS-PCE) reduced a 37-variable electromagnetic design to converged sensitivity analysis with only 30 analysis points, and sparse PCE on a concrete face rockfill dam used 50 to 75% fewer samples with superior accuracy. UQ workflows report 40 to 94% computational reductions versus standard MC.

Global sensitivity analysis. Sobol' indices, first-order direct effects and total-order including interactions, computed from PCE surrogates enable variance decomposition at negligible cost after surrogate construction. AEDA makes Sobol'-index global sensitivity analysis a standing output so the Optimization and Decision layers can rank which architecture parameters actually drive output variability and prune the decision space before invoking expensive optimization.

Risk-aware decision calculus. Scenario-based two-stage stochastic optimization with LSTM-XGBoost forecasting (Monte Carlo Dropout plus quantile regression for UQ) outperformed deterministic and rule-based dispatch, achieving a ZAR 15 billion (0.9%) cost reduction with improved reliability (1625 MWh versus 3538 MWh load shedding) over a seven-day horizon. Risk appetite is encoded explicitly in the objective via Conditional Value-at-Risk (CVaR) and variance constraints rather than optimizing expected value alone, so the Decision Layer avoids extreme-case outcomes. Pre-posterior Bayesian value-of-information analysis quantifies the expected gain from acquiring data before irreversible decisions; for structural health monitoring, sensor location and configuration mattered more for decision value than sensor quantity, giving the Executive Support layer a normative basis for recommending whether to act or gather more data, and where to place new data sources.

Calibrated uncertainty as the human-in-the-loop trigger. Every model output is instrumented with calibrated uncertainty so low-confidence cases route to human review, preserving accountability and override. Probabilistic OCR with Monte Carlo Dropout achieved 99.13% accuracy with a mean confidence-interval width of +/- 1.22 for financial fields and an expected calibration error of 2.9%, flagging uncertain outputs for human review. A hybrid forecasting engine achieved 94.3% forecast accuracy at the 1-day horizon and 74.4% at 150 days, making horizon-dependent confidence explicit. Strategic Governance Intelligence couples sentence-embedding semantic similarity for redundancy detection, LLM-based strategic reasoning for explainable insight, and Monte Carlo probabilistic risk modeling, presenting radar-chart risk visualization and ISO 31000-compliant risk matrices to executives. Model-agnostic

explainability (SHAP, LIME) supports interrogation and regulatory traceability from raw data to recommendation.

Architectural Guidance

Build the Simulation layer as a DEVS-specified hybrid engine over a multifidelity MC hierarchy, with non-intrusive sparse PCE plus Latin Hypercube Sampling as the default surrogate-and-sampling layer and Sobol'-index sensitivity analysis as a standing output. Express the Optimization layer as a constrained mathematical program over the enterprise ontology, using capability value rather than asset cost as the objective, EVM telemetry (including uncertainty-aware GEDM and ZEVM variants) as the measurement substrate, and reinforcement learning or multi-objective RL for continuous rebalancing under cost, performance, and SLA constraints. Encode risk appetite with CVaR and variance constraints, couple Bayesian networks with value-of-information analysis, and instrument every output with calibrated uncertainty thresholds that route low-confidence cases to human review. This separation of concerns aligns with the layered enterprise AI integration pattern (LEAIM), which formally separates data acquisition, model lifecycle management, model serving, orchestration, and governance with explicit dependency constraints, and establishes governance, explainability, and compliance as built-in design requirements rather than after-the-fact controls.

Decision Layer, Explainability, and Governance

The Decision Layer is where AEDA's upstream computation, GraphRAG retrieval, agent swarms, the enterprise digital twin, simulation, and optimization, resolves into outputs a human decision authority can act on, defend, and audit. It is the point at which BEA/DoDAF DM2 documentation completes its evolution into a computational decision stack, and it is also the point at which the enterprise incurs its accountability obligations. This section specifies the decision-intelligence outputs the layer produces, the explainability and provenance controls that make those outputs defensible, the human-in-the-loop structure that preserves command authority, and the evidence base for the autonomous-enterprise end state.

Decision-Intelligence Outputs

The Decision Layer does not emit point predictions. Each decision object carries a structured payload that an executive or commander can interrogate:

- **Recommended actions.** Concrete, executable courses of action derived from the optimization layer. Natural-language-to-formulation translation (OptLLM-style) and solver-informed reinforcement learning let non-expert users formulate optimization problems and receive executable recommendations with explained trade-offs. The solver

functions as both executor and reward signal, so recommendations are grounded in a formal model rather than an unconstrained generative surface.

- **Risk scores.** Quantified risk derived from knowledge-graph reasoning and causal inference. Knowledge-graph-based risk management has identified hidden risk-propagation paths and delivered early warning, with a reported 35% improvement in decision-making efficiency in financial-services applications; knowledge-graph plus causal inference reached root-cause identification accuracy above 90% in quality-management settings.
- **Confidence intervals.** Probabilistic bounds on each recommendation, surfaced rather than hidden, so the decision authority can distinguish a high-confidence recommendation from a marginal one.
- **Sensitivity.** Identification of the inputs and assumptions to which a recommendation is most responsive, so operators understand which conditions would change the answer.
- **Alternative futures.** Multiple candidate outcomes rather than a single deterministic projection. The enterprise digital twin supports inverse inference through multi-objective Bayesian optimization: targets are specified and the policies required to reach them are derived, allowing the layer to present alternative futures and the policies that produce them.
- **Cost and schedule impacts.** Quantified operational consequences of each recommended action. Digital-twin-driven logistics has demonstrated a 2.1-day reduction in average delivery time, a 12 percentage point improvement in on-time delivery rates, and nearly 20% cost reduction while enhancing robustness to disruptions, illustrating the cost and schedule deltas this layer is intended to expose.

These outputs are produced inside a closed loop. Every agent decision, environmental response, and outcome is captured and fed back to refine decision policies, so the decision objects above are continuously revalidated rather than fixed at deployment.

Explainability Controls

Once decisions are produced by GraphRAG, agent swarms, simulation, and optimization, they must be explainable to government accountability standards. AEDA embeds explainability at the core of the lifecycle, not as an afterthought.

The governing tension is the explainability-performance paradox: the most performant models are inherently opaque, while high-stakes defense, eligibility, and law-enforcement-support contexts require both accuracy and interpretability. DARPA's Explainable AI program (2016-2021) framed this gap. The literature shows the paradox is navigable rather than absolute: a hybrid framework integrating rule-based models with deep learning, using Layer-Wise Relevance Propagation and SHAP, achieved 94.3% accuracy while reaching a trustworthiness index of 92.1%. AEDA therefore prefers hybrid intrinsically-interpretable-plus-post-hoc architectures.

Each Decision Layer output should carry:

- **Local explanations** (LIME, SHAP) attributing a specific recommendation to its driving features.
- **Global explanations** (SHAP) describing model behavior across the decision space.
- **Counterfactual explanations** identifying the minimal change in inputs that would change the recommendation.

Explanations are engineered to reduce, not increase, cognitive load, and to align with operator and domain reasoning. This is a deliberate design constraint: post-hoc explanations frequently increase cognitive load and misalign with domain reasoning, which degrades rather than supports the decision authority.

Provenance and Auditability

Beneath the Knowledge Graph and Data Sources layers, AEDA maintains a provenance and lineage substrate that makes every decision traceable from data source to inference. Provenance systems deliver machine-checkable evidence rather than narrative assurances.

- **Lineage and lifecycle evidence.** Immutable data lineage graphs and feature-store governance capture lifecycle evidence from ingestion to inference.
- **AI-Receipts-style ledger.** An AI Commit Ledger approach generates structured receipts per action; the reference implementation reported 94.2% attribution accuracy with less than 340ms overhead, so every agent action and decision is attributable and replayable for post-incident analysis.
- **Cryptographic provenance.** Blockchain-powered provenance with cryptographic verification, zero-knowledge proofs, and federated logging reduced fraudulent activities by more than 50% versus traditional audit approaches in simulation.
- **Fairness provenance.** An AI Fairness Provenance Record documents data origin, model choices, and bias metrics so auditors can run provenance-based fairness audits and trace any disparate decision to its source. Provenance-based auditing has surfaced statistically significant bias: in clinical models, logistic regression exhibited significant gender bias (EOD = +0.256, p = 0.0080) while a random forest's smaller disparity was not statistically significant. Fairness auditing is warranted: in automated resume evaluation, GPT-4 awarded higher scores to female candidates with comparable qualifications and lower scores to Black male candidates, with biases translating to 1-3 percentage point differences in hiring probability, and algorithmic fairness constraints often prove mutually exclusive, so improving one dimension can degrade another.

To make provenance consumable by the Executive Support tier, AEDA separates technical evidence generation from governance consumption through explicit emit, store, and query interfaces, and maintains a Technical-Regulatory Correspondence Matrix mapping regulatory anchors (EU AI Act, GDPR, and DoD and federal equivalents) to the concrete evidence artifacts the executive layer can surface on demand.

Assurance Cases

AEDA treats assurance cases as first-class artifacts: structured, evidence-backed arguments for the safety and ethics of its decisions, of the kind a DoD enterprise needs for certification.

- **Overarching Properties** (Intent, Correctness, Innocuity) align AI/ML component properties to system-level safety, following the aerospace approach.
- **Continuous Assurance.** A continuous-assurance workflow integrates design-time, runtime, and evolution-time assurance using formal verification, RoboChart for functional correctness and PRISM for probabilistic risk, and auto-regenerates assurance arguments when specifications or verification results change.
- **Ethics argument pattern.** PRAISE provides a principles-based ethics assurance argument pattern (justice, beneficence, non-maleficence, respect for human autonomy, with transparency supporting).

Two assurance philosophies inform how much trust AEDA places in its own components. The dependability perspective minimizes trust in AI/ML elements through defense-in-depth and a hierarchy of simpler guard systems and micro-Operational Design Domains; the trustworthy perspective applies assurance to the AI/ML elements themselves. For agent swarms and the digital twin, AEDA applies defense-in-depth: opaque AI/ML components are wrapped in simpler, verifiable guard layers and micro-ODDs, and full trust is reserved for points where assurance evidence supports it.

Risk-Tiered Governance and Human-in-the-Loop

AEDA implements risk-tiered governance in the Ontology Governance and Decision layers. Decision flows are classified as low, medium, or high risk. High-risk flows (eligibility decisions, law-enforcement support, biometric identification, defense targeting) are gated with named accountability ownership, meaningful human oversight, pre-deployment impact assessment, fairness testing, audit trails, security controls, enforceable procurement clauses for vendor accountability, and accessible grievance and review mechanisms.

Governance is not advisory. Governance effectiveness is empirically measurable: organizations combining explainable AI with empowered ethics boards experienced 48% fewer instances of bias and regulatory violations, and 35% fewer regulatory investigations when third-party audits were employed, while advisory-only ethics boards showed limited impact. Effectiveness requires integration into core decision-making.

Human-in-the-loop is realized through three cooperating agent classes mapped onto the Agent Swarm layer:

- **Orchestrating agents** perceive enterprise state and coordinate forecasting, risk, and optimization specialists.
- **Learning agents** extract causal insight from each decision episode.

- **Governance agents** monitor drift, constraint violations, and ethical boundaries, and trigger human oversight near decision boundaries.

For defense, aerospace, and critical-infrastructure deployments, AEDA is designed for human command-and-control subordination, with adversarial red-teaming and penetration testing, explainable autonomous cyber-defense layers, and recovery procedures that restore human control if autonomy fails. In simulated military network environments, autonomous cyber-defense systems using multi-agent reinforcement learning, natural-language processing, and rule-based reasoning outperformed conventional intrusion detection in detecting stealthy intrusions and lateral movement while maintaining explainability through integrated XAI layers.

Executive Support and Calibrated Trust

The Executive Support tier consumes the decision objects, explanations, provenance, and assurance arguments above and presents them to leadership under a calibrated-trust discipline. Calibrated trust is built through training, transparency, human-machine teaming, and robust data governance. AEDA guards against skill fade and over-reliance, surfaces explanations that align with domain reasoning, and preserves human agency rather than displacing it. Short-term hazards include skill fade and corrupted data; long-term risks involve adversarial tampering, and the design addresses both.

Value is demonstrated through measurable operational gains within privacy and equity constraints. Public-sector predictive analytics have delivered an 18% reduction in eldercare backlog waitlists within nine months, a reduction in crime response lead times from 14 days to under three hours, and an improvement in budget variance accuracy from plus/minus 15% to plus/minus 4%. These are the kinds of decision-quality gains the Executive Support tier is intended to surface and defend.

The Autonomous-Enterprise End State

The end state AEDA targets is the third phase of a three-phase maturity model. Phase 1, the Data-Driven enterprise, is human-centric with a data-to-insight lag of days to weeks; this is where BEA/DoDAF DM2 documentation sits today. Phase 2, the AI-Augmented enterprise, applies machine learning, predictive analytics, and robotic process automation, remains reactive, and compresses the lag to hours. Phase 3, the Continuously Optimized AI-Assisted enterprise, is a closed-loop agentic system performing real-time continuous optimization with no human intervention until performance degrades or objectives change. AEDA's computational decision stack is the Phase 2-to-Phase 3 evolution of the enterprise architecture.

The end state is supported by evidence rather than asserted. Autonomous Enterprise Decision Systems are validated as a coherent, empirically supported integration of knowledge graphs, ontologies, large language models, simulation engines, optimization algorithms, and digital twins into closed-loop architectures in which autonomous agents perceive, reason, and execute

with minimal human intervention. Retrieval-augmented generation raised enterprise SQL question-answering accuracy from 16% to 54%. Hybrid optimization combining transformer-based deep learning with rule-based reasoning and reinforcement learning achieved 15% better accuracy, 30% lower computational overhead, and improved resilience to anomalies versus conventional deep reinforcement learning.

Maturity in governance, not only in technology, separates outcomes. The Agentic AI Governance Maturity Model defines five levels across twelve governance domains; Level 4-5 organizations achieved 94.3% lower sprawl indices, 96.4% fewer risk incidents, and 32.6% higher effective task completion rates versus Level 1. The adoption barrier is principally organizational rather than technical: technical implementation of agentic AI consumes roughly 20% of effort while organizational change management, stakeholder alignment, and governance establishment consume roughly 80%, and infrastructure gaps, real-time data deficits, and skill shortages are root-level drivers while organizational resistance is a dependent outcome. AEDA therefore treats governance and explainability as deployment gates, not retrofits, and sequences adoption by closing root-level infrastructure, data, and skills gaps before attempting to overcome organizational resistance. The autonomous enterprise is reached by maturing the governance spine in lockstep with the computational stack, never ahead of it and never behind it.

Space Domain Extension and Ontology Mapping

The Autonomous Enterprise Decision Architecture (AEDA) is defined against the defense enterprise, where its substrate is the Business Enterprise Architecture (BEA) and the DoDAF DM2 metamodel rendered computational. The same ten-layer progression, from Data Sources through Ontology Discovery, Ontology Governance, Enterprise Ontology, Knowledge Graph, GraphRAG, Agent Swarms, Enterprise Digital Twin, Simulation, Optimization, and Decision/Executive Support, generalizes to the space domain without structural change. What changes is the content of each layer, not its shape. The space domain substitutes its own authoritative data sources, its own domain ontologies, and its own mission and decision-support semantics into the stack, while inheriting AEDA's central thesis intact: a formal, governed ontology is the trust-and-accuracy enforcement mechanism for every reasoning layer above it. This section sketches that generalization and is intended as the seed for a separate Space Decision Architecture paper, not as a complete treatment.

Why the Generalization Holds

AEDA's Enterprise Ontology layer establishes that ontology grounding is the mechanism that converts descriptive architecture documentation into machine-reasonable, auditable inference, with the supporting literature reporting that ontology-grounded knowledge graphs raised accuracy from 37% to 98% and reduced hallucination from 63% to 1.7% in clinical question answering (Ali, Taha, Morsey 2026). That result is domain-transferable in principle: it depends on the presence of a formal, governed ontology and a closed-loop validation pipeline (SPARQL

constraint queries plus a symbolic consistency check), not on any feature specific to the clinical domain. The space domain is a strong candidate for the same treatment because it is data-rich, safety-critical, multi-organizational, and already partially formalized through existing space ontologies. The Ontology Discovery literature names space explicitly as an emerging frontier, with ontology discovery efforts targeting NASA, ESA, and international space agency resources and addressing cislunar operations, satellite systems, and mission planning. AEDA's discovery-to-governance progression therefore has a live application surface in space without new methodology.

Mapping AEDA Layers to Space Enterprise Actors and Systems

The table below maps AEDA's substrate and selected layers onto representative space-domain actors and systems. The mapping is illustrative of where each enterprise function resides in the space ecosystem; it does not assert any particular integration or data-sharing arrangement among these organizations.

AEDA layer	Defense substrate	Space-domain counterpart
Data Sources	DoD authoritative systems	NASA mission and engineering data; SDA catalog and tracking feeds; STM observational and conjunction data; SCaN and LunaNet network telemetry; Space ISAC threat and anomaly reporting
Ontology Discovery	DoDAF/BEA documentation corpus	NASA, agency, and operator corpora on cislunar operations, satellite systems, and mission planning
Enterprise Ontology	BEA / DoDAF DM2 made computational	Governed Space Enterprise Ontology built on existing space object and orbital/SSA ontologies
Knowledge Graph	DoD enterprise KG	Resident space object, orbit, conjunction, and mission knowledge graph
Agent Swarms	DoD multi-agent layer	Conjunction-assessment, traffic-coordination, and anomaly-response agents sharing the space ontology as protocol

AEDA layer	Defense substrate	Space-domain counterpart
Decision / Executive Support	DoD decision layer	Conjunction mitigation, traffic management, and mission-assurance decision support

Representative space-enterprise actors and the function each anchors:

- NASA: mission, engineering, and science data sources, and the agency ontology-discovery corpus on satellite systems, cislunar operations, and mission planning.
- TraCSS (Traffic Coordination System for Space): the civil space traffic coordination function, a primary consumer of the Decision/Executive Support layer for traffic management.
- CARA (Conjunction Assessment Risk Analysis): conjunction screening and risk analysis, mapping to the Agent Swarm and Decision layers for collision-avoidance decision support.
- SDA (Space Domain Awareness): catalog, detection, and tracking feeds into the Data Sources and Knowledge Graph layers.
- STM (Space Traffic Management): the governing decision function over conjunction and traffic data, mapping to the Decision/Executive Support layer.
- SCaN (Space Communications and Navigation) and LunaNet: network and telemetry data sources, and, for cislunar operations, a discovery surface for the LunaNet interoperability and lunar-mission ontology.
- Space ISAC (Information Sharing and Analysis Center): threat, anomaly, and incident reporting feeding the Data Sources layer and the anomaly-response agents.
- JPL (Jet Propulsion Laboratory): deep-space mission, engineering, and science data sources, and a mission-planning discovery corpus.

Existing Space Ontologies as the Discovery Seed

AEDA does not require the Space Enterprise Ontology to be built from scratch. The Ontology Discovery layer is explicitly designed to reuse and align existing formal vocabularies rather than invent isolated ones, and the space domain already supplies candidates. Rovetto's Space Object Ontology and related orbital and space-situational-awareness ontologies (resources held at D:/Documents/MITRE/AI/AEDA) provide a formal starting vocabulary for resident space objects, orbits, and conjunction-relevant concepts. Under AEDA's discovery-to-governance flow, these existing ontologies enter as high-value seed inputs: the Discovery layer aligns and extends them against current NASA, agency, and operator data, and the Governance layer reconciles them with newly discovered concepts through ontology alignment, fusion with conflict resolution, and multi-agent consensus validation before promotion to the authoritative Space Enterprise Ontology. This is the same reuse principle AEDA applies to standardized upper and domain ontologies generally, reusing established vocabularies to guarantee semantic interoperability across organizational and system boundaries rather than inventing isolated vocabularies. The

combination of named space ontologies and the explicitly identified space ontology-discovery agenda (NASA, ESA, and international agency resources; cislunar operations; satellite systems; mission planning) constitutes the discovery seed for the space stack.

Proposed Canonical Space Enterprise Ontology Stack

We propose a layered Space Enterprise Ontology, mirroring AEDA's enterprise ontology construction and built on the same W3C standards stack (RDF/RDFS triples, OWL 2 DL/QL axioms and constraints, SPARQL for multi-hop query, with description logics providing decidable, auditable reasoning). The five tiers move from domain-neutral foundations to mission-specific and decision-support semantics:

1. **Foundation tier.** Domain-neutral upper-ontology and vocabulary constructs (the RDF/RDFS/OWL 2 backbone, with reuse of standardized upper and reference ontologies as in AEDA generally). This tier carries no space-specific content; it supplies the formal logical substrate and interoperability primitives.
2. **Enterprise tier.** The governed enterprise semantics inherited from AEDA: actors, organizations, capabilities, processes, and systems, the space-domain analog of BEA/DoDAF DM2 entities made computational. This tier represents the space enterprise as an enterprise, independent of any single mission.
3. **Space Domain tier.** The core space vocabulary seeded from existing space ontologies: resident space objects, orbits, orbital regimes, conjunctions, and space-situational-awareness concepts, sourced from Rovetto's Space Object Ontology and related orbital/SSA ontologies and aligned and extended through the Discovery and Governance layers. This is the tier that distinguishes the space stack from the generic enterprise stack.
4. **Mission tier.** Mission-specific concepts spanning cislunar operations, satellite systems, deep-space missions, and the network and navigation context (SCaN, LunaNet), drawn from the named space ontology-discovery agenda. This tier specializes the Space Domain tier for particular operational and mission contexts.
5. **Decision Support tier.** The semantics that drive the Decision/Executive Support layer: conjunction-assessment, collision-avoidance, traffic-coordination, and mission-assurance concepts that connect the lower tiers to executive decision outputs, mapping to the actor functions anchored by CARA, TraCSS, STM, SDA, and Space ISAC. Every output at this tier is required to carry a traceable logical justification path, satisfying the same explainability and auditability requirement AEDA imposes on its Executive Support layer.

Each tier is governed under the same discovery-to-governance discipline AEDA prescribes for the defense enterprise: candidate concepts from the Discovery layer are reconciled through

alignment, fusion, and multi-agent consensus validation, with human-in-the-loop checkpoints before promotion, and the resulting ontology serves as the consistency-enforcing schema for the Knowledge Graph, the semantic protocol for the Agent Swarms, and the grounding-and-explainability layer for space decision support. A full elaboration of this stack, including its formal axioms, its governance cadence, and its mapping to operational space decision workflows, is the subject of the forthcoming Space Decision Architecture paper for which this section is the seed.

Implementation Roadmap and Build Division

AEDA does not arrive as a single delivery. It is built in phases, each of which produces a usable artifact that the next phase consumes. The sequence follows the ten-layer stack from the bottom up: an enterprise ontology must exist before a knowledge graph can populate it, the knowledge graph must exist before GraphRAG can retrieve over it, retrieval must exist before agent swarms can reason against it, and so on through the digital twin, simulation, optimization, and decision layers. Each phase is scoped so that it stands on its own as a verifiable increment rather than a partial commitment to an unproven whole.

Two horizons govern the roadmap. The near-term horizon expands the existing single-layer BEA brain into a ten-layer corpus and restructures its schema. The FY27 horizon stands up the full ten-layer software build, the working computational decision stack. The phasing below marks each phase against these two horizons.

Build Division: Claude and Codex

The work divides along a clean seam. Claude packages knowledge and designs schema. Codex, the internal GPT-5.5 instance on MITRE GitLab, executes the brain integration and the ten-layer software build.

Concretely, Claude is responsible for the upstream, design-intensive work: curating and structuring the source corpus, authoring the DM2-to-OWL translation logic and the enterprise ontology schema, defining the knowledge-graph entity and relation model, specifying retrieval and agent-reasoning patterns, and producing the design packages that hand off to engineering. Codex is responsible for the downstream, execution-intensive work: wiring the brain integration, standing up the graph and retrieval services, and constructing the ten layers as running software inside the MITRE GitLab environment.

This seam holds across every phase. Where a phase has both a schema-and-knowledge component and a build component, Claude produces the former and Codex executes the latter.

Near-Term Horizon: Corpus Expansion and Schema Restructure

The existing BEA brain is single-layer. The near-term objective is to expand that corpus to cover all ten layers of the architecture and to restructure the schema accordingly. This work is design-led and corpus-led, and it is the precondition for the FY27 software build.

Phase 1: Enterprise Ontology MVP and DM2 to OWL. Establish the enterprise ontology as a minimum viable product and define the translation from the DoDAF DM2 data model into OWL. This phase moves enterprise architecture from documentation into a formal, machine-readable ontology and sets the semantic foundation every higher layer depends on.

- Claude packages: curate the DM2 source material, author the DM2-to-OWL translation logic, and design the enterprise ontology schema for the MVP.
- Codex builds: integrate the ontology into the brain and stand up the MVP ontology as a running artifact.

Phase 2: Knowledge Graph. Populate the enterprise ontology into a knowledge graph, instantiating entities and relations over the formal schema.

- Claude packages: design the knowledge-graph entity and relation model and prepare the corpus for population.
- Codex builds: construct and populate the knowledge graph service.

Phase 3: GraphRAG. Layer graph-aware retrieval over the knowledge graph so that queries traverse entities and relations rather than flat text.

- Claude packages: specify the retrieval patterns and the graph-grounded query design.
- Codex builds: implement the GraphRAG retrieval service against the knowledge graph.

These first three phases, together with the corpus expansion across all ten layers and the schema restructure, constitute the near-term horizon. They convert the single-layer BEA brain into a ten-layer ontology, knowledge graph, and retrieval surface.

FY27 Horizon: The Full Ten-Layer Software Build

The phases below stand up AEDA as running software, the computational decision stack in full. They are the FY27 horizon. Each continues to follow the build division: Claude designs and packages, Codex executes the build inside MITRE GitLab.

Phase 4: Agent Swarm. Stand up agent swarms that reason against the GraphRAG retrieval surface, decomposing and executing analytical tasks over the enterprise knowledge.

- Claude packages: specify agent roles, reasoning patterns, and orchestration design.
- Codex builds: construct and deploy the agent swarm against the retrieval layer.

Phase 5: Digital Twin. Build the enterprise digital twin, a live computational representation of the enterprise grounded in the knowledge graph and exercised by the agent swarm.

- Claude packages: design the digital twin model and its correspondence to the enterprise ontology.
- Codex builds: implement the enterprise digital twin.

Phase 6: Simulation. Add a simulation layer over the digital twin so that courses of action can be exercised against the enterprise representation before they are taken.

- Claude packages: design the simulation model and scenario structure.
- Codex builds: implement the simulation layer against the digital twin.

Phase 7: Optimization. Add an optimization layer that searches over simulated outcomes to identify preferred courses of action.

- Claude packages: design the optimization formulation and objective structure.
- Codex builds: implement the optimization layer against the simulation outputs.

Phase 8: Decision Layer. Stand up the decision layer and executive support surface, where optimized courses of action are presented for decision and the stack delivers executive-grade support.

- Claude packages: design the decision-layer logic and the executive support presentation.
- Codex builds: implement the decision layer and executive support surface.

Phase 9: Autonomous Enterprise. Close the loop. With all prior layers in place, the stack operates as an autonomous enterprise decision architecture, moving from data sources through ontology, knowledge graph, retrieval, agent swarms, digital twin, simulation, and optimization to the decision and executive support layers as a single computational decision stack.

- Claude packages: design the closed-loop integration and the governance patterns that keep the autonomous stack accountable.
- Codex builds: integrate the full ten-layer stack into the autonomous enterprise decision architecture.

Summary

The near-term horizon is corpus and schema work: expand the single-layer BEA brain to all ten layers and restructure the schema, delivering the enterprise ontology MVP with DM2-to-OWL translation, the knowledge graph, and GraphRAG. The FY27 horizon is the full ten-layer software build: agent swarms, digital twin, simulation, optimization, decision layer, and the autonomous enterprise. Across both horizons the division holds. Claude packages knowledge and designs schema; Codex executes the brain integration and the ten-layer build inside MITRE GitLab.

Figure Inventory

Figure 1. Ontology Methods Timeline. Chronological mapping of ontology engineering and learning methods drawn from the ontology discovery and governance synthesis. Supports the Ontology Discovery and Ontology Governance layers of AEDA.

Figure 2. Techniques Taxonomy. Hierarchical classification of ontology construction and learning techniques from the ontology synthesis. Supports the Ontology Discovery and Enterprise Ontology layers.

Figure 3. Performance Comparison. Comparative view of ontology method performance across the dimensions reported in the ontology synthesis. Supports the Ontology Discovery and Enterprise Ontology layers.

Figure 4. Application Domains (Ontology). Survey of domains in which the surveyed ontology methods have been applied, from the ontology synthesis. Supports the Enterprise Ontology layer.

Figure 5. Challenges and Solutions. Pairing of recurring challenges with proposed solutions identified in the ontology synthesis. Supports the Ontology Governance and Enterprise Ontology layers.

Figure 6. GraphRAG Architecture. Reference architecture for graph-based retrieval-augmented generation from the GraphRAG synthesis. Supports the Knowledge Graph and GraphRAG layers.

Figure 7. Knowledge Graph Construction Pipeline. End-to-end pipeline for constructing a knowledge graph from source data, drawn from the GraphRAG synthesis. Supports the Knowledge Graph layer and connects upstream Data Sources to GraphRAG.

Figure 8. RAG Approaches Comparison. Comparison of retrieval-augmented generation approaches, including graph-based and conventional retrieval, from the GraphRAG synthesis. Supports the GraphRAG layer.

Figure 9. Application Domains (GraphRAG). Survey of domains applying GraphRAG and knowledge-graph retrieval, from the GraphRAG synthesis. Supports the GraphRAG and Knowledge Graph layers.

Figure 10. Best Practices Framework. Framework of best practices for knowledge-graph and GraphRAG implementation from the GraphRAG synthesis. Supports the Knowledge Graph and GraphRAG layers.

Figure 11. Performance Metrics. Metrics used to evaluate GraphRAG and knowledge-graph retrieval performance, from the GraphRAG synthesis. Supports the GraphRAG layer.

Figure 12. Uncertainty Quantification Methods Comparison. Comparison of uncertainty quantification methods from the decision-science synthesis. Supports the Simulation, Optimization, and Decision layers.

Figure 13. Bayesian Decision Framework. Structure of a Bayesian approach to decision-making under uncertainty, from the decision-science synthesis. Supports the Decision Layer and Executive Support layer.

Figure 14. Portfolio Optimization Evolution. Evolution of portfolio optimization methods drawn from the decision-science synthesis. Supports the Optimization and Decision layers.

Figure 15. Enterprise Decision Architecture. Integrated view of the enterprise decision stack from the decision-science synthesis. Supports the Decision Layer and Executive Support layer, and frames the AEDA stack as a whole.

Figure 16. Uncertainty Propagation and Sensitivity Analysis. Depiction of uncertainty propagation and sensitivity analysis across the decision pipeline, from the decision-science synthesis. Supports the Simulation, Optimization, and Decision layers.

Embedded Figures

Figure 1. `figure1_ontology_methods_timeline.png`

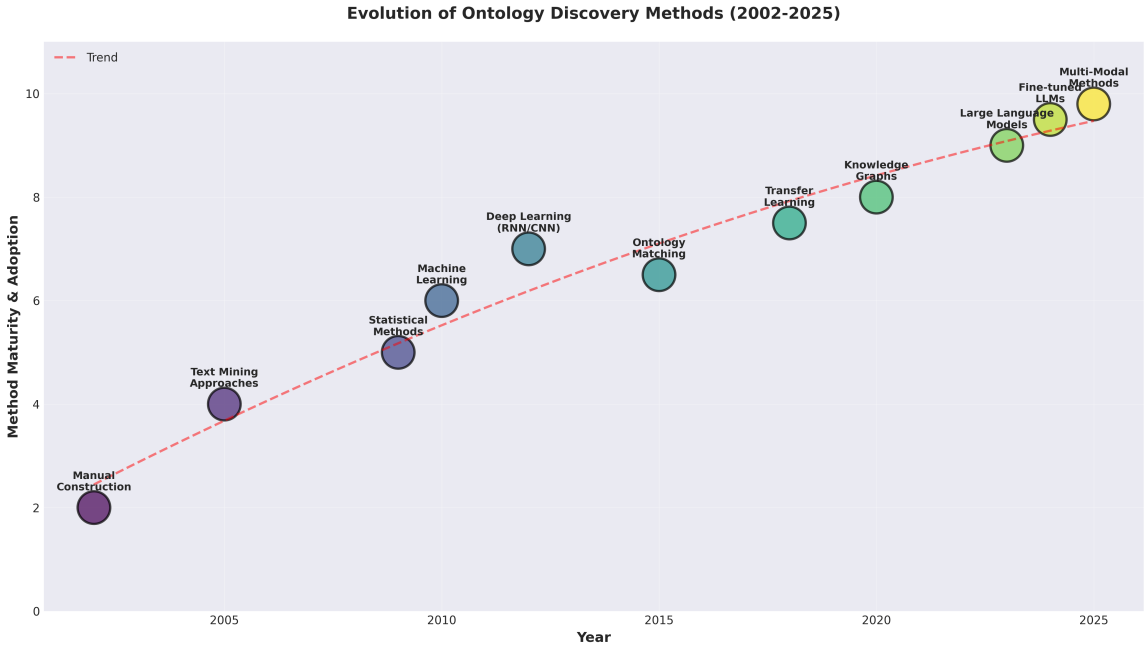


Figure 2. `figure2_techniques_taxonomy.png`

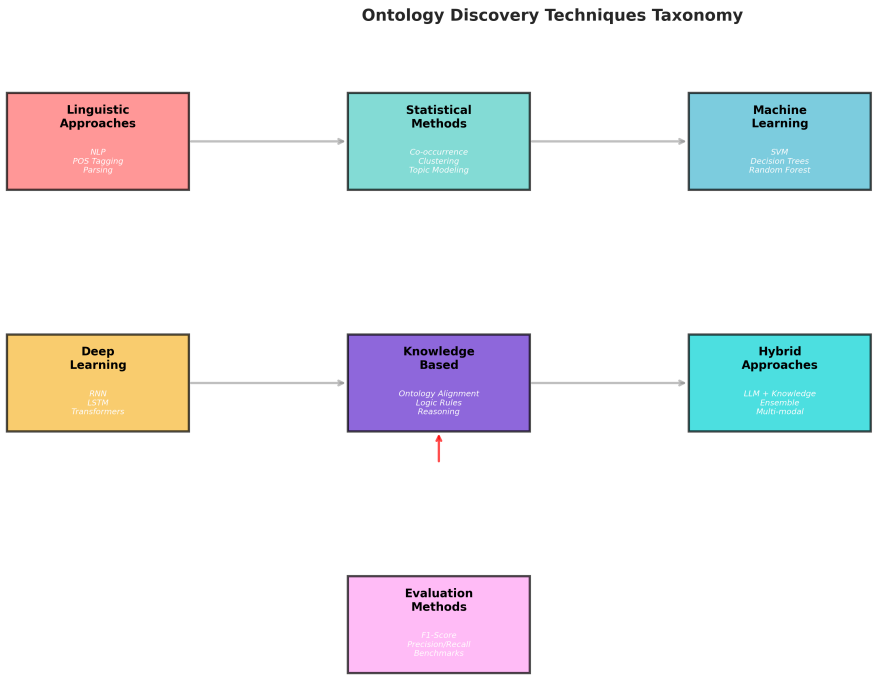


Figure 3. `figure3_performance_comparison.png`

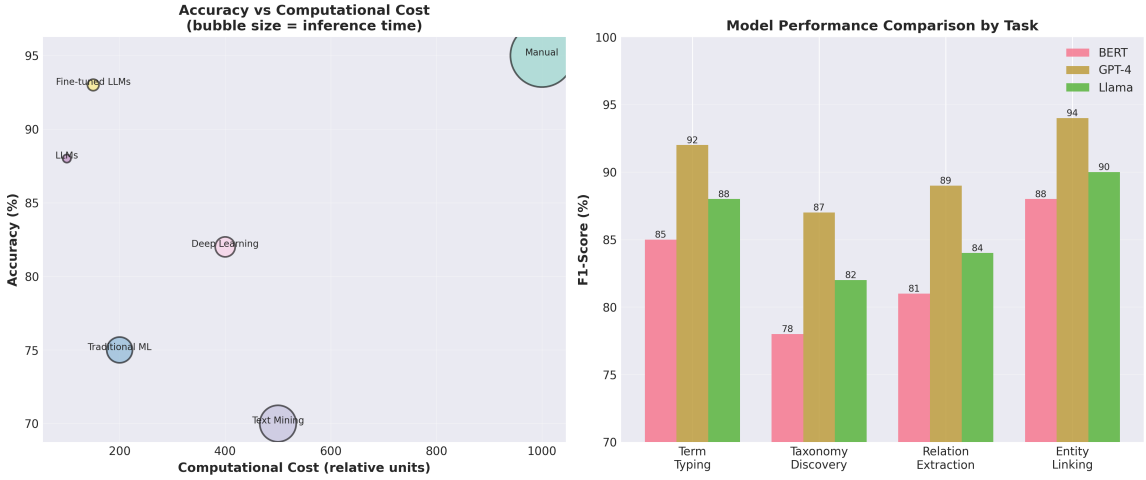


Figure 4. `figure4_application_domains.png`

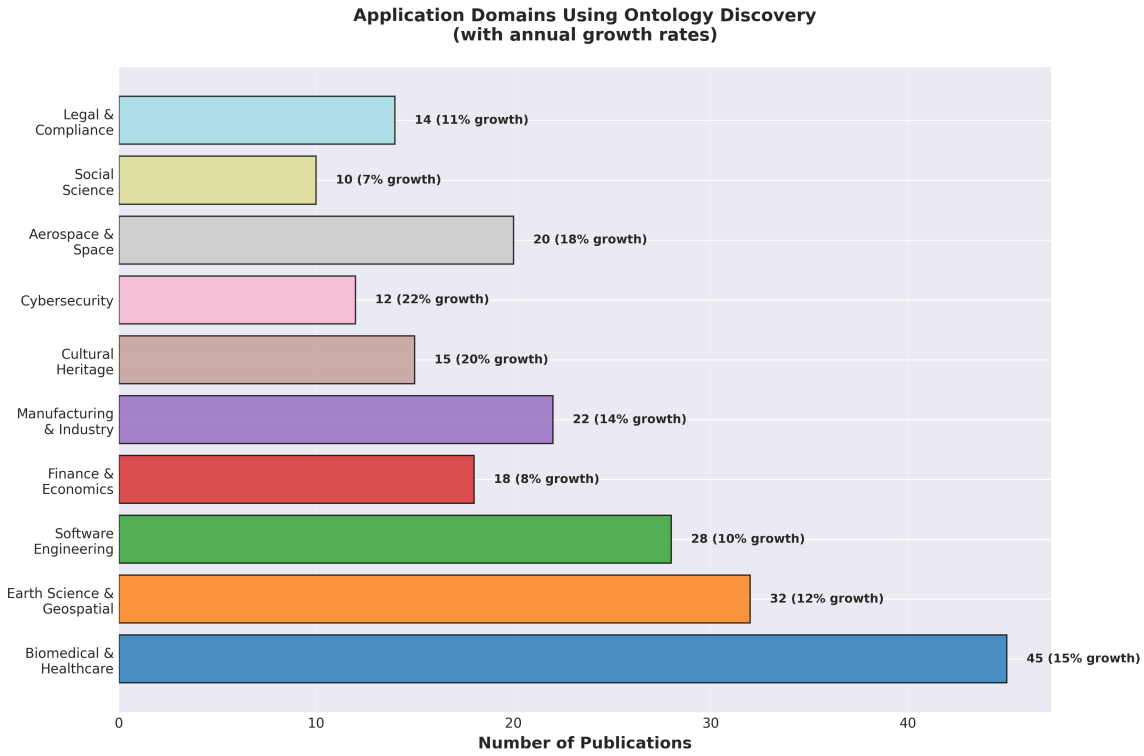


Figure 5. `figure5_challenges_solutions.png`

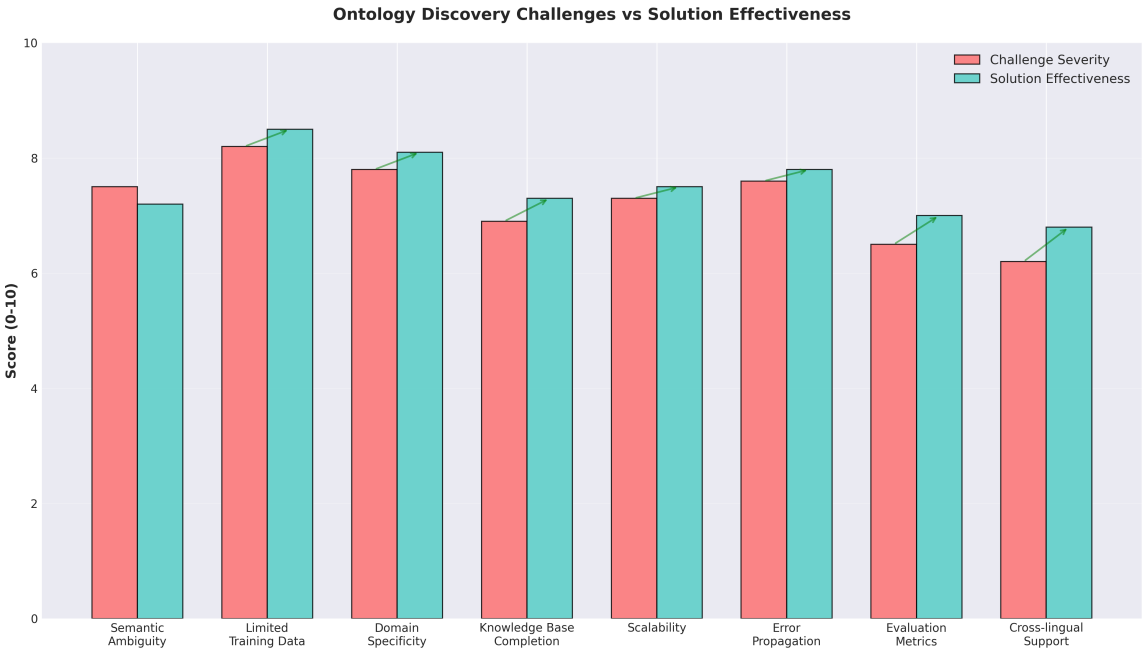


Figure 6. `graphrag_architecture.png`

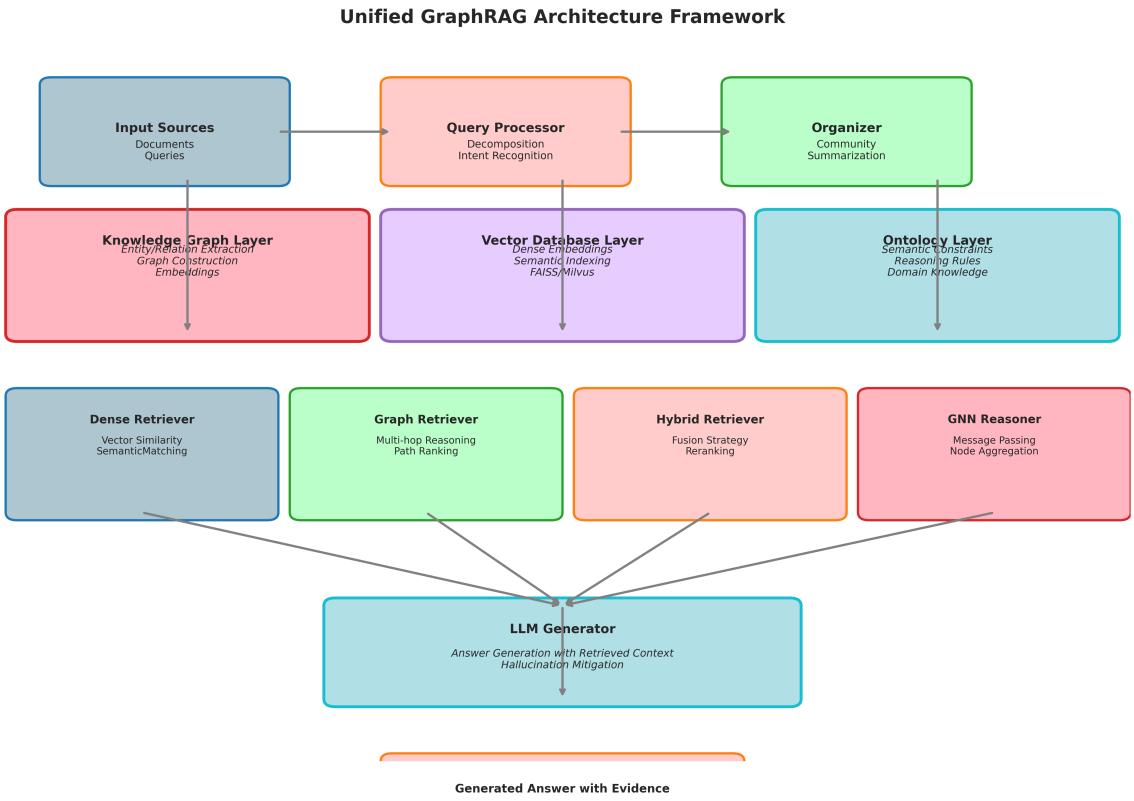


Figure 7. `kg_construction_pipeline.png`

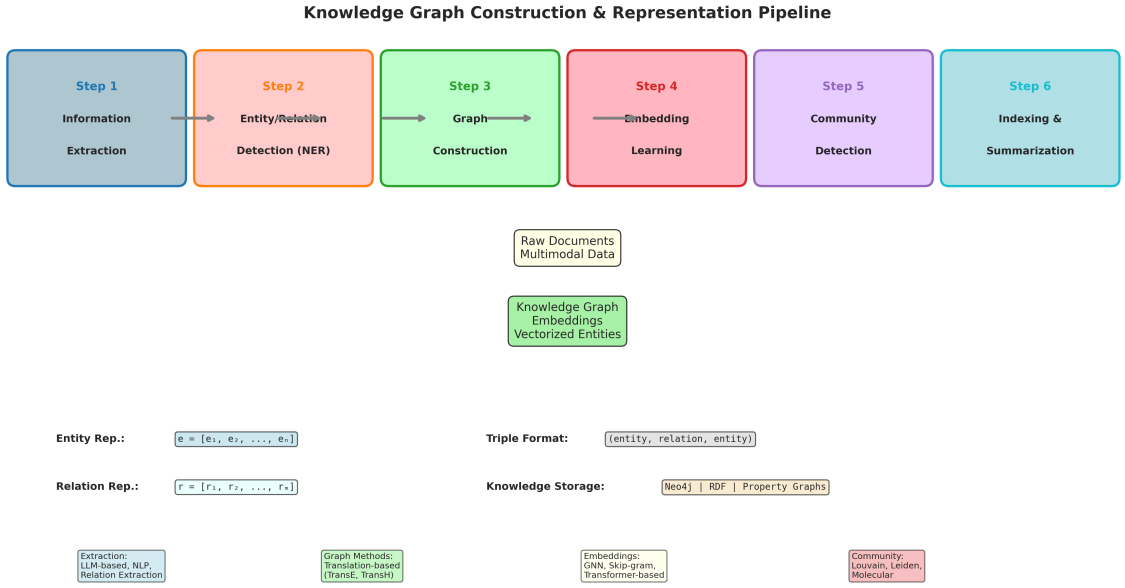


Figure 8. `rag_approaches_comparison.png`

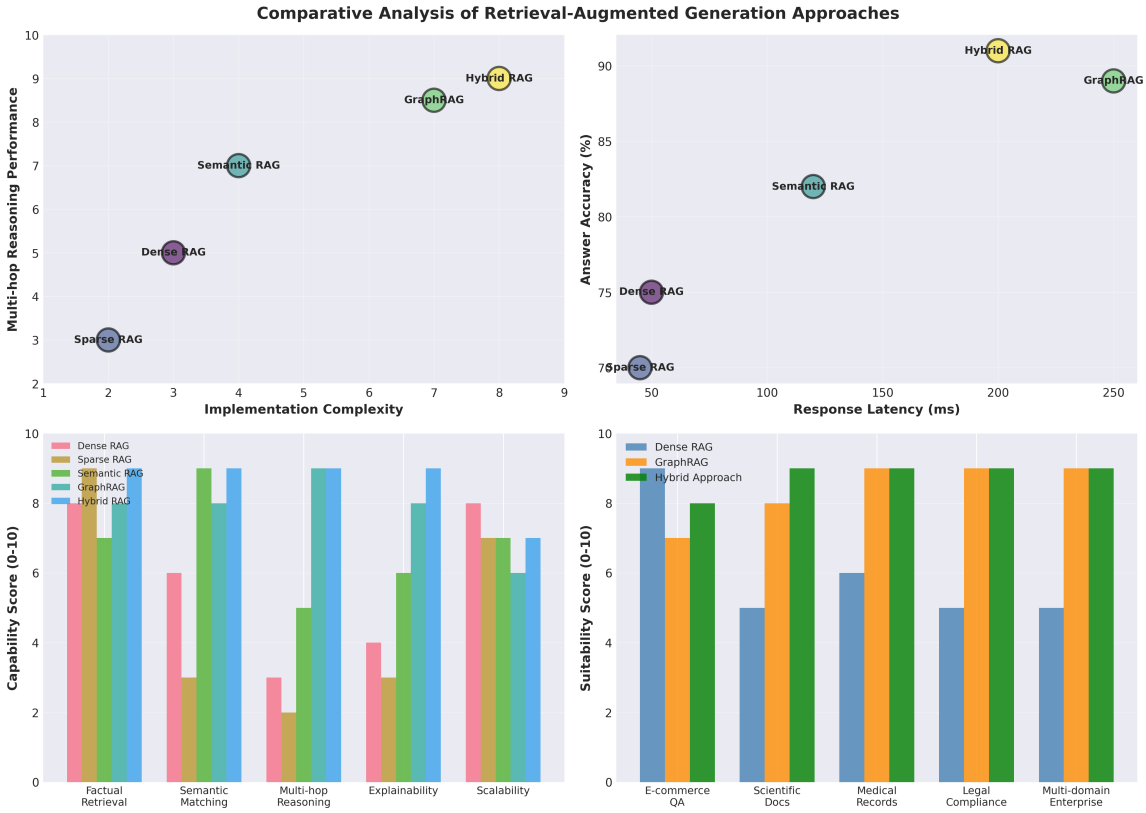


Figure 9. `application_domains.png`

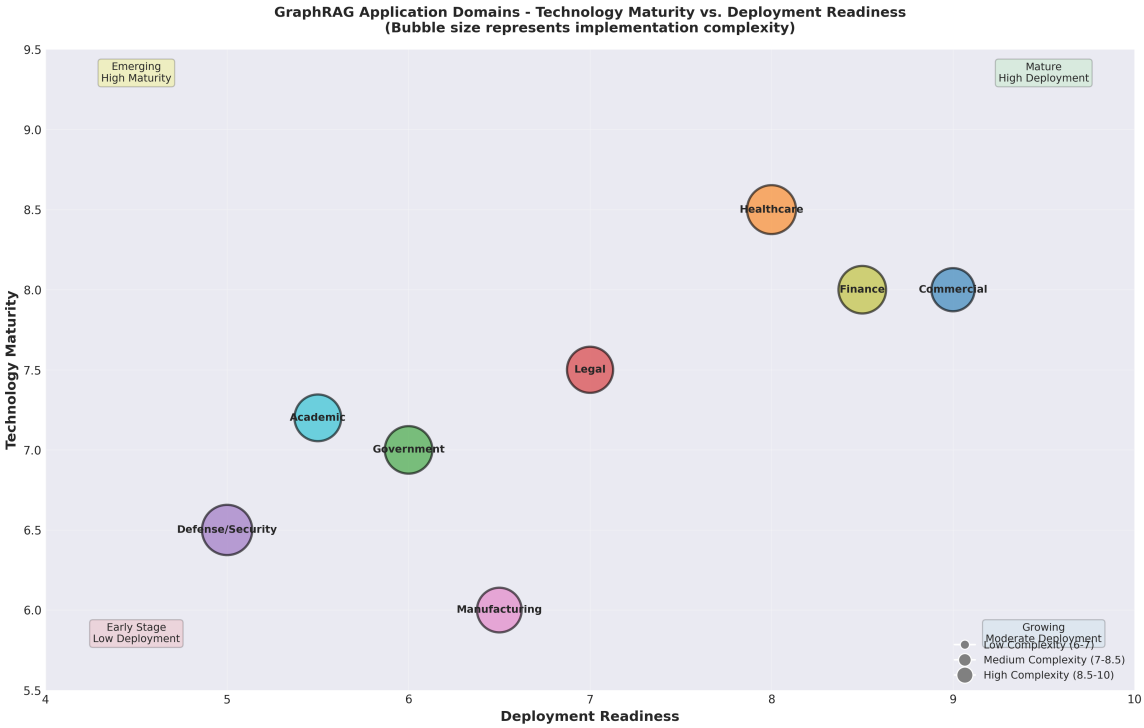


Figure 10. `best_practices_framework.png`

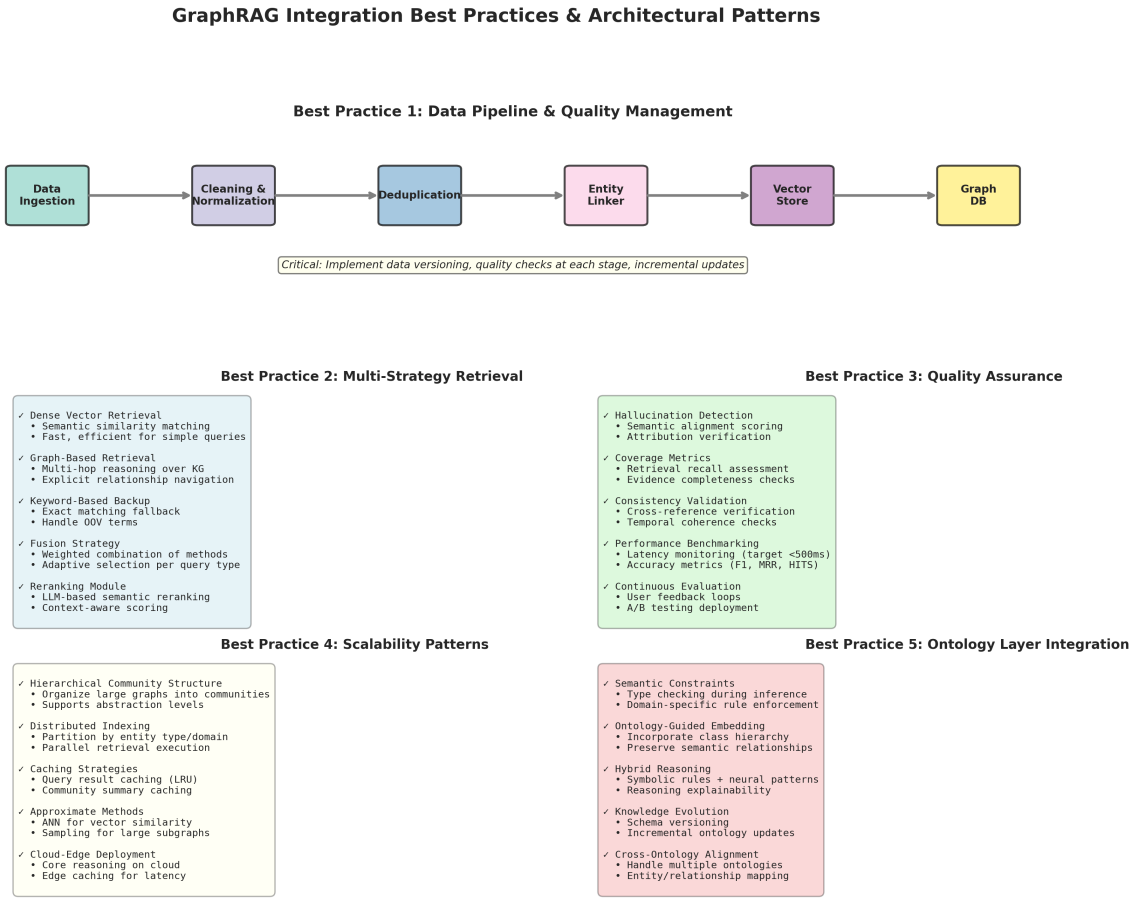


Figure 11. `performance_metrics.png`

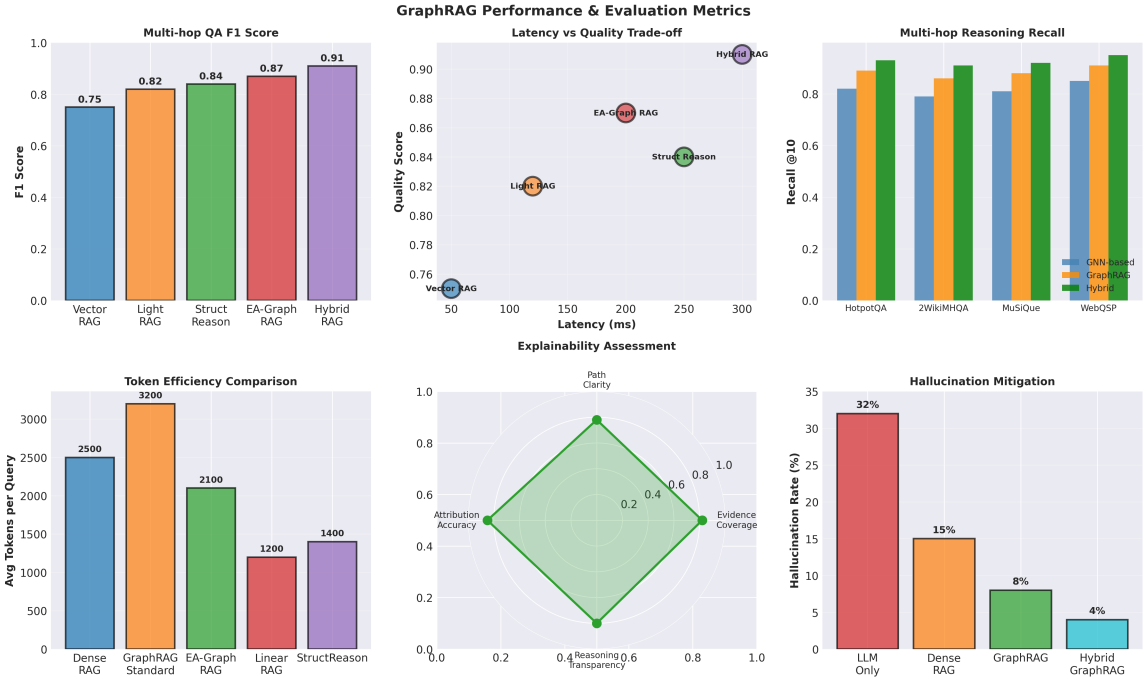


Figure 12. `01_UQ_Methods_Comparison.png`

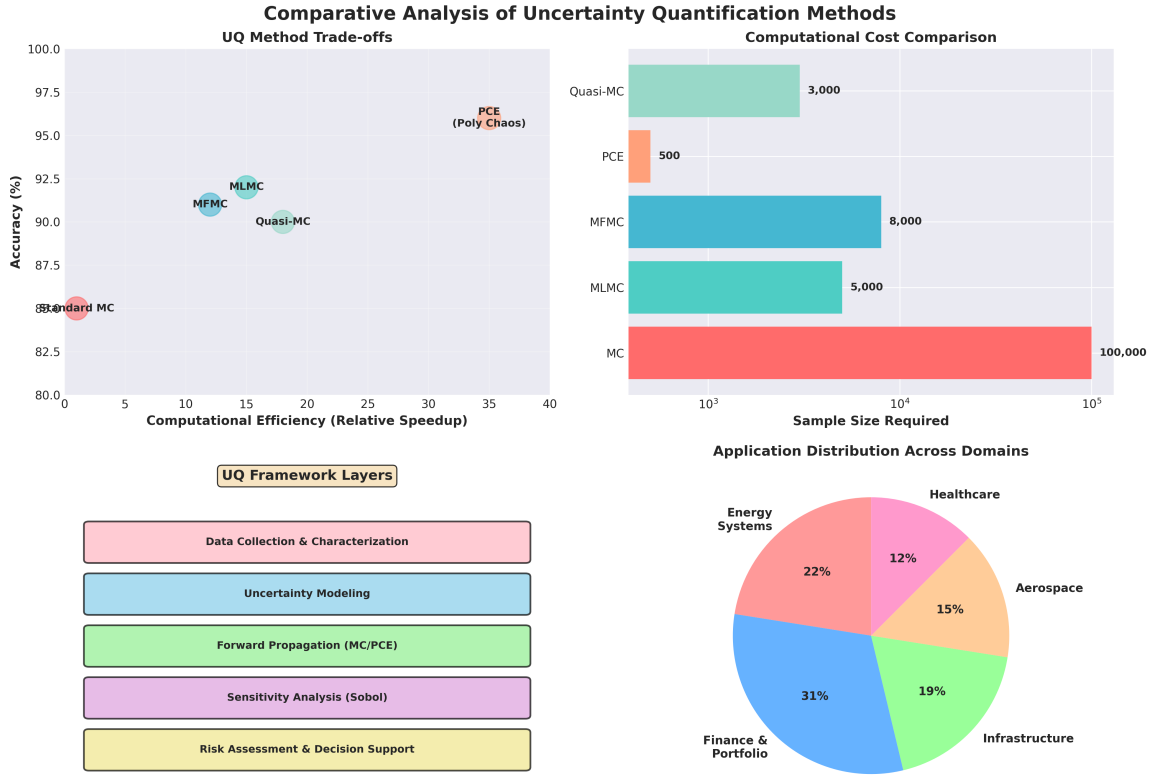


Figure 13. `02_Bayesian_Decision_Framework.png`

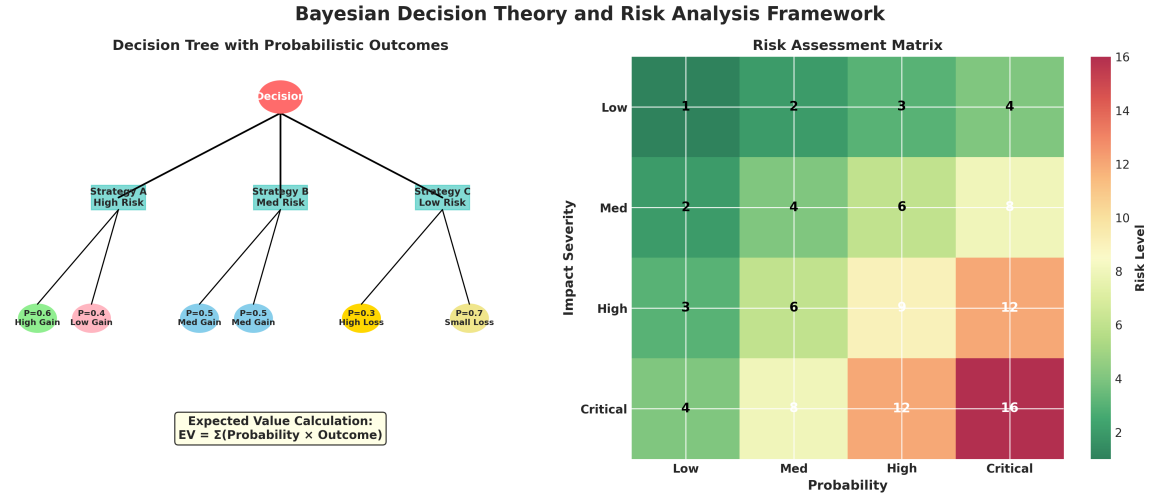


Figure 14. `03_Portfolio_Optimization_Evolution.png`

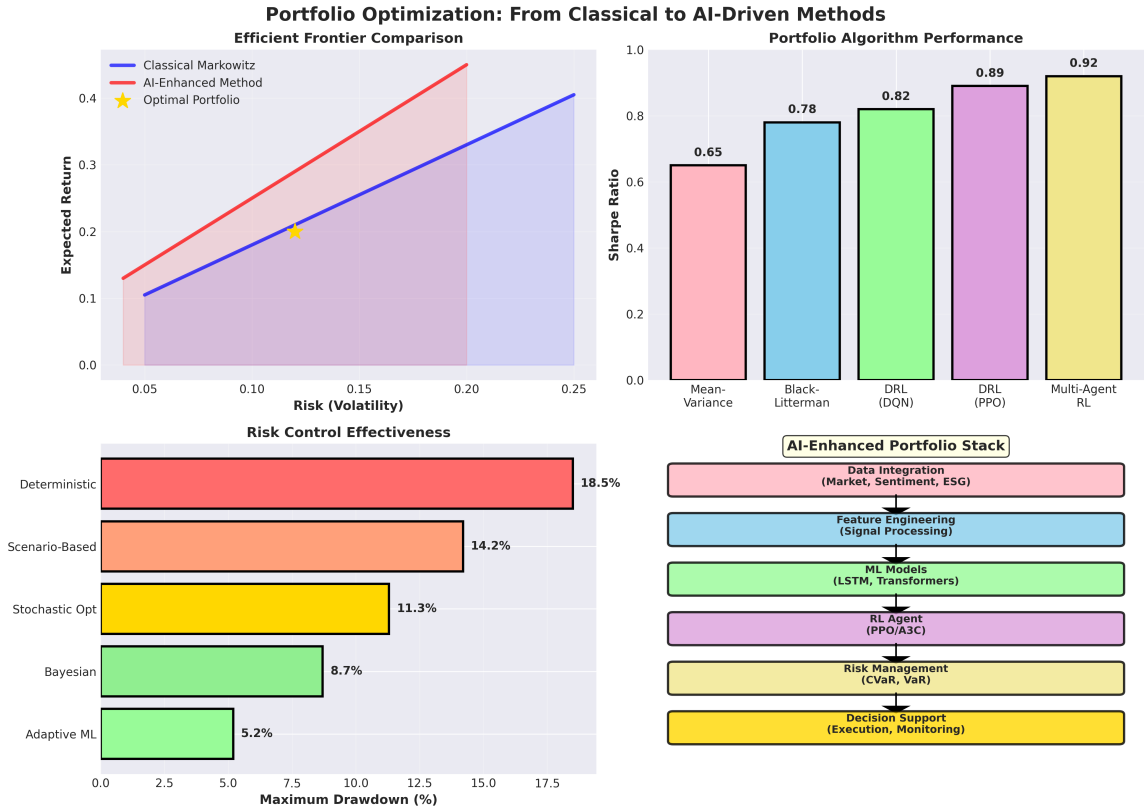


Figure 15. `04_Enterprise_Decision_Architecture.png`

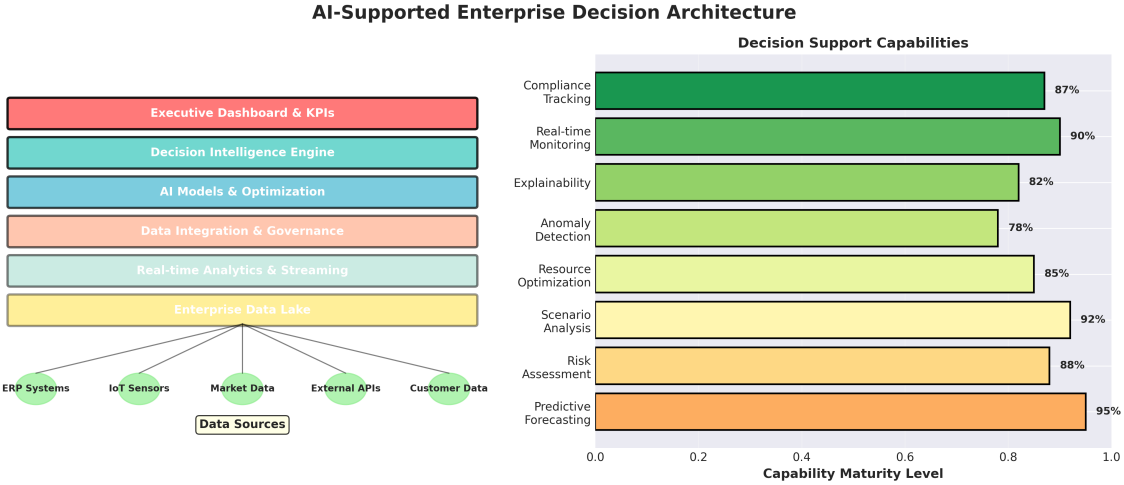
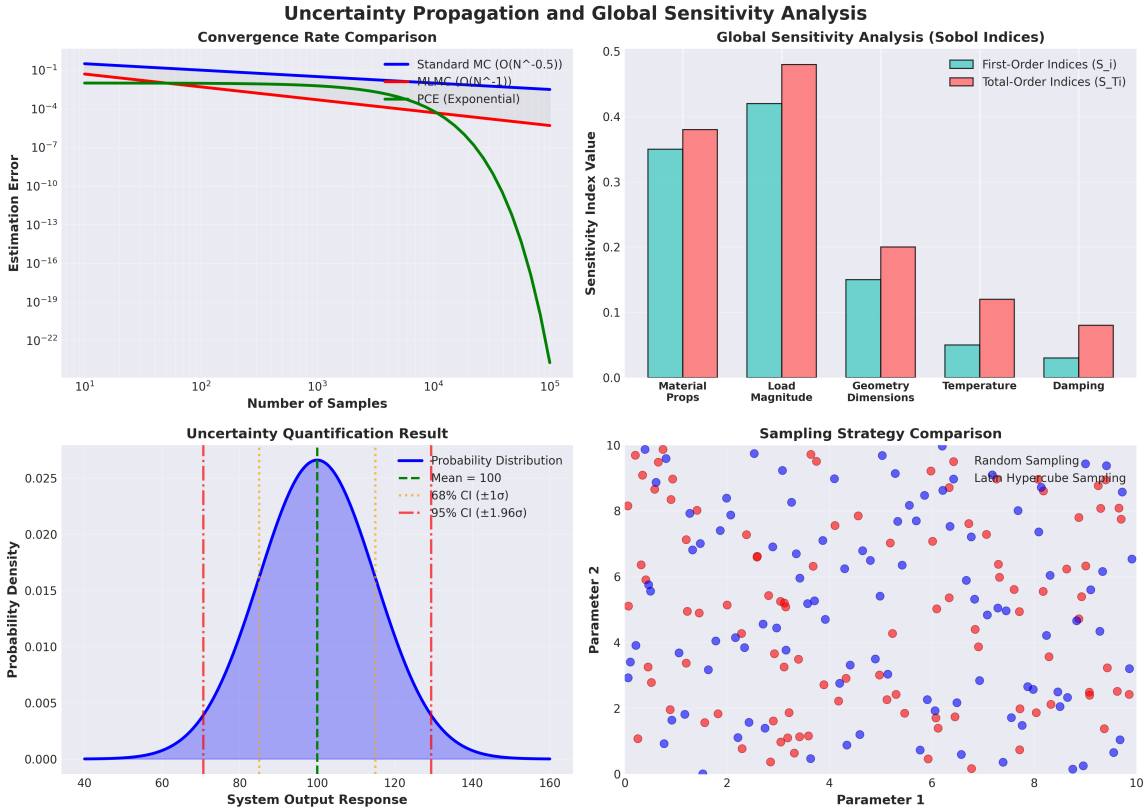


Figure 16. `05_Uncertainty_Propagation_SA.png`



Codex Implementation Instructions

This package is canonical project memory. Treat it as the authoritative source of record for the Autonomous Enterprise Decision Architecture (AEDA) program. The brain lives in MITRE GitLab; Codex integrates from this package. The working baseline is `D:/Claude_Code/archi`.

Primary Objective

Produce two coupled deliverables: the first implementation-ready AEDA Reference Architecture specification, and the brain corpus expansion from its current scope to full ten-layer coverage. AEDA evolves DoD enterprise architecture (BEA, DoDAF DM2) from documentation into a computational decision stack. The specification must be precise enough to build against, not a conceptual overview. Every layer in the stack must arrive with defined interfaces, data contracts, and a path from the layer above it to the layer below it.

The ten-layer stack, in order, is: Data Sources, Ontology Discovery, Ontology Governance, Enterprise Ontology, Knowledge Graph, GraphRAG, Agent Swarms, Enterprise Digital Twin,

Simulation, Optimization, Decision Layer, Executive Support. Hold this ordering as the canonical reference throughout implementation.

Do Not Regenerate Research

The literature reviews are complete and are not to be reproduced, re-run, or re-summarized. All 11 AEDA syntheses and all 23 DCIO syntheses are final inputs. Treat them as fixed evidence. Cite from them; do not extend, replace, or supplement them with new searches unless an implementation decision is blocked by a gap that the existing syntheses demonstrably do not cover. If a gap exists, flag it for human review rather than filling it with fabricated evidence. Never invent metrics, systems, citations, or source claims beyond what the provided syntheses contain.

Implementation Priorities

Direct effort to the build, in this order:

1. **Ontology design.** Define the enterprise ontology that anchors the stack. Resolve scope, modularity, and the boundary between discovered and governed concepts.
2. **DM2 to OWL mapping.** Specify the formal mapping from the DoDAF DM2 metamodel to OWL. Document the translation rules, the constructs that map cleanly, and the constructs that require interpretation or do not map.
3. **Knowledge Graph schema.** Define the graph schema that materializes the enterprise ontology, including node and edge types, property contracts, and provenance.
4. **GraphRAG implementation.** Specify retrieval over the knowledge graph: indexing, query construction, grounding, and the contract that GraphRAG presents to the agent layer.
5. **Agent orchestration.** Define the agent swarm design: roles, coordination, task decomposition, and the interface between agents and both GraphRAG below and the digital twin above.
6. **Enterprise Digital Twin.** Specify the digital twin representation, its synchronization with the knowledge graph, and the state it exposes to simulation.
7. **Simulation.** Define the simulation layer driven by the digital twin: scenario construction, execution, and output contracts.
8. **Optimization.** Specify the optimization layer over simulation outputs: objective formulation, constraint handling, and solution interfaces.
9. **Decision Layer.** Define how optimization results become decision options surfaced to Executive Support, including traceability back through every lower layer.

Working Constraints

- Keep this package as the single canonical reference. Update it in place as decisions are made; do not fork parallel memory.
- Maintain the MITRE technical register: precise, faithful to source, no em dashes.
- Preserve full provenance. Every architectural assertion must trace to a layer interface or to one of the completed syntheses.
- When a design decision requires a judgment the syntheses do not settle, surface it explicitly for human decision rather than resolving it silently.

Appendix A. Research Corpus Inventory (AEDA-track reviews)

Eleven AEDA-track systematic reviews, one per layer plus ontology discovery.

LR01: AI-Native Enterprise Architecture: Comprehensive Literature Review and State-of-the-Art Assessment

AEDA layer: Foundation: AI-native EA, DM2/DoDAF/MBSE

Scope. State-of-the-art assessment of AI-enabled enterprise architecture, model-based systems engineering (MBSE), and digital engineering. Examines the documentation-centric EA/SE frameworks (DoDAF, DM2, UAF, NAF, TOGAF, FEAF, BEA) and assesses how LLMs, GraphRAG, knowledge graphs, ontology-based reasoning, agentic systems, and digital twins can convert static architecture artifacts into dynamic, reasoning-based decision-support systems. Synthesizes 50+ peer-reviewed publications from 2022-2026; maps directly to the AEDA foundation thesis that EA must evolve from documentation into a computational decision stack.

Key findings. - Framework gap thesis: DoDAF, DM2, UAF, NAF, TOGAF, and FEAF excel at specifying WHAT to document (viewpoints, metamodels, SysML traceability) but provide limited guidance on HOW to reason over architectural artifacts for decision support. This gap is where AI-augmented approaches create transformative value, shifting EA from periodic review cycles (typically quarterly or semi-annual) to continuous near-real-time governance. - MBSE + GraphRAG question-answering: a proof-of-concept GraphRAG-enhanced approach to MBSE model retrieval achieved 93% accuracy in answering complex system architecture questions [4]. LLM+RAG over MBSE model APIs lets engineers create, query, and analyze SysML models through conversational interfaces rather than specialized modeling languages [2]. - GraphRAG outperforms vector-only RAG on multi-hop reasoning: on HotpotQA and MuSiQue, GraphRAG systems achieve F1 scores exceeding 0.59 versus 0.40-0.45 for vector-only approaches [15], by capturing reasoning paths across multiple sources and reducing information dilution. - Automated knowledge-graph construction: the RAKG framework achieves 95.91% accuracy on medical ontology mapping, a 6.2 percentage point improvement over traditional

GraphRAG, using pre-extracted entities as RAG queries to preserve global context under token limits [17]. LinearRAG constructs lightweight hierarchical graphs using only entity extraction and semantic linking, achieving linear scalability with corpus size and reducing token consumption by 85% versus relation-intensive approaches [18]. - Hallucination mitigation and detection: GraphRAG grounding reduces but does not eliminate hallucination; attention-pattern and semantic-alignment detection achieves 90%+ AUC for identifying hallucinated outputs [19], enabling guardrails (flag for human review, regenerate evidence, escalate). Ontology constraints (e.g., SNOMED CT, ICD-11) act as active governance mechanisms, not passive references [20, 42]. - Agentic enterprise governance metrics: a production agentic governance implementation reported Governance Efficiency of 0.961 (policy adherence rate), Resilience Index of 0.993 (availability under adverse conditions), and Downtime Reduction of 0.988, substantially outperforming traditional review-based governance ($p < 0.0001$) [26]. Autonomous multi-agent systems across supply chain, manufacturing, and ERP show 15-47% improvements in decision cycle time and operational throughput; hybrid BDI plus market-like coordination proved optimal [25]. - Semantic integration and retrieval gains: generative-AI semantic integration across hybrid SAP landscapes delivered 73% reduction in semantic mapping errors versus manual mappings [21]. TagRAG (tag-guided hierarchical KG retrieval) achieved a 78.36% winning rate against traditional GraphRAG while reducing storage 14.6x and retrieval time 1.9x [23]. GraphRAG-R1 (process-constrained RL) reached state-of-the-art multi-hop performance while reducing computational cost by 30% [24]. - Digital twin operational intelligence: SysML Opaque Actions can embed lightweight automation to pull telemetry and update model properties for model-centric visibility [31]. Federated supply-chain DTs propagate ripple effects through 50 simulation scenarios at 98% coverage [35]. A cloud-edge collaborative DT for renewable energy achieved 3-order-of-magnitude simulation-efficiency improvement (10 days to 182.6 seconds), modeling error below 0.6%, and a three-fold reduction in equipment unplanned outages [36]. - Maturity-readiness correlation: EA maturity assessed for AI-readiness correlates strongly with successful scaled AI deployment (Spearman rho = 0.88) using integrated TOGAF-IndEA assessment; transformative-level architecture (Maturity Score 4.75/5) achieves 3-4x faster AI deployment timelines than basic maturity (2.25/5) [7]. - Persistent challenges: a systematic review of 149 primary studies found organizations lack common understanding of how formal V&V of behavior fits MBSE lifecycles, with highly variable approaches [37]. Hallucination is partially irreducible; knowledge graphs require continuous curation; ontologies drift as business terminology evolves; multi-stakeholder governance of shared semantic resources remains organizationally hard [40].

Named frameworks/systems/standards. DoDAF (US DoD Architecture Framework), DM2 (Data Management 2.0), UAF (Unified Architecture Framework), NAF (NATO Architecture Framework), TOGAF + ADM (Architecture Development Method), FEAF (Federal Enterprise Architecture Framework), BEA (Business Enterprise Architecture), SysML, ISO/IEC/IEEE 15288 (lifecycle management), SEREA (Systems Engineering Reference Enterprise Architecture, INCOSE technical product, UAF-based), IndEA (integrated with TOGAF for maturity assessment), MBSE (Model-Based Systems Engineering), Digital Thread / Cognitive Digital Thread, GraphRAG (Graph-based Retrieval-Augmented Generation), RAG

(Retrieval-Augmented Generation), RAKG (document-level RAG knowledge graph construction), LinearRAG, TagRAG (tag-guided hierarchical KG retrieval), GraphRAG-R1 (process-constrained reinforcement learning), RAG-x (density-adaptive path sampling), OntologyRAG, SNOMED CT / ICD-11 (ontology constraints), AI-SME (AI Systems Modeling Enhancer, ChatGPT-enabled MBSE assistant), MBSA (Model-Based Safety Assessment), Cognitive Digital Twin (CDT), Autonomous Digital Process Twin, Systems of Systems (SoS) digital twin layered architecture, BDI (Belief-Desire-Intention) agent architecture, Seven Pillars of Agentic AI Implementation framework, Runtime constitutional framework (self-regulating AI agents), Cognitive Master Data Management (MDM), Enterprise Agentic Mesh, TRiSM (Trust, Risk, and Security Management) for agentic AI, AgriTrust (federated semantic governance framework), Benchmarks: HotpotQA, MuSiQue

Top sources. 1. V. Quast, G. Jacobs, S. Dehn, G. Hoepfner. Enabling humans and AI systems to retrieve information from system architectures in model-based systems engineering. 2026. doi: 10.3390/systems14010083 2. T. Esho, C. Hoyt, J. Marshall, J. Gadewadikar. Artificial intelligence enabled systems engineering modeling with retrieval augmented generation. 2026. doi: 10.1002/sys.70032 3. H. Zhang et al.. RAKG: document-level retrieval augmented knowledge graph construction. 2025. doi: 10.48550/arXiv.2504.09823 4. A. M. Ogunmolu, A. D. Popoola, O. Z. Adesokan, A. A. Abdulmalik, S. A. Joseph. Runtime policy orchestration for autonomous industrial control and smart manufacturing systems: A unified framework for governance, compliance, and adaptive resilience. 2026. doi: 10.9734/jerr/2026/v28i41862 5. S. Raza, R. Sapkota, M. Karkee, C. Emmanouilidis. TRiSM for agentic AI: A review of trust, risk, and security management in LLM-based agentic multi-agent systems. 2025. doi: 10.48550/arXiv.2506.04133

Contribution to AEDA. This review establishes the foundational thesis for AEDA: that institutionalized DoD and federal EA frameworks (DoDAF, DM2, UAF, NAF, TOGAF, FEAF, BEA) are documentation-centric and stop short of computational decision support, and that LLM/GraphRAG/knowledge-graph/ontology/agentic technology can close that gap. It supplies the literature backbone for AEDA's early stack layers, anchoring Ontology Discovery/Governance and the Enterprise Ontology in ontology-driven semantic integration and automated KG construction, the Knowledge Graph and GraphRAG layers in measured multi-hop retrieval gains, the Agent Swarms layer in BDI/multi-agent governance results, and the Digital Twin/Simulation layers in cognitive and SoS digital-twin evidence. It also frames the Decision Layer and Executive Support tiers around the documented shift from periodic to continuous, near-real-time governance, and provides quantitative justification (rho = 0.88 maturity correlation, 93% MBSE QA accuracy, 0.961 governance efficiency) for treating EA as an operational decision capability rather than a planning artifact.

Design implications. - Ground every AEDA brain output in an authoritative knowledge graph: adopt GraphRAG over flat-text RAG for the GraphRAG layer, since graph-structured representation, query-aware subgraph traversal, and structure-aware integration enable the multi-hop reasoning DoDAF/DM2 systems-of-systems queries require (F1 $>$ 0.59 vs 0.40-0.45 vector-only). - Build the Enterprise Ontology and Knowledge Graph layers with LLM-assisted

automated construction (RAKG-style pre-extracted-entity queries; LinearRAG-style entity-only lightweight hierarchical graphs) to control token cost (85% reduction) and achieve linear scalability, rather than relying on manual ontology engineering that suffers knowledge decay. - Treat ontologies as active governance mechanisms in the Ontology Governance layer: embed domain constraints (analogous to SNOMED CT/ICD-11) directly into retrieval and generation so the constraints enforce factual accuracy at generation time, not as passive reference structures. - Install hallucination guardrails as a first-class brain component: attention-pattern/semantic-alignment detection at 90%+ AUC to flag low-confidence outputs for human review, trigger evidence regeneration, or escalate to specialized reasoning, since hallucination is partially irreducible even with KG grounding. - Architect the Agent Swarms layer on hybrid BDI plus market-like coordination and the Seven Pillars (perception-reasoning-action loops, shared coordination protocols, RL-based continuous learning, data governance, self-healing resilience, human-AI co-governance, scalable infrastructure); embed compliance policies as machine-readable rules so violations are detected in minutes, not weeks. - Connect formal architecture models to live telemetry for the Digital Twin and Simulation layers using SysML Opaque Actions (or equivalent) to pull monitoring data and update model properties, giving model-centric visibility of designed-versus-actual behavior and supporting SoS layered twins where constituent systems retain autonomy. - Make AI-readiness maturity assessment an explicit gate before scaling: because EA maturity correlates with AI deployment success (rho = 0.88) and transformative maturity yields 3-4x faster deployment, AEDA adoption should advance method maturity, tool interoperability, organizational capability, and governance frameworks in parallel. - Adopt the orchestrator pattern at the Decision/Executive Support layers: position architects and executives as orchestrators of AI reasoning with explainability as a foundational constraint and auditable evidence trails linking decisions to reasoning, preserving human decision authority and regulatory-audit readiness.

Risks/limitations. - Hallucination is partially irreducible: even with knowledge-graph grounding, LLMs occasionally generate responses inconsistent with retrieved evidence; multi-agent chains add cascading hallucinations, agent-coordination failures, and systematic bias amplification through RL (TRiSM-for-agentic-AI concerns) [40, 42]. - Knowledge-graph and ontology maintenance burden: KGs require continuous curation and ontologies drift as business terminology evolves; large-scale production deployments expose this as a persistent operational cost [40]. - MBSE V&V immaturity: a systematic review of 149 studies found most organizations lack a common understanding of how formal verification of behavior fits the MBSE lifecycle, with highly variable approaches that rarely address simplifications, assumptions, and tool-integration challenges [37]. - AI-generated models are imperfect: LLM-produced MBSE models frequently exhibit redundancy, lack cohesiveness, and require significant manual refinement [11]. - Multi-stakeholder semantic governance is organizationally hard: governing shared semantic resources across business units, regions, and platforms remains an unsolved organizational challenge, not merely a technical one [40]. - Standardization gaps for agentic systems: protocols for multi-agent coordination, security-attestation mechanisms for autonomous decision-makers, and observability standards for distributed reasoning remain largely undefined; the Enterprise Agentic Mesh is proposed but broad standardization is still needed [41]. - Fragmented EA

standards: DoDAF, UAF, NAF, and TOGAF each define distinct metamodels and viewpoints, so AEDA must reconcile divergent metamodels and lacks standardized formats for representing architectural reasoning across them [95]. - Adoption barriers persist: MBSE tools present steep learning curves and data-accessibility issues for non-technical stakeholders, and cloud-migration strategy effectiveness depends on contextual factors (regulation, technical debt, cloud maturity, business objectives) [1, 19, 38].

LR02: Ontology-Grounded Reasoning Systems

AEDA layer: Enterprise Ontology layer

Scope. Reviews ontology-based AI, semantic web technologies (RDF/RDFS, OWL, SPARQL), description logics, ontology engineering methodologies, enterprise ontologies and model-driven engineering, and the integration of formal ontologies with large language models, retrieval-augmented generation, neuro-symbolic reasoning, and agentic architectures. Evaluates how formal semantic representations enable architectural inference, impact analysis, and mission-level decision support. Maps to the AEDA Enterprise Ontology layer (formal, governed semantic backbone of BEA/DoDAF DM2 turned computational), spanning upstream into Ontology Governance and downstream into Knowledge Graph, GraphRAG, Agent Swarms, and Decision/Executive Support layers.

Key findings. - Ontology-grounded knowledge graphs deliver the review's headline quantitative result: in clinical question answering, ontology grounding achieved 98% accuracy versus 37% for baseline ChatGPT-4, and reduced hallucination rates from 63% to 1.7% (Ali, Taha, Morsey 2026, ref [15]). This is the single strongest empirical case for ontology grounding as a hallucination-mitigation and accuracy mechanism. - Description Logics (DLs) are the formal foundation: decidable fragments of first-order logic balancing expressive power with computational tractability, providing the logical underpinning of the W3C OWL standard and enabling sound, complete reasoning (satisfiability checking, entailment) with well-defined semantics required for explainability and auditability in mission-critical systems. - Semantic-web-enabled enterprise architecture produced measured operational gains: in multi-school curriculum-governance case studies, semantic-web-driven EA frameworks reduced documentation inconsistencies by up to 24% and improved query efficiency by approximately 50% through structured semantic representations (Silalahi, Indrajit, Mantoro 2025, ref [14]). - The RDF/RDFS/OWL/SPARQL stack supports incremental adoption: organizations start with basic RDF triple graphs (subject-predicate-object), add RDFS class/property vocabulary, then advance to OWL 2 profiles (OWL 2 DL for full expressiveness, OWL 2 QL optimized for data access) as capability matures. SPARQL enables multi-hop reasoning, filtering, and aggregation over the graph. - Ontology-mediated query answering (OMQA) provides virtual semantic access to relational/legacy databases via ontological mappings and query rewriting, avoiding full materialization of databases as RDF while maintaining data currency (Xiao, Corman 2021, ref [13]). This is directly relevant to wrapping existing DoD authoritative data sources. - Neuro-symbolic integration is the dominant architectural pattern for combining

neural learning with symbolic rigor. Compiling DL ontologies into probabilistic circuits yields three capabilities: synthetic dataset generation capturing ontology semantics, GPU-accelerated deductive reasoning with runtimes substantially faster than traditional reasoners, and neuro-symbolic classification producing predictions consistent with ontological constraints (Lazzari, Presutti, Vergari 2026, ref [18]). - LLMs can automate knowledge-graph construction from unstructured text (entity extraction, relationship inference, semantic enrichment), and when paired with ontology-aligned RDF/OWL schema generation and multi-LLM consensus validation produce interpretable, SPARQL-compatible knowledge graphs (Das et al. 2026, ref [16]). Multi-agent LLM frameworks with adaptive ontology mapping to standards (SNOMED CT, ICD-11) preserve semantic interoperability at scale (Ho et al. 2024, ref [17]). - Hallucination mitigation works through multi-layered validation: formal ontologies encode ground truth, SPARQL queries the constraints, symbolic reasoners (e.g., HermiT) verify logical consistency, and LLM candidate outputs are filtered through semantic constraints in an iterative closed-loop so only factually grounded outputs reach users. - Architectural inference and impact analysis are enabled by dynamic multi-layer/hierarchical knowledge graphs that automatically parse change-propagation paths and identify critical elements affected by engineering changes; graph-aware learning (graph neural networks over enterprise data) predicts system impacts with high accuracy, outperforming traditional approaches by capturing essential domain semantics. - In financial services, ontological knowledge graphs (ontology-based data ingestion + KG building + rule-based inference + real-time dashboards) automated Basel III liquidity / regulatory compliance reporting with real-time reporting, semantic interoperability, auditability, accuracy improvements, reduced latency, and significant reductions in data-reconciliation errors versus traditional systems (Katipelly, Thalary 2024, ref [8]). - Explainability is a primary motivation, not a byproduct: formal logic enables traceable reasoning chains and natural-language explanations with logical justification paths, delivering transparency required for regulated, high-stakes domains and improving logical coherence, reasoning accuracy, and explanation fidelity over purely neural baselines.

Named frameworks/systems/standards. RDF (Resource Description Framework), RDFS (RDF Schema), OWL / OWL 2 DL / OWL 2 QL (Web Ontology Language), SPARQL (SPARQL Protocol and RDF Query Language), Description Logics (DLs), HermiT reasoner, Protege / WebProtege / VocBench, Uschold-Gruninger methodology, Guarino formal classifications, CIDOC CRM, SKOS, Ontology-Mediated Query Answering (OMQA), GraphRAG (graph-based retrieval-augmented generation), RAG (retrieval-augmented generation), Logic Tensor Networks / differentiable reasoning, Probabilistic circuits (compiled from DL ontologies), Neuro-symbolic AI architectures, Graph Neural Networks (GNNs), Knowledge graphs, Linked Data (URIs), Model-Driven Engineering (MDE) / meta-modeling / model transformation, Domain-Specific Languages (DSLs), Enterprise Architecture (EA) frameworks, SNOMED CT, ICD-11, ChatGPT-4 / GPT-4 (baseline), Basel III liquidity reporting framework, Multi-agent LLM systems

Top sources. 1. M. Ali, Z. Taha, M. M. Morsey. Ontology-grounded knowledge graphs for mitigating hallucinations in large language models for clinical question answering. 2026. doi: 10.1016/j.jbi.2026.104993 2. S. M. B. Silalahi, R. Indrajit, T. Mantoro. Development of

a semantic-web enabled enterprise architecture framework for curriculum governance. 2025. doi: 10.1109/ICCED68324.2025.11324880 3. N. Lazzari, V. Presutti, A. Vergari. To neuro-symbolic classification and beyond by compiling description logic ontologies to probabilistic circuits. 2026. doi: 10.48550/arXiv.2601.14894 4. G. Xiao, J. Corman. Ontology-mediated SPARQL query answering over knowledge graphs. 2021. doi: 10.1016/j.bdr.2020.100177 5. A. Katipelly, S. Thalary. Semantic automation of Basel III liquidity reporting: Utilizing ontological knowledge graphs for real-time regulatory compliance and auditability. 2024. doi: 10.63282/3050-922x.ijeret-v5i2p115

Contribution to AEDA. This review supplies the formal, computational substance of the AEDA Enterprise Ontology layer, the layer that converts BEA/DoDAF DM2 from descriptive documentation into a machine-reasonable semantic backbone. It establishes the canonical standards stack (RDF/RDFS triples, OWL 2 DL/QL for axioms and constraints, SPARQL for multi-hop query, DLs for decidable reasoning) that the enterprise ontology should be built on, and grounds it in well-defined logical semantics enabling sound, complete, auditable inference. It provides the upstream connection to Ontology Governance (engineering methodologies, multi-dimensional quality assessment, HermiT consistency checking, SPARQL competency questions) and the downstream connections to Knowledge Graph (ontologies as the consistency-enforcing schema), GraphRAG (graph-structured retrieval with entity relationships and hierarchies), Agent Swarms (shared ontology as the inter-agent semantic protocol), and Decision/Executive Support (explainable, traceable reasoning chains). Critically, it gives AEDA its evidentiary thesis: ontology grounding measurably reduces LLM hallucination (63% to 1.7%) and raises accuracy (37% to 98%), making the enterprise ontology not optional metadata but the trust-and-accuracy enforcement mechanism for every layer above it.

Design implications. - Build the AEDA enterprise ontology on the W3C standards stack (RDF/RDFS, OWL 2, SPARQL) rather than ad hoc schemas, selecting OWL 2 profiles by need: OWL 2 DL where full axiomatic expressiveness is required, OWL 2 QL where high-volume data access dominates. Adopt incrementally (RDF graph first, then RDFS vocabulary, then OWL axioms) to match organizational maturity. - Treat the ontology as the hallucination-control and factual-grounding layer for the GraphRAG and Agent Swarm layers: route LLM candidate outputs through a closed-loop validation pipeline (SPARQL constraint queries + symbolic reasoner consistency check such as HermiT + iterative corrective feedback) so only ontology-consistent outputs surface to decision-makers. - Use ontology-mediated query answering (OMQA) with query rewriting to wrap existing DoD authoritative relational/legacy data sources with a semantic access layer, avoiding full RDF materialization while preserving data currency, this is the bridge from the Data Sources layer into the ontology. - Standardize on a neuro-symbolic pattern for the reasoning and optimization layers: layered representation where neural components extract features from raw data, symbolic layers encode DM2-derived domain axioms and rules, and bidirectional feedback enforces consistency; consider compiling DL ontologies to probabilistic circuits for GPU-accelerated deductive reasoning where throughput matters. - Engineer architectural impact analysis as a dynamic multi-layer / hierarchical knowledge graph plus graph-aware learning (GNNs over enterprise data) to auto-parse change-propagation paths and flag critical elements, moving AEDA from

reactive documentation to proactive predictive impact analysis at the Digital Twin / Simulation layers. - Invest in ontology engineering and governance tooling early: collaborative editors (Protege/WebProtege/VocBench), established methodologies (Uschold-Gruninger plus Guarino classifications), multidimensional quality assessment, automated consistency checking (HermiT), and SPARQL competency questions to validate the ontology represents required knowledge. - Reuse standardized upper/vocabulary ontologies (e.g., CIDOC CRM, SKOS analogues, and domain standards like SNOMED CT/ICD-11 in their respective fields) to guarantee semantic interoperability across organizational and system boundaries rather than inventing isolated vocabularies. - Use LLM-assisted knowledge-graph construction (entity extraction, relationship inference) with ontology-aligned RDF/OWL schema generation and multi-LLM consensus validation to scale graph population, but pair automation with human expert curation to maintain clinical/mission-grade quality. - Make explainability a first-class design requirement: every reasoning output should carry a traceable logical justification path and natural-language explanation, satisfying the auditability and accountability demands of regulated DoD decision support and feeding directly into the Executive Support layer.

Risks/limitations. - Expressiveness-versus-tractability tension: richer OWL/DL axioms increase reasoning cost. The review explicitly notes maintaining balance between model complexity and computational efficiency remains an ongoing challenge; over-axiomatized ontologies can become computationally intractable for real-time decision support. - Ontology engineering and maintenance burden: high-quality ontologies require structured methodologies, multidimensional quality assessment, and continuous curation. Semantic integration success depends on addressing data quality, consistency, and maintenance challenges, automation complements but does not fully replace manual curation. - Headline metrics are domain-specific and single-study: the 98% accuracy / 63%-to-1.7% hallucination reduction and the 24% inconsistency / 50% query-efficiency gains come from specific clinical and curriculum-governance studies; generalization to DoD enterprise architecture is not yet demonstrated in the reviewed literature and should not be assumed. - Explaining DL reasoning to humans remains hard: the review notes the challenge of explaining reasoning results requires integration with cognitive science principles, formal soundness does not automatically yield human-comprehensible explanations. - Neuro-symbolic interface design is non-trivial: combining statistical neural strengths with formal symbolic requirements raises fundamental open questions about knowledge representation and the interfaces enabling effective bidirectional communication between paradigms. - Semantic interoperability across heterogeneous, hybrid (structured/semi-structured/unstructured) infrastructures and legacy-to-cloud-native integration is a persistent, unresolved challenge driving (not yet fully solved by) layered knowledge-graph architectures. - LLM-driven knowledge-graph construction can introduce errors at scale; reliance on multi-LLM consensus validation and ontology alignment mitigates but does not eliminate the risk of propagating extraction mistakes into the authoritative graph.

LR03: GraphRAG and Knowledge Graph Architectures

AEDA layer: Knowledge Graph + GraphRAG layers

Scope. State-of-the-art review of GraphRAG, knowledge graphs, graph neural networks, semantic retrieval, multi-hop reasoning, and graph-based decision support, comparing architectures across commercial, government, academic, and defense applications, and identifying best practices for combining vector databases, graph databases, and ontology layers into unified reasoning systems. Maps directly to AEDA's Knowledge Graph and GraphRAG layers, which sit between the Enterprise Ontology layer and the Agent Swarm layer. 45 cited references, generated June 07, 2026.

Key findings. - GraphRAG addresses the core limitation of traditional dense-vector RAG: conventional RAG treats documents as flat, atomized chunks ranked by semantic similarity alone, which cannot capture relational structure or support multi-hop reasoning. GraphRAG organizes knowledge as explicit structured graphs of entities, relationships, and hierarchical semantic dependencies, improving reasoning reliability and interpretability. - A unified GraphRAG framework has five primary components: query processor, retriever, organizer, generator, and data source. The retriever spans dense retrievers (semantic embeddings), graph retrievers (BFS or personalized PageRank over subgraphs), and hybrid retrievers (learned fusion). - Adaptive query routing outperforms rigid GraphRAG-for-everything. EA-GraphRAG computes continuous complexity scores from syntactic features, routing simple factual queries to dense retrieval (50ms latency, ~75% accuracy) and complex multi-hop queries to GraphRAG (200ms latency, ~88% accuracy), with reported token cost reductions of up to 90.71% versus baseline. - KG construction quality is improvable with LLMs and RAG-style methods. GPT-4o achieved precision 93.75% and F1 96.26% on medical ontology mapping combined with vector databases. RAKG (document-level retrieval-augmented KG construction) achieved 95.91% accuracy on the MINE dataset, a 6.2 percentage point improvement over the GraphRAG baseline, by treating KG construction itself as a retrieval-augmented task to mitigate long-context forgetting and coreference complexity. - Graph embeddings (translation-based TransE/TransH/TransR; semantic matching; hyperbolic geometries; neural-symbolic fusion) map entities/relations into low-dimensional spaces. Hyperbolic embeddings (HEM) better capture hierarchical biomedical structure than Euclidean. Temporal KG embeddings reason over evolving facts and concept drift. - Ontology-guided KG construction outperforms pure vector-based RAG while reducing LLM usage cost, supplying formal semantic constraints, consistency, and rule-based reasoning. FAIR GraphRAG uses FAIR Digital Objects (core data, metadata, persistent identifiers, semantic links) as fundamental graph units. Hybrid neural+formal approaches are especially valuable in regulated domains (healthcare, finance, legal). - GNN and multi-hop advances: MAGNA (multi-hop attention) achieved 5.7% relative error reduction on KG completion and up to 10 percentage point improvement on node classification. ScaleGNN addresses over-smoothing via per-hop pure-neighbor matrices and learnable sparsity masking on billion-scale graphs. KIFGraph (multi-granularity fusion graphs with masked attention) outperforms standard GNN methods on HotpotQA. StepChain GraphRAG (question decomposition + BFS) achieved 4.70% EM and

3.44% F1 improvement on HotpotQA. DRKG (LLM-guided hop-constrained reasoning plans) achieved 1-5% accuracy improvement with higher interpretability. - Integrated vector-graph retrieval: HybRAG fuses semantic node-level and structural path-level retrieval, outperforming single-retriever baselines on WebQSP and CWQ. Neural-symbolic dual-indexing uses Prize-Collecting Steiner Trees plus Personalized PageRank; selective skeleton construction from the top 20% of chunks (by eigenvector centrality) achieves 10x cost reduction versus exhaustive KG construction while improving generation quality by 32.4% and retrieval coverage by 92.4%. - Multimodal GraphRAG extends graphs to text plus illustrations. MMGraphRAG uses spectral clustering for cross-modal entity linking and path-based retrieval, achieving state-of-the-art on CMEL, DocBench, and MMLongBench. - Application-domain evidence: Commercial business DSS (LSTM + XGBoost + business KG) achieved 23.5% sales-forecast error reduction, risk warnings 2-4 weeks in advance, 15.2% click-through improvement, 22.5% conversion increase, 12.6% inventory-holding-cost reduction; recruitment DSS reached 7.9% maximum prediction error. Healthcare: an EHR-oriented multicenter KG across 1,185 patients in three hospitals identified 124 chronic-kidney-disease patients (86% clinician-confirmed CKD-positive) that single-hospital data could not; hypertension medication CDSS reached 91% recall, 83% hit@3, 77% MRR on 124 cases. Government: RAGulating Compliance (multi-agent ontology-free regulatory-triplet KG) and e-government multi-agent GraphRAG improve compliance QA and reduce hallucinations. Defense/security: Neo4j-enabled GraphRAG augmenting a multimodal LLM (LLaVA) improved orbital-debris detection with reduced hallucinations; TE-G-SAGE edge-aware GNN on NF-UNSW-NB15-v3 NetFlow delivers explainable network intrusion detection with SHAP attribution. - Hallucination mitigation: GraphRAG reduces but does not eliminate hallucination. Graph Grounding and Alignment (GGA) uses mechanistic interpretability via path reliance degree (PRD) and semantic alignment score (SAS) for lightweight post-hoc detection that outperforms semantic and confidence baselines on AUC and F1. Other mechanisms: semantic alignment scoring, path reliance analysis (detect over-reliance on shortest-path triples), attribution verification with explicit evidence chains, confidence scoring via Bayesian/ensemble disagreement. - Scalability and efficiency: hierarchical community structure, distributed indexing by entity/domain, LRU and community-summary caching, Approximate Nearest Neighbor reducing dense retrieval from O(n) to O(log n), and cloud-edge deployment for interactive latency (<500ms). StructReason uses PCST-based structural refinement to cut token consumption 40-70% while improving F1 versus standard GraphRAG and LightRAG. Joint KG-LLM models gave up to 12.0% accuracy and 8.6% F1 improvement on mental-health etiology, stress, and emotion tasks versus standalone LLMs. - Benchmarking: knowledge-intensive multi-hop benchmarks (HotpotQA, MuSiQue, 2WikiMultiHopQA) plus retrieval metrics (Recall@k, MRR, NDCG). GraphRAG-Bench shows GraphRAG surpasses dense RAG when questions require explicit relational reasoning, hierarchical organization, or multi-source integration; dense RAG stays superior and more efficient for simple factual retrieval. Context relevance (whether the retrieved subgraph answers the query) is more predictive of downstream performance than abstract metrics. - Summary performance table by method (latency / accuracy / hallucination reduction): Dense RAG ~50ms / 75-80% / Moderate; Sparse RAG ~45ms / 70-75% / Low; Semantic RAG ~120ms / 80-85% / Moderate; GraphRAG ~250ms / 85-89% / High; EA-GraphRAG ~180ms / 88-92% / Very High; Hybrid

RAG ~200ms / 89-93% / Very High; Neural-Symbolic ~300ms / 89-95% / Very High.

Named frameworks/systems/standards. GraphRAG (Graph-based Retrieval-Augmented Generation), EA-GraphRAG (adaptive dense/graph routing, continuous complexity scores), Youtu-GraphRAG (vertically unified agents), RAKG (document-level retrieval-augmented KG construction), TransE / TransH / TransR (translation-based KG embeddings), HEM (Hyperbolic Embedding Model for biomedical KGs), FAIR GraphRAG / FAIR Digital Objects, MAGNA (Multi-hop Attention Graph Neural Network), ScaleGNN, KIFGraph, StepChain GraphRAG, BDTR (bridge-guided dual-thought-based retrieval), DRKG (Decomposed Reasoning over Knowledge Graph), EASE-DR (enhanced sentence embeddings for dense retrieval), HybRAG (Hybrid Retrieval Framework), Neural-symbolic dual-indexing (Prize-Collecting Steiner Trees + Personalized PageRank), MMGraphRAG / CMEL dataset, RAGulating Compliance (multi-agent regulatory-triplet KG), TE-G-SAGE (edge-aware GNN on GraphSAGE), GGA (Graph Grounding and Alignment; PRD, SAS), StructReason (PCST structural refinement), T-GRAG (dynamic temporal GraphRAG), KG-anchored RAG, GraphRAG-Bench, Neo4j / Cypher (graph database), BERT, DeBERTa, T5, GPT-4o, LLaVA (encoders/LLMs), GraphSAGE, Personalized PageRank, BFS, Prize-Collecting Steiner Tree, eigenvector centrality, spectral clustering, ANN, Benchmarks: HotpotQA, MuSiQue, 2WikiMultiHopQA, WebQSP, CWQ, MINE, DocBench, MMLongBench, NF-UNSW-NB15-v3; metrics Recall@k, MRR, NDCG, EM, F1; ARES-inspired evaluation

Top sources. 1. H. Han et al.. Retrieval-augmented generation with graphs (GraphRAG). 2024. doi: 10.48550/arXiv.2501.00309 2. S. Dong, Q. Zhang, Y. Xiao, S. Chen, C. Zhou, X. Huang. Use graph when it needs: Efficiently and adaptively integrating retrieval-augmented generation with graphs (EA-GraphRAG). 2026. doi: 10.48550/arXiv.2602.03578 3. B. Peng et al.. Graph retrieval-augmented generation: A survey. 2025. doi: 10.1145/3777378 4. Z. Xiang et al.. When to use graphs in RAG: A comprehensive analysis for graph retrieval-augmented generation (GraphRAG-Bench). 2025. doi: 10.48550/arXiv.2506.05690 5. J.-S. Yang, Z. Zeng, Z. Shen. Neural-symbolic dual-indexing architectures for scalable retrieval-augmented generation. 2025. doi: 10.1109/ACCESS.2025.3638761

Contribution to AEDA. Supplies the architectural and empirical foundation for AEDA's Knowledge Graph and GraphRAG layers, the bridge between the Enterprise Ontology layer above and the Agent Swarm layer below. It establishes that the AEDA brain should not be a single vector store but a unified reasoning substrate combining vector databases, graph databases, and an ontology layer, with adaptive routing that selects dense retrieval for simple factual queries and graph traversal for multi-hop enterprise questions. It validates ontology-guided construction (formal constraints plus neural flexibility) as the right pattern for a regulated DoD enterprise-architecture context, and it documents the retrieval, embedding, multi-hop reasoning, hallucination-detection, and benchmarking machinery AEDA needs to turn BEA/DoDAF DM2 content from documentation into a queryable, explainable decision stack.

Design implications. - Build the AEDA brain as a unified vector + graph + ontology stack, not a vector store alone; treat the Enterprise Ontology layer as the formal constraint

provider that drives KG construction (ontology-guided construction outperforms pure vector RAG while cutting LLM cost). - Implement adaptive query routing at the GraphRAG layer (EA-GraphRAG pattern): syntactic/complexity scoring sends simple factual lookups to dense retrieval (~50ms) and multi-hop architecture questions to graph traversal (~200ms); budget for the documented latency tiers and pursue the reported token-cost reductions (up to 90.71%). - Adopt the five-component framework (query processor, retriever, organizer, generator, data source) as the AEDA GraphRAG reference decomposition; use hybrid retrievers (HybRAG-style semantic node + structural path fusion) rather than choosing one modality. - Use Neo4j-class graph databases with Cypher and Personalized PageRank/BFS traversal for the Knowledge Graph layer; apply Prize-Collecting Steiner Tree subgraph extraction and eigenvector-centrality skeleton construction to keep cost tractable at enterprise scale (10x cost reduction, +32.4% generation quality, +92.4% retrieval coverage reported). - Treat KG construction as a retrieval-augmented task (RAKG pattern) to handle entity disambiguation, coreference, and cross-document integration across the BEA/DoDAF corpus; pair with LLM extraction (GPT-4o-class) validated by ontology mapping. - Engineer hallucination detection into the stack from the start: implement GGA-style path reliance degree and semantic alignment scoring plus explicit evidence-chain attribution, because explainability and traceability are mandatory for DoD decision support. - Plan for scale and freshness: hierarchical community structure, distributed indexing by domain, ANN (O(log n)) dense retrieval, caching, and incremental/temporal updates (T-GRAG-style) with versioning and rollback to verified KG states; cloud-edge split for <500ms interactivity. - Instrument the stack with a benchmark harness (GraphRAG-Bench plus Recall@k, MRR, NDCG, EM/F1) and weight context-relevance metrics, which are more predictive of downstream quality than abstract scores; codify when GraphRAG beats dense RAG (relational, hierarchical, multi-source) versus when dense RAG suffices. - Favor neuro-symbolic integration with constraint-aware planning for the high-stakes AEDA decision layer; the literature reports up to 12.0% accuracy and 8.6% F1 gains for joint KG-LLM over standalone LLMs and best hallucination control in the Neural-Symbolic configuration. - Support multimodal KGs (MMGraphRAG/spectral-clustering cross-modal linking) so AEDA can fuse diagrams, illustrations, and text from architecture artifacts, not text alone.

Risks/limitations. - GraphRAG reduces but does not eliminate hallucination; residual hallucinations persist and require dedicated post-hoc detection (GGA / path reliance / attribution), adding system complexity. - Knowledge incompleteness and missing bridge entities break multi-hop reasoning; if bridge evidence is buried too deep it cannot support reasoning chains (motivating iterative retrieval / BDTR), a real risk for an incomplete or evolving enterprise-architecture KG. - Latency-accuracy trade-off is intrinsic: graph traversal incurs 100-300ms versus sub-100ms dense retrieval; over-applying GraphRAG to all queries wastes latency and tokens, which is why adaptive routing is required. - Deep GNNs suffer over-smoothing where node representations become indistinguishable; mitigations (ScaleGNN selective fusion, pruning) are needed for billion-scale graphs. - Many reported gains come from specific academic/healthcare/biochemical benchmarks (HotpotQA, WebQSP, CWQ, MINE, NF-UNSW-NB15-v3) and may not transfer directly to DoD enterprise-architecture data; metrics are domain- and dataset-

specific. - Exhaustive KG construction is computationally expensive at billion-token scale; without selective skeleton construction / structural refinement, cost and token consumption are prohibitive. - Evaluation is unsettled: reference-based metrics can mislead, and context relevance must be measured directly; abstract metrics are weaker predictors of downstream performance. - Stated future-work gaps remain open: handling noisy real-world KGs with incomplete or erroneous data, lightweight systems for resource-constrained/edge environments, temporal reasoning over evolving knowledge, and federated approaches for privacy-sensitive applications (directly relevant to multi-classification or multi-agency DoD deployment).

LR04: Agentic Enterprise Intelligence

AEDA layer: Agent Swarm layer

Scope. Comprehensive review of agentic AI systems, autonomous planning, multi-agent architectures, cognitive architectures, tool-using agents, and orchestration frameworks, and how agent swarms perform architecture analysis, portfolio optimization, risk assessment, red-team evaluation, and strategic planning. Covers governance, human-in-the-loop oversight, explainability, trust, and safety/red-teaming. Organized in six sections: (I) Foundations of agentic AI, (II) Multi-agent and cognitive architectures, (III) Tool integration and orchestration, (IV) Enterprise applications, (V) Governance/HITL/trust, (VI) Safety/red-teaming/evaluation. 41 references.

Key findings. - Agentic AI is framed as a paradigm shift from stateless, prompt-driven generative models to goal-directed systems that perceive, reason over extended horizons, plan, and act through external tools via iterative control loops with minimal human intervention [1][2]. Core capabilities: perception, memory, planning/reasoning, tool-use, and multi-agent coordination [1]. - Cognitive architecture grounding: the CoALA framework (Cognitive Architectures for Language Agents) organizes agents via modular memory systems, structured action spaces for internal-memory and external-environment interaction, and generalized decision-making to choose actions [6]. Historical grounding includes reactive agents, deliberative architectures, and Belief-Desire-Intention (BDI) models [2]. - Hierarchical multi-agent frameworks use supervisor agents that decompose user goals into executable subtasks via hierarchical task networks, combining chain-of-thought reasoning, contextual memory, and task decomposition to improve task completion accuracy, planning efficiency, and adaptability over single-agent and flat multi-agent designs [7]. - Memory infrastructure: G-Memory is a hierarchical agentic memory system inspired by organizational memory theory, managing multi-agent interactions through three-tier graph hierarchies, insight graphs (high-level generalizable knowledge), query graphs (efficient retrieval), and interaction graphs (fine-grained collaboration trajectories), with bi-directional traversal supporting cross-trial knowledge leverage [8]. - Tool integration core patterns: the ReAct framework interleaves reasoning and acting to call tools, observe results, and dynamically adjust, improving multi-step QA, logical inference, and knowledge retrieval [3]. Agentic RAG embeds autonomous agents into the retrieval pipeline using reflection, planning, tool use, and multi-agent collaboration to dynamically manage retrieval and iteratively refine context [10]. -

Model Context Protocol (MCP) is positioned as an emerging open standard enabling agents to access tools, data sources, and external systems through standardized interfaces, moving from fixed API calls to dynamic, runtime-discovered capabilities and cross-framework interoperability [4]. - Agent-Ready Architecture (ARA) wraps legacy System APIs through MCP-compliant orchestration layers that enrich deterministic endpoints with natural-language tool descriptions and structured parameter schemas, achieving 90% semantic discovery precision and reducing multi-tool chain latency by 60.5% [12]. - Portfolio optimization: a graph attention-based heterogeneous multi-agent DRL framework models time-varying asset correlations with specialized agents for risk assessment, return prediction, and market perception, achieving 16.8% annualized returns, 1.34 Sharpe ratio, and 8.2% maximum drawdown, outperforming mean-variance and equal-weight portfolios [14]. Hierarchical DRL with auxiliary agents achieves Sharpe ratio improvements exceeding 8.2% over traditional strategies [15]. Augmented Lagrangian Multiplier methods enforce hard risk constraints with zero constraint violations during testing [16]. - Enterprise architecture orchestration: multi-layer agentic AI architectures demonstrate 3-10x workflow acceleration with 60-80% reductions in mean time to resolution (MTTR) for critical tasks [17][37]. The Enterprise Agentic Architecture Framework (EAAF) defines six layers (infrastructure, enterprise integration, orchestration/coordination, governance/safety, agent intelligence, interaction) with a central Control Plane managing policies, identity, scheduling, observability, and agent lifecycle [17]. - Governance maturity: the Agentic AI Governance Maturity Model (AAGMM) is a five-level framework over 12 governance domains grounded in NIST AI RMF and ISO/IEC 42001, validated through 750 simulation runs across five enterprise scenarios, showing statistically significant differences ($p<0.001$, effect sizes $d>2.0$) between maturity levels; Level 4-5 organizations achieve 94.3% lower sprawl indices, 96.4% fewer risk incidents, and 32.6% higher effective task completion rates [23]. Industry context: only 21% of enterprises have mature governance for autonomous agents, and 40% of agentic AI projects are projected to fail by 2027 due to inadequate governance [23]. - Human-in-the-loop: the Adaptive Oversight Calibration Model (AOCM) treats oversight as a continuous, context-sensitive function via six formal propositions (task criticality, AI competency boundaries, human cognitive capacity, institutional constraints, trust dynamics, feedback loops) across eight high-stakes sectors [24]. The TRACE Framework (Trust, Review, Accountability, Critique, Explainability) embeds governance anchors at the agent level, dedicated Critic agents for meta-validation, and an Overall System Confidence score driving automated action, human escalation, and continuous learning [25]. - Explainability metrics: XAI substantially improves decision quality, with LIME increasing accuracy by 12.1% and SHAP by 14.8% versus no-explanation systems [28]; post-hoc SHAP/LIME/Grad-CAM in water resource management yield 21% forecast accuracy improvement and 34% reduction in false alarms [29]; HITL XAI in paperless GMP validation achieves 32% cycle-time reductions and improves inter-rater agreement from kappa=0.71 to kappa=0.85 [26]; clinical SDOH extraction exceeds 95% confidence [19]; clinical workflow orchestration shows 60% faster ambulance response, 38% shorter door-to-clinician intervals, and 22% higher OR throughput [20]. - Policy-compliant orchestration: CAMCO (Safe and Policy-Compliant Multi-Agent Orchestration) models multi-agent decision-making as constrained optimization with constraint projection engines, adaptive risk-weighted Lagrangian utility shaping, and iterative negotiation, demonstrating zero policy violations, risk

exposure below threshold (mean ratio 0.71), 92-97% utility retention, and mean convergence in 2.4 iterations [33]. Multi-Agent Orchestration Protocols embed jurisdictional rules, access controls, and tamper-evident audit logs via central orchestrators consulting Agent Registries and Policy Engines [34]. - Red-teaming and safety: AJAR (Adaptive Jailbreak Architecture for Red-teaming) exposes multi-turn jailbreak algorithms as callable MCP services, improving X-Teaming attack success from 65.0% to 76.0%, reaching 80% cumulative success one turn earlier, and reproducing Crescendo better than PyRIT (91.0% vs 87.5%) [35]. Trustworthy Agentic AI defines a seven-layer trust taxonomy (identity, planning, communication, memory, retrieval, execution, oversight) yielding six reusable secure-coordination design patterns [37]. The SAFE-AI Framework emphasizes Safety, Auditability, Feedback, and Explainability with a taxonomy of suggestive, generative, autonomous, and destructive AI behaviors [38]. Standard agent benchmarks named: GAIA, AssistantBench, WebArena [39].

Named frameworks/systems/standards. CoALA (Cognitive Architectures for Language Agents) framework [6], Belief-Desire-Intention (BDI) model; ACT-R (cognitive architectures) [2][8 outline], ReAct framework (reasoning + acting) [3], Agentic RAG (Retrieval-Augmented Generation) [10], Model Context Protocol (MCP) [4], LangGraph (stateful orchestration, exactly-once semantics, explicit state machines) [11/29-text], CrewAI (role-based multi-agent) [29-text], AutoGen (conversational multi-agent) [29-text], GradientSys (multi-agent LLM scheduler with ReAct orchestration) [11], Agent-Ready Architecture (ARA), MCP-compliant legacy-API wrapping [12], G-Memory (hierarchical three-tier graph agentic memory) [8], Contract Net Protocol; Agent-to-Agent (A2A) communication [24-text], AlphaAgents (role-based equity portfolio multi-agent) [35-text], TradingAgents framework (LLM financial trading) [21], Graph attention-based heterogeneous multi-agent DRL portfolio framework [14], Augmented Lagrangian Multiplier risk-constrained RL [16], Enterprise Agentic Architecture Framework (EAAF) with central Control Plane [17], Agentic AI Governance Maturity Model (AAGMM); 12 governance domains; NIST AI RMF; ISO/IEC 42001 [23], Adaptive Oversight Calibration Model (AOCM) [24], TRACE Framework (Trust, Review, Accountability, Critique, Explainability) [25], LOKA Protocol (Universal Agent Identity Layers, Decentralized Ethical Consensus, Decentralized Identifiers, post-quantum cryptography) [32], CAMCO (Safe and Policy-Compliant Multi-Agent Orchestration) [33], Multi-Agent Orchestration Protocol with Agent Registries and Policy Engines [34], AJAR (Adaptive Jailbreak Architecture for Red-teaming); Crescendo, ActorAttack, X-Teaming attacks; PyRIT baseline [35], RedTWIZ multi-turn red-teaming framework [36], Trustworthy Agentic AI seven-layer trust taxonomy [37], SAFE-AI Framework (Safety, Auditability, Feedback, Explainability) [38], Magentic-One generalist multi-agent system [39], Benchmarks: GAIA, AssistantBench, WebArena [39], SHAP, LIME, Grad-CAM explainability techniques [27][28][29], Standards/regulations: EU AI Act, NIST AI RMF, ISO/IEC 42001, GDPR, HIPAA, HL7 FHIR [11][20][23][33]

Top sources. 1. T. Sumers, S. Yao, K. Narasimhan, T. L. Griffiths. Cognitive architectures for language agents (CoALA). 2023. doi: 10.48550/arXiv.2309.02427 2. V. Arunkumar, G. G. R., R. Buyya. Agentic artificial intelligence (AI): Architectures, taxonomies, and evaluation of large language model agents. 2026. doi: 10.48550/arXiv.2601.12560 3. P. Venkiteela. An enterprise agentic architecture framework for agentic AI governance and scalable autonomy (EAAF).

2026. doi: 10.64539/sjcs.v2i1.2026.368 4. V. Acharya. Governing the agentic enterprise: A governance maturity model for managing AI agent sprawl in business operations (AAGMM). 2026. doi: (no DOI; Semantic Scholar paper cec9745d5c3c0418ba1fd2d368a9e40dbf305f8a) 5. N. Sinha. Building trust in agentic AI: TRACE framework for policy-driven multi-agent system design. 2026. doi: 10.47191/ijcsrr/v9-i2-46

Contribution to AEDA. This review supplies the empirical and architectural backbone for AEDA's Agent Swarm layer, the computational tier where specialized autonomous agents collaborate to execute architecture analysis, portfolio optimization, risk assessment, red-team evaluation, and strategic planning over the lower AEDA layers (Knowledge Graph, GraphRAG, Enterprise Ontology). It establishes a concrete design vocabulary for the swarm: cognitive-architecture grounding (CoALA, BDI) for individual agent structure; hierarchical supervisor/subtask decomposition for swarm topology; G-Memory-style three-tier graph memory for cross-agent and cross-trial knowledge (directly compatible with AEDA's Knowledge Graph layer); ReAct + Agentic RAG as the tool-and-retrieval pattern binding agents to the GraphRAG layer; and MCP/ARA as the interoperability and legacy-integration standard for agent-to-tool access. Critically, it provides quantified governance evidence (AAGMM, EAAF, TRACE, AOCM, CAMCO) that lets AEDA treat governance, human-in-the-loop oversight, and policy compliance as first-class layer-level requirements rather than afterthoughts, including measured payoffs (94.3% lower sprawl, 96.4% fewer risk incidents, 32.6% higher task completion at governance Levels 4-5) and a Control Plane reference pattern for identity, scheduling, observability, and lifecycle. It also names the evaluation and red-team apparatus (GAIA, AssistantBench, WebArena, AJAR, RedTWIZ, SAFE-AI seven-layer taxonomy) needed to validate the swarm before it feeds the Simulation, Optimization, and Decision layers.

Design implications. - Structure each AEDA swarm agent on a cognitive-architecture pattern (CoALA-style modular memory, structured action space, generalized decision-making) and adopt a hierarchical supervisor topology: a planner/supervisor agent decomposes enterprise-architecture goals (BEA/DoDAF analysis, portfolio optimization, risk scoring) into subtasks routed to specialized agents (analyst, risk, optimizer, critic). - Implement a G-Memory-style three-tier graph memory (insight / query / interaction graphs) as the swarm's shared memory, and back it directly with the AEDA Knowledge Graph layer so agent insights and collaboration trajectories persist and support cross-trial knowledge reuse. - Bind agents to tools and data via the Model Context Protocol (MCP) as the standard interface, and use an Agent-Ready Architecture (ARA) wrapper to expose existing deterministic enterprise/system APIs (the documentation-era BEA/DoDAF data sources) as natural-language, schema-typed tools; expect ARA-class gains (90% semantic discovery precision, 60.5% lower multi-tool chain latency) as design targets. - Use ReAct interleaved reasoning-acting plus Agentic RAG to ground every agent action in the AEDA GraphRAG and Knowledge Graph layers, with reflection and iterative retrieval refinement rather than single-shot retrieval. - Adopt a six-layer EAAF-style reference structure with a central Control Plane (policies, identity, scheduling, observability, lifecycle governance) as the AEDA swarm's coordination substrate; treat this Control Plane as the integration seam between the Agent Swarm layer and the Decision/Executive Support layers. - Make governance a layer-level requirement: instrument an

AAGMM-style maturity model (12 domains, grounded in NIST AI RMF and ISO/IEC 42001) and actively monitor and prevent agent sprawl (functional duplication, shadow/orphaned agents, permission creep, unmonitored delegation chains). - Embed human-in-the-loop as a continuous, context-sensitive function (AOCM) rather than a binary gate: calibrate oversight by task criticality, AI competency boundaries, and trust dynamics, and add dedicated Critic agents (TRACE) producing an Overall System Confidence score that routes outputs to automated action, human escalation, or continuous learning. - Enforce policy compliance at runtime with a CAMCO-style constrained-optimization layer (constraint projection, risk-weighted Lagrangian utility shaping, iterative negotiation) plus tamper-evident audit logs and Agent Registry/Policy Engine consultation, targeting zero policy violations and high utility retention. - Instrument explainability as core infrastructure (SHAP/LIME/Grad-CAM, uncertainty-aware reasoning, bias/fairness/robustness auditing); the literature shows measurable decision-quality gains (SHAP +14.8%, LIME +12.1% accuracy) and improved trust, which the AEDA Decision and Executive Support layers should surface. - Build a standing red-team and evaluation harness for the swarm before promotion to Simulation/Optimization/Decision: benchmark on GAIA, AssistantBench, and WebArena; run AJAR/RedTWIZ multi-turn adversarial testing; and structure trust controls along the seven-layer taxonomy (identity, planning, communication, memory, retrieval, execution, oversight) with SAFE-AI guardrails, sandboxing, and runtime verification.

Risks/limitations. - Governance immaturity is the dominant deployment risk: only 21% of enterprises have mature governance for autonomous agents and 40% of agentic AI projects are projected to fail by 2027 due to inadequate governance and risk controls [23]. AEDA's Agent Swarm layer is unusable without the accompanying governance/Control Plane scaffolding. - Agent sprawl, functional duplication, shadow agents, orphaned agents, permission creep, and unmonitored delegation chains, degrades safety and cost and must be actively quantified and contained [23]. - Hallucination propagation across agent boundaries: agents can fabricate information and spread false outputs through the multi-agent system, requiring multi-level control, secure-coordination patterns, and a layered trust taxonomy [37][38]. - Adversarial vulnerability of tool-enabled swarms: as agents gain persistent state, tool access, and autonomous control loops, action-security attacks (multi-turn jailbreaks via Crescendo, ActorAttack, X-Teaming) become potent; demonstrated X-Teaming success rises to 76.0% and Crescendo reproduction reaches 91.0%, so red-teaming must target action security, not just content moderation [35][36]. - Persistent oversight tensions across high-stakes sectors: explainability-performance tradeoffs, autonomy-accountability gaps, over-trust/under-trust dualities, and participation-effectiveness paradoxes; oversight is not a solved binary [24]. - Speed-quality tradeoffs in autonomous decision-making introduce accuracy, fairness, and safety risks, and XAI itself carries an accuracy-interpretability tradeoff plus unresolved human-AI trust calibration challenges [30][60-text]. - Reported quantitative gains (portfolio returns, MTTR/workflow acceleration, clinical throughput, governance effect sizes) come largely from simulation runs and single-study or domain-specific empirical evaluations, not standardized cross-domain benchmarks; the review itself flags the need for standardization of agentic architectures and cross-domain evaluation benchmarks before these figures generalize [79]. - Open

standardization and ethical-pluralism gaps remain: lack of architecture standardization, ethical pluralism in multi-stakeholder settings, and mechanisms for preventing emergent adversarial behaviors at scale are named as unresolved future-research problems [79]. - Source-strength caveat for AEDA citation use: a large share of references are 2026 preprints, journals of varying tier, and one Semantic Scholar entry without a DOI [23], so quantitative claims should be treated as indicative and verified before being cited as established results in MITRE work product.

LR05: Enterprise Digital Twins: Architectures, Simulation Methodologies, Data Integration, and AI-Driven Organizational Performance Forecasting

AEDA layer: Enterprise Digital Twin layer

Scope. Comprehensive review of enterprise digital twins (EDTs), organizational digital twins, mission/cyber-physical digital twin ecosystems, and digital engineering environments. Covers reference architectures, data integration strategies, simulation methodologies, embedded AI/ML techniques, organizational performance forecasting and executive decision-support applications, and implementation challenges. The review frames EDTs as continuously updating virtual replicas of organizational processes, assets, and business systems that enable real-time monitoring, simulation-driven experimentation, and evidence-based strategic decision-making, distinct from offline business intelligence by their bidirectional, continuously synchronized interaction with physical and organizational systems.

Key findings. - EDTs are defined as continuously updating virtual replicas of organizational processes, assets, and entire business systems that capture not just physical assets but business processes, workflows, organizational structures, and market-facing operations; the distinguishing feature versus traditional BI is bidirectional interaction (continuous data streams in, actionable insights out influencing systems in real-time). Framed within Industry 4.0/5.0 human-centric plus autonomous data-driven paradigms. - Reference architectures follow standardized multi-layer patterns: data ingestion layers, virtual representation layers, analytical processing layers, and application/user interface layers. A critical innovation is a dedicated data layer that decouples physical and cyber components and acts as a universal translator converting heterogeneous formats into standardized representations. - Deployment patterns: cloud-plus-edge hybrid (cloud for centralized data repositories and computationally intensive simulations; edge for local real-time processing to minimize latency), plus service-oriented/microservices and Platform-as-a-Service architectures enabling modular composition, service catalogs with standardized interaction protocols, and governance-controlled evolution. - Knowledge graphs are a critical enabling technology for semantic interoperability: entity layers (organizational concepts), relationship layers (dependencies/interactions), and reasoning layers (deriving new insights). Multi-ontology networks combine domain-specific ontologies with standardized ontologies including OWL and SOSA; graph neural networks and reasoning engines enable pattern recognition and anomaly detection. - Simulation methodologies span physics-based, data-driven, and hybrid (physics-informed ML) models, plus discrete event

simulation (DES) for supply chain/logistics/operational processes and agent-based simulation (ABS) for autonomous adaptive entities and emergent organizational behavior. Real-time synchronization updates virtual representations with minimal latency (typically milliseconds to seconds depending on application), with automated/continuous parameter calibration to counter model drift. - Embedded AI/ML: LSTM networks for time-series forecasting (cited at 94% accuracy in water demand forecasting); ensemble methods (random forests, gradient boosting, NN ensembles); anomaly detection via isolation forests, autoencoders, variational autoencoders, and multi-stage architectures combining sequence modeling with statistical process control and operating-envelope constraints; reinforcement learning (A3C, PPO) for autonomous decision-making in safe simulated training grounds; and generative AI/LLMs plus agentic digital twins for interpretability, natural-language summarization, scenario generation, and autonomous analysis. - Quantified application results reported: hospitality digital twin predicted occupancy with R2 = 0.86, energy consumption within 8.3% accuracy, and staff efficiency improvements of 14.7%; financial digital twins achieved 30-40% improvements in customer experience metrics while reducing operational risk; graph-based supply chain twins achieved significantly improved scalability over traditional simulation while enabling proactive disruption management. - Executive decision-support use cases: rapid simulation-based evaluation of strategic alternatives, what-if/Monte-Carlo scenario analysis (simulating thousands of scenarios to stress-test strategies and surface vulnerabilities), and risk-free exploration of organizational transformation/change management before implementation. - Cyber-physical ecosystem evolution: system-of-systems integration via the data layer; coupled digital twins with semantic lifting for reusable instantiable components; bidirectional control via Intelligent Acting Digital Twins (IADTs) that move beyond passive monitoring to autonomous control; and edge-cloud distributed architectures that add resilience (continued degraded operation if cloud connectivity is lost) and privacy (sensitive data retained at edge). - Digital engineering enablers: Digital-Twin-as-a-Service (DTaaS) platforms, the ISO 23247 digital twin reference architecture standard, Model-Based Systems Engineering (MBSE) for formal requirements/interface capture and validation, and advanced visualization (3D immersive, AR overlays, interactive dashboards, gesture and natural-language interfaces).

Named frameworks/systems/standards. LSTM (Long Short-Term Memory) networks, Random forests / gradient boosting machines / neural network ensembles, Isolation forests, autoencoders, variational autoencoders (anomaly detection), A3C (Asynchronous Advantage Actor-Critic), PPO (Proximal Policy Optimization), Discrete Event Simulation (DES), Agent-Based Simulation (ABS), Physics-informed machine learning (hybrid physics-based plus ML), Knowledge graphs / Graph Neural Networks, Multi-ontology networks (OWL, SOSA ontologies), ISO 23247 digital twin reference architecture standard, Digital-Twin-as-a-Service (DTaaS), Platform-as-a-Service (PaaS), Microservices / Service-Oriented Architecture, Model-Based Systems Engineering (MBSE), Intelligent Acting Digital Twins (IADTs), Coupled digital twins with semantic lifting, Large Language Models (LLMs) / agentic digital twins, Federated learning, differential privacy, secure multi-party computation (privacy-preserving), Statistical process control / control charts (data and model validation), Edge-cloud distributed architectures, Extended reality / augmented reality interfaces

Top sources. 1. F. Edrisi, D. Perez-Palacin, M. Caporuscio, S. Giussani. Developing and evolving a digital twin of the organization. 2024. doi: 10.1109/access.2024.3381778 2. S. Barat, V. Kulkarni, K. Bhattacharya. Enterprise digital twins for risk free business experimentations. 2022. doi: 10.1109/WSC57314.2022.10015412 3. C. Qian, Y. Guo, A. Hussaini, A. Musa, A. Sai, W. Yu. A new layer structure of cyber-physical systems under the era of digital twin. 2024. doi: 10.1145/3674974 4. J. Li, J. Zhao, X. Shi, X. Huang, R. Li. A digital twin and knowledge graph fusion framework for industrial intelligence in the context of industry 5.0 and industrial IoT. 2026. doi: 10.1002/itl2.70278 5. G. Antonesi, T. Cioara, I. Anghel, V. Michalakopoulos, E. Sarmas, L. Toderean. From transformers to large language models: A systematic review of AI applications in the energy sector towards agentic digital twins. 2025. doi: 10.48550/arXiv.2506.06359

Contribution to AEDA. This review supplies the conceptual and architectural blueprint for AEDA's Enterprise Digital Twin layer, the stage where the stack transitions from representing the enterprise (ontology, knowledge graph, GraphRAG) to simulating it as a living, continuously synchronized virtual replica. It establishes that an enterprise/organizational digital twin can model not only physical assets but business processes, workflows, organizational structures, and decision-makers themselves, which is exactly what AEDA needs to evolve BEA/DoDAF documentation into a computational decision stack. It validates the upstream AEDA layers as direct feeders: the multi-layer data layer maps to AEDA Data Sources and Ontology layers; knowledge-graph plus multi-ontology semantic integration maps to AEDA Enterprise Ontology and Knowledge Graph layers; agentic/LLM digital twins map to AEDA Agent Swarms and GraphRAG; and DES/ABS/hybrid simulation plus RL optimization map directly to AEDA's Simulation and Optimization layers. It also confirms the downstream value proposition: simulation-driven what-if experimentation and forecasting feed the Decision and Executive Support layers, enabling evidence-based strategic decisions that the BEA-to-AEDA transformation is meant to deliver.

Design implications. - Build the EDT layer on an explicit four-tier reference architecture (data ingestion, virtual representation, analytical processing, application/UI) with a dedicated data layer acting as a universal translator that decouples physical/cyber components, so AEDA's twin can ingest heterogeneous DoD enterprise sources without point-to-point coupling. - Reuse the AEDA knowledge graph and enterprise ontology directly as the twin's semantic substrate; adopt multi-ontology networks (OWL, SOSA plus domain ontologies) and graph reasoning so the digital twin inherits semantic interoperability rather than rebuilding it. - Provision a hybrid edge-cloud, microservices/DTaaS deployment: cloud for long-horizon enterprise-wide simulation and optimization, edge for low-latency local monitoring, with a service catalog and governance workflow for controlled evolution. This gives resilience (degraded-mode operation on connectivity loss) and privacy (sensitive data kept at the edge). - Combine simulation paradigms rather than picking one: DES for process/logistics flows, ABS for emergent organizational and multi-actor behavior, and physics-informed/hybrid models where governing laws are known, with continuous automated calibration to keep the twin synchronized as the enterprise drifts. - Embed the AI stack the literature validates: LSTM/ensemble forecasters for performance projection, autoencoder/isolation-forest multi-stage anomaly detection, and RL (A3C/PPO) trained inside

the twin as a safe sandbox before any real-world action, aligning with AEDA's Optimization and Agent Swarm layers. - Integrate LLM/agentic digital twins as the bridge to AEDA's Executive Support layer: use LLMs to translate simulation outputs into natural-language explanations, generate scenarios, and reason over unstructured enterprise information, keeping outputs explainable for executive trust. - Anchor to standards and rigorous engineering: adopt the ISO 23247 reference architecture for interoperability and MBSE for formal requirements, interface, and validation capture so the AEDA twin is auditable and traceable, not ad hoc. - Treat continuous data governance and validation as a first-class subsystem (statistical process control plus ML-based anomaly detection on incoming data), since poor data quality cascades through simulation into corrupted decision support. - Preserve a human-in-the-loop posture: position the twin to enhance rather than replace executive judgment, with explainable AI safeguards on any autonomous (IADT-style) actuation given accountability and organizational-values concerns.

Risks/limitations. - Data quality is named the primary impediment to successful deployment: incomplete, inconsistent, or erroneous data corrupts simulation accuracy and cascades through analytical pipelines into unreliable decision support. - Legacy system integration friction: older enterprise systems use proprietary data formats and communication protocols, requiring mapping/translation layers that must preserve consistency and performance. - Cybersecurity exposure including adversarial attacks on embedded ML models that degrade forecasting accuracy and lead to poor organizational decisions (the review explicitly cites adversarial attacks on AI-driven water forecasting), requiring adversarial training, anomaly monitoring, and secure pipelines. - Privacy risk when processing customer, employee, or regulated data, necessitating federated learning, differential privacy, and secure multi-party computation. - Organizational/change-management risk: success requires technical skills, data-culture maturity, and executive sponsorship; workforces must be trained to interpret and trust model-based recommendations, and the shift from intuition-based to evidence-based decision-making is itself a barrier. - Accountability and oversight risk from delegating organizational decisions to autonomous AI agents, raising questions about human oversight, accountability, and organizational values; explainable AI is positioned as a critical but incomplete safeguard. - Field immaturity: the literature notes the field remains young, with substantial open research on autonomous organization simulation, knowledge integration, explainable decision support, and federated digital twin networks, implying limited validated precedent for full enterprise-scale deployment. - Many quantified results are domain-specific point findings (for example hospitality $R2 = 0.86$, 8.3% energy accuracy, 14.7% staff efficiency; 94% water-demand LSTM accuracy; 30-40% financial customer-experience gains) drawn from narrow case studies and should not be assumed to transfer directly to a DoD enterprise context without re-validation. - Real-time synchronization and continuous calibration impose latency and compute demands; maintaining millisecond-to-second fidelity at enterprise scale across distributed instances is a non-trivial engineering and cost constraint.

LR06: Discrete-Event Simulation, Agent-Based Simulation, Mission Engineering, and Cyber-Physical System Integration

AEDA layer: Simulation layer (discrete-event simulation, agent-based simulation, system dynamics, hybrid simulation, and digital-twin cyber-physical coupling). Within AEDA this layer turns the Enterprise Digital Twin into executable behavior over time, feeding the Optimization and Decision layers.

Scope. Comprehensive review of discrete-event simulation and the DEVS formalism, agent-based simulation, system dynamics, hybrid simulation, mission engineering and operational planning, defense simulation/campaign analysis/wargaming, enterprise-architecture-to-executable-model transformation, digital twins and cyber-physical systems, model verification/validation/continuous evolution, resource-scheduling/dependency/risk modeling, and the integration of machine learning, multi-agent RL, and LLMs into simulation frameworks. 37 references spanning 2015-2026.

Key findings. - DEVS (Discrete-Event System Specification) provides a rigorous mathematical formalism for modeling discrete-event systems, with stated advantages of completeness, verifiability, extensibility, and maintainability, and implementations across C++, Java, and Python [1]. - The xDEVS framework combines traditional DEVS implementations with cloud deployment and parallel/distributed (PDES) execution, achieving a stated speedup of 15.95x on distributed systems while preserving model reusability and semantic rigor [2]. PDES uses both optimistic and conservative synchronization mechanisms [3]. - Validation studies report discrete-event models achieving less than 5% deviation from actual plant data, confirming precision in capturing manufacturing process behavior [5]. - Agent-based modeling (ABM) captures decentralized decision-making, inter-agent interactions, bounded rationality, and emergent collective phenomena; military applications include UAV swarm precision strikes (target destruction rates, attrition as functions of communication, maneuverability, firepower, sensing) [8] and network-centric warfare under communication-failure conditions [9]. - The FLAME-GPU framework accelerates agent-based transport models via GPU, achieving a stated 100x improvement over CPU-based simulation while scaling to vehicle populations of tens of thousands [10]. - System Dynamics (SD) captures causal feedback loops, time-delayed responses, and nonlinear relationships for strategic planning, resource management, and organizational behavior; it models supply-chain vulnerability risk contagion [11] and quantifies nonlinear competency-development/risk/efficiency relationships in HR development [12]. - Hybrid simulation (DES + SD + ABM) integrates macroscopic aggregate dynamics with microscopic behavioral heterogeneity, applied to healthcare, supply chains, and manufacturing; the synthesis stresses no single methodology dominates across all contexts [6]. - Digital transformation in project/operations management reduced operational expenses by 25% while improving schedule precision by 40% [13]. - SysML integrated with the IMPRINT discrete-event human-performance tool creates traceability from system-design models to human-performance predictions (operator workload, mission effectiveness), keeping SysML as authoritative source of truth [14]. - UAV logistics hierarchical channel management increased channel utilization

by 25% while reducing conflict rates by 35% [15]. - Real-time digital-twin intrusion detection for industrial SCADA achieved 96.3% attack-detection F1-score with sub-500-millisecond latency, outperforming rule-based approaches [25]. - 6G-enabled DT framework achieved 0.8ms end-to-end latency using terahertz communications and intelligent reflecting surfaces for ultra-reliable industrial control [27]. - MBSE proof-of-concept digital twin (LEGO physical system, Raspberry Pi, FlexSim DES, Capella/Arcadia, MQTT) demonstrated bidirectional physical-digital synchronization and virtual commissioning before physical deployment [21]; spacecraft digital twins via mechanism-data fusion support on-orbit simulation of rendezvous, docking, and crewed operations [26]. - ML/RL/LLM integration into simulation: ASSUME agent-based electricity-market framework uses RL for adaptive bidding without specifying dominant strategies a priori [34]; DES+RL hybrids optimize factory routing while preserving process-flow and resource-contention dynamics [35]; Simio (COTS DES) now embeds ML without external programming [35]; process mining extracts behavioral models from execution data for DT construction [36]; LLM-driven wargaming generates injects and feedback aligned to MITRE ATT&CK [19][37].

Named frameworks/systems/standards. DEVS (Discrete-Event System Specification) formalism, xDEVS toolkit (cloud-enabled parallel/distributed DEVS), PDES (Parallel Discrete-Event Simulation) with optimistic and conservative synchronization, FLAME-GPU (GPU-accelerated agent-based simulation framework), System Dynamics (SD), Agent-Based Modeling (ABM), Hybrid simulation (DES + SD + ABM), SysML (Systems Modeling Language), IMPRINT (human-performance discrete-event tool), Model-Based Systems Engineering (MBSE), Capella / Arcadia (MBSE method and tool), FlexSim (discrete-event simulation), Simio (COTS DES with embedded ML), ASSUME (agent-based electricity-market simulation with RL), Digital Twin / Cyber-Physical Systems, Quintuple helix DT verification framework, Opal-RT and Typhoon HIL real-time simulators, MQTT, OPC-UA, ANSI C37.118, IEC 61499 function blocks protocols/standards, MITRE ATT&CK (referenced for AI wargaming inject alignment), Angloval tactical military scenario/experimentation environment, Process mining for DT development, BIM (Building Information Modeling) integrated with DES and game theory, Large Language Models for wargaming/national-security applications

Top sources. 1. J. L. Risco-Martin, S. Mittal, K. Henares, R. Cardenas, P. Arroba. xDEVS: A toolkit for interoperable modeling and simulation of formal discrete event systems. 2022. doi: 10.1002/spe.3168 2. J. L. Risco-Martin, K. Henares, S. Mittal, L. F. Almendras, K. Olcoz. A unified cloud-enabled discrete event parallel and distributed simulation architecture. 2022. doi: 10.1016/j.simpat.2022.102539 3. S. Brailsford, T. Eldabi, M. Kunc, N. Mustafee, A. Osorio. Hybrid simulation modelling in operational research: A state-of-the-art review. 2019. doi: 10.1016/J.EJOR.2018.10.025 4. A. Sayghe. Digital twin-driven intrusion detection for industrial SCADA: A cyber-physical case study. 2025. doi: 10.3390/s25164963 5. B. Soykan, G. Blanc, G. Rabadi. A proof-of-concept digital twin for real-time simulation: Leveraging a model-based systems engineering approach. 2025. doi: 10.1109/ACCESS.2025.3557367

Contribution to AEDA. This layer is the engine that converts AEDA's static Enterprise Digital Twin into executable, time-advancing behavior. The literature establishes a methodological

spine for the Simulation layer: DEVS supplies a formal, verifiable, composable specification (the bridge from the Enterprise Ontology/Knowledge Graph to runnable model components), ABM supplies decentralized actor and emergent-behavior modeling (the natural execution substrate for the Agent Swarms layer), and System Dynamics supplies aggregate feedback/policy dynamics for strategic decision support. Hybrid simulation is the key design pattern, because BEA/DoDAF enterprise behavior spans aggregate flows, discrete resource contention, and heterogeneous decision-making actors simultaneously. MBSE/SysML-to-simulation traceability (Capella/Arcadia, SysML-to-IMPRINT) is the concrete mechanism for AEDA's core claim of evolving DoDAF DM2 from documentation into computation: it keeps the architecture model as authoritative source of truth while generating executable predictions. The DT/cyber-physical findings (sub-500ms SCADA detection, 0.8ms 6G loops, spacecraft mechanism-data fusion) show the bidirectional real-time coupling AEDA's Enterprise Digital Twin requires, and the ML/RL/LLM integration findings connect this layer forward to the Optimization and Decision layers (RL for strategy search, surrogate models for speed, LLMs for scenario generation and executive-facing narrative).

Design implications. - Adopt DEVS as the formal specification standard for AEDA simulation components so models inherit completeness, verifiability, extensibility, and maintainability, and so architecture elements map cleanly to composable, reusable simulation entities. Use a DEVS-class engine (xDEVS pattern) to enable cloud-distributed PDES execution when campaign-scale models demand it. - Build the Simulation layer as a hybrid engine, not a single method: DES for resource queues/scheduling/dependencies, ABM for organizational actors and emergent behavior feeding the Agent Swarms layer, and SD for aggregate policy/feedback dynamics. Method choice should be driven by aggregation level, actor heterogeneity, temporal resolution, and data availability. - Make MBSE/SysML the authoritative source of truth and generate executable models from it (SysML-to-IMPRINT, Capella/Arcadia patterns) so the AEDA brain preserves architecture-to-simulation traceability rather than maintaining a divergent parallel model. This operationalizes the 'documentation to computation' thesis directly. - Engineer the Enterprise Digital Twin for bidirectional real-time coupling with latency budgets in mind (sub-500ms detection and sub-millisecond control loops are demonstrated), and use DT reference models for continuous behavioral validation and anomaly/attack detection against normal operating envelopes. - Plan for continuous model evolution: reuse model-based-design validation methods for continuous DT validation, use parameter estimation from historical data for automatic drift correction, and use process mining to auto-construct behavioral models from execution logs, reducing manual specification bottlenecks. - Embed ML/RL/LLM as first-class simulation capabilities: RL agents for strategy exploration inside DES/ABM environments (feeding Optimization), surrogate/ML models for speed, and LLM-driven scenario/inject generation aligned to threat frameworks (e.g., MITRE ATT&CK) for the Decision/Executive Support layers. Prefer COTS DES that already embeds ML (Simio pattern) to lower integration cost. - Standardize on interoperability protocols and treat simulation output as an organizational knowledge asset: adopt OPC-UA/MQTT/IEC 61499/ANSI C37.118 where relevant, and use two-tier data architectures separating representation data (model structure/parameters) from operational data (results/outcomes) so simulation runs become

queryable organizational knowledge rather than isolated studies. - Use hardware-in-the-loop / co-simulation testbeds (Opal-RT, Typhoon HIL pattern) and formal reachability analysis for FMI/black-box components to establish ground-truth validation and safety certificates before trusting simulation outputs in mission-critical decisions.

Risks/limitations. - No single methodology dominates: choosing the wrong method for the aggregation level, actor heterogeneity, temporal resolution, or data availability degrades fidelity. The synthesis explicitly warns method selection is problem-dependent. - Validation complexity is highest exactly where AEDA most needs it: the summary table rates hybrid simulation and digital twins as Very High validation complexity. Model validation is described as a critical bottleneck in DT deployment, requiring continuous validation, parameter drift detection, reachability analysis, and HIL testing. - Quantitative results are domain-specific and not universal: the <5% plant-data deviation [5], 15.95x [2] and 100x [10] speedups, 25%/40% cost/schedule gains [13], 96.3% F1 / sub-500ms [25], and 0.8ms [27] figures come from specific case studies (manufacturing, traffic, SCADA, 6G bearing-fault detection) and should not be assumed to transfer to enterprise/mission contexts without revalidation. - Real-time and distributed performance carry hidden cost: PDES correctness depends on optimistic/conservative synchronization, and ultra-low-latency results (0.8ms) assume advanced infrastructure (terahertz comms, intelligent reflecting surfaces) not present in typical enterprises. - Black-box and FMI-compliant components limit verifiability; reachability analysis must rely on data-driven dynamic sensitivity without full source access, yielding probabilistic rather than absolute guarantees. - ML/RL/LLM-in-the-loop introduces new failure modes: RL strategy search and LLM-generated injects can produce unvetted or non-reproducible behaviors, and several cited results derive from proof-of-concept or non-peer-reviewed sources (e.g., LEGO/Raspberry Pi PoC [21], arXiv preprint [27]), so maturity for mission-critical DoD use is not yet established. - Enterprise architectures frequently remain abstract and disconnected from execution; transforming EA into executable models requires systematic mapping of architectural elements to simulation entities, constraints, and decision rules, which is non-trivial and a primary integration risk for the AEDA stack.

LR07: Monte Carlo Decision Optimization, Uncertainty Quantification, and AI-Supported Enterprise Decision Architecture

AEDA layer: Simulation + Optimization (Monte Carlo, uncertainty quantification, sensitivity analysis), feeding the Decision Layer and Executive Support

Scope. Comprehensive review of Monte Carlo simulation, probabilistic risk analysis, uncertainty quantification (UQ), stochastic optimization, Bayesian decision theory, portfolio optimization, and AI-supported decision intelligence, with emphasis on integrating uncertainty modeling into enterprise architectures and executive decision environments. Covers foundational MC methods and variance reduction; polynomial chaos and non-intrusive UQ; Bayesian networks and probabilistic risk assessment; scenario-based, robust, and reliability-based stochastic optimization; portfolio optimization from mean-variance through deep reinforcement learning;

AI-driven decision support architecture, data/governance foundations, and human-AI teaming; sampling strategies and global sensitivity analysis; and applications across energy, finance, healthcare, and infrastructure. 104 references.

Key findings. - Classical Monte Carlo convergence is O(N^-0.5), so computational cost grows quadratically with desired accuracy, which is prohibitive for high-fidelity simulations requiring thousands of forward-model evaluations (e.g., aerospace). - Multilevel Monte Carlo (MLMC) and multifidelity Monte Carlo (MFMC) extend control-variate ideas to do most evaluations on low-accuracy/low-cost models and few on high-accuracy/high-cost models, reducing computational burden while maintaining statistical accuracy. Randomized quasi-Monte Carlo with scrambled Sobol' sequences shows smaller bias and RMSE than standard MC for risk-averse stochastic optimization. - Sparse polynomial chaos expansion (PCE) combined with Latin Hypercube Sampling achieved 40-60% computational time reductions versus traditional approaches while improving prediction accuracy by 15-25% [ref 1]. PCE generally shows superior accuracy and efficiency versus MC for smooth stochastic responses and is non-intrusive (no code modification needed). - Global sensitivity analysis via Sobol' indices (first-order direct effects and total-order including interactions) computed from PCE surrogates enables variance decomposition at negligible cost after surrogate construction; UQ workflows report 40-94% computational reductions versus standard MC. Sparse PCE on a concrete face rockfill dam used 50-75% fewer samples with superior accuracy. - PLS-PCE (Partial Least Squares PCE) reduced a 37-variable high-dimensional electromagnetic design to converged sensitivity analysis with only 30 analysis points; WAFP (weighted approximate Fekete points) least-squares PCE matched MC results far more efficiently for cardiovascular models. - Bayesian networks enable quantitative risk assessment via probabilistic graphical reasoning. Bayesian Belief Network + 5D BIM found probabilistic cash-flow ranges deviate 11-130% from deterministic estimates when risk impacts are included. Markov-chain + Bayesian dynamic risk framework achieved 78% predictive accuracy in forecasting risk-state evolution. - Pre-posterior Bayesian (value-of-information) analysis quantifies expected gain from acquiring data before irreversible decisions; for structural health monitoring, sensor location/configuration mattered more for decision value than sensor quantity, and excessive measurement noise should be controlled via careful sensor selection. - Scenario-based two-stage stochastic optimization with LSTM-XGBoost forecasting (Monte Carlo dropout + quantile regression for UQ) outperformed deterministic/rule-based dispatch: a South African case achieved ZAR 15 billion (0.9%) cost reduction with improved reliability (1625 MWh vs 3538 MWh load shedding) over a seven-day horizon. Risk-averse formulations use CVaR and variance constraints. - Deep reinforcement learning (PPO, DQN, A3C) outperforms classical mean-variance/Markowitz optimization for dynamic allocation under regime shifts. NLP-enhanced predictive analytics reached Sharpe ratio 1.37 vs 0.74 for classical Markowitz. FD-RLPO improved annualized returns 7.98-19.80% at given risk levels; hybrid LSTM-TI-Adaptive gave +4.2 pp annual return and -3.1 pp max drawdown versus fixed-parameter systems in 2022-2024 Russian-market conditions. - AI-driven decision support: a hybrid engine (Multiresolution Dynamic Time Warping + Hierarchical Quantum Recurrent Reservoirs + Multi-Head Hypergraph Attention) achieved 94.3% forecast accuracy at 1-day and 74.4% at 150-day horizons. Probabilistic OCR with Monte Carlo Dropout

achieved 99.13% accuracy, mean confidence-interval width +/-1.22 for financial fields, expected calibration error 2.9%, flagging uncertain outputs for human review. - Layered AI integration is the architectural consensus: the LEAIM model defines five formally separated layers (data acquisition, model lifecycle management, model serving, orchestration, governance) with explicit dependency constraints to prevent coupling and enable scalability, resilience, and governance. Governance, explainability, security, and compliance must be built-in design requirements, not after-the-fact controls. - Distributed AI control for procurement/supply chain (edge-to-cloud fusion, deep-learning forecasting, RL policies, federated multi-agent coordination) achieved 22% improvement in data-fusion accuracy, 75% reduction in coordination delays, and >90% decrease in procurement exceptions. Healthcare SNGP UQ reached AUROC ~0.85 / AUPRC ~0.52 for in-hospital mortality; a Bayesian-network survival model reached AUC 0.880, F1 0.779.

Named frameworks/systems/standards. Monte Carlo (MC) simulation, Multilevel Monte Carlo (MLMC), Multifidelity Monte Carlo (MFMC), Randomized quasi-Monte Carlo (scrambled Sobol' sequences), Polynomial Chaos Expansion (PCE), Sparse PCE / sparse polynomial chaos, Latin Hypercube Sampling (LHS), Partial Least Squares PCE (PLS-PCE), Sobol' indices (global sensitivity analysis), Stochastic Reduced Order Models (SROMs), Kriging metamodels / surrogate models, Stochastic Response Surface Method (SRSM) with ADIFOR automatic differentiation, Weighted Approximate Fekete Points (WAFP), Coherence-optimal sampling, Two-phase MC / Non-Intrusive Polynomial Chaos (MSC/NIPC) for QMMU, Bayesian networks / Bayesian Belief Networks, 5D Building Information Modeling (BIM), Fuzzy Dynamic Bayesian Networks, EDIB model (Event Tree-DEMATEL-ISM-Bayesian Network), Adversarial Risk Analysis (ARA), Pre-posterior Bayesian value-of-information analysis, Markov chain risk frameworks, Conditional Value-at-Risk (CVaR), Two-stage stochastic programming, LSTM-XGBoost forecasting with Monte Carlo Dropout and quantile regression, NSGA-II (simheuristic with Monte Carlo), GLUE method, Black-Litterman model, Deep Reinforcement Learning: PPO, DQN, A3C, PPO-HER, Dual-Model PPO, FD-RLPO (Feature Domain-based RL Portfolio Optimization), Multi-agent RL (centralized training, decentralized execution), Spectral Normalized Neural Gaussian Process (SNGP), Monte Carlo Dropout, LEAIM (Layered Enterprise AI Integration Model), Federated learning (differential privacy, secure aggregation, homomorphic encryption, multiparty computation), Retrieval-Augmented Generation + Knowledge Graphs, SHAP and LIME (explainability), QAOA and quantum annealing, Apache Spark, Python/R, TensorFlow, PyTorch, Kubernetes, PyThia, UQTk (UQ toolkits), Oracle Crystal Ball, GDPR, CCPA, NIST (governance standards)

Top sources. 1. J. Zhang. Modern Monte Carlo methods for efficient uncertainty quantification and propagation: A survey. 2020. doi: 10.1002/wics.1539 2. F. Menhorn, G. Geraci, D. Seidl, Y. Marzouk, M. Eldred, H. Bungartz. Multilevel Monte Carlo Estimators for Derivative-Free Optimization Under Uncertainty. 2023. doi: 10.1615/int.j.uncertaintyquantification.2023048049 3. M. Ehre, I. Papaioannou, D. Straub. Global sensitivity analysis in high dimensions with PLS-PCE. 2020. doi: 10.1016/j.ress.2020.106861 4. S. Yadav, D. Mewani. LEAIM: A layered enterprise architecture model for scalable and governed integration of artificial intelligence in distributed systems. 2026. doi: 10.63282/3050-9262.ijaidsml-v7i1p130 5. L. Long, M. Dohler,

S. Thons. Determination of structural and damage detection system influencing parameters on the value of information. 2020. doi: 10.1177/1475921719900918

Contribution to AEDA. This review supplies the quantitative engine for the AEDA Simulation and Optimization layers and the probabilistic substrate that the Decision Layer and Executive Support consume. It establishes that an autonomous enterprise decision stack must propagate uncertainty rather than rely on point estimates: Monte Carlo (with MLMC/MFMC and quasi-MC variance reduction) and polynomial chaos surrogates provide the forward-propagation mechanics for the digital-twin/simulation layers, while Sobol'-index global sensitivity analysis identifies which architecture parameters actually drive output variability (the inputs the optimization layer should act on). For optimization, it shows scenario-based and risk-averse stochastic programming (CVaR), reliability-based design, and deep RL allocation as the methods for choosing actions under uncertainty. For the Decision and Executive Support layers, Bayesian decision theory and pre-posterior value-of-information analysis give the normative calculus for expected-utility maximization and for deciding when to gather more data. Critically, it maps the layered enterprise AI integration pattern (LEAIM) directly onto AEDA's separation of concerns, and it operationalizes calibrated uncertainty (confidence intervals, expected calibration error, Monte Carlo Dropout) as the trigger for human-in-the-loop review, which is the governance backbone for autonomous decision support.

Design implications. - Build the AEDA Simulation layer on a multifidelity hierarchy: use MLMC/MFMC to run most evaluations on cheap low-fidelity surrogates and reserve high-fidelity runs for refinement, since classical MC at O(N^-0.5) does not scale to high-dimensional architecture evaluation. - Standardize on non-intrusive sparse PCE plus Latin Hypercube Sampling as the default surrogate-and-sampling layer (40-60% time reduction, 15-25% accuracy gain reported); non-intrusiveness lets the brain wrap existing models without code modification. - Make global sensitivity analysis (Sobol' indices from PCE surrogates) a standing output so the Optimization and Decision layers can rank parameter importance and prune the decision space cheaply after surrogate construction. - For high-dimensional architecture spaces, adopt dimension-reduction surrogates (PLS-PCE) that converge with very few samples (37 variables to 30 points) before invoking expensive optimization. - Represent dependencies and causal risk structure with Bayesian networks; couple them with value-of-information (pre-posterior) analysis so the Executive Support layer can recommend whether to act or gather more data, and where to place new sensors/data sources for maximum decision value. - Encode risk appetite explicitly in the optimization objective via CVaR/variance constraints and two-stage stochastic programs rather than optimizing expected value alone, so the Decision Layer avoids extreme-case outcomes. - Adopt the LEAIM five-layer separation (data acquisition, model lifecycle, model serving, orchestration, governance) with explicit dependency constraints as the AEDA reference layering, embedding governance/explainability/compliance as design requirements, not bolt-ons. - Instrument every model output with calibrated uncertainty (confidence intervals, predictive entropy via Monte Carlo Dropout, expected calibration error) and use uncertainty thresholds to route low-confidence cases to human review, preserving accountability and override. - Provide model-agnostic explainability (SHAP, LIME) and RAG-plus-knowledge-graph interfaces at the Executive Support layer to support interrogation, trust calibration, and regulatory traceability

from raw data to recommendation. - Provision a cloud-native, containerized stack (Apache Spark, Python/R, TensorFlow/PyTorch, Kubernetes) with dedicated UQ libraries (PyThia, UQTk) and use federated learning with layered privacy controls (differential privacy, secure aggregation, homomorphic encryption) when integrating siloed enterprise data.

Risks/limitations. - Classical Monte Carlo is computationally prohibitive at high fidelity (O(N^-0.5) convergence); without variance reduction or surrogates the simulation layer will not scale. - Bayesian methods require prior specification, introducing subjectivity that may not reflect true uncertainty; epistemic uncertainty from incomplete knowledge and data scarcity remains hard to characterize. - Non-stationarity in financial markets and infrastructure systems demands adaptive models that detect and respond to regime shifts; static models degrade, and high training accuracy does not guarantee performance on out-of-distribution samples in critical applications. - Explainability remains insufficient: many organizations report AI systems lack transparency adequate for stakeholder trust and regulatory compliance; algorithmic bias and ethical-governance gaps are ongoing. - Cultural and organizational resistance frequently outweighs technological limits; data quality, integration complexity, governance requirements, and limited in-house analytical expertise impede deployment and delay value realization. - Quantum optimization approaches (QAOA, quantum annealing) are limited by current hardware maturity, so only hybrid quantum-classical approaches are practical near-term; treat as forward-looking, not production-ready. - Surrogate/metamodel methods (PCE, Kriging, SROMs) trade exactness for speed; accuracy depends on smoothness assumptions and adequate sampling, so validation and sensitivity analysis are required to build stakeholder confidence before scaling. - Most reported quantitative gains (Sharpe ratios, accuracy, cost reductions) are single-domain case studies (energy, finance, healthcare, specific national markets); generalization to a DoD enterprise architecture context is unproven and should be validated before adoption.

LR08: AI-Augmented Portfolio Management: State-of-the-Art Review

AEDA layer: Optimization layer (portfolio management, capability-based planning, strategic investment analysis, mission value assessment, earned value management, optimization algorithms)

Scope. State-of-the-art review of how AI systems continuously evaluate portfolios, recommend investment strategies, identify redundancies, and maximize mission outcomes under resource constraints. Spans traditional mean-variance portfolio theory and its AI extensions; capability-based planning (asset-centric to outcome-centric); strategic investment analysis and mission value assessment; earned value management (EVM) evolution including uncertainty-aware and AI-integrated variants; reinforcement-learning and metaheuristic portfolio optimization; explainable AI decision support; redundancy detection via semantic similarity; resource allocation under multi-dimensional constraints; and sector applications across defense/space, healthcare/public sector, and infrastructure/emergency management. 52 references.

Key findings. - Portfolio management integrates cost, schedule, and performance dimensions to optimize asset allocation across constrained resource environments; traditional fundamental/technical analysis is being augmented by ML that analyzes large datasets more efficiently, with convex optimization plus multi-factor models and multi-armed bandit approaches enhancing performance during market volatility [1][2][3]. - Capability-based planning marks a shift from asset-centric to outcome-centric portfolio management; integrating project portfolio management with enterprise architecture and capability-based planning enables multi-criteria selection grounded in strategic objectives, using AHP and linear programming [6]. Defense research quantifies capability value by combining investment portfolio measures with technological aging metrics, linking investment projects to operational systems and mission outcomes [7]. - Stochastic project portfolio selection maximizes expected value via genetic algorithms plus Monte Carlo simulation, handling schedule interdependencies, budget constraints, risk registers, and portfolio reliability constraints [8]; bi-objective formulations minimize total cost while maximizing stakeholder interest fulfillment, with nonlinear programming and heuristics exploring Pareto-optimal sets [8][9]. - EVM remains foundational for integrated project control via PV, EV, AC, CPI, and SPI [4]. Technique effectiveness varies by project type and stage: Earned Schedule predicts more accurately in early stages, Earned Duration more reliably at later stages [12]. Uncertainty-aware variants: Grey Earned Duration Management (GEDM) gives upper/lower completion bounds [13]; Z-number based EVM (ZEVM) combines possibility and reliability via fuzzy logic to improve cost-duration accuracy under high uncertainty [14]; agent-based EVM adapts to agile with real-time task-board tracking [15]; sustainable EVM adds environmental metrics [16]. - Reinforcement learning enables dynamic adaptive portfolio optimization: DDPG, PPO, and A2C optimize asset allocation by learning from market interactions and outperform traditional methods on Yahoo Finance data [17]. Deep RL stock trading generates cumulative returns of 85.12% with 25-40% maximum drawdown reduction [18]; actor-critic methods deliver superior risk-adjusted performance via continuous rebalancing [19]. - SLA-aware multi-objective RL (SLA-MORL) achieves 67.2% reduction in training time for deadline-critical jobs and 68.8% reduction in costs for budget-constrained workloads while maintaining 73.4% SLA compliance improvement; intelligent initialization and dynamic weight adaptation eliminate cold-start, reducing initial exploration overhead by 60% [20]. - Strategic Governance Intelligence uses sentence-embedding semantic similarity to detect redundant/overlapping project proposals, LLM-based strategic reasoning for explainable governance insights, and Monte Carlo probabilistic risk modeling; interactive web implementation provides radar-chart risk visualization and ISO 31000-compliant risk matrices [21]. - Human-AI collaboration is most effective as augmentation, not replacement: hybrid models let AI handle data-intensive analysis while humans focus on judgment, ethics, and strategic foresight [22][23]. An AI-orchestrated hybrid framework (neuro-fuzzy executive strategy, quantum-inspired multi-objective evolutionary planning, moral-normative policy alignment) achieves 90.2% decision accuracy and 95.6% ethical compliance; graph-attentive influence modeling plus GAN counterfactual simulation enables policy impact prediction [24]. - Metaheuristic and quantum-inspired optimizers: genetic-algorithm robust mean-variance portfolios are more stable and higher-return than traditional MV across Hang Seng/DAX/FTSE/S&P 100/Nikkei 225 [1]; NSGA-II plus constraint programming cuts labor costs 13.2% while improving satisfaction

[25]; Enhanced Egret Swarm + beetle antennae search achieves superior Sharpe ratios [26]; HSL-DFA achieves 93 Mbps throughput at the 200th node [28]; Quantum SVM achieves 89.65% portfolio performance, surpassing other quantum algorithms by 25.15% [29]. - Mission-critical AI resource allocation: attention-augmented multi-agent RL with Decaying Attention Guidance achieves 34% faster convergence, 51.5% +/-1.7% improvement in cost-effectiveness vs rule-based baselines, 94.9% +/-0.6% critical-target elimination, real-time response (<0.3s), and 95.2% resource utilization [34]. For space, mixed-integer programming and a multi-period precedence-constrained knapsack maximize stakeholder interest fulfillment under periodic budget constraints while sequencing capability development [36]. - Quantitative impact summary: AI-driven strategies achieve 15-20% reductions in portfolio volatility and 30% faster rebalancing [50]; dynamic resource allocation reaches 91.3% utilization with 99.2% demand satisfaction [51]; multi-objective optimization reduces operational costs 10-23% [38]; AI predictive models improve forecasting accuracy 32-45% [42]; real-time systems cut decision cycles from days to milliseconds [34]; anomaly detection identifies financial irregularities with 98% accuracy in real time [52]. - Implementation challenges center on data quality, algorithmic transparency, and model bias; explainable AI and fairness-aware algorithms are essential; longitudinal performance and user-trust evaluation remains limited; GDPR, NIST AI RMF, and ISO/IEC standards are frequently referenced but only limitedly implemented in live systems [22][42]. Emerging integration approaches include physics-informed ML (Hamilton-Jacobi-Bellman value approximation with conformal-prediction verification for formal safety) [47], ML+MILP hybrids replacing nonlinear constraints (validated on space logistics) [48], and federated/decentralized optimization with centralized training [49].

Named frameworks/systems/standards. Mean-variance optimization (Modern Portfolio Theory), Multi-armed bandit algorithm, Multi-factor models, Analytical Hierarchy Process (AHP), Linear programming, Capability-based planning (CBP), Enterprise architecture integration with project portfolio management, Defense capability metric (investment portfolio + technological aging), Genetic algorithm simheuristic (stochastic project portfolio selection with reliability constraints), Monte Carlo simulation, Nonlinear programming with heuristic algorithms (Pareto-optimal sets), Real-World Value Assessment (RWVA), Earned Value Management (EVM): PV, EV, AC, CPI, SPI, Earned Duration (ED), Earned Schedule (ES), Grey Earned Duration Management (GEDM), Z-number based EVM (ZEVM), Agent-based EVM simulation model, Sustainable EVM, Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), Advantage Actor-Critic (A2C), Actor-critic algorithms, SLA-MORL (SLA-aware multi-objective reinforcement learning), Strategic Governance Intelligence (sentence embeddings + LLM reasoning + Monte Carlo), ISO 31000 risk matrices, AI-orchestrated hybrid optimization (neuro-fuzzy + quantum-inspired evolutionary + moral-normative alignment), Graph-attentive influence modeling + GAN counterfactual simulation, NSGA-II (multi-objective genetic algorithm) + constraint programming, Enhanced/NBESOA Egret Swarm Optimization + beetle antennae search, Particle Swarm Optimization (PSO) + simulated annealing, Hybrid Snow Leopard and Dark Forest Algorithm (HSL-DFA), Quantum Support Vector Machine (QSVM), Attention-augmented multi-agent RL with Decaying Attention Guidance, Integer linear programming + hybrid variable neighborhood search +

simulated annealing, Mixed-integer programming (space systems acquisition); multi-period precedence-constrained knapsack, Improved particle swarm optimization (ammunition support task allocation), Economic and Clinical Intelligence framework, AI-enhanced smart KPIs (descriptive, predictive, prescriptive), Physics-informed ML (Hamilton-Jacobi-Bellman value approximation + conformal prediction), ML + MILP integration (space logistics network optimization), Federated learning / decentralized optimization (centralized training), GDPR, NIST AI RMF, ISO/IEC standards, Multi-criteria decision analysis (MCDA)

Top sources. 1. S. Zhu, B. E. Robertson, D. N. Mavris. Space systems development and acquisition modeling using mixed integer programming. 2022. doi: 10.2514/6.2022-4354 2. R. Chessex, Z. Mathews. A defense capability metric from investment portfolio and technological aging. 2024. doi: 10.1109/ACCESS.2024.3458094 3. A. Aldea, M. Iacob, M. Daneva, L. H. Masyhur. Multi-criteria and model-based analysis for project selection: An integration of capability-based planning, project portfolio management and enterprise architecture. 2019. doi: 10.1109/EDOCW.2019.00032 4. M. Saiz, D. Lopez-Lopez, L. Calvet, A. A. Juan. A genetic algorithm simheuristic for solving the stochastic project portfolio selection problem with portfolio reliability constraints. 2025. doi: 10.1111/itor.70064 5. K. Ho, Y. Shimane, M. Isaji. Generalizing space logistics network optimization with integrated machine learning and mathematical programming. 2024. doi: 10.2514/1.A36122

Contribution to AEDA. Provides the evidence base for AEDA's Optimization layer, the stage that converts the Enterprise Digital Twin and Simulation outputs into ranked investment and capability decisions under resource constraints. The review supplies four assets directly transferable to AEDA: (1) a capability-based planning spine that links investment projects to operational systems and mission outcomes [6][7], aligning AEDA optimization with BEA/DoDAF capability and outcome views rather than asset inventories; (2) defense/space-native formulations (mixed-integer programming, multi-period precedence-constrained knapsack for stakeholder-interest maximization under periodic budgets [36]; ML+MILP for space logistics [48]) that show portfolio optimization can be expressed as solvable mathematical programs over the enterprise ontology; (3) continuous-evaluation machinery (RL/multi-objective RL for dynamic rebalancing [17][18][20], semantic-similarity redundancy detection [21]) that operationalizes AEDA's promise of a portfolio that is continuously re-optimized rather than documented; and (4) an explainable, human-in-the-loop governance pattern [21][22][23][24] that the Decision and Executive Support layers can consume. It establishes EVM (CPI/SPI plus uncertainty-aware GEDM/ZEVM variants) as the measurement substrate feeding optimization, and quantifies the achievable gains (forecasting accuracy +32-45%, cost -10-23%, volatility -15-20%, sub-second decision latency) that justify the computational stack.

Design implications. - Model the optimization layer as a constrained mathematical program over the enterprise ontology: encode capability-based planning as multi-criteria selection (AHP + linear/mixed-integer programming) and adopt the multi-period precedence-constrained knapsack formulation [36] so the brain maximizes stakeholder/mission-interest fulfillment subject to periodic budget constraints and capability-development sequencing. - Make capability value, not asset cost, the optimization objective: implement a capability metric combining

investment-portfolio measures with technological-aging signals [7] so AEDA recommendations target mission outcomes and surface obsolescence-driven reinvestment. - Build the EVM telemetry feed first: instrument PV/EV/AC/CPI/SPI as the measurement substrate [4], select forecasting technique by project maturity (Earned Schedule early, Earned Duration late) [12], and adopt uncertainty-aware variants (GEDM bounds [13], Z-number EVM [14]) so the optimizer consumes ranges, not single-point estimates. - Use reinforcement learning / multi-objective RL for continuous rebalancing under constraints (cost, performance, SLA) [17][20], and pre-seed with historical-learning initialization and dynamic weight adaptation to avoid cold-start exploration overhead [20]; pair with Monte Carlo / genetic-algorithm simheuristics for stochastic selection with reliability constraints [8]. - Implement redundancy detection as a first-class portfolio function: sentence-embedding semantic similarity plus LLM strategic reasoning to flag overlapping initiatives and surface consolidation opportunities with explanations [21], feeding the Decision layer. - Keep humans in the loop by design: architect AEDA optimization as an augmentative layer where AI handles data-intensive analysis and humans own judgment, ethics, and foresight [22][23]; expose explainable outputs (radar-chart risk views, ISO 31000 risk matrices) [21] for Executive Support. - Wire optimization to the Simulation/Digital Twin layers via ML+MILP hybrids that replace nonlinear constraints with trained models (proven on space logistics) [48] and physics-informed ML with conformal-prediction verification where formal safety guarantees are required [47]. - Plan for decentralized/federated optimization (centralized training, decentralized execution) [49] to scale across enterprise units while preserving data governance, and bake GDPR/NIST AI RMF/ISO-IEC alignment plus data-lineage and audit trails into the stack from the outset [22][42].

Risks/limitations. - Data quality, algorithmic transparency, and model bias are the dominant implementation barriers; without explainable-AI and fairness-aware methods plus data-governance integration, AI-driven portfolio decisions are unreliable and inequitable [42]. - Longitudinal evaluation is scarce: few studies track system impact or user trust over extended periods, so reported gains may not persist; multidisciplinary, long-term performance tracking is needed before operational reliance [22]. - Regulatory and ethical frameworks (GDPR, NIST AI RMF, ISO/IEC) are frequently cited but only limitedly implemented in live systems, leaving a compliance and trust gap for production deployment [22][76 context]. - Most reported quantitative results (returns of 85.12%, 89.65% QSVM performance, 90.2% decision accuracy, 98% anomaly-detection accuracy, etc.) come from finance, HPC, telecom, healthcare, and civil-infrastructure domains, not validated DoD/space enterprise-architecture portfolios; transfer to AEDA mission-value optimization is unproven and benchmarks are heterogeneous and non-comparable. - EVM technique effectiveness varies by project type and progress stage [12], so a single forecasting method risks inaccurate predictions; the optimizer must select methods by maturity and handle the indeterministic nature of outcomes that deterministic models miss [13]. - Cold-start and exploration overhead in reinforcement-learning optimizers degrade early performance unless mitigated by historical-learning initialization and dynamic weighting [20]; market/operational volatility can destabilize learned policies. - AI-orchestrated hybrid frameworks reporting very high ethical-compliance and decision-accuracy rates [24] rely on novel composite methods (neuro-fuzzy, quantum-inspired, GAN counterfactuals) whose

reproducibility and real-world validation are not established. - Over-automation risk: the literature is consistent that replacement (rather than augmentation) of human judgment degrades outcomes for tacit-knowledge, ethical, and contextual decisions [22][23]; an AEDA optimizer that removes the human-in-the-loop forfeits this benefit.

LR09: Explainable AI for Government Decision Support

AEDA layer: Decision/Trust layer (XAI, provenance, governance) — Decision Layer and Executive Support

Scope. A comprehensive review of explainable AI (XAI), trusted AI, AI governance, provenance and traceability systems, assurance cases, auditability, and decision transparency, evaluating how government and defense organizations can deploy AI decision-support systems while maintaining accountability, explainability, and regulatory compliance. Spans 47 references across seven sections: foundational concepts, XAI methodologies, governance/regulatory frameworks, provenance and data governance, assurance cases and certification, government/defense-specific applications, and critical gaps/future directions.

Key findings. - Central tension is the explainability-performance paradox: the most performant models (deep learning, ensembles, complex feature interactions) are inherently opaque, while high-stakes domains (healthcare, finance, criminal justice, defense) require both exceptional accuracy and clear interpretability. DARPA's XAI program (2016-2021) framed this gap, noting early symbolic-reasoning AI was inherently explainable but modern ML created powerful opaque models that cannot explain decisions to human users in mission-critical DoD/allied contexts. - XAI methods are categorized along scope (local vs. global), timing (post-hoc vs. intrinsic), and architecture dependency. Core techniques: LIME (local) and SHAP (global), plus Layer-Wise Relevance Propagation (LRP), Grad-CAM, counterfactual reasoning. A hybrid XAI framework integrating rule-based models with deep learning using LRP and SHAP achieved 94.3% accuracy while maintaining explainability and reaching a trustworthiness index of 92.1%. - A four-axis study of explainability (data, model, prediction, and explanation evaluation) covered 410 peer-reviewed articles published January 2016 to October 2022. - Risk-tiered governance is the dominant pattern: public-sector AI is classified into low-, medium-, and high-risk uses. High-risk systems (eligibility decisions, law-enforcement support, biometric identification) require named accountability ownership, meaningful human oversight, pre-deployment impact assessment, proportionate explainability, data quality/fairness testing, security controls with audit trails, enforceable procurement clauses for vendor accountability, and accessible grievance/review mechanisms. - Governance effectiveness is empirically measurable: organizations combining explainable AI with empowered ethics boards experienced 48% fewer instances of bias and regulatory violations, and 35% fewer regulatory investigations when third-party audits were employed. Advisory-only ethics boards showed limited impact; effectiveness requires integration into core decision-making. - Provenance systems deliver machine-checkable evidence. AI Commit Ledger (Git-native, local-first) generates structured AI Receipts per commit, achieving 94.2% attribution accuracy with less than 340ms overhead

per commit. Blockchain-powered provenance with cryptographic verification, zero-knowledge proofs, and federated logging reduced fraudulent activities by more than 50% versus traditional audit approaches in simulation. - Provenance-based auditing enables fairness audits: in clinical models, logistic regression exhibited statistically significant gender bias (EOD = +0.256, p = 0.0080) while random forest's smaller disparity was not statistically significant. The AI Fairness Provenance Record documents data origin, model choices, and bias metrics so auditors can trace decisions to their source. - Assurance cases provide structured, evidence-backed arguments for AI safety. The Overarching Properties approach (Intent, Correctness, Innocuity) aligns AI/ML component properties to system-level safety for aerospace. A Continuous Assurance Framework integrates design-time, runtime, and evolution-time assurance using formal verification (RoboChart for functional correctness, PRISM for probabilistic risk) and auto-regenerates assurance arguments when specifications or verification results change. PRAISE is a principles-based ethics assurance argument pattern (justice, beneficence, non-maleficence, respect for human autonomy, with transparency supporting). - Two contrasting assurance philosophies: the 'dependability' perspective minimizes trust in AI/ML via defense-in-depth and a hierarchy of simpler 'guard' systems and micro-ODDs, versus the 'trustworthy' perspective that applies assurance to the AI/ML elements themselves. - Public-sector predictive analytics deliver concrete operational gains: 18% reduction in eldercare backlog waitlists within nine months, reduction in crime response lead times from 14 days to under three hours, and improvement in budget variance accuracy from plus/minus 15% to plus/minus 4%. - Defense/military AI requires calibrated trust via training, transparency, human-machine teaming, and robust data governance; short-term hazards include skill fade and corrupted data, long-term risks involve adversarial tampering. Decentralized storage (blockchain) and XAI are proposed to enhance reliability and transparency in military health applications. - Persistent gaps: over 80% of clinical studies use post-hoc, model-agnostic approaches with clinician sample sizes below 25 participants; explanations generally improve trust but frequently increase cognitive load and misalign with domain reasoning. XAI methods carry computational and accuracy trade-offs. 'Compliance asymmetry' (structural friction between regulatory ambition and institutional capacity) particularly burdens low-capacity actors (SMEs, public authorities), compounded by friction between the EU AI Act and GDPR. - Standardization is fragmented: ISO/IEC 24027 and 24368 aim to embed fairness, explainability, and risk control, but enforceability varies across the EU risk-tiered model, China's social-stability focus, and the U.S. decentralized model. Recommendations include mandatory audits, region-specific standard annexes, mutual recognition agreements (MRAs), and alignment with OECD AI Principles.

Named frameworks/systems/standards. DARPA Explainable AI (XAI) program (2016-2021), LIME (Local Interpretable Model-agnostic Explanations), SHAP (SHapley Additive exPlanations), Layer-Wise Relevance Propagation (LRP), Grad-CAM, Counterfactual reasoning / counterfactual explanations, AI Governance Strategic Framework (AIGSF), Five-layer AI governance framework, Five-dimensional trust model (quality, security, privacy, fairness, explainability), AI Commit Ledger (Git-native provenance, AI Receipts), Blockchain-powered data provenance (zero-knowledge proofs, federated logging), AI Product Passport, FUTURE-AI principles (Fairness, Universality, Traceability, Usability, Robustness, Explainability), XAI-

Compliance-by-Design + Technical-Regulatory Correspondence Matrix, AI Fairness Provenance Record, Retrieval-augmented generation (RAG) with provenance metadata for clinical decision support, Overarching Properties approach (Intent, Correctness, Innocuity), Landscape of AI Safety Concerns methodology, Continuous Assurance Framework (RoboChart, PRISM formal verification), PRAISE (principles-based ethics assurance argument pattern), Defense-in-depth / micro-Operational Design Domains (micro-ODDs) guard architecture, Trustworthiness Assurance Assessment (EU AI Act, seven interconnected requirements), EU AI Act, GDPR, HIPAA, FDA AI/ML guidelines, ISO/IEC 24027, ISO/IEC 24368, OECD AI Principles, Mutual Recognition Agreements (MRAs)

Top sources. 1. D. Gunning, E. S. Vorm, Y. Wang, M. Turek. DARPA's explainable AI (XAI) program: A retrospective. 2021. doi: 10.1002/ail2.61 2. R. Dwivedi et al.. Explainable AI (XAI): Core ideas, techniques, and solutions. 2022. doi: 10.1145/3561048 3. S. Ali et al.. Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence. 2023. doi: 10.1016/j.inffus.2023.101805 4. A. Agarwal, M. Nene. A five-layer framework for AI governance: Integrating regulation, standards, and certification. 2025. doi: 10.1108/TG-03-2025-0065 5. Z. Porter, I. Habli, J. McDermid, M. H. L. Kaas. A principles-based ethics assurance argument pattern for AI and autonomous systems (PRAISE). 2022. doi: 10.1007/s43681-023-00297-2

Contribution to AEDA. This review supplies the trust, transparency, and accountability spine for AEDA's Decision Layer and Executive Support tier, and the governance overlay for the entire stack. AEDA evolves BEA/DoDAF from documentation into a computational decision stack; once decisions are produced by GraphRAG, agent swarms, simulation, and optimization, they must be explainable, auditable, and defensible to government accountability standards. The literature operationalizes this: risk-tiered governance assigns named accountability and proportionate explainability to high-stakes decisions; provenance and lineage architecture (lineage graphs, feature-store governance, AI Receipts, fairness provenance records) make every AEDA decision traceable from data source to inference; assurance cases (Overarching Properties, Continuous Assurance, PRAISE) provide the structured, evidence-backed safety/ethics arguments a DoD enterprise needs for certification; and the dependability vs. trustworthy assurance distinction informs how much trust AEDA should place in autonomous agent components versus guard layers. It also maps directly to AEDA's Ontology Governance layer (data quality, fairness testing, runtime policy enforcement) and gives Executive Support its calibrated-trust and human-oversight requirements.

Design implications. - Embed explainability at the core of the AEDA lifecycle, not as an afterthought: every Decision Layer output should carry local (LIME/SHAP) and global explanations plus counterfactuals, and prefer hybrid intrinsically-interpretable-plus-post-hoc architectures (the hybrid LRP+SHAP design reached 94.3% accuracy and a 92.1% trustworthiness index, showing performance and explainability can coexist). - Build a provenance and lineage substrate beneath the Knowledge Graph and Data Sources layers: capture immutable data lineage graphs, feature-store governance, and machine-checkable lifecycle evidence from ingestion to inference. Adopt an AI-Receipts-style ledger (94.2% attribution accuracy, <340ms

overhead) so every agent action and decision is attributable and replayable for post-incident analysis. - Implement risk-tiered governance in the Ontology Governance and Decision layers: classify AEDA decision flows into low/medium/high-risk, and gate high-risk flows (eligibility, law-enforcement support, biometric, defense targeting) with named accountability ownership, meaningful human oversight, pre-deployment impact assessment, fairness testing, audit trails, and accessible redress. - Adopt a compliance-by-design evidence architecture: separate technical evidence generation from governance consumption with explicit emit/store/query interfaces, and maintain a Technical-Regulatory Correspondence Matrix mapping regulatory anchors (EU AI Act, GDPR, and DoD/federal equivalents) to concrete evidence artifacts the executive support layer can surface on demand. - Use assurance cases as a first-class AEDA artifact: structure system-level safety arguments with Overarching Properties (Intent, Correctness, Innocuity) and an ethics argument pattern (PRAISE), and adopt a continuous-assurance workflow that auto-regenerates arguments when specifications or verification results change (formal methods such as PRISM for probabilistic risk, RoboChart for functional correctness). - For agent swarms and the digital twin, apply defense-in-depth: minimize trust in opaque AI/ML components by wrapping them in simpler, verifiable guard layers and micro-ODDs rather than assuming the AI/ML elements themselves are trustworthy. Reserve full trust only where assurance evidence supports it. - Engineer fairness auditing into the pipeline: maintain an AI Fairness Provenance Record (data origin, model choices, bias metrics such as EOD) so auditors can run provenance-based fairness simulations and trace any disparate decision back to its source. - Calibrate trust and preserve human agency in Executive Support: design for human-machine teaming, guard against skill fade and over-reliance, surface explanations that reduce rather than increase cognitive load, and align explanations to operator/domain reasoning. Demonstrate value with measurable operational gains (e.g., backlog and lead-time reductions, budget-variance accuracy improvements) while staying within privacy and equity constraints.

Risks/limitations. - Explainability-performance paradox: the most accurate models remain the least interpretable; AEDA must accept accuracy/computational trade-offs when XAI is added, and cannot assume opacity is removable for free. - Explanations can backfire: over 80% of clinical studies use post-hoc model-agnostic methods with clinician samples below 25; explanations frequently increase cognitive load and misalign with domain reasoning, so naive XAI surfacing in Executive Support can degrade rather than improve decisions. - Evaluation immaturity: XAI explanation quality lacks robust, user-centered evaluation methodologies; small sample sizes and weak capture of human-AI interaction patterns limit confidence that explanations actually support correct decisions. - Compliance asymmetry and enforceability gaps: structural friction between regulatory ambition and institutional capacity burdens low-capacity actors, and frictions between instruments (EU AI Act vs. GDPR) create proportionality and auditability problems AEDA governance must reconcile. - Standardization fragmentation: ISO/IEC 24027 and 24368 and regional regimes (EU risk-tiered, China social-stability, U.S. decentralized) lack consistent enforceability and often fail to accommodate local values; cross-jurisdiction AEDA deployments face harmonization risk absent MRAs and OECD alignment. - Advisory-only governance is ineffective: ethics boards without integration into core decision-

making show limited impact; AEDA governance bodies must hold real authority and pair with third-party audits to realize the measured 48% bias-reduction / 35% fewer-investigations benefits. - Defense-specific threats: short-term skill fade and corrupted data, long-term adversarial tampering of AI-augmented decision systems; AEDA must assure data integrity and maintain human agency under contested conditions. - Provenance/audit overhead and dependencies: ledger and blockchain provenance add latency and operational complexity, and some auditing toolchains remain immature ('AI ethics tools not yet fit for purpose'), risking false assurance if evidence artifacts are incomplete or unverified.

LR10: Autonomous Enterprise Decision Systems: Evolution from Data-Driven to Continuously Optimized AI-Assisted Enterprises

AEDA layer: Autonomous Enterprise synthesis layer (capstone integrating Data Sources, Ontology, Knowledge Graph, GraphRAG, Agent Swarms, Digital Twin, Simulation, Optimization, Decision, and Executive Support layers into a single closed-loop decision stack)

Scope. Forward-looking research assessment synthesizing 150+ peer-reviewed papers (published 2020-2026) on autonomous enterprise decision systems that integrate knowledge graphs, ontologies, large language models, simulations, optimization engines, and digital twins. Covers a three-phase maturity trajectory (data-driven to AI-augmented to continuously optimized AI-assisted enterprise), enabling technologies and architectural integration, semantic interoperability challenges, research gaps (explainability, fairness, robustness), barriers to adoption (organizational, data, workforce), and sector implications for government, defense, aerospace, and critical infrastructure.

Key findings. - AEDS are defined as the integration of knowledge graphs, ontologies, LLMs, simulation engines, optimization algorithms, and digital twins into cohesive closed-loop architectures where autonomous agents perceive environmental states, reason, and execute operational changes with minimal human intervention, distinguished from data-driven enterprises that depend on human interpretation of dashboards. - A three-phase maturity model frames the transition: Phase 1 Data-Driven (human-centric, data-to-insight lag of days to weeks); Phase 2 AI-Augmented (ML/predictive/RPA, reactive, lag compressed to hours); Phase 3 Continuously Optimized AI-Assisted (closed-loop agentic systems, real-time continuous optimization, no human intervention until performance degrades or objectives change). - Knowledge graph plus causal inference achieved root-cause identification accuracy above 90% in quality management versus conventional statistical methods [11]. - Knowledge graph-based risk management in financial services achieved 35% improvement in decision-making efficiency via hidden risk propagation path identification and early warning [12]. - RAG-enhanced LLMs achieved 54% accuracy in question-answering against enterprise SQL databases, a sharp improvement from 16% accuracy using zero-shot LLM queries on raw databases [16]. - LLM-integrated supply chain frameworks reduced time-to-decision from weeks to minutes by enabling natural language formulation of optimization problems [17]; OptLLM translates natural language queries into mathematical formulations and invokes solvers [19]; solver-informed RL uses optimization

solvers as reward signals to reach state-of-the-art executable model generation [20]. - Digital twins drove a 2.1-day reduction in average delivery time, 12 percentage point improvement in on-time delivery rates, and nearly 20% cost reduction while enhancing robustness to disruptions [23]; multi-objective Bayesian optimization enables inverse inference where targets are specified and policies are derived [10]. - Hybrid traffic optimization combining transformer-based deep learning with rule-based reasoning and RL achieved 15% better accuracy, 30% lower computational overhead, and improved resilience to anomalies versus conventional deep RL [27]. - GPT-4, despite being more advanced than GPT-3.5, awarded higher scores to female candidates with comparable qualifications and lower scores to Black male candidates, with biases translating to 1-3 percentage point differences in hiring probability [41]; algorithmic fairness constraints often prove mutually exclusive (improving one dimension degrades another). - The Agentic AI Governance Maturity Model (AAGMM) provides a five-level framework across 12 governance domains; Level 4-5 organizations achieved 94.3% lower sprawl indices, 96.4% fewer risk incidents, and 32.6% higher effective task completion rates versus Level 1 [48]. - Adoption barriers are more often organizational than technical: a healthcare study found governance-related barriers more salient than technological or people-related barriers [46]; a hierarchical analysis found infrastructure gaps, real-time data deficits, and skill shortages are root-level drivers while organizational resistance is a dependent outcome [49]. - Organizations report that technical implementation of agentic AI consumes 20% of effort while organizational change management, stakeholder alignment, and governance establishment consume 80%; the 'AI Data Eclipse' and 'pilot purgatory' phenomena block scaling from proof-of-concept to production. - In simulated military network environments, autonomous cyber defense (ACD) systems using multi-agent RL, NLP, and rule-based reasoning outperformed conventional intrusion detection in detecting stealthy intrusions and lateral assaults while maintaining explainability via integrated XAI layers [56].

Named frameworks/systems/standards. Autonomous Enterprise Decision Systems (AEDS) conceptual framework, Three-phase enterprise maturity model (Data-Driven / AI-Augmented / Continuously Optimized AI-Assisted), Retrieval-Augmented Generation (RAG), OptLLM (natural language to optimization formulation), Solver-informed Reinforcement Learning (solver as reward signal), Multi-objective Bayesian optimization (inverse inference in digital twins), Hierarchical / deep reinforcement learning (DRL) architectures, Multi-agent frameworks (orchestrating, learning, governance agents), Model Context Protocol (agentic digital twins for urban logistics) [31], Edge-cloud orchestration architectures (millisecond inference latency), SHAP (SHapley Additive exPlanations) and counterfactual explanations, Federated learning and differential privacy, Agentic AI Governance Maturity Model (AAGMM) — five levels, 12 domains [48], Unified Control Framework for enterprise AI governance [62], FHIR (Fast Healthcare Interoperability Resources) standard, ISO/IEC 42001 (AI governance), ISO/IEC 25012 (data quality), ISO/IEC 21823-3 (IoT semantic interoperability), NIST AI Risk Management Framework, EU AI Act, GDPR; US Executive Order on AI, Autonomous Cyber Defense (ACD) systems, API-first integration architectures, PID control principles adapted for AI closed-loop stability, Demand sensing with LSTM, temporal convolutional networks, transformer-based predictors

Top sources. 1. S. Wasserkrug et al.. Enhancing decision making through the integration of large language models and operations research optimization. 2025. doi: 10.1609/aaai.v39i27.35090 2. J. Sequeda, D. Allemang, B. Jacob. A benchmark to understand the role of knowledge graphs on large language model's accuracy for question answering on enterprise SQL databases. 2023. doi: 10.1145/3661304.3661901 3. V. Acharya. Governing the agentic enterprise: A governance maturity model for managing AI agent sprawl in business operations. 2026. doi: https://www.semanticscholar.org/paper/cec9745d5c3c0418ba1fd2d368a9e40dbf305f8a 4. H. L. Boddupally. Self-improving enterprise platforms using learning loops and AI-driven orchestration. 2025. doi: 10.69888/ftsin.2025.000538 5. J. An, D. Huang, C. Lin, M. Tai. Measuring gender and racial biases in large language models: Intersectional evidence from automated resume evaluation. 2025. doi: 10.1093/pnasnexus/pgaf089

Contribution to AEDA. This review is the synthesis capstone for AEDA: it names and validates the exact stack AEDA proposes (knowledge graphs + ontologies + LLMs/GraphRAG + simulation/digital twins + optimization + agent swarms + closed-loop decision) as a coherent, empirically supported architecture rather than an aspiration. It supplies AEDA's organizing narrative, the maturity-model spine that lets MITRE position BEA/DoDAF DM2 documentation as Phase 1 and AEDA's computational decision stack as the Phase 2-to-3 evolution. It provides quantified evidence for each AEDA layer (KG causal root-cause >90%, RAG SQL QA 54% vs 16%, digital-twin logistics gains, hybrid RL 15%/30%, AAGMM governance outcomes), and it foregrounds the multi-agent feedback-loop architecture (orchestrating, learning, governance agents) that maps directly onto AEDA's Agent Swarms plus Decision/Executive Support layers. Critically for a defense/aerospace FFRDC audience, it grounds the brain in critical-infrastructure, military-network, and aerospace governance findings, including human-command subordination requirements.

Design implications. - Build the AEDA brain as an explicit closed-loop, not a static model store: capture every agent decision, environmental response, and outcome and feed them back to refine decision policies in real time [6][30]. Treat models as continuously refined artifacts, not deploy-once assets. - Architect three cooperating agent classes onto the Agent Swarm layer: orchestrating agents (perceive state, coordinate forecasting/risk/optimization specialists), learning agents (extract causal insights from each decision episode), and governance agents (monitor drift, constraint violations, ethical boundaries and trigger human oversight) [31][32]. - Ground the KG/Ontology layers in causal inference, not just correlation: causal-reasoning-over-KG yielded root-cause accuracy above 90% [11], and causal/counterfactual capability is named a top open research gap. Model domain context as a first-class reasoning component (domain-contextualized concept graphs) [14]. - Pair every LLM/GraphRAG surface with retrieval grounding and a formal solver: RAG lifted enterprise SQL QA from 16% to 54% [16], and solver-informed RL plus OptLLM-style natural-language-to-math translation let non-experts formulate and solve optimization problems with explained trade-offs [16][17][19][20]. Use the solver as both executor and reward signal. - Stand up an enterprise digital twin for inverse inference (specify targets, derive policies) via multi-objective Bayesian optimization, and embed DRL agents for predictive maintenance, process optimization, and quality control

[10][23][24]. - Engineer closed-loop stability explicitly: adopt control-theory mechanisms (PID-style, stability-aware optimization) to prevent oscillation/divergence, and continuous validation to detect model drift before decision quality degrades silently [28][33][43]. - Invest in semantic standardization first: canonical data models, ontology mappings, metadata/data catalogs documenting definitions, lineage, and quality, plus API-first wrappers over legacy systems. The standards landscape (ISO/IEC 42001, 25012, 21823-3; NIST AI RMF) is siloed, so the brain must integrate them into one governance surface [34][37][52]. - Bake in governance and explainability as deployment gates: implement a maturity model (AAGMM-style, five levels across 12 domains) with auditability, bias auditing, counterfactual explanations interpretable to domain experts, and human-in-the-loop triggers near decision boundaries [40][41][48]. - For defense/aerospace/critical-infrastructure deployments, design for human command-and-control subordination, adversarial red-teaming/penetration testing, explainable autonomous cyber-defense layers, and recovery procedures that restore human control if autonomy fails [54][55][56][57][59]. - Plan for the 80/20 effort split: budget the majority of program effort for organizational change management, stakeholder alignment, and governance establishment, and sequence adoption by fixing root-level infrastructure, real-time data, and skills gaps before attempting to overcome organizational resistance [46][49][53].

Risks/limitations. - LLM-specific risks remain inadequately mitigated: probabilistic hallucination, stochastic behavior in edge cases, inconsistent reasoning in high-dimensional spaces, opacity, and difficulty incorporating hard constraints (regulatory, safety) for mission-critical use [21][22]. - Measured fairness harms: state-of-the-art LLMs exhibit gender and racial bias in high-stakes decisions (GPT-4 resume bias, 1-3 percentage point hiring-probability differences) [41]; fairness constraints are often mutually exclusive, and feedback loops can amplify historical human bias. - Closed-loop instability: naive feedback mechanisms can oscillate or diverge; model drift can degrade decision quality undetected if monitoring is inadequate [33][43]. - Adversarial fragility: small imperceptible input perturbations can flip outputs; enterprise attack surfaces include manipulated supply-chain forecasts, adversarial resumes, and fraudulent credit inputs; defenses remain underdeveloped [44][66 context]. - Reinforcement learning deployment constraints: exploration-exploitation trade-offs are risky in high-stakes domains, sample efficiency is poor (millions of interactions to converge), and the reward-specification problem for complex organizational objectives is unresolved [26][31]. - Data foundation failures: 'AI Data Eclipse' (incomplete, biased, inaccessible data) and 'pilot purgatory' block scaling; legacy systems were built for documentation/billing, not ML training, with inconsistent temporal resolution and weak quality metrics [35][50]. - Accountability and agency gaps: unresolved questions of who is responsible when an autonomous system causes harm (developer, deploying organization, or approving regulator), creating governance crises in critical infrastructure and defense [46][71 context]. - Governance frameworks lag capability: EU AI Act, US EO, and ISO/NIST standards are often inadequate for highly autonomous systems making rapid consequential decisions and remain siloed by technical domain [38][52][62]. - Sim-to-real and adversarial-military gap: agents trained in simulation underperform in real operations, and adversaries actively manipulate agent behavior; maintaining human command while preserving autonomy creates unresolved architectural tensions [59]. - Workforce and macro risk: displacement of routine

decision roles, acute skill shortages (data/ML engineering, AI ethics/governance), and arms-race policy pressure toward rapid deployment without adequate safeguards [52][61].

LR11: Ontology Discovery Research

AEDA layer: Ontology Discovery + Ontology Governance

Scope. Comprehensive literature review (generated June 07, 2026; 27 references) of automated ontology discovery at the intersection of AI, knowledge engineering, and semantic web. Covers the methodological evolution 2002-2025 (linguistic/NLP, statistical/ML, deep learning, LLM-based), the core ontology learning tasks (term extraction and typing, taxonomy discovery, non-taxonomic relation extraction, entity linking and disambiguation), application domains (biomedical, Earth science/geospatial, cultural heritage, cybersecurity/maritime/space), advanced directions (knowledge graph construction, multi-modal discovery, ontology alignment, dynamic evolution), and challenges. Directly maps to the AEDA Ontology Discovery layer (extracting candidate concepts/relations from data sources) and feeds the Ontology Governance and Enterprise Ontology layers.

Key findings. - Methodological evolution is staged: linguistic/NLP-dominated early work, then statistical/ML (LDA topic models, association rule mining, hierarchical clustering), then a deep learning revolution circa 2015-2018 (RNN/LSTM/CNN, then BERT and transformer variants), then an LLM paradigm shift since 2023 (GPT-3, GPT-4, fine-tuned variants enabling zero-shot/few-shot discovery). - Modern LLMs combined with retrieval-augmented generation (RAG) and fine-tuning achieve F1-scores exceeding 90% on multiple ontology learning benchmarks. - Term extraction/typing: BERT-based models achieve 86-90% accuracy on diverse domain corpora; LLM-based approaches achieve 88-94% accuracy via pre-trained linguistic knowledge plus fine-tuning or few-shot demonstrations. - Domain-specific LLM fine-tuning achieves F1-scores exceeding 96% on medical ontology mapping tasks. - Joint entity-relation extraction (multi-task learning with shared representations) yields 15-20% performance improvements over pipeline methods and achieves 85-93% precision/F1 on domain-specific corpora, mitigating error propagation that plagues two-phase pipelines. - Overall the field reports modern systems achieving 88-96% accuracy across multiple benchmarks. - Joint/end-to-end optimization across ontology discovery tasks outperforms two-phase pipeline approaches (e.g., graph-based optimal branching on weighted hypernym graphs learns concepts and relations from scratch). - Advanced frontier directions: LLM-driven knowledge graph construction with prompt engineering, ontology alignment verification, and multi-agent consensus validation; physics-regularized knowledge graphs incorporating domain constraints and causal reasoning for trustworthy, interpretable representations. - Space is named as an emerging frontier, with ontology discovery efforts targeting NASA, ESA, and international space agency resources, addressing cislunar operations, satellite systems, and mission planning.

Named frameworks/systems/standards. BERT (and variants) for NER, relation extraction, semantic similarity, GPT-3 / GPT-4 and domain-specific fine-tuned LLM variants,

RNN / LSTM / CNN deep learning architectures, BiLSTM-CRF sequence labeling with BIO (Begin-Inside-Outside) tagging, Latent Dirichlet Allocation (LDA) topic modeling, Retrieval-Augmented Generation (RAG); fuzzy RAG variant for industrial process KG construction, Chain-of-thought reasoning, in-context examples, multi-stage / multi-stage prompting, TF-IDF, C-value, NTF term-weighting statistical measures, Semantic Role Labeling (SRL) and frame semantics for n-ary relation extraction, Personalized PageRank (PPR) plus semantic similarity for entity linking, Graph Neural Networks (GNNs) and reinforcement learning for taxonomy/relation discovery, Semantic web standards: RDF, OWL, SPARQL, Gene Ontology (GO), CIDOC Conceptual Reference Model (CRM), LLMs4OL Challenge datasets / benchmark (term typing, taxonomy discovery, non-taxonomic relation extraction), Physics-regularized knowledge graphs (e.g., PhyGeo-KG); Remote Sensing Indices Knowledge Graph (RSIKG), Multi-agent consensus validation and agent-based ontology curation, Neuro-symbolic integration (ontology-aware neural architectures)

Top sources. 1. H. B. Giglou, J. D'Souza, S. Auer. LLMs4OL: Large language models for ontology learning. 2023. doi: 10.48550/arXiv.2307.16648 2. M. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, H. M. Abbasi. A survey of ontology learning techniques and applications. 2018. doi: 10.1093/database/bay101 3. T. Zengeya, J. V. Fonou-Dombeu. A review of state of the art deep learning models for ontology construction. 2024. doi: 10.1109/ACCESS.2024.3406426 4. A. Mavridis, S. Tegos, C. Anastasiou, M. Papoutsoglou, G. Meditskos. Large language models for intelligent RDF knowledge graph construction: Results from medical ontology mapping. 2025. doi: 10.3389/frai.2025.1546179 5. U. Das, K. Atmakuri, D. H. Ho, C. Lee, Y. Lee. Clinical knowledge graph construction and evaluation with multi-LLMs via retrieval-augmented generation. 2026. doi: 10.48550/arXiv.2601.01844

Contribution to AEDA. Provides the evidentiary basis for AEDA's Ontology Discovery layer, the step that converts the Data Sources layer into candidate concepts, types, taxonomies, and non-taxonomic relations that the Enterprise Ontology and Knowledge Graph layers consume. It establishes that automated, LLM-plus-RAG ontology discovery is now performant enough (88-96% accuracy across benchmarks, 96%+ F1 on domain-specific mapping) to replace manual, expert-driven enterprise architecture documentation with a computational pipeline. It validates the AEDA progression from Discovery to Governance: discovery tasks (term extraction/typing, taxonomy induction, relation extraction, entity linking) produce candidates, and governance functions (ontology alignment, fusion, multi-agent consensus validation, dynamic evolution) curate and reconcile them. It also names space and aerospace/cislunar as live application frontiers, anchoring AEDA's defense/space enterprise relevance.

Design implications. - Build the Discovery layer as a multi-task / joint extraction pipeline, not a sequential NER -> relation-extraction -> entity-linking pipeline: the literature reports 15-20% gains and avoidance of error propagation from end-to-end joint optimization. - Pair LLMs with RAG against the existing enterprise ontology so discovery is ontology-aware and respects domain constraints (DoDAF DM2, BEA structures) rather than hallucinating free-form concepts; RAG-grounded extraction is what drives the 90%+ and 96%+ figures. - Use BERT-class models for high-throughput term typing/NER (86-90%) and reserve LLM

few-shot/fine-tuned calls for harder, low-frequency, or cross-domain terminology (88-94%) to manage cost; this directly informs the cost-aware tiering of the discovery stack. - Treat Ontology Governance as a first-class layer implementing ontology alignment, fusion with conflict resolution, and multi-agent consensus validation, since independently discovered DoD enterprise ontologies will be heterogeneous and must be reconciled before entering the Enterprise Ontology. - Adopt temporal / dynamic ontology evolution (temporal knowledge graphs) so the enterprise ontology tracks how doctrine, systems, and policy change over time, supporting the downstream Simulation and Decision layers. - Move toward neuro-symbolic, physics/constraint-regularized construction (domain constraints, causal reasoning) to make the discovered ontology interpretable and trustworthy enough to drive an executive Decision layer. - Plan for human-in-the-loop governance checkpoints: discovery accuracy of 88-96% means roughly 4-12% of candidates are wrong, so governance must include verification and explainability before promotion to the authoritative ontology.

Risks/limitations. - Semantic ambiguity and polysemy remain a fundamental, unsolved challenge; word sense disambiguation only partially helps, and domain-specific polysemy still causes errors, a direct risk for defense terminology with overloaded acronyms. - Data scarcity and domain specificity constrain supervised approaches; specialized DoD/space domains have limited training data and often require fine-tuning or adaptation, and one cited source questions whether LLMs really adapt to domains. - Evaluation metrics and benchmarking for ontology quality are underdeveloped: precision/recall/F1 do not capture ontology coherence, utility, or downstream impact, complicating any claim that a discovered enterprise ontology is fit for decision use. - Error propagation persists in multi-stage pipelines; even joint learning faces difficulty balancing optimization across interconnected tasks. - Computational efficiency and scalability: LLM inference cost limits practical deployment at web/enterprise scale, motivating distillation and edge strategies before stack-wide rollout. - Cross-lingual / multilingual coverage is weak (research is English-centric), a limitation for international/coalition space and defense ontologies. - Reported headline figures (90%+, 96%+ F1, 85-93%, 15-20% gains, 88-96% accuracy) are drawn from heterogeneous domain-specific benchmarks and are not standardized across methods, so they should not be read as guaranteed enterprise-scale performance. - Reference [2], cited for the space-domain and several foundational claims, is an anonymized 'Text block 1, Unknown Year, Available: None' with no verifiable provenance, weakening the citation chain for the space-frontier assertion.

Appendix B. Known Refinement Items (for the Codex pass)

Adversarial review found fidelity clean (no fabricated metrics, systems, or sources) across the core package and the v4 additions. Open consistency items for Codex:

Core package assessment. The drafted sections are largely faithful to the source syntheses. Spot-checking every load-bearing quantitative metric and named citation against LR01-LR11 found no fabricated numbers, systems, or sources: the headline figures (ontology grounding 37%-to-98% accuracy and 63%-to-1.7% hallucination from Ali/Taha/Morsey; rho=0.88 maturity

correlation; GraphRAG latency/accuracy tiers and 90.71% token reduction; 10x/32.4%/92.4% neural-symbolic dual-indexing; 15.95x xDEVS speedup; 48%/35% governance-board results; portfolio 16.8%/1.34 Sharpe/8.2%; 35% KG efficiency and >90% root-cause) all trace correctly to their cited syntheses. No em dashes were found in any drafted section (the prose consistently uses commas, semicolons, and parenthetical clauses, e.g. the decision section's "GraphRAG retrieval, agent swarms, the enterprise digital twin, simulation, and optimization, resolves into"). The one substantive problem is a structural naming inconsistency: the architecture is branded "ten-layer" throughout, but the canonical layer list enumerated in the header, the refarch section, the roadmap, and the codex section contains 11-12 named layers (Data Sources, Ontology Discovery, Ontology Governance, Enterprise Ontology, Knowledge Graph, GraphRAG, Agent Swarms, Enterprise Digital Twin, Simulation, Optimization, Decision Layer, Executive Support). This count mismatch should be reconciled before the package is treated as canonical. Coverage is otherwise complete across all layers, with each layer specified by purpose, inputs, outputs, technologies, and grounding evidence. Verdict: substantively clean on fabrication and em dashes; one numbering inconsistency and a few minor traceability/ordering items to resolve. - INCONSISTENCY (refarch, codex, roadmap, origin, definitions, space): The package is named the 'ten-layer reference architecture' but the canonical layer enumeration lists 11-12 named elements (Data Sources, Ontology Discovery, Ontology Governance, Enterprise Ontology, Knowledge Graph, GraphRAG, Agent Swarms, Enterprise Digital Twin, Simulation, Optimization, Decision Layer, Executive Support). Even collapsing Decision Layer + Executive Support yields 11, and counting Data Sources yields 12. Either rename to match the true count or explicitly state which elements are 'layers' versus substrate/tier so 'ten' is defensible. - INCONSISTENCY (codex): The Implementation Priorities list and the 'ten-layer stack, in order' restatement in the codex section omit the 'Data Sources' layer at the front in the priorities list (priorities start at Ontology design) while the inline stack restatement includes Data Sources, and also appends both 'Decision Layer' and 'Executive Support' as separate trailing items. This compounds the ten-versus-twelve ambiguity within a single section that declares itself the 'authoritative source of record.' Reconcile the canonical ordering in one place and reference it. - INCONSISTENCY (origin): The origin section narrates a 'seven-phase' discovery arc, but the package elsewhere describes 11 AEDA syntheses (LR01-LR11) and a ten/twelve-layer stack. The seven-phase framing maps reviews to phases in a way that does not obviously align with the 11 reviews or the layer count; verify the phase-to-review mapping is internally consistent and that no review is silently dropped (the section is truncated at phase seven, so confirm phases for Optimization/Decision/Executive Support and the discovery/governance reviews are accounted for). - TRACEABILITY (twin): The Enterprise Digital Twin section states EDTs capture 'the decision-makers themselves' and lists representable entities (Organizations, Programs, Capabilities, Budgets, Schedules, Resources, Risks, Dependencies). LR05 supports modeling organizational structures, processes, and decision-makers, but the specific enumerated DoD entity set (Budgets, Schedules as first-class) is an AEDA design mapping, not a cited LR05 claim. Acceptable as design, but confirm it is framed as architecture rather than implying it is an evidenced result. - TRACEABILITY (dm2_owl): The DM2-to-OWL named object properties (performs, supports, realizes, consumes, produces, exchanges) and the specific BFO alignment (Performer/Resource as independent continuants, Activity as occurrent/process, Capability as

realizable disposition, InformationExchange as information artifact) are design constructs not drawn from any synthesis. This is appropriate for a mapping-strategy section, but it should be explicitly labeled as proposed AEDA design so it is not mistaken for evidence-grounded DM2 canon; the syntheses do not validate this particular property set. - MINOR (refarch Layer 2): Lists 'risk-tiered governance with named accountability [LR09]' as a candidate technology for the Ontology Governance layer. LR09's risk-tiered governance is framed for decision flows (eligibility, law-enforcement, biometric, defense targeting), not ontology-candidate promotion. The borrowing is reasonable but slightly stretches LR09's scope; confirm the cross-application is intended. - MINOR (definitions): The definitions section notes representative measured results 'where they substantiate the concept, not as guarantees of any specific deployment' which is good hygiene, but the Enterprise Ontology definition states ontology grounding 'is the trust-and-accuracy enforcement mechanism for every layer above it' as a flat assertion. This is AEDA's thesis claim extrapolated from a single clinical-QA result (LR02); ensure it is presented as architectural rationale rather than empirically generalized fact.

v4 additions assessment. CLEAN on fabrication and em dashes; minor consistency/framing items only. I verified all source files LR05-LR10. Every quantitative metric and named-source attribution in both sections traces to the cited review: >90% root-cause accuracy and 35% decision-efficiency gain (LR10 line 15); 16%-to-54% graph-grounded QA (LR10 line 19); <5% DEVS deviation and 15.95x speedup (LR06 lines 9, 8); 40-94% UQ cost reduction and CVaR (LR07 lines 74, 32); 51.5% cost-effectiveness ratio and 95.2% resource utilization (LR08 line 56); 48% fewer bias/regulatory-violation instances with the advisory-vs-integrated framing (LR09 line 23); compliance-by-design routing explainability into audit-ready evidence (LR09 line 31); hard-constraint incorporation difficulty (LR10 line 21); IADTs moving beyond passive monitoring (LR05 line 71); SysML-as-authoritative-source while a DES tool computes behavior (LR06 line 20); precedence-constrained budget-bounded portfolio selection (LR08 line 60); RL sample-efficiency and reward-specification cautions (LR10 line 31). No metric, system, or citation was invented. No em dashes or en dashes appear in either section; hyphenated compounds and word-form ranges ("40 to 94 percent") are correct. The closed-loop control framing is coherent with the ten-layer model and is internally consistent with the world-model section: both treat the Enterprise Digital Twin as the state holder, the simulation/Monte Carlo/optimization layers as the forward-evaluation path, and RL/learning agents plus governance agents as the loop-closing mechanism, which is faithful to LR10's Phase-3 closed-loop description (lines 37, 40-43) and LR06/LR07/LR08 layer roles. The implementation-maturity notes are honest and appropriately scoped. The world-model section is implementable as described because it binds the six state-space primitives to the existing ontology/Knowledge Graph layers rather than introducing an unsourced new substrate. Remaining items are framing-level, not fabrication. - TRUNCATION (not a content defect, but must be fixed before publication): the controlloop section ends mid-sentence and mid-word at 'autonomously influence physical coun'. The Actuation/Observation/Learning-Loop subsection that closes the control loop is incomplete, so the section as submitted does not actually demonstrate loop closure in prose, the very claim the section is built on. Complete the actuation-observation-update paragraph, grounding it in LR05 IADTs (line 71) and LR10 closed-loop learning (lines 40-43). - CONSISTENCY

(ten-layer model): controlloop assigns control-theoretic roles to layers but the AEDA stack named in the task header lists the upper layers as Agent Swarm -> Enterprise Digital Twin -> Simulation -> Optimization -> Decision, whereas controlloop introduces a 'Policy Engine' stage and 'Monte Carlo Engine' and 'Optimization Engine' as named components. These are defensible refinements, but the section should state explicitly how Policy Engine maps to the canonical Decision layer boundary (it sits between Optimization and Decision), otherwise a reader reconciling the prose against the ten-layer diagram will see an apparent eleven-stage stack. Add one sentence binding Policy Engine + Decision Layer to the single canonical 'Decision' layer. - ATTRIBUTION PRECISION (low risk): controlloop states 'parallel discrete-event execution has achieved speedups of 15.95 times.' LR06 (line 8) attributes 15.95x specifically to the xDEVS framework on distributed/cloud systems, not to parallel DES generically. The number is correct; tighten the attribution to xDEVS to avoid overgeneralizing one framework's result to all PDES. - FRAMING (verify against v4 intent, not a fabrication): controlloop calls CVaR a risk measure the Monte Carlo Engine computes. LR07 supports CVaR as a risk-averse optimization objective/constraint (lines 32, 23) but does not present it as a standard Monte Carlo Engine output per se. The claim is within the literature's scope; consider softening to 'computing risk measures such as Conditional Value-at-Risk' as already worded, which is acceptable, but ensure the Optimization Engine (not the Monte Carlo Engine) is where CVaR enters as an objective, to stay faithful to LR07's placement. - WORLD-MODEL / SOURCE FIDELITY (minor): worldmodels states 'LR10 identifies the semantic-integration barrier' for automated state extraction from heterogeneous sources. LR10 Section IV (lines 44-49) does treat semantic integration and data-quality at integration boundaries as a primary barrier, so this is supported; however LR10 frames it as data/semantic-integration and infrastructure barriers broadly (line 75 root-level drivers), not a single named 'semantic-integration barrier.' Reword to 'the semantic-integration and data-quality barriers LR10 identifies' to match the source's phrasing and avoid implying a single coined term. - COHERENCE CHECK (passes, noted for the record): the two sections are mutually consistent. worldmodels recasting the Enterprise Digital Twin as the world model that the simulation/optimization/RL layers read and write is the same state-holding role controlloop assigns it. No contradiction between the sections on which layer holds enterprise state or where the loop closes.