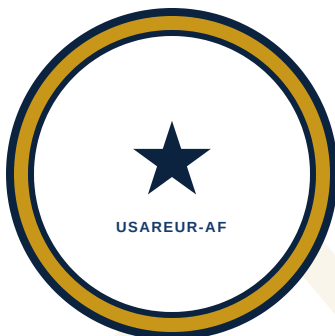


DRAFT — UNOFFICIAL — NOT FOR OPERATIONAL USE

ARCHITECTURE REFERENCE

ODT-EA



Enterprise Architecture — 03: Artifacts and Views

Architecture Reference

HEADQUARTERS
UNITED STATES ARMY EUROPE AND AFRICA
(USAREUR-AF)
Wiesbaden, Germany

DRAFT — NOT FOR OFFICIAL USE. FOR TRAINING PLANNING PURPOSES ONLY.

20 MARCH 2026

DRAFT — UNOFFICIAL — NOT FOR OPERATIONAL USE

sidebar_position: 4 title: "Artifacts & Views"

ENTERPRISE ARCHITECTURE — 03: ARTIFACTS AND VIEWS

The things EA produces, what they're for, and how they connect

THE CRITICAL DISTINCTION: ARTIFACTS VS. ARCHITECTURE

Many EA programs confuse producing artifacts with *doing architecture*. The difference:

- **Artifacts** are documents, diagrams, models — the recorded outputs of architectural thinking
- **Architecture** is the set of decisions, principles, and structures that govern how the enterprise works

Artifacts have no value if they don't drive decisions. The question to ask about any EA artifact is: "**What decision does this inform or govern?**" If the answer is "none," the artifact shouldn't exist.

THE STANDARD VIEW MODEL (ISO/IEC/IEEE 42010)

Formal EA practice organizes artifacts into **architectural descriptions** composed of **views**. A view is a representation of the enterprise from a specific stakeholder perspective, addressing specific concerns.

The classic four-layer view model (used by TOGAF, DODAF, and most military frameworks):



TECHNOLOGY / INFRASTRUCTURE

← What platforms and infrastructure exist

Each layer informs the one below it. Strategy drives capability requirements. Capabilities drive process design. Processes drive data requirements. Data requirements drive application design. Application requirements drive infrastructure decisions.

The most common EA failure: building the bottom layers first (technology, systems) without alignment to the top layers (strategy, capabilities, processes).

THE CORE ARTIFACTS

Capability Map

What it is: A structured, hierarchical inventory of *what the enterprise does*, independent of *how* it does it.

What it's for: The most foundational EA artifact. Provides a stable vocabulary for discussing the enterprise that doesn't change every time a re-org happens.

Key principle: Capabilities are *business functions*, not org boxes. "Intelligence Analysis" is a capability. "J2 Directorate" is an org structure. Orgs change; capabilities are more stable.

Example structure:

```

Mission Command
├── Situational Awareness
│   ├── Intelligence Collection
│   ├── Sensor Integration
│   └── Common Operating Picture
├── Operations Planning
│   ├── Course of Action Development
│   ├── Synchronization Matrix Management
│   └── Decision Support
└── Communications
    ├── Network Management
    └── Cross-Domain Data Sharing
  
```

Decision it informs: Investment prioritization, capability gap analysis, portfolio rationalization.

Value Stream Map

What it is: End-to-end representation of how value flows through the enterprise from input to outcome, crossing org and system boundaries.

What it's for: Identifies where the actual work happens, where the handoffs break down, where data gets created or transformed, and where the delays and waste accumulate.

Key principle: Value streams are *horizontal* (cross-org, cross-system). Org charts are *vertical*. Most enterprise problems live at the horizontal seams.

Decision it informs: Process redesign, integration priorities, data ownership assignment, automation targeting.

Capability Heat Map

What it is: A capability map overlaid with an assessment dimension — typically investment level, performance rating, or maturity score.

What it's for: Makes the gap between current state and target state visible at a glance. Enables leadership to see where capability is over/under-invested relative to strategic priority.

Common dimensions: - Performance (red/yellow/green by capability) - Investment level (over/under/right-sized) - Strategic importance (high/medium/low) - Technology debt (high/medium/low)

Decision it informs: Budget allocation, sequencing of modernization efforts, risk prioritization.

Application / System Inventory

What it is: A structured catalog of systems and applications, with metadata about purpose, capability alignment, owner, lifecycle status, and integration points.

What it's for: Answers "what do we actually have and what does it do?" before making any technology decisions. Foundation for rationalization, consolidation, and migration planning.

Key fields: - System name and ID - Capability(ies) it supports - Owner / authoritative authority - Lifecycle status (current / sunset / emerging) - Integration dependencies - Data it produces/consumes - Classification level(s)

Decision it informs: Application rationalization, consolidation targets, risk assessment (systems past end-of-life), investment sequencing.

Data / Information Map

What it is: A structured view of the information the enterprise creates, uses, and depends on — including ownership, authoritative sources, quality, and flow.

What it's for: The most underbuilt view in most EA programs, and the most important in data-intensive environments. Without it, you can't do semantic interoperability, authoritative source governance, or AI/ML at scale.

Key questions it answers: - What data entities does the enterprise depend on? - Who is the authoritative source for each? - Where does data originate vs. where is it consumed? - Where does the same data exist in multiple places with conflicting values? - What data is required for each critical decision?

Decision it informs: Data platform investment, ontology design, API/integration standards, AI/ML feasibility, authoritative source designation.

Technology Reference Architecture

What it is: The approved set of technology patterns, platforms, and standards the enterprise uses to deliver solutions.

What it's for: Reduces the decision space for solution architects and program teams. "You don't have to decide what database to use — here are the approved options and when to use each."

Typical components: - Approved platform tiers (cloud, on-prem, classified) - Integration patterns (event-driven, API, file-based) - Data storage patterns (operational, analytical, archival) - Security patterns (zero-trust, identity, encryption standards) - Development standards (languages, frameworks, CI/CD patterns)

Decision it informs: Acquisition requirements, solution design reviews, vendor selection, security compliance.

Roadmap

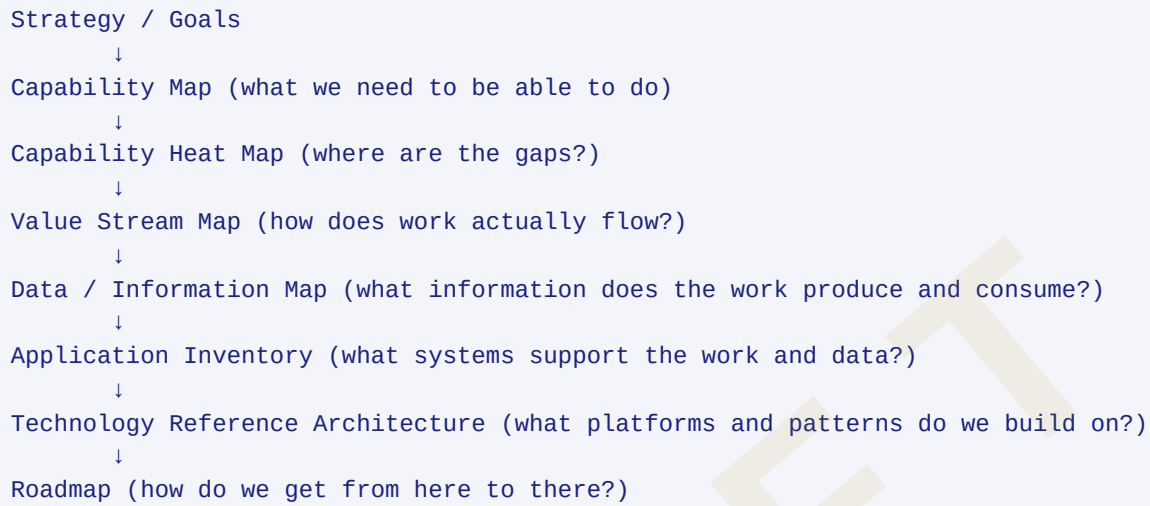
What it is: A sequenced, dependency-aware plan for moving from current state to target state across multiple dimensions (capability, data, systems, people).

What it's for: Makes transformation tangible. Turns strategy into a time-phased set of initiatives with dependencies visible.

Key principle: A roadmap without dependency modeling is just a list of projects. The value of the roadmap is in surfacing what has to come *before* something else — and therefore where the critical path runs.

Decision it informs: Portfolio sequencing, program funding, resource allocation, risk management.

HOW THE ARTIFACTS CONNECT



Each artifact answers a question. The questions are ordered. You cannot answer question 5 well without having answered questions 1–4.

THE MINIMUM VIABLE EA ARTIFACT SET

For a command-level CTO establishing EA governance, the minimum set that enables real decisions:

Priority	Artifact	What it unlocks
1	Capability Map	Common vocabulary; portfolio alignment
2	Application / System Inventory	Rationalization; dependency visibility
3	Data / Information Map	Semantic governance; authoritative sources
4	Capability Heat Map	Investment prioritization
5	Technology Reference Architecture	Acquisition guidance; solution review governance
6	Roadmap	Sequenced transformation plan

Value streams and operating model views can come later — they're important but not blocking.

DODAF CONTEXT

In military contexts, these views are formalized in DODAF (Department of Defense Architecture Framework):

DODAF View Set	What It Covers
AV (All Views)	Overview, dictionary, standards
CV (Capability Views)	Capability taxonomy, relationships, evolution
OV (Operational Views)	Operational nodes, activities, information flows
SV (Systems Views)	Systems, interfaces, functions
DIV (Data and Information Views)	Conceptual, logical, physical data models
TV (Technical Standards Views)	Standards profiles and forecasts
SvcV (Services Views)	SOA/service interfaces and behaviors

The conceptual mapping: CV ≈ Capability Map, OV ≈ Value Stream / Process, DIV ≈ Data Map, SV ≈ Application Inventory, TV ≈ Technology Reference Architecture.

DODAF is the mandated format; the underlying purpose is identical to the generic view model described above.

Previous: [02 — Schools of Thought](#) | Next: [04 — Governance](#)