C4ISR Architectures and Software Architectures

Rich Hilliard
rh@mitre.org
IEEE Architecture Working Group
http://www.pithecanthropus.com/~awg/

Circa 1996?
updated info:
 r.hilliard@computer.org
http://www.iso-architecture.org/42010
Contents

- “Architecture” in Context
- What is the C4ISR Architecture Framework?
- Issues with the Framework
- How to fix it?
- How does it relate to “Software Architecture”?
<Adjective> Architectures

- Application Architectures
- Data Architectures
- Enterprise Architectures
- Logical Architecture
- Makefile Architectures
- Operational Architectures
- Physical Architectures
- Security Architectures
- Systems Architectures
- Technical Architectures
- Occupant Architectures
- Heating and Lighting Architectures
- Building Code Architectures
Two Definitions We Like

- An architecture is the highest level (essential, unifying) concept of a system in its environment
  - IEEE Architecture Working Group, 1995
- The structure of the components of a program/system, their interrelationships, and principles and guidelines governing their design and evolution over time.
- SEI, 1995
What is the Framework?

- The Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) Architecture Framework
  - “... provides the rules, guidance, and product descriptions for developing and presenting architecture descriptions that ensure a common denominator for understanding, comparing and integrating architectures.”
  - OSD recently directed “that all on-going and, planned C4ISR or related architectures be developed in accordance with Version 2.0”
What is the Framework?

- “There are three major perspectives, i.e., views, that logically combine to describe an architecture ... the operational, systems and technical views.”

- All Framework quotations and examples in this briefing are taken from Framework 2.0 document (available from http://www.cisa.osd.mil)
What is the Framework?

**Operational View**
- Identifies Warfighter Relationships and Information Needs

**Systems View**
- Relates Capabilities and Characteristics to Operational Requirements

**Technical View**
- Prescribes Standards and Conventions
- Technical Criteria Governing Interoperable Implementation/Procurement of the Selected System Capabilities

Specific Capabilities Identified to Satisfy Information-Exchange Levels and Other Operational Requirements
What is the **Framework**? (Operational Architecture View)

- “The operational architecture view is a description of the tasks and activities, operational elements, and information flows required to accomplish or support a military operation.”

- Key concepts:
  - operational elements, activities and tasks, information exchange requirements
What is the *Framework*? (Systems Architecture View)

- “The systems architecture view is a description, including graphics, of systems and interconnections providing for, or supporting, warfighting functions.”

- Key concepts:
  - systems, nodes, locations, physical connections, performance parameters
What is the Framework? (Technical Architecture View)

- "The technical architecture view is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements, whose purpose is to ensure that a conformant system satisfies a specified set of requirements."

- Key concepts:
  - implementation guidelines, technical standards, conventions, rules and criteria organized into profiles
What is the Framework?

- The Framework identifies 26 architecture products
- 7 essential (i.e., required)
- 19 supporting (i.e., optional)

<table>
<thead>
<tr>
<th>Applicable Architecture View</th>
<th>Product Reference</th>
<th>Architecture Product</th>
<th>Essential or Supporting</th>
<th>General Nature</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Views (General)</td>
<td>AV-1</td>
<td>Overview and Summary Information</td>
<td>Essential</td>
<td>Scope, purpose, intended users, environment depicted, analytical findings, if applicable. (4.2.1.2)</td>
</tr>
<tr>
<td>All Views (Terms)</td>
<td>AV-2</td>
<td>Integrated Dictionary</td>
<td>Essential</td>
<td>Definitions of all terms used in all products. (4.2.1.2)</td>
</tr>
<tr>
<td>Operational OV-1</td>
<td>High-level Operational Concept Graphics</td>
<td>Essential</td>
<td>High-level graphical description of operational concept (high-level organizations, missions, geographic configuration, connectivity, etc.). (4.2.2.1)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-2</td>
<td>Operational Node Connectivity Description</td>
<td>Essential</td>
<td>Operational nodes, activities performed at each node, connectivity &amp; information flow between nodes. (4.2.2.2)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-3</td>
<td>Operational Information Exchange Matrix</td>
<td>Essential</td>
<td>Information exchanged between nodes and the relevant attributes of that exchange such as media, quality, quantity, and the level of interoperability required. (4.2.2.3)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-4</td>
<td>Command Relationships Chart</td>
<td>Supporting</td>
<td>Command, control, coordination relationships among organizations (e.g., inter-operability), interfaces, and mechanisms that perform those activities. (4.2.2.4)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-5</td>
<td>Activity Model</td>
<td>Supporting</td>
<td>Activities, relationships among activities, O(OMA) (instructions, e.g., plans, scenarios), and mechanisms that perform those activities. (4.2.2.5)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-6</td>
<td>Operational Rules Model</td>
<td>Supporting</td>
<td>One of the three products used to describe operational activity sequence and timing that identifies the business rules that constrain the operation. (4.2.2.6)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-7</td>
<td>Operational Event/Trace Description</td>
<td>Supporting</td>
<td>One of the three products used to describe operational activity sequence and timing that traces the actions in a scenario or critical sequence of events. (4.2.2.7)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-8</td>
<td>Operational Event/Trace Description</td>
<td>Supporting</td>
<td>One of the three products used to describe operational activity sequence and timing that identifies responses of a business process to events. (4.2.2.8)</td>
<td></td>
</tr>
<tr>
<td>Operational OV-9</td>
<td>Logical Data Model</td>
<td>Supporting</td>
<td>Documentation of the data requirements and structural business process roles of the Operational View. (4.2.2.9)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-1</td>
<td>System Interface Description</td>
<td>Essential</td>
<td>Identification of systems and system components and their interfaces, within and between nodes. (4.2.2.10)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-2</td>
<td>Systems Communications Description</td>
<td>Supporting</td>
<td>Physical nodes and their related communications linkages. (4.2.2.11)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-3</td>
<td>Systems/Matrix</td>
<td>Supporting</td>
<td>Relationships among systems in a given architecture, can be designed to show the relationships of interest, e.g., system interfaces, planned vs. existing interfaces, etc. (4.2.2.12)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-4</td>
<td>System Functionality Description</td>
<td>Supporting</td>
<td>Functions performed by systems and the information flow among system functions. (4.2.2.13)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-5</td>
<td>Operational Activity to System Function Traceability Matrix</td>
<td>Supporting</td>
<td>Mapping of system functions to operational activities. (4.2.2.14)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-6</td>
<td>System Information Exchange Matrix</td>
<td>Supporting</td>
<td>Mapping of information exchanges among system elements, applications and HW allocated to system elements. (4.2.2.15)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-7</td>
<td>System Performance Parameters Matrix</td>
<td>Supporting</td>
<td>Performance characteristics of each system's hardware and software elements, for the appropriate timeframes. (4.2.2.16)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-8</td>
<td>Logical Data Description</td>
<td>Supporting</td>
<td>Planned incremental approach toward migrating a state of systems to a more efficient state, or toward evolving a current system to a future implementation. (4.2.2.17)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-9</td>
<td>System Technology Forecast</td>
<td>Supporting</td>
<td>Emerging technologies and software/hardware products that are expected to be available in a given set of timeframes, and that will affect future development of the architecture. (4.2.2.18)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-10</td>
<td>Systems Rules Model</td>
<td>Supporting</td>
<td>One of the three products used to describe systems activity sequence and timing, System-specific refinements of critical sequences of events that identify the business rules that constrain the operation. (4.2.2.19)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-11</td>
<td>System State Transition Description</td>
<td>Supporting</td>
<td>One of the three products used to describe systems activity sequence and timing, System-specific refinements of critical sequences of events that identify the business rules that constrain the operation. (4.2.2.20)</td>
<td></td>
</tr>
<tr>
<td>Systems SV-12</td>
<td>Physical Data Model</td>
<td>Supporting</td>
<td>Physical implementation of the information of the Logical Data Model, e.g., message formats, file structures, attached schema. (4.2.2.21)</td>
<td></td>
</tr>
<tr>
<td>Technical TV-1</td>
<td>Technical Architecture Profile</td>
<td>Essential</td>
<td>Extraction of standards that apply to the given architecture. (4.2.2.22)</td>
<td></td>
</tr>
<tr>
<td>Technical TV-2</td>
<td>Standards Technology Forecast</td>
<td>Supporting</td>
<td>Description of emerging standards that are expected to apply to the given architecture, within an appropriate set of timeframes. (4.2.2.23)</td>
<td></td>
</tr>
</tbody>
</table>
Issues with the Framework

- Do we need another MIL Standard?
  - in the face of recent DOD direction to use best commercial practices (such as IEEE, ISO, and de facto industry standards like UML)
  - Its products are summary in nature rather than “developmental”
  - against trend toward contractor-defined products
  - “... the Framework does not provide guidance in how to design or implement a specific architecture or how to develop and acquire systems-of-systems.”
Issues with the Framework

- What does it mean to conform to the Framework?
  - Produce all essential, applicable products

- No provisions in *Framework* for user comment, document revision and maintenance

- The authors haven’t done their homework
  - The *Framework* ignores the substantial literature and emerging practice in architecture from the commercial world, industrial, research and standards communities
  - Mistakenly cites a definition of “architecture” as from IEEE (actually the definition appears to be from SEI)
Issues with the Framework (Conceptually)

- Everything isn’t an architecture. As Mary Shaw put it:
  - “Let’s not dilute the term architecture by applying it to everything in sight” (April 1995)
  - “Architecture” as used in the Framework is synonymous with “model”
- Does not distinguish problem and solution space orientations
- Systems Architecture View adopts a stove-piped mind set
- Operational-Systems-Technical split cuts across usual engineering partitions
How to fix it?

- Change the name to *C4ISR Modeling Framework*
- Make the notion of *view* more rigorous, following community practices (ISO, IEEE)
- Generalize “operational architecture view” to address multiple stakeholders of a DOD (or any system)
- Eliminate redundant architecture products, and allow best commercial practices in selection of notations and tools, managing at the viewpoint level
How does the Framework relate to Software Architecture?

- Software architecture “defines [a] system in terms of computational components and interactions among those components” (Shaw and Garlan)
- Key elements: components (clients, servers, databases, filters, layers, ...); connectors (procedure calls, shared variables, protocols, ...)
Issues with “Software Architecture”

- Emphasis on software structure makes it difficult to deal with other system fundamentals, e.g.,
  - Distribution
  - Security
  - Behavior and dynamics
IEEE Architecture Working Group: Goals and Objectives

◆ Take a “wide scope” interpretation of architecture as applicable to software-intensive systems
◆ Establish a conceptual framework and vocabulary for talking about architectural issues of systems
◆ Identify and promulgate sound architectural practices
◆ Allow for the evolution of those practices as relevant technologies mature
Conceptual Framework

Architecture
- has documents
- fits within
- lives in
- has concerns for
- has

System
- has concerns for
- fulfills
- solves
- has role in

Mission
- achieves within
- participates in
- has

Context
- achieves
- participates
- has

System Stakeholder
- has

Architectural Description
- conforms to
- comprises

Viewpoint
- satisfies
- comprises

Architectural View
- comprises

Model
- represents concerns of

System Stakeholder
Example Architecture Viewpoints

- Distribution View
- Capability View
- Production View
- Data View
- Security View
- Communications View

Architecture (set of abstractions)
Current Status and Plans

- Version 1.0 of P1471 *Recommended Practice*, has circulated to reviewers
- First ballot of P1471, October 1998
- Interested reviewers/participants contact:
  - Basil Sherlund (chair) b.sherlund@computer.org, or
  - ieee-awg@spectre.mitre.org
  - http://www.pithecanthropus.com/~awg