14th International Command and Control Research and Technology Symposium

C2 and Agility

Attaining Value from Actionable DoD Enterprise Architectures

Paper #53

Topic 9: C2 Architecture and Technologies

Steven J. Ring
The MITRE Corporation
202 Burlington Rd
Bedford, MA 01730
781-271-8613
sring@mitre.org

14th International Command and Control Research and Technology Symposium

C2 and Agility

Attaining Value from Actionable DoD Enterprise Architectures

Paper #53

Topic 9: C2 Architecture and Technologies

Steven J. Ring
The MITRE Corporation
202 Burlington Rd
Bedford, MA 01730
781-271-8613
sring@mitre.org

Abstract

This paper describes how DoD and federal government sectors can attain value from their architectures that will lead to better understanding and improving their business and warfighting capabilities. The Encarta Dictionary defines **value** as worth, desirability, importance, or usefulness of something to somebody. Within the context of an Enterprise Architecture, *value* can be defined as: somebody (*agencies, etc*) who performs actions (*develops architectures*) to obtain results (*analytics*) that are useful (*planning, decision making, and improving operational execution*) to somebody (*businesses*, *warfighters*). Architectures are reusable assets that you invest in to attain value. If you use an architecture only once, it becomes an expense. When you use an architecture for value, it becomes **actionable,** hence the term **actionable architectures**. Actionable architectures enhance an organizations ability to be agile in reacting to a rapidly changing operational environment. In this paper, practical benefits, strategies, and techniques are presented for analyzing static and dynamic "executable" architectures to make them actionable. Two federal government instruments that measure how well organizations and agencies plan, develop, and use their architectures are discussed as another path to achieve value. Ultimately, the value of architectures is measured by how well they are used to help organizations and agencies be successful in achieving their business and warfighter goals and objectives.

Introduction

The first mention of the word "architecture" within federal legislation came in the Clinger Cohen Act (CCA) of 1996 which recognized the need for federal agencies to improve the way they select and manage information technology (IT) resources [CCA, 1996]. CCA states that a Chief Information Officer (CIO) shall be responsible for "*developing, maintaining, and facilitating the implementation of a sound and integrated information technology architecture for the executive agency*" [CCA 5125(b), 1996]. CCA defines IT architecture as "*an integrated framework for evolving or maintaining existing IT and acquiring new IT to achieve the agency's strategic goals and information resources management goals*".

Presidential Executive Order 13011 of 1996 on *Federal Information Technology* established a federal CIO Council to serve as a forum to share ideas and practices and develop recommendations for overall federal IT management policy, procedures, and standards [PEO 13011, 1996]. In 2001, the CIO Council published *A Practical Guide to Federal Enterprise Architecture*, a step-by-step process guide to assist agencies in defining, maintaining, and implementing Enterprise Architectures [CIO, 2001]. The Guide defines an Enterprise Architecture (EA) as the descriptions of a current baseline environment (as-is) and future target environment (to-be) and the transition plan for changing the enterprise from current to target.

Today, architecting is being applied throughout the DOD not only to support improvements in Command and Control (C2) operational concepts and joint force interoperability, but to the broader functions of: defining the force and force capabilities; allocating resources to acquire and improve capabilities; equipping the force; and supporting the transformation to an agile net-centric force. Command and Control (C2) is recognized as a critical element of successful military operations. C2 is characterized by the strong direct link between human and organizational issues where the organizational design reflects the interaction of tasks to be done, the people available to perform them, and the systems and other material resources that support them. It is this human dimension that largely distinguishes C2 assessments from other military operational assessments. All of these elements require an understanding of highly complex military doctrine, organizations, missions and processes, as defined in an integrated C2 enterprise architecture, where the communication of information, within complicated endeavors, becomes critical.

The Encarta Dictionary defines **value** as worth, desirability, importance, or usefulness of something to somebody. Within the context of an Enterprise Architecture, *value* can be defined as: somebody (*agencies, etc*) who performs actions (*develops architectures*) to obtain results (*analytics*) that are useful (*planning, decision making, and improving operational execution*) to somebody (*businesses*, *warfighters*). Architectures are reusable assets that you invest in to attain value. If an architecture is used only once, it becomes an expense. When an architecture is used for value, it becomes **actionable,** hence the term **actionable architectures**.

John Zachman defines three Returns on Investments (ROI) of an EA to be: 1) *alignment* of systems and networks with management intentions and requirements for the enterprise; 2) *integration* of interfaces, interoperation, information, and business rules across the organization; and 3) *managing change* due to complexities of the enterprise [Zachman, 2001]. As today's government agencies are asked to provide more services to citizens [PMA, 2002] and as the Department of Defense (DoD) undergoes the transformation to a net-centric environment [QDR, 2006] *actionable architectures* that provide *value* are the only way to manage that alignment, integration and change.

Current State of Attaining Architecture Value

An EA offers tangible benefits to an organization by enabling strategic acquisition planning and informed investment decisions, and by improving operational process execution. An EA is the foundation of the decision making process. However, for architectures to be of value they must be used in the decision making process supporting a business' or warfighter mission's outcomes and goals (see Figure 1). These outcomes and goals, in turn, become drivers for the architecture, thus completing the loop.

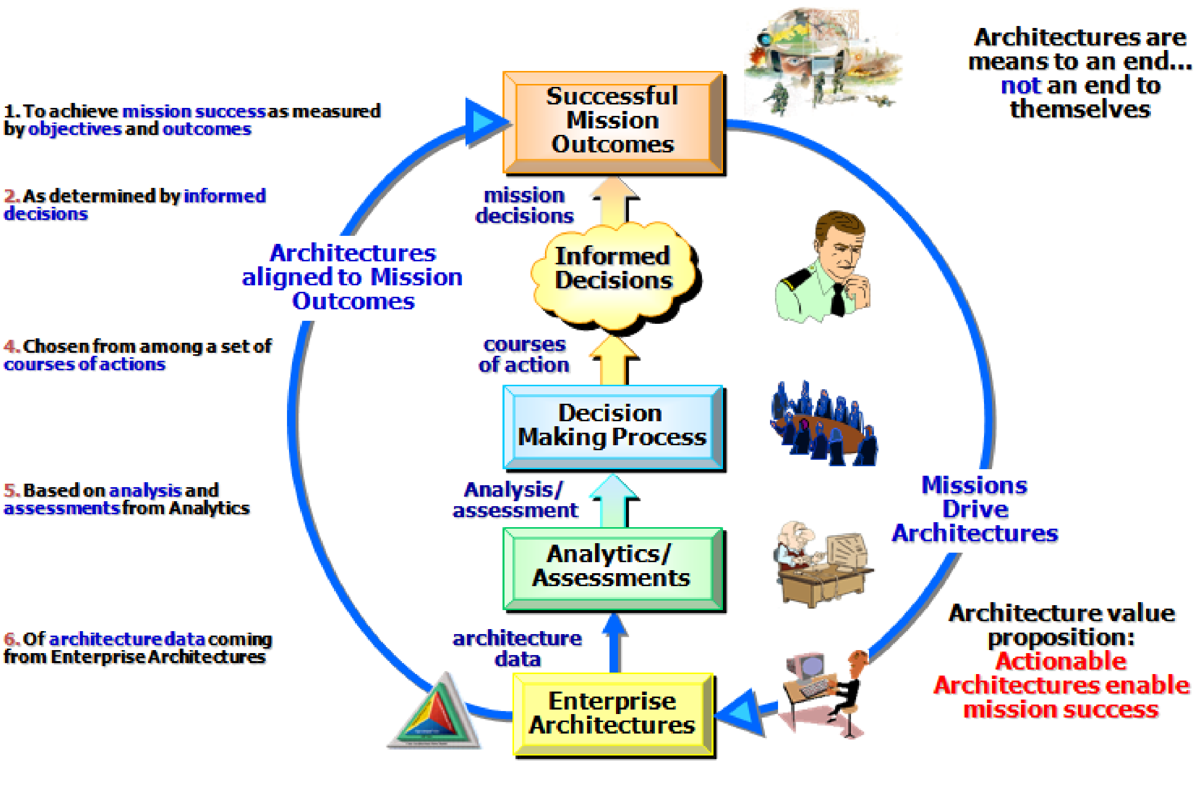


Figure 1 Using Architectures to Attain Value

Agencies and organizations spend significant time and resources on the planning and development life cycle phases of their EAs. However, they rarely get value from their architectures because architecture use for the purposes of planning, investment decision making, and improving operational execution generally ranges from minimal to non-existent (although pockets of good usage do exist).

Too often, architectures are viewed as a mandated, "*check the box*," compliance requirement that is costly and takes too long to produce. Although this is certainly not the intent of the DoD Joint Capabilities Integration and Development System (JCIDS) process [JCIDS, 2007], misinterpretation of JCIDS is a prime example of this architecting. JCIDS calls for developing certain DoD Architecture Framework (DoDAF) products [DoDAF, 2007]. These DoDAF products should be not be viewed as an "end" by themselves, that is "*checking the box*," but as a means to an end for attaining value. Unfortunately, JCIDS' emphasis on discrete DoDAF product development leads some to obscure the real purpose in describing architectures which results in "*architecture for architecture's sake*." When this viewpoint is adopted, some organizations and agencies even terminate all architecture work once JCIDS Milestone Development B has been reached. The architecture then becomes nothing more than *wallpaper* and *shelfware (i.e., unused)*. These agencies miss out on attaining real-world value and ROI when they fail to use their architecture products to analyze and assess processes, operations, and resource utilization.

How to Attain Value

Agencies must understand the value of their EA. The General Accounting Office's (GAO) Information Management Technology report recognizes this explicit fact [GAO-04-49, 2004]. It states: "*attempts to modernize IT environments without blueprints—models simplifying the complexities of how agencies operate today, how they want to operate in the future, and how they will get there—often result in unconstrained investment and systems that are duplicative and ineffective*."

Many agencies often feel that architectures only document what is already known – that is, inventories of present conditions, processes and resources. In fact, architecture descriptions can help communicate complexity and aid in their understanding. They can model both current and future capabilities and detail the plans to get to that future state. As such, an EA should be seen as an overall planning and management tool and not strictly as a technology project [Hardy, June 2008]. Within that planning and management function, actionable architectures enhance an organizations ability to be agile in reacting to a rapidly changing operational environment. Jan Popkin has stated "*Enterprise Architecture is a mechanism to provide results – whether it's agility, alignment, collaboration – and so on … it is an enabler in itself*" [Popkin, 2008].

Integrated Architectures

Architecture descriptions usually take one of two forms: 1) structured models characterized by hierarchical activity decompositions like IDEF0 [IDEF0, 1993] or 2) object-oriented Unified Modeling Language (UML) models that describe system behavior from a user's perspective [UML, OMG]. In a 2005 State of DoD Architecting study, it was reported that structured modeling accounts for 80% of DoD architectures [Ring, 2005].

However, before using architecture descriptions for any purposes, one must start with an architecture that is integrated, unambiguous, and consistent. The **Activity Based Methodology** (ABM) was developed to improve the practice of DoD architecting [Ring, 2007]. ABM represents a new paradigm for architecture development and analysis. It establishes a tool-independent, disciplined, repeatable approach to developing, maintaining, and analyzing fully integrated, unambiguous, traceable, and consistent structured enterprise and mission level architectures. Architectures developed with ABM answer the six Zachman interrogatives: Who, What, Where, When, Why and How [Zachman, 1987]. With ABM, architectures can be developed in a more efficient, streamlined, and speedier process that ensures data consistency and results in quality architecture products. The resultant analysis, thus, is more accurate, not subject to misinterpretation, and has more value to the decision maker. ABM has been implemented by three major enterprise architecture technology corporations whose products are widely used throughout DoD and government sectors today: 1) *System Architect* by IBM [IBM, 2009], 2) *Provision* by Metastorm (Metastorm, 2008), and 3) *Troux 8* by Troux Technologies [Troux, 2009].

Measuring the Quality and "Goodliness" of an Architecture

There is little attention paid today to measuring the "goodliness" of an architecture or to assess its quality or to validate and verify (v&v) its architecture data content. Just because you have OVs and SVs doesn't guarantee quality and consistency in your architecture and its data content. How do you measure architecture verifiability, traceability, compliance and completeness? How do you know when you are "done" and how do measure the quality of your architecture when that "done" state is reached. By what rationale do you claim your architecture is complete and can be used for analysis and decision making purposes? What Quality Control/Quality Assurance process was the architecture subjected to assess its quality?

Presently, we have no way of measuring architecture compliance with any "good, known, or standard" architecture concepts. For example, have OV-2 nodes been defined for which there are no Information Exchanges either to or from that Node? Have Mechanism Roles been defined yet they neither produce nor consume any type of information, data, or material product? Is there inconsistency or physical/ logical impossibilities in the OV-3 (Information Exchanges) or SV-6 (System Data Exchanges) products? This would be the case where exchanges might have been defined between two Nodes, where, it is **physically or logically impossible** for them to exchange anything – e.g., an Air Force AWACS node with an Army Tank Unit node or an Aegis naval ship Node with an Army Tactical Operations Center Node. Architecture tools cannot prevent this from happening because they have no way of knowing any physical or logical impossibilities. Yet we must at least consider doing some basic quality checks on our architectures to uncover these discrepancies that, ultimately, results in invalid analysis. Being able to measure the quality and "goodliness" of architectures will increase their value.

Iterative Development/Analysis Approach

Another important way to achieve value is to use an iterative approach to architecture development/analysis – i.e., build a little, use a little, build a little more, use a little more, etc. This enables architects to achieve small but reachable goals early throughout the architecture life-cycle process and then build on those successes.

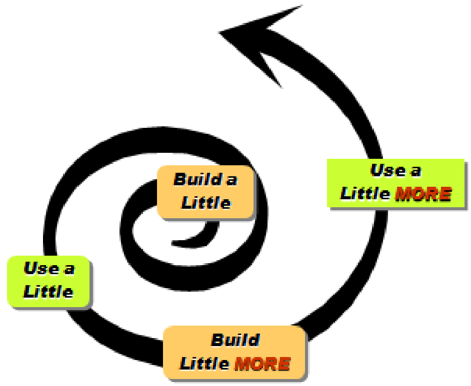


Figure 2 Iterative Approach

What is also needed is a set of stable and repeatable analytics so that as the architecture evolves and matures, the analysis can be repeatedly applied to continuously obtain increasing value. This avoids the "big-bang" approach where you wait until you are done (whatever that means) to begin the analysis process in attaining value. This was the subject of a recent article by Mike Rosen in "*Death by Architecture*" where he makes this very case [Rosen, 2009].

Architecture Analysis

Architecture analytics are those sets of processes, practices, and procedures that transform architecture data into actionable information. However, there is little consistency in architecture analytics today and thus, analysis data when obtained, has not been used consistently to support any purpose. If value is not attained from architectures, then there is uncertainty about how they impact future agency/business capability and warfighter performance. Consequently, it is easy to understand the difficulty of acceptance, support and increased resistance to architectures by senior management and military leaders.

The reality of architecture analytics is that it is non-trivial, sophisticated, "hard" work. It takes skill and knowledge on one's part to know the nature of data at hand and the integrity of data to be analyzed. It requires an iterative process of discovering patterns by clustering and filtering important data from the unimportant. It is about using a variety of analysis tools/ techniques where one selects the best technique to achieve relevant accurate and believable results based on the data being analyzed. Along these lines, there are two categories of architecture analytics: static (non-time varying) and dynamic (time-varying).

Static Architecture Analysis

Static architecture analysis can locate, identify, and resolve definitions, properties, facts, constraints, inferences (examining elements related to other elements so as to draw conclusions), associations (correlations between element and their attributes), and issues both within and across architecture descriptions that are redundant, conflicting, false, missing, and/or obsolete. Three key static techniques include 1) architecture data mining, 2) military Doctrine, Organization, Training, Material, Leadership, Personnel and Facility (DOTMLPF) analysis, and 3) visual representations.

Architecture data mining reveals and helps discover hidden rules, practices, gaps, seams, relationships, requirements, and patterns on how an enterprise conducts its business. One could say all the answers are in the architecture data to be mined – the hard part is determining the appropriate questions to ask. Three DoDAF products – OV-3, SV-5 (Operational Activity to Systems Function Traceability Matrix), and SV-6 – are the basis for DoD architecture data mining. Together, they identify connections between producers and consumers at the operational activity and system functional leaf levels. These connections are essential 1) for conducting "what if" and "if what" impact assessments between what is required and what is delivered – e.g., assessing the impact of losing a system or a system node, and 2) for examining usage (who/what affects something) and references (who/what is affected by something).

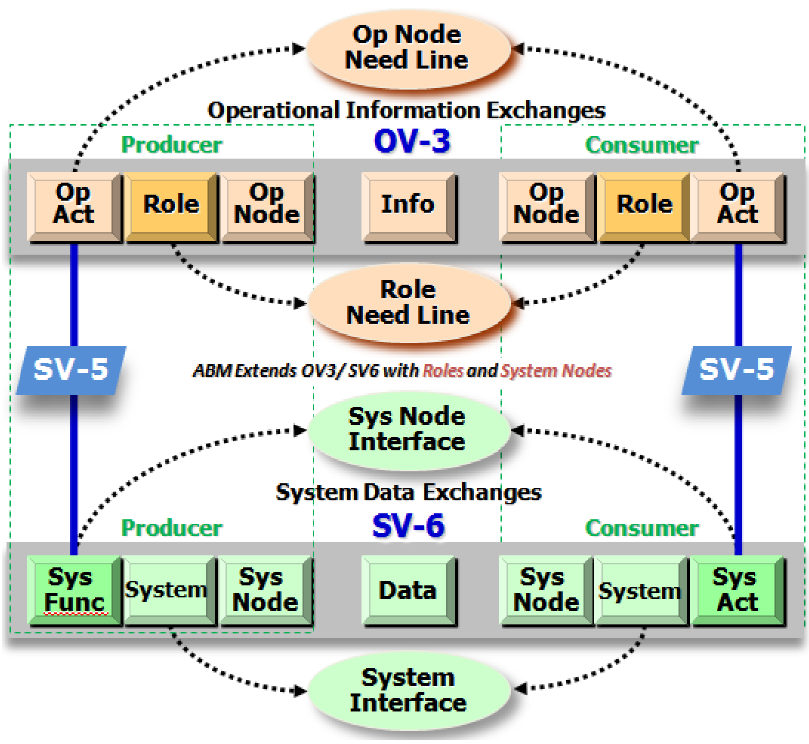


Figure 3. ABM OV-3, SV-5 and SV-6 Used for Architecture Data Mining

It should be pointed out that ABM uncovered a missing DoDAF operational need line – the *Role to Role Need Line*. This is the operational equivalent to the System to System Interface.

In addition, one can analyze critical resource (roles, systems) requirements, their dependencies, and their responsibilities by examining each resource allocation and aggregating its set of outputs (i.e., who produces what outputs from what activities/ system functions to determine the set of responsibilities (i.e., outputs) required of each resource. From that, one can discover if any requirements are redundant, conflicting, and/or obsolete. Likewise, one can aggregate a resource's entire set of inputs (i.e., who consumes what inputs) to uncover any hidden or missing requirements.

DOTMLPF analysis: DOTMLPF is the military description of a warfighting capability that consists of **O**rganizations of **T**rained **P**eople led by effective and competent **L**eaders (*human resources who do work*) performing **D**octrinal operations (*how work is performed*) at **F**acilities (*locations – where work is performed*) using **M**aterial (*system resources that do work*). DOTMLPF analysis (table 1 below) leads to better definitions of agile warfighting capabilities. The most critical characteristic of a net-centric transformed force, in terms of DOTMLPF, involves having not only the right Material (e.g., sensors, infostructure, combat systems) but also the right Doctrine, Organization, Personnel, Training and Leadership [Alberts, 2004].

Domain	Analysis	Architecture Elements
Doctrine	Examine Tactics, Techniques and Procedures	Activities, Nodes, People (Roles), Systems
Organization	Examine organizational structure	Organizational Units, People (Roles)
Training	Examine non-material educational and training issues of personnel and the systems they use in the performance of their activities	Activities, People (Roles), Systems
Material	Examine materiel solutions – a new system or more systems?	System Functions, Systems
Leadership	Examine leadership and reporting relationship issues – direct, indirect, task, matrix	Organizational Units, People (Roles)
Personnel	Examine people solutions and their roles (i.e. job titles) – new personnel or personnel with better qualifications	People (Roles), Organizational Units
Facilities	Examine places where activities are performed and, possibly, fixing, building or modifying facilities	Locations (places) where Activities are performed by People (Roles) (using systems)

Table 1. DOTMLPF Analysis

DOTMLPF analysis can lead to both material (e.g., we need more stuff) and non-material (e.g., we need better educated people or people with different skills) solutions. One area overlooked is the organization solution – the "O" in DOTMLPF – where the solution might lie in reorganizing, retraining, or reequipping an organization. This was the subject of a Joint Task Force Command and Control Strategy Research Project at the Army War College [Dickens, 2005].

In addition, DOTMLPF can discover:

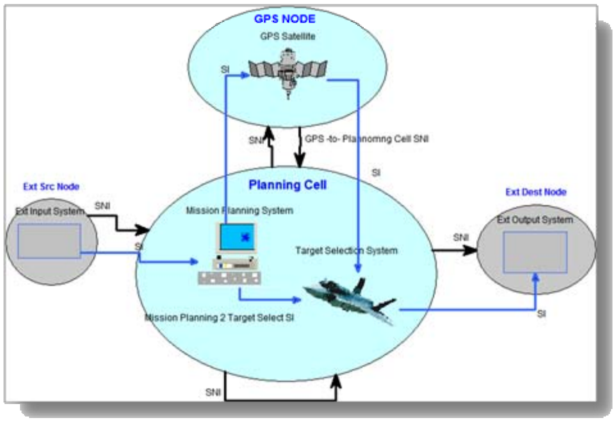
- **Gaps** (complete or partial absence of something) – For example, when a materiel or non-materiel solutions does not exist to fulfill DOTMLPF requirements.
- **Shortfalls** (amount by which something falls short of what is required) – For example, when DOTMLPF requirements have not been fulfilled to the required level for a specific task or when existing materiel or non-materiel solutions lacks sufficient interfaces to fulfill DOTMLPF requirements
- **Duplications** (exact copy of something) – For example, when more than one materiel or non-materiel solution fulfills the same DOTMLPF requirement

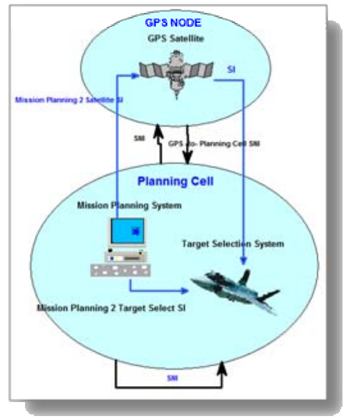
Unfortunately, consistent and standard analysis guidelines for each of the Doctrine, Organization, Material, Training, Leadership, Person, and Facility areas are lacking (although pockets of good DOTMLPF analysis do exist). This is another reason why agencies and organizations rarely get value from their architectures. (This should be the subject of future considerations.)

Visual Representations: How do you educate senior management and military leaders on the value of an EA? How do you create a "*Management View*" for visually presenting and communicating architectures to them in short, concise ways and in their language and in terms they understand? This was a need highlighted in the 2005 State of DoD Architecture study and it is as relevant today as it was when it first surfaced then.

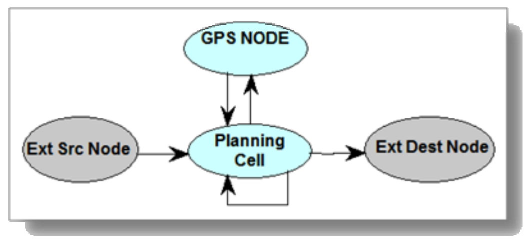
In DoDAF architectures, visual analysis representations can be accomplished two ways: 1) by selectively depicting certain element and relationship sets in multiple versions of the same product, and 2) by structuring a single product multiple ways. These visual representations provide value to different stakeholders depending on their focus and purpose and constitute that '*Management View*'.

SV-1 is a good example of **selectively depicting** certain elements and relations. While there is only one master SV-1 depicting all system nodes and systems (which is what is usually produced), you can have multiple SV-1s by choosing different sets of System Nodes and Systems to depict. The interface relationships and exchanges always remain the same. Below on the left is a typical master SV-1 showing all System Nodes, Systems, and their interfaces. Below on the right is a second SV-1 but only showing ***Internal*** Systems Nodes, Systems and their internal Interfaces.

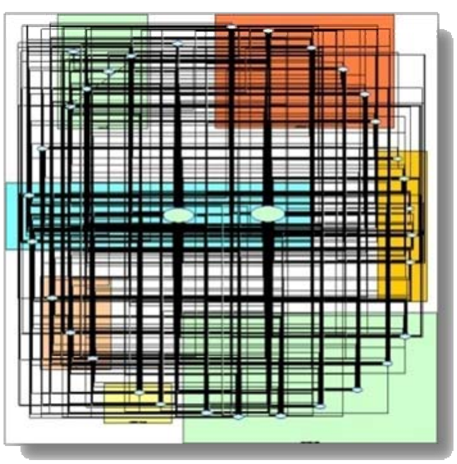


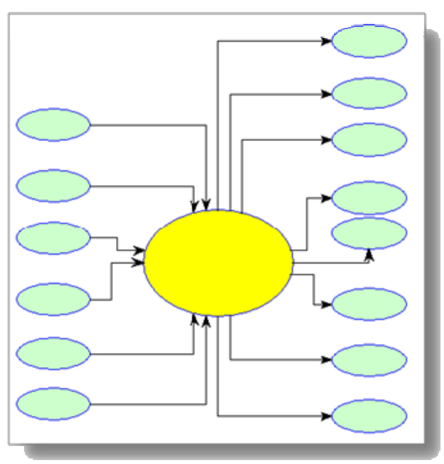


Going further, on the left below is a third SV-1 but only showing Systems and their Interfaces and on the right below is a fourth SV-1 but only showing System Nodes and their interfaces. The end result is that you can have as many SV-1s as needed to convey meaning and provide value.



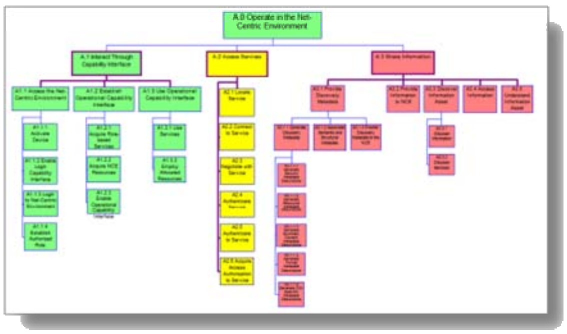
OV-2 is another example of this same type of visual analysis. Typically, an OV-2 diagram (on the left) looks like a ball of string that is unintelligible and, for all practical purposes, totally useless. A better approach (on the right) is to have each Node be the center spoke of a wheel where only the *first level source and destination node neighbors* are show. You can have as many of these OV-2 diagrams as you have operational nodes – one diagram per node.

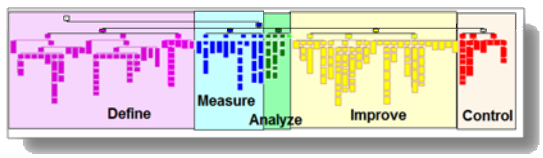




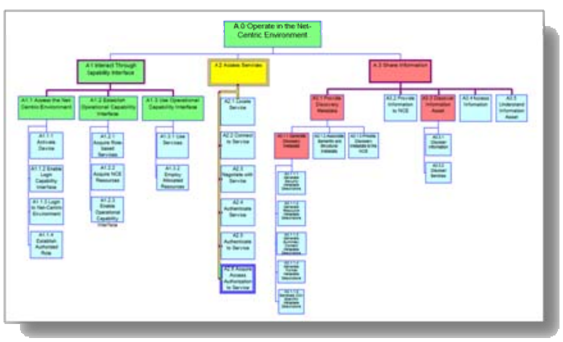
The OV-5 Node Tree diagram is an example of **structuring a single product** multiple ways. An OV-5 Node Tree diagram shows a complex hierarchical OV-5 activity model within a single 2 dimensional diagram. It is an enabler for helping people understand the organization and composition of that activity model. You can structure a Node Tree diagram three ways: classification, organization, and grouping.

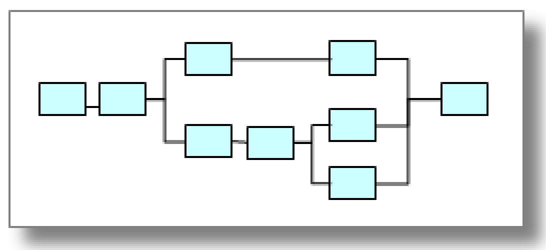
Classification: A Node Tree diagram (shown on the left below) can be used to show functional decomposition where each functional area is colored differently. On the right is a more complicated Node Tree Diagram from a different source but still showing functionality.



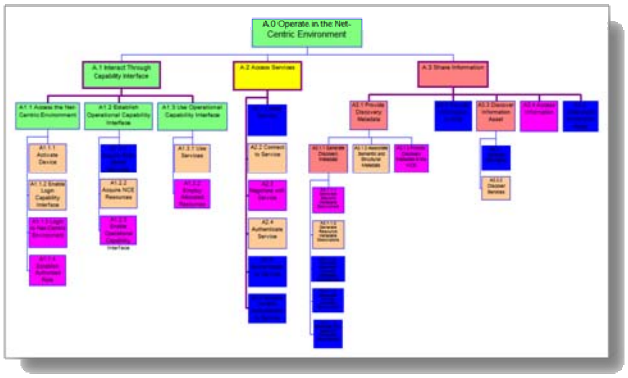


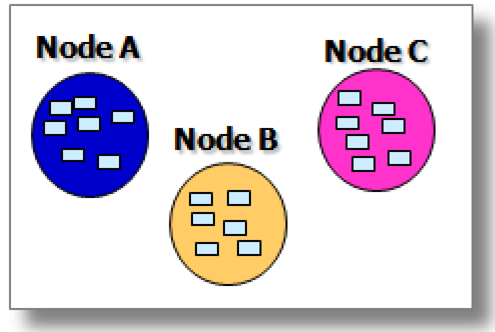
Organization: The same Node Tree diagram (on the left below) can be used again to show those leaf activities (color blue) that constitute an OV-6C process scenario thread (shown on the right as one of a number of candidate thread diagrams).





Grouping: The same Node Tree diagram (on the left below) can be used even again to show activity associations with nodes, or with roles, or with organizations, etc. Here activities are color coded depending on their node association. On the right is the same activity-node association but from an OV-2 Node perspective.





OV-3 is another example of a single product that can be visualized different ways. Typically, the OV-3 is produced and shown as a spreadsheet (see below).

	A	B	C	D	E	F	G	H
1	IER	From Activity	From Node	From OpNodeActivity Roles	ICOM Arrow	To Activity	To Node	To OpNodeActivity Roles
2	Travel Directions Detected	Receive Travel Directions	Baltimore	"Travel Clerk"	Travel Directions Detected	Notify all Travel Departments	United States	
3	Correlated Travel Data	Process Travel Assessment	Boston	"Travel	Correlated Travel Data	Update Common Travel Picture	Cleveland	"Travel Director"
4	Request for Travel Assessment	Requested Travel Assessment	Boston	"Travel	Request for Travel Assessment	Re-Redirect Travel Plans	Cleveland	"Travel Director"
5	Merged Travel Data	Coordinate Effects	Boston	"Travel	Merged Travel Data	Decide to Look at Travel Plan	Denver	"Travel Planner"
6	Correlated Travel Data	Process Travel Assessment	Boston	"Travel	Correlated Travel Data	Update Internal Information	United States	
7	Confirmed ID	Report Target Data	Chicago	"Travel Manager"	Confirmed ID	Coordinate Effects	Boston	"Travel
8	Travel Assessment	Report Travel Assessment	Chicago	"Travel Manager"	Travel Assessment	Process Travel Assessment	Boston	"Travel
9	Confirmed ID	Report Target Data	Chicago	"Travel Manager"	Confirmed ID	ReAnalyze Travel Directions	Cleveland	"Travel Director"
10	Travel Assessment	Report Travel Assessment	Chicago	"Travel Manager"	Travel Assessment	Process Travel Assessment	Cleveland	"Travel Director"
11	Confirmed ID	Report Target Data	Chicago	"Travel Manager"	Confirmed ID	Receive Travel Data	Houston	"Travel Helper"
12	Travel Assessment	Report Travel Assessment	Chicago	"Travel Manager"	Travel Assessment	Process Travel Assessment	Houston	"Travel Helper"
13	Confirmed ID	Report Target Data	Chicago	"Travel Manager"	Confirmed ID	Conduct Travel Analysis	Miami	"Travel Leader"
14	Travel Assessment	Report Travel Assessment	Chicago	"Travel Manager"	Travel Assessment	Process Travel Assessment	New York	"Travel Analyzer"
15	No Travel ID	Report Target Data	Chicago	"Travel Manager"	No Travel ID	No Traveling Identification	United States	
16	Correlated Travel Data	ReAnalyze Travel Directions	Cleveland	"Travel Director"	Correlated Travel Data	Coordinate Effects	Boston	"Travel
17	Destination Configuration Data	Redirect Travel Plans	Cleveland	"Travel Director"	Destination Configuration Data	Report Target Data	Chicago	"Travel Manager"
18	Travel Configuration Data	Re-Redirect Travel Plans	Cleveland	"Travel Director"	Travel Configuration Data	Report Travel Assessment	Chicago	"Travel Manager"
19	Traveling Configuration Data	Redirect My Traveling Plans	Cleveland	"Travel Director"	Traveling Configuration Data	Report Target Data	Chicago	"Travel Manager"

However, it can be multiply sorted and color coded to provide a much more meaningful visual result. The depiction below (in report format) shows the same ordering of the OV-3 spreadsheet from above but in a far more easily understood format where the Need Lines and exchanges (and their separate counts) clearly stand out. This is another "*Management View*" example.

Page 1 of 3 **Travel Example OV3 Matrix in 'TravelEx'** 11

	Input/Output	FromActivity	FromNode	ToNode		ToActivity	
1	NodePairs=1		**Baltimore**				
	1		IERCnt=1	1 **United States**			
1	1 Travel Directions Detected	2.7 Receive Travel Directions			L	7 Notify all Travel Departments	E
2	NodePairs=4		**Boston**				
	2		IERCnt=2	1 **Cleveland**			
2	1 Request for Travel Assessment	5.1 Requested Travel Assessment			L	5.2 Re-Redirect Travel Plans	L
3	2 Correlated Travel Data	5.3 Process Travel Assessment			L	5.8 Update Common Travel Picture	L
	3		IERCnt=1	2 **Denver**			
4	1 Merged Travel Data	3.3 Coordinate Effects			L	3.4 Decide to Look at Travel Plan	L
	4		IERCnt=1	3 **United States**			
5	1 Correlated Travel Data	5.3 Process Travel Assessment			L	4 Update Internal Infomation	E
3	NodePairs=9		**Chicago**				
	5		IERCnt=2	1 **Boston**			
6	1 Confirmed ID	3.1 Report Target Data			L	3.3 Coordinate Effects	L
7	2 Travel Assessment	5.7 Report Travel Assessment			L	5.3 Process Travel Assessment	L
	6		IERCnt=2	2 **Cleveland**			
8	1 Confirmed ID	3.1 Report Target Data			L	3.2 ReAnalyze Travel Directions	L
9	2 Travel Assessment	5.7 Report Travel Assessment			L	5.3 Process Travel Assessment	L
	7		IERCnt=2	3 **Houston**			
10	1 Confirmed ID	3.1 Report Target Data			L	3.5 Receive Travel Data	L
11	2 Travel Assessment	5.7 Report Travel Assessment			L	5.3 Process Travel Assessment	L
	8		IERCnt=1	4 **Miami**			
12	1 Confirmed ID	3.1 Report Target Data			L	4.1 Conduct Travel Analysis	L
	9		IERCnt=1	5 **New York**			
13	1 Travel Assessment	5.7 Report Travel Assessment			L	5.3 Process Travel Assessment	L
	10		IERCnt=1	6 **United States**			
14	1 No Travel ID	3.1 Report Target Data			L	6 No Traveling Identification	E
4	NodePairs=10		**Cleveland**				
	11		IERCnt=1	1 **Boston**			
15	1 Correlated Travel Data	3.2 ReAnalyze Travel Directions			L	3.3 Coordinate Effects	L
	12		IERCnt=3	2 **Chicago**			
16	1 Traveling Configuration Data	1.3 Redirect My Traveling Plans			L	3.1 Report Target Data	L
17	2 Destination Configuration Data	2.3 Redirect Travel Plans			L	3.1 Report Target Data	L
18	3 Travel Configuration Data	5.2 Re-Redirect Travel Plans			L	5.7 Report Travel Assessment	L

Dynamic "Executable Architecture" Analysis

Most DoDAF products are "static" representations where architecture descriptions only show that activities and system functions "*must be capable of*" producing and consuming information and data products. They do not provide details on event sequencing or on how or under what conditions products are produced and consumed. They also do not explicitly identify timing details, or the number (capacity – how many) and assignment (ordering – who does what and in which order) of resources used.

Two DoDAF products have been defined for event sequencing – OV-6C (Operational Event Trace Description) and SV-10C (Systems Event Trace Description). OV-6C is used to describe operational activity trace actions in a scenario or sequence of events. SV-10C is used to identify system specific refinements of critical sequences of system function events described in the System View. However, both products lack sufficient details on timing dependencies, behaviors, performances and resource allocations.

Dynamic "Executable Architecture" simulation models differ from static operational models by introducing the time element and the accompanying risk, cost and dynamic behavior assessments. They go beyond "*must be capable of*" and define precisely, when and under what conditions, products (inputs/ outputs) are produced and consumed and the exact number and ordering (for when there is a team of resources performing the same task and, in baseball vernacular, who is leading off? who is batting second? etc.). An executable architecture, in DoDAF terms, can be defined as a *dynamic (over time) model of sequenced processes/ events (concurrent or sequential) performed at a node (location) by roles (within organizations) using resources (systems) to produce and consume information (data).*

Executable process models, the Zachman *When* interrogative, enable time-dependent, repeatable behavior and performance analysis of complex, dynamic operations and of human and system resource interactions that cannot be identified or properly understood using pure static models. These dynamic models enable time varying processes, their resources, cost, and the relationships among them to be understood, simplified, measured and optimized for efficiency, effectiveness, and performance. They show how to transform and evolve organizations, processes, and modes of operation over time to adapt to new roles, relationships, technologies, and threats.

Four prevalent technologies used to model executable architectures include: 1) token-based Coloured Petri Nets [CPN, 2009], 2) Discrete Event Simulations [Fishman, 2001], 3) executable UML based on event sequencing diagrams (Executable UML, Wikipedia), and 4) Business Process Modeling Notation (BPMN) [BPMI, 2009] together with the Business Process Execution Language (BPEL) [BPMN-BPEL. 2009].

Dynamic analysis can assess processes and organization structure to identify bottlenecks, delays, and lags and to optimize resource allocation. Responses to a single occurrence or synchronous/ asynchronous (random) events can be understood. The executable model can be stressed to the breaking point and beyond to assess impacts from conditions, events, and scenarios. Dynamic analysis can also be used to help determine how to recover from single or multiple process and resource failures and how to balance resources and minimize queuing times.

When unacceptable performance and queuing conditions occur, there are three solutions:

- First, one can **decrease the time duration** of a queued task (i.e., reduce rate of service). This implies that a solution was found by either 1) making the task more efficient so that it takes less time, 2) automating some part of the task or the entire task itself or 3) replacing the person (role) responsible for performing the task with someone who has better education and/or training and can, thus, perform the task more efficiently – that is, the *non-material* DOTMLPF solution.
- Second, one can **increase the number** (capacity) of available resources performing the queued task. However, there might be some physical or cost consideration that makes this solution unattainable. For example, certain military vehicles can only hold so many soldiers. There is no physical room to add more soldiers. In addition, it might be totally cost prohibitive to add more system resources (i.e., put more satellites in space).
- Third, one can **decrease the rate of arrival** or the number of incoming jobs for a queued task – that is, provide less demand per time period on the resource. This implies that either 1) some upstream operation was improved resulting in fewer jobs being passed to the resource, 2) some jobs were diverted to an idle resource, or 3) doctrine/tactics were changed by possibly splitting the task or by redefining it with different resources.

Two forms of dynamic performance assessments measure 1) the operational success of a warfighting forces' effort relative to the objective of the mission or operation being evaluated – *Measures of Effectiveness (MOE)* and 2) individual measures of human and system resource ability to function in that operational environment - *Measures of Performance (MOP)*. They both help determine overall operational mission success in accomplishing mission objectives. MOP and MOE, together with resource utilization over time, can be categorized into three different measurement areas: time-related, resource-related, and reliability related.

- **Time-related measures** can assess how long it takes to complete a single process thread or groups of process threads looking for latency times and variances under different loading and (extreme) stress conditions. Process times can be increased (or decreased) to assess impact on overall thread times. Delays due to unavailable resources or resource contention can also be discovered.
- **Resource-related measures** can assess resource utilization by measuring how busy (or not busy) a resource is to identify whether it is a bottleneck (overutilized and thus not available) or idle (underutilized and thus available). In addition, one can measure the utilization and operational costs of a resource in the context of when it performs its designed function within an overall process thread. One can also examine the utility of adding resources to support bottlenecked operation and balance that with the cost of that resource. Tradeoffs and impact analysis between various types of resources can be assessed and one can experiment with employing available (idle) resources to lighten the load on overutilized resources.
- **Reliability-related measures** can assess the health of an operation and its recoverability in terms of the impact of a single point of failure due to some unintended termination (perhaps a resource or sensor is no longer available). These measures may be qualitative assessments such as minimal impact, task failure, mission failure, or even loss of life. One can also assess the availability of an alternate or backup resource. If an alternate resource is not available when it is needed then the impact could again be described in the same qualitative measures as the impact of a single point of failure. Recoverability issues can be measured in terms of 1) the time to recover from a failure, 2) the time to adapt to changes in the environment (measured in time, quality, mission success, or losses), 3) the time to assess the impact of unexpected or unintended disruptions, and 4) the time to perform graceful degradation (what tasks need to be completed in how much time prior to a shutdown). This can be measured in terms such as whether the mission tasks were completed prior to shutdown, or whether the mission was accomplished prior to the organization's status changing to combat ineffective.

A major advantage of executable architecture models is that operational concepts and process threads can be safely (and cheaply) simulated and refined in a laboratory environment. Dynamic models can validate and verify original operational assumptions and can be used to predict and meet future force and organizational capabilities and data requirements. Lessons can be learned in a laboratory environment while avoiding costly mistakes that would have necessarily occurred in an actual live operational environment with real people, real systems, and real events.

One can extend a single executable architecture model to link with 1) a combat war-gaming simulation model as a mission scenario generator to provide operational context and 2) a communication simulation model to incorporate system and network delays of exchanges and information flows into the overall processing time. This was the focus of a MITRE research effort [Pawlowski, 2004] where a federation of executable architecture simulation models were linked, via the High Level Architecture [HLA, Wikipedia], within the context of a mission thread to dynamically access and measure process performance and force effectiveness, organizational work efforts, resource allocation, communication capability, impact of network and communication failures, and ultimately, mission success.

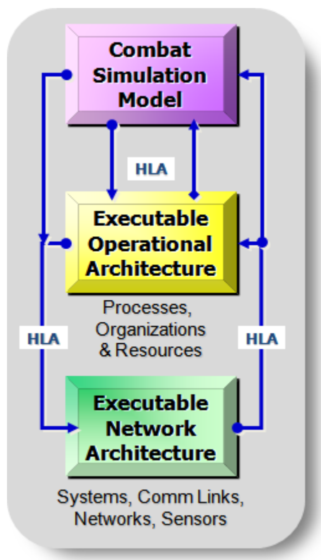


Figure 4. Federation of Simulation Models

Governance Measurement Instruments

Two governance measurement instruments assess federal agencies and their EA programs. These instruments complement each other and provide measurements for 1) the completeness of an agency's EA, 2) how well agencies are performing in developing their EA, and 3) how well they are using their EA. Knowing these measurements contributes to the overall value of an architecture.

The first is the Office of Management and Budget's (OMB's) Enterprise Architecture Assessment Framework [EAAF, 2008]. EAAF assesses the capability and maturity of federal EA programs to guide and inform strategic IT investments. OMB uses EAAF to better understand the current state of an agency's EA and agencies use EAAF to help them identify their EA programs' strengths and weaknesses and to assess the capability of their EA programs. The latest version of EAAF (v3.0) spans planning, investment, and operations activities required to work in concert to improve agency performance through the management and use of information and information technology. Its focus is on using architectures to drive agency performance [Hardy, Sep 2008].

The second is the GAO's Enterprise Architecture Management Maturity Framework (EAMMF), which assesses the maturity of an agency's EA program and its management [GAO-03-584G, 2003]. EAMMF outlines steps to achieve a stable and mature process for managing the development, maintenance, and implementation of an EA. Although the EAMMF is prevalent within federal government sectors, it is rarely used within DoD project and system program offices. If it were, it could provide valuable insight and perspectives on how well DoD architectures are being planned, developed and used. It should be considered across all DoD sectors and possibly, be tailored for DoD usage. (This should be a subject of future considerations).

Conclusion

As we move towards a more agile net-centric force and especially within C2 operations characterized by human and organizational issues, the importance and relevance of architectures to deliver the ROI of managing alignment, integration and change becomes clear. Architectures are living electronic documents, evolving over time that, through various analysis techniques, can provide real value to the decision maker. When you use an architecture for value, it becomes **actionable.** Actionable architectures enhance an organizations ability to be agile in reacting to a rapidly changing operational environment.

However, to attain the full value of architectures, agencies need to start with integrated, consistent, and unambiguous architectures, use the iterative development approach, and then continuously apply static and dynamic analytics to gain increasing value. Static architecture data mining helps discover hidden and/or missing rules, gaps, requirements, etc. Static DOTMLPF analysis leads to better definitions of warfighter capabilities. Static visual representations provide a "Management" view for understanding and analyzing complex architectures. While not prevalent today, executable dynamic architecture simulations provide invaluable feedback about the interpretation and implementation of time-based operational needs and allocation of resources, which helps improve the probability of overall mission success. Together, static and dynamic analytics are complementary in helping achieve the architecture value proposition – that *actionable architectures enable business and mission success*. Clear communication of real world cases in which architectures have provided credible and identifiable value to organizational and agency goals is necessary to build interest and support for those who continue to question their effort and worth. The two government architecture measurement instruments, EAAF and EAMMF, should be more prevalent within DoD to assess how well agencies and organizations are planning, developing, and using their architectures. Ultimately, the success of an Enterprise Architecture will be measured by how well it helped an agency meets its business and warfighter goals and objectives in accomplishing a mission.

References

Alberts, D.S., Hayes, R.E., (2004). *Power to the Edge*, CCRP Publication Series, pg 126, 2004

BPMI. (2009). *Business Process Management Initiative*, http://www.bpmi.org/

BPMN-BPEL. (2009). *Business Process Execution Language for Web Services*, v1.1, http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/

CIO, (2001). *A Practical Guide to Federal Enterprise Architecture*. Chief Information Officer Council, v1.0. February 2001.

CCA (1996). *Clinger-Cohen Act (CCA) of 1996: Information Technology Management Reform*,

CCA 5125(b), (1996). *Clinger-Cohen Act of 1996: Information Technology Management* Reform, Sec 5125(b).

Color Petri Nets (CPN). (2009). Retrieved January 15, 2009 from http://www.informatik.uni-hamburg.de/TGI/PetriNets/

Dickens, Lt COL James F., (2005). *Putting the "O" in Joint DOTMLPF Organizational Capabilities for Joint Task Force Command and Control*, U.S. Army War College Research project, 18 March 2005

DoDAF (2007). *DOD Architecture Framework (DoDAF)*, v1.5, Volumes I, II, and III, 23 April 2007

EAAF (2008). *Enterprise Architecture Assessment Framework (EAAF),* v3.0, Dec 2008, http://www.whitehouse.gov/omb/egov/a-2-EAAssessment.html

Executable UML. (Wikipedia) Retrieved January 15, 2009 from http://en.wikipedia.org/wiki/Executable_UML

Fishman, G., (2001). *Discrete-Event Simulation: Modeling, Programming, and Analysis*, Berlin: Springer-Verlag, ISBN 0-387-95160-1

GAO-03-584G, (2003). *A Framework for Assessing and Improving Enterprise Architecture Management* (Version 1.1), April 2003. http://www.gao.gov/new.items/d03584g.pdf

GAO-4-49. (2004). *Government Wide Strategic Planning, Performance Measurement, and Investment Management Can Be Further Improved*, January 2004.

Hardy, Michael, (June, 2008). *Enterprise Architects Talk the Business Talk*, Federal Computer Week, June 30, 2008

Hardy, Michael, (Sep, 2008). *Get Ready to Show Your EA Work*, Federal Computer Week, September 29, 2008

HLA (Wikipedia). *High Level Architecture*, Retrieved January 15, 2009 from http://en.wikipedia.org/wiki/High_Level_Architecture

IBM (2009). Retrieved January 15, 2009 from http://www-01.ibm.com/software/awdtools/systemarchitect/

IDEF0, (1993). *Integration Definition for Function Modeling (IDEF0),* Federal Information Processing Standards Publication 183, 1993 December 21, http://www.idef.com/pdf/idef0.pdf

JCIDS. (2007). *Joint Capabilities Integration and Development System (JCIDS).* Chairman of the Joint Chiefs of Staff Manual (CJCSM) 3170.01C, 1 May 2007.

Metastorm, (2008). *ABM Generation Utility*, v6.1, www.metastorm.com, 2008

PEO 13011, (1996). *The President Executive Order 13011, Federal Information Technology, Federal Register*, Vol. 61, No. 140, July 19, 1996

PMA, (2002). *The President's Management Agenda*, Fiscal Year 2002, Office of Management and Budget, http://www.ssa.gov/pma/

QDR, (2006). *Quadrennial Defense Review Report*, Feb 6, 2006

Pawlowski, T., Barr, P., & Ring, S. J., (2004), *Applying Executable Architectures to Support Dynamic Analysis of C2 Systems*, #113, 2004 Command and Control Research and Technology Symposium, San Diego, CA

Popkin, Jan., (2008). *Enterprise Architecture: a Journey, not a Destination*, Government Computer News, April 4, 2008

Ring, S. J., & Nicholson, D. (2007). *Activity-Based Methodology for Development and Analysis of Integrated DoD Architectures*. in P. Saha (Ed.), Handbook Of Enterprise Systems Architecture In Practice (Chapter 5, pp. 85-133). Hershey, PA: Information Science Reference, ISBN 978-1-59904-189-6

Ring, S.J., Johnson, M. (2005). *State of DoD Architecting, Winter 2005 Command Information Superiority Architectures (CISA),* Worldwide Conference, Omaha, NE, December 1, 2005

Rosen, M., (2009). *Death by Architecture*, Cutter Consortium Enterprise Architecture Practice, 25 Feb 2009

Troux 8. (2009). Retrieved January 15, 2009 from http://www.troux.com/products/troux_8/

UML (OMG). *The Unified Modeling Language™ - UML*, Object Management Group, http://www.uml.org/

Zachman, J. A., (1987*). A Framework for Information Systems Architecture*. IBM System Journal, 26(3). www.zifa.com

Zachman, J., (2001). *You Can't "Cost Justify" an Architecture*, John Zachman International, 2001

Author Biography

STEVEN J RING is an Enterprise Architect supporting U.S. Air Force, U.S. Army, and OSD/NII. He is a Principal Information Engineer at the MITRE Corporation in Bedford, MA. His experience includes engineering management and technical leadership roles in commercial and military product development and integration. He served in the Army and then received his BEE from Cleveland State University and MS in Systems Engineering from Case Institute of Technology. He holds both an Enterprise Architecture Certificate and Chief Information Officer (CIO) Certificate from the National Defense University. He is TOGAF v8 certified. Mr. Ring has focused on applying information and knowledge-based technology in support of interoperability, netcentrincity, and architecture modeling and simulation for DoD programs and systems. He has concentrated on improving and advancing the principles, practices, and governance of DoD architecting to attain value and has contributed in the areas of techniques and methodologies for integrating, analyzing and validating Enterprise Architectures. Mr. Ring co-authored the Activity Based Methodology (ABM), an efficient and effective new paradigm for development and analysis of DoDAF integrated architectures. ABM has been implemented by three major enterprise architecture technology corporations whose products are widely used throughout DoD and government sectors today.