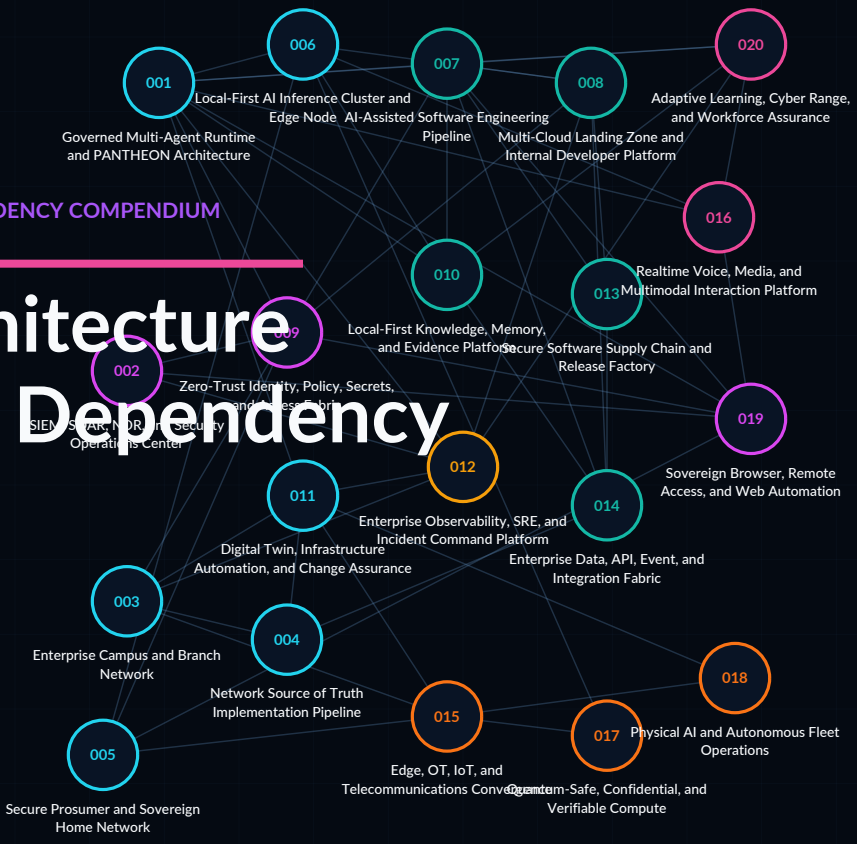
REFERENCE ARCHITECTURE LANDSCAPE / DEPENDENCY COMPENDIUM

Reference Architecture Landscape and Dependency Compendium



Cross-architecture relationships, shared trust boundaries, dependency contracts, integration failure modes, deployment interactions, observability surfaces, and portfolio-level verification.

PERMANENT DOCUMENT ID	ARCHITECTURES	RELATIONSHIPS
MYTHOS-RA-COMP-0001	20	50

Landscape summary

ARTIFICIAL INTELLIGENCE

2 RA-001, RA-006

CYBERSECURITY AND NETWORK SECURITY

3 RA-002, RA-009, RA-019

NETWORK ENGINEERING

4 RA-003, RA-004, RA-005, RA-011

PLATFORM ENGINEERING

3 RA-007, RA-008, RA-013

DATA, KNOWLEDGE, AND EVIDENCE

2 RA-010, RA-014

RELIABILITY AND OPERATIONS

1 RA-012

EDGE, OT, AND FUTURE COMPUTE

3 RA-015, RA-017, RA-018

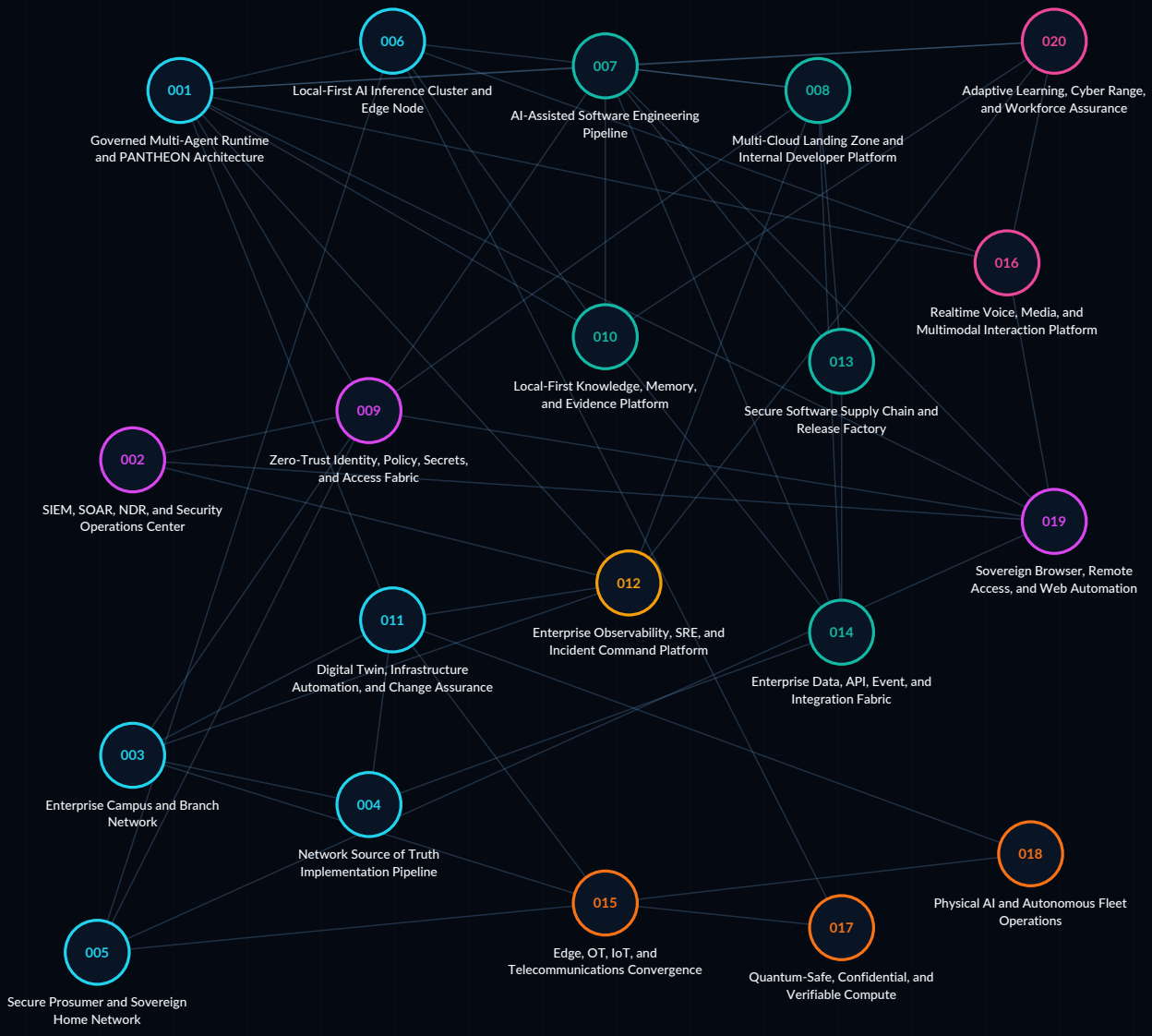
HUMAN SYSTEMS AND LEARNING

2 RA-016, RA-020

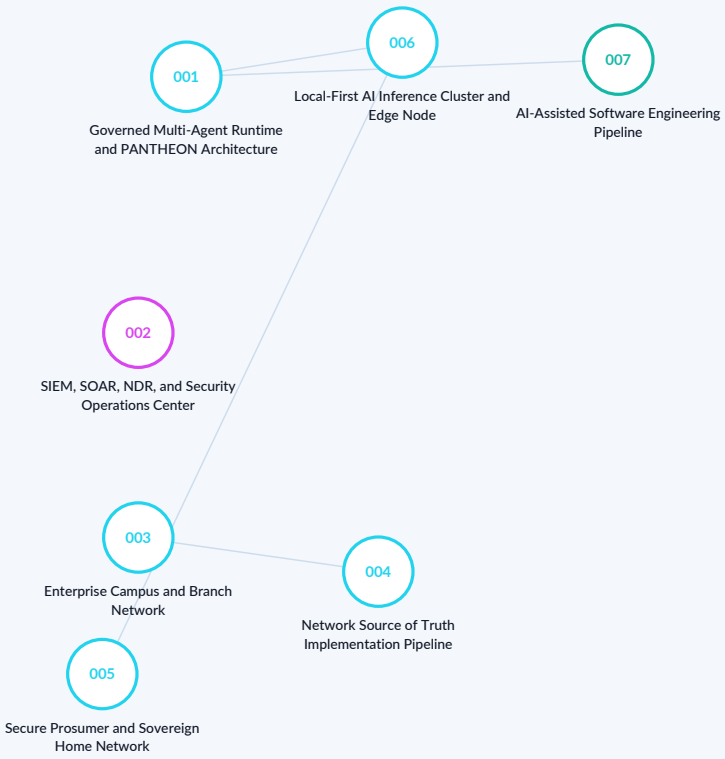
LANDSCAPE POSITION

The architecture system is intentionally compositional. Individual architectures remain independently citable, while shared dependencies are explicit and must not create hidden authority, circular trust, or unbounded failure propagation.

Complete architecture map

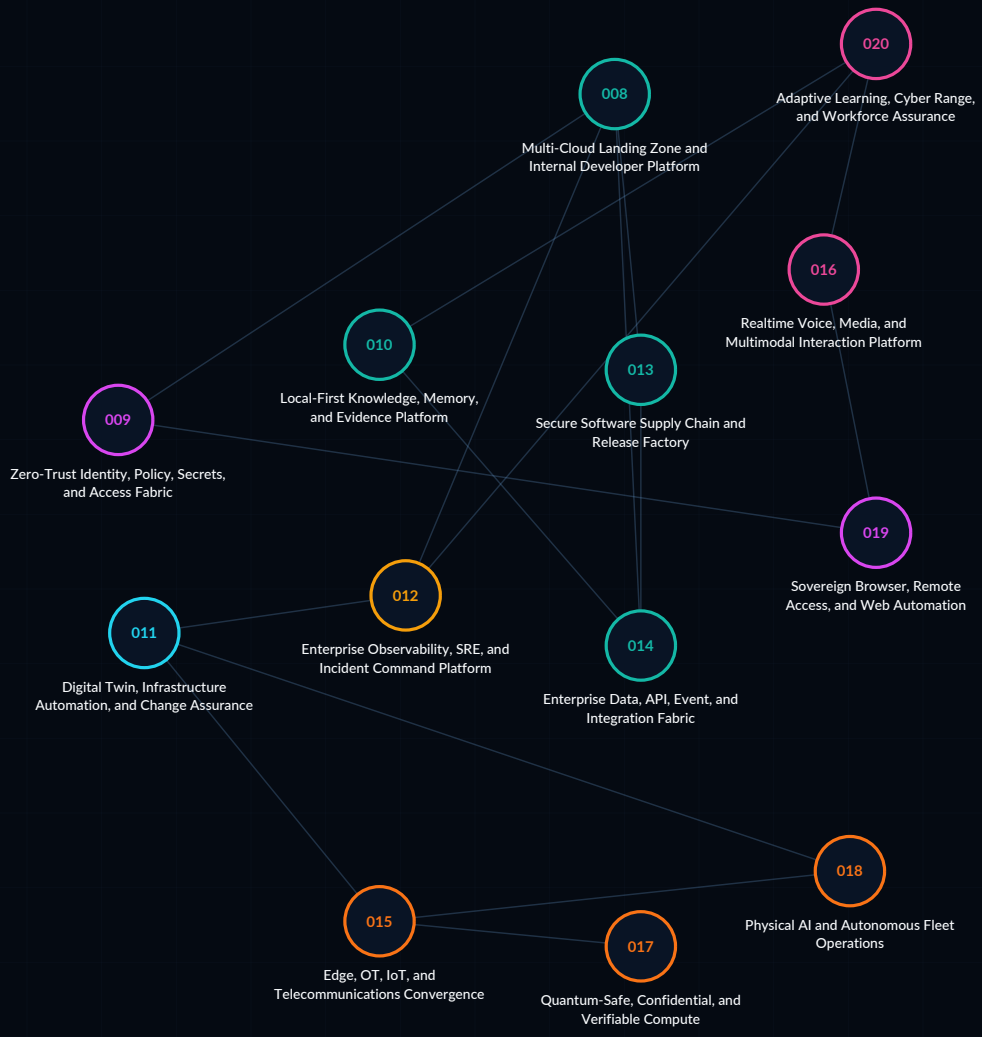


Canonical architecture set

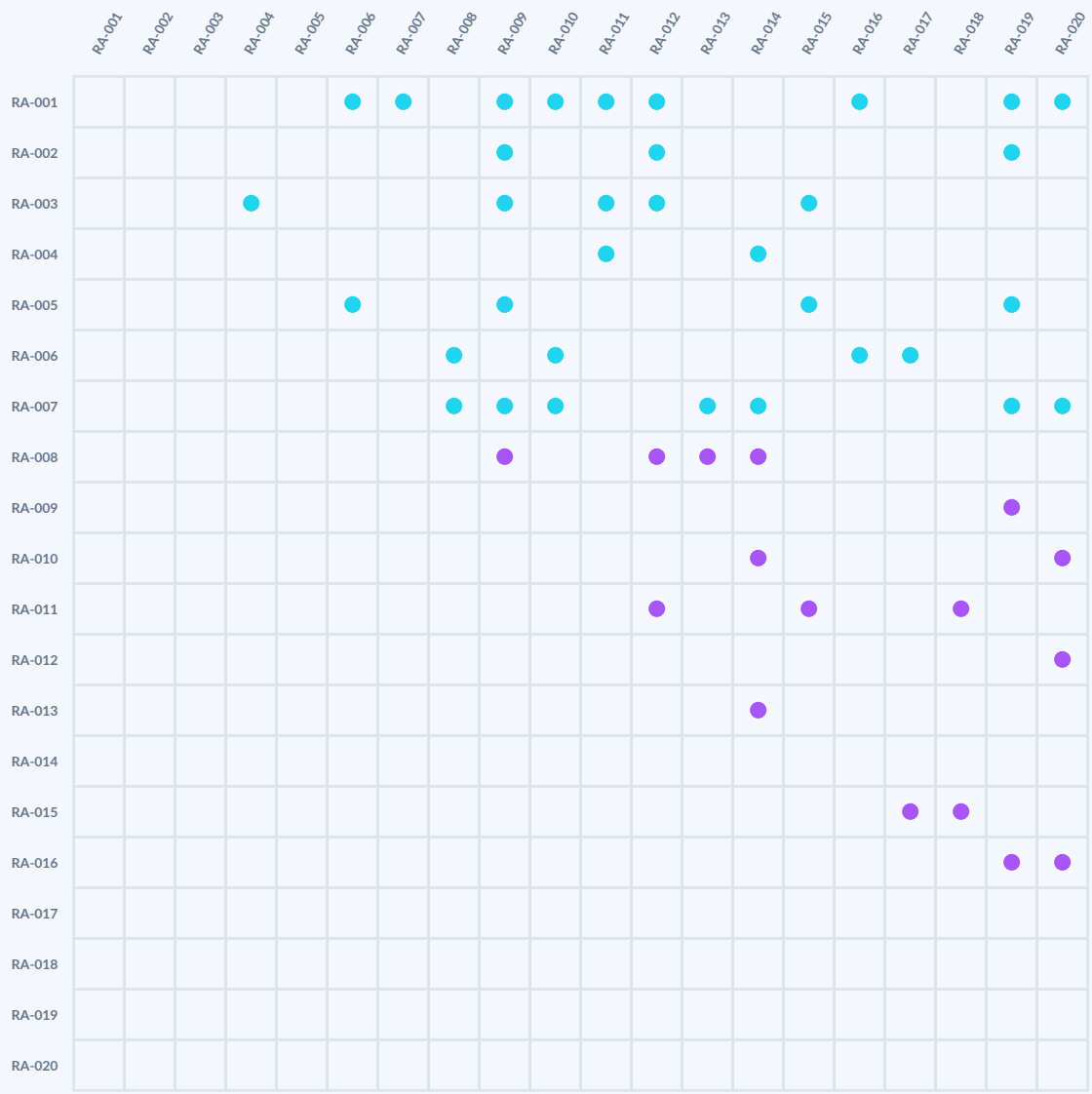


The canonical set establishes the base operating patterns for governed agent runtime, security operations, enterprise networking, authoritative network state, sovereign edge networking, local AI inference, and AI-assisted software engineering.

Approved future-domain architecture set



Directional dependency matrix



A marked cell indicates an explicit integration/dependency relationship from row architecture to column architecture. Relationships are directional and require interface, authority, data, failure, recovery, and evidence contracts.

Shared trust-boundary doctrine

IDENTITY	Authenticate and bind human, service, workload, device, agent, and external identities.
AUTHORITY	Authorize the exact action, scope, duration, resource, and consequence class.
DATA	Classify data, purpose, provenance, retention, residency, privacy, and egress.
PROTOCOL	Pin interface, schema, version, encryption, validation, retry, timeout, and compatibility.
FAILURE	Define loss, degradation, backpressure, isolation, containment, rollback, and recovery.
EVIDENCE	Record decision, versions, state changes, output, validation, timing, owner, and disposition.

Cross-architecture failure propagation

IDENTITY / POLICY

RA-009 impairment can block or weaken downstream execution, remote access, software delivery, and browser automation.

OBSERVABILITY

RA-012 impairment can hide degradation across RA-003, RA-008, RA-011, RA-015, RA-018, and RA-020.

DATA / INTEGRATION

RA-014 impairment can stall knowledge, supply chain, automation, telemetry, and cross-domain workflows.

EDGE / CONNECTIVITY

RA-015 impairment can isolate physical systems, autonomous fleets, local compute, or field telemetry.

SOFTWARE SUPPLY CHAIN

RA-013 compromise can propagate trusted-looking artifacts into RA-007, RA-008, RA-019, and dependent runtime environments.

AGENT / MODEL RUNTIME

RA-001 or RA-006 faults can corrupt decisions or automation unless authority, deterministic execution, verification, and evidence remain independently enforceable.

Portfolio observability and verification

OUTCOME

Mission and user outcomes

SERVICE

SLOs, capacity, performance

CONTROL

Identity, policy, approvals

STATE

Desired, runtime, observed

SECURITY

Threats, trust, abuse

DEPENDENCY

External service and supply chain

RECOVERY

Restore, failover, rollback

EVIDENCE

Trace, artifacts, receipts, decisions

Verification is architecture-specific, but acceptance always requires defined scope, testable criteria, relevant negative/degraded/recovery cases, current evidence, accountable ownership, and explicit residual risk.

Lifecycle, revision, and limitations

PERMANENT IDENTITY	Architecture IDs remain stable across ordinary revision.
CANDIDATE STATUS	Architecture publications are not certification, legal compliance, or proof of implementation.
VERSIONED CHANGE	Material architecture changes require rationale, affected relationship analysis, source refresh, and regression verification.
FRESHNESS	External authorities and implementation assumptions must be revalidated at adoption and scheduled review.
SCOPE LIMITS	Reference architectures are patterns. Product, sector, jurisdiction, workload, and environment constraints still require engineering decisions.