1. Executive Summary

At the December 2011 NDIA SE Division Annual Planning Meeting, several architecture related topics were raised. One of the topics came from the Navy, who asked: "Is the use of authoritative DoDAF-like architectures critical for a successful systems integration effort?"

Subsequently, a small working group was assembled and a number of issues related to DoDAF were discussed, including: 1) industry and others who observed that use of DoDAF in the acquisition and Joint Capability Integration and Development System (JCIDS) process could become disjointed from program technical work and/or performed in duplication with a programs' other systems engineering efforts. If performed improperly it could result in limited utility of the DoDAF data and artifacts, such as traceability of systems engineering artifacts to operational capability requirements; 2) appropriate application of DoDAF in conjunction with bigger picture SE and System of Systems Engineering (SoSE) efforts is often misunderstood; and 3) lack of documented, measurable benefits of a unified architecture framework and systems engineering artifact and document standards and its applicability to future DoD program and mission success. To address these questions, it was determined that we need to increase industry and DoD awareness and understanding of how architecture and DoDAF are an integral part of the systems engineering process to enable model based design, and define a path forward to address possible current architecture and systems engineering governance improvements, including their roles in acquisition, JCIDS, and PPBE. As all these issues are anecdotal, the working group attempted to address them by applying their individual and collective experiences having used the DoDAF for many years to address each issue raised.

A proposal was developed to: a) identify recommendations for adjustments needed to current architecture and systems engineering paths including topics of governance processes, standards, languages, tools, etc.; b) describe how architecture processes as represented by DoDAF should be integrated into larger SoSE efforts, including SoS and capability-based test and evalution and what DoD guidance is needed in this area; and c) recommend priorities for ongoing architecture framework and systems engineering artifact and document standards (e.g., DODD 8270 (still in DRAFT), 8310, DUSD (A&TL) Systems Engineering Guide for Systems of Systems, SECNAVINST 5000.2E, Mil-Std 498 DIDs, IDEAS Group, OMG UPDM, ISO 42010) and innovations (e.g., Ontology working groups, INCOSE MBSE activities, capabilities based acquisition and T&E, DoDAF "fit for purpose" via the DoDAF Meta Model (DM2) ontology).

A significant number of meetings and individual work proceeded to develop this report. However, no significant independent data sources for the usage of DoDAF were found available. Hence, only the results and conclusions that could be derived from work represent the best attempt at determining the utility of DoDAF to date.

In December 2012 at the NDIA SE Division Annual Planning Meeting progress on this task was reported. With changes in the principals who expressed the concerns about DoDAF, these concerns were no longer deemed high priority; hence the task itself was not validated. As the team who had been working this for many months believed that a summary of that work was needed, this paper presents results of the working group's investigation.

The conclusions include:

1. DoDAF has begun to enhance systems integration across the Department by providing a useful means of communications between the Uniformed Services, other US Government Departments and Agencies, and coalition partners;
2. The use of DoDAF will enable greater reuse of architectures and the architectural information for projects throughout the system acquisition process , particularly if that work used model-based systems engineering (MBSE) techniques, processes, and tools to capture the information in a reusable form;
3. Attempts to turn DoDAF into an all-encompassing methodology should be resisted and instead the focus of the effort should be to support existing and new methodologies, such as Systems Modeling Language (SysML), Lifecycle Modeling Language (LML), and others;
4. Continuing work is needed to define effective metrics for architectures and DoDAF;
5. A closer working relationship between the DoD CIO and DASD (SE) offices within OSD needs to be fostered to reduce the potential for duplicative policies and procedures, particularly between the System of Systems and early systems engineering efforts with DoDAF;
6. A survey is needed to quantify the benefits of the DoDAF 2.0 to DoD in the context of providing benefits to the conduct of systems engineering.

2. Background

This section provides an overview of DoDAF 2.02, including its history and structure.

What is DoDAF 2?

DoDAF 2 provides an architecture framework that includes an ontology-based meta model defining all of the key concepts and terms relevant to architecting various perspectives of a problem and solution specific to DoD. The framework is general enough to be used outside of the specific defense domain, but the primary focus of this whitepaper is the DoD application of the framework. As an architecture framework, DoDAF in its current 2.02 form includes the DoDAF 2.0 Meta Model (DM2) as three categories of model re-ification or realization: Conceptual, Logical and Physical.

The DoDAF 2.02 specification has evolved to the point where the framework emphasizes more of the underlying model than it does the specific products that are generated from the underlying model. A set of viewpoints and related products have been defined as part of the baseline, however, the current emphasis is on “fit for purpose” use of the underlying model elements for the use of or presentation to the specific stakeholder in support of their decision making process.

Origin of DoDAF, original purpose and scope

As long ago as 1995, DoD components identified a lack of a single structure to define the components’ or the DoD’s technical infrastructure. Disparate principles, assumptions, and terminology which were in use for developing service/agency architectures, was identified as one of the impediments to interoperable systems. Consequently, a DepSecDef Tasking Memo was issued on October 10, 1995: “…I am directing the acceleration of the development of C4I integration and architecture efforts through the creation of a DoD-wide C4I Integrated product

Team." The result was the publication of a C4ISR Architecture Framework (AF), which defined a "minimal set of rules and guidance that will improve the development, integration, and assessment of C4ISR architectures in context with joint/combined mission operations." This minimal set was derived from previously separate DoD efforts that grew out of the cancellation of MIL-STD-490, 498 and 499 in 1994, in Business Process Re-engineering (BPR) (e.g., IDEF), IT standards (e.g., TAFIM), and systems engineering. The merger of these resulted in the Operational (BPR oriented), Systems View (SE oriented) and Technical (standard oriented). The separation of operational architecture from the systems and technical part was originally intended to better define the roles of the operational requirements community as distinguished from the systems engineering / acquisition community (e.g., Joint staff versys SYSCOMS). One of the major intents was to improve the requirement generation process and communications between the operational and engineering communities. One off shoot of this was the establishment of the JCIDS process in 2003 at the direction of the SECDEF. A DOD Software Engineering office was also established, but only later became elevated to be the DoD Systems Engineering office.

C4ISR was intended to:

- Provide the capability to leverage, compare, contrast, and integrate different architectures
- Make architectures more understandable across the community and improve our capability to create a joint, integrated C4ISR environment
- Enable the use of architectures to improve operational effectiveness, interoperability, efficiency, etc.

Follow on tasking resulting in the publication of DoDAF v1.0 in 2003, which was aimed to expand the scope of architecture description from C4ISR systems to include all DoD systems: "The Department of Defense (DoD) Architecture Framework (DoDAF), Version 1.0, defines a common approach for DoD architecture description development, presentation, and integration for both warfighting operations and business operations and processes. The Framework is intended to ensure that architecture descriptions can be compared and related across organizational boundaries, including Joint and multinational boundaries."

The purpose of the Department of Defense (DoD) Architecture Framework (DoDAF), Version 1.0, was to provide guidance for describing architectures for both warfighting operations and business operations and processes. The Framework provides the guidance, rules, and product descriptions for developing and presenting architecture descriptions that ensure a common denominator for understanding, comparing, and integrating Families of Systems (FOSs), Systems of Systems (SoSs), and interoperating and interacting architectures."

DoDAF v2.0 was published in 2009 and includes the statement: "The Department of Defense Architecture Framework (DoDAF) serves as the overarching, comprehensive framework and conceptual model enabling the development of architectures to facilitate the ability of Department of Defense (DoD) managers at all levels to make key decisions through organized information sharing across the Department, Joint Capability Areas (JCAs), Mission, Component, and Program boundaries. DoD Components are expected to conform to DoDAF in development of architectures within the Department. Conformance ensures that reuse of information,

architecture artifacts, models, and viewpoints can be shared with common understanding.[1]" New and expanded goals have evolved, probably to the point of total over scope or beyond what the framework alone can support. Other branches of DoD and more standards need to be tailored to more specific processes within DoD. In this regard, the DCMO has issued BPR instructions to address standard business processes. The Joint Staff has issued JCAs, JCSFLs, Capability Test Methodology based on Joint Mission Threads, etc. Also, different SYSCOMs and agencies have also created architecture and SE handbooks related to DoDAF and various SE methodologies. More effort is required to synchronize these efforts.

In summary, the scope of this DoD-defined framework has continued to expand. Future DoD plans include defining a Unified Architecture Framework (UAF) that is jointly defined and adopted by NATO, as well as Australia, Sweden, and Switzerland. The objective of the participating countries is to move towards the use of a standard Architecture Framework to help achieve system interoperability.

Systems Engineering as Defined by INCOSE

Systems Engineering (SE) as defined by the INCOSE in the "INCOSE Systems Engineering Handbook v.3.2.2[2]" is defined as

> *"…an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem: operations, cost and schedule, performance, training and support, test, manufacturing, and disposal. SE considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. (INCOSE)"*

The Systems Engineering "V" Model (see Figure 2-1) is used to visualize the systems engineering focus throughout the life cycle stages (as defined by INCOSE) including Concept, Development, Production, Utilization/Support and Retirement. Both the "V" Model and the lifecycle stages create an important context for the discussion of architecting and in particular the role of DoDAF in this end to end systems engineering life cycle.

As indicated in the INCOSE SE Handbook, the life cycle begins with early concept level interaction and studies to understand stakeholder needs. A critical step in these early interactions and studies is the development of a conceptual architecture and design to help explore the stakeholder needs and identify early on and risk factors associated with cost, schedule, performance, technology maturity, etc. The architecture helps the systems engineering team generate early cost and schedule projections that can be assessed throughout the lifecycle.

[1] http://dodcio.defense.gov/dodaf20.aspx

[2] INCOSE, Systems Engineering Handbook, October, 2011, v. 3.2.2

Systems Architecting as Defined by Rechtin & Maier

Systems Architecting as discussed by Rechtin & Maier in their book "The Art of Systems Architecting,"[3] provides the following guidance for systems architecting,

> *"…taking a systems approach means paying close attention to results, the reasons we build a system. Architecture must be grounded in the client's/user's/ customer's purpose. Architecture is not just about the structure of components. One of the essential distinguishing features of architectural design versus other sorts of engineering design is the degree to which architectural design embraces results from the perspective of the client/user/customer. The architect does not assume some particular problem formulation, as "requirements" is fixed. The architect engages in joint exploration, ideally directly with the client/user/customer, of what system attributes will yield results worth paying for."*

[3] Maier, Mark; Rechtin, Eberhardt; "The Art of Systems Architecting"; Third Edition

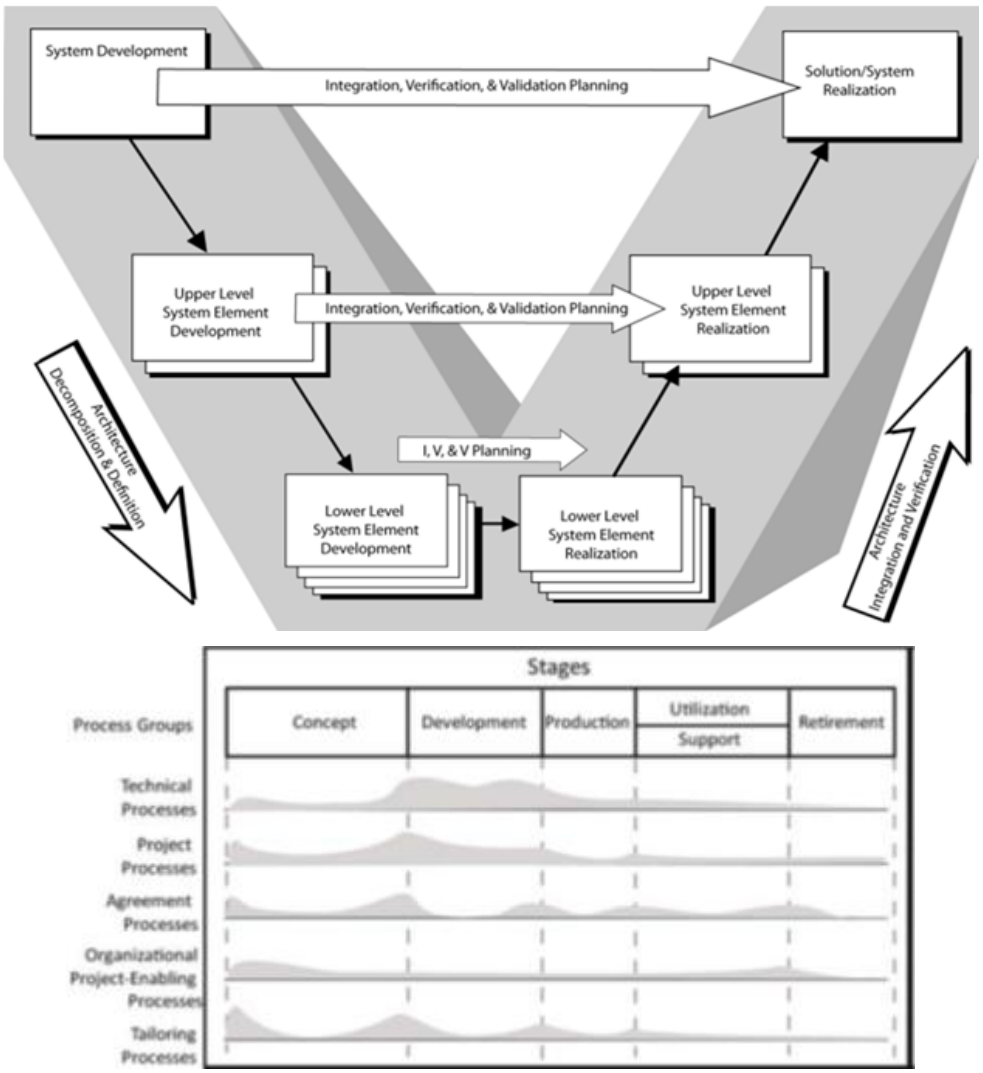

Figure 2-1. Systems Engineering "V" Lifecycle Model from the INCOSE SE Handbook

Systems Architecting can begin with a set of requirements "as a formal characterization of a systems functional and non-functional characteristics, but also includes the process of validating these "constraints" as early as possible. This early validation helps provide the subsequent engineering stages with a better understanding of the customer's intent and needs that are not always adequately represented by a set of textual requirements.

DoDAF structure

Views (levels of abstraction), DoDAF Viewpoints and Models

DoDAF v2.0 defines several viewpoints. Each viewpoint has a particular purpose. However, it should be emphasized that DoDAF is fundamentally about creating a coherent model of the enterprise to enable effective decision-making. DoDAF v2.0 defines the following viewpoints:

- The All Viewpoint describes the overarching aspects of architecture context that relate to all viewpoints.
- The Capability Viewpoint articulates the capability requirements, the delivery timing, and the deployed capability.
- The Data and Information Viewpoint articulates the data relationships and alignment structures in the architecture content for the capability and operational requirements, system engineering processes, and systems and services.
- The Operational Viewpoint includes the operational scenarios, activities, and requirements that support capabilities.
- The Project Viewpoint describes the relationships between operational and capability requirements and the various projects being implemented. The Project Viewpoint also details dependencies among capability and operational requirements, system engineering processes, systems design, and services design within the Defense Acquisition System process. An example is the Vcharts in Chapter 4 of the Defense Acquisition Guide.
- The Services Viewpoint is the design for solutions articulating the Performers, Activities, Services, and their Exchanges, providing for or supporting operational and capability functions.
- The Standards Viewpoint articulates the applicable operational, business, technical, and industry policies, standards, guidance, constraints, and forecasts that apply to capability and operational requirements, system engineering processes, and systems and services.
- The Systems Viewpoint, for Legacy support, is the design for solutions articulating the systems, their composition, interconnectivity, and context providing for or supporting operational and capability functions.[4]

Key elements and relationships - who, what, where, when, and how

DoDAF defines adequate coverage for the six interrogatives: Who, What, Why, When, Where, How, as shown in Figure 2-2.

[4] http://dodcio.defense.gov/dodaf20.aspx

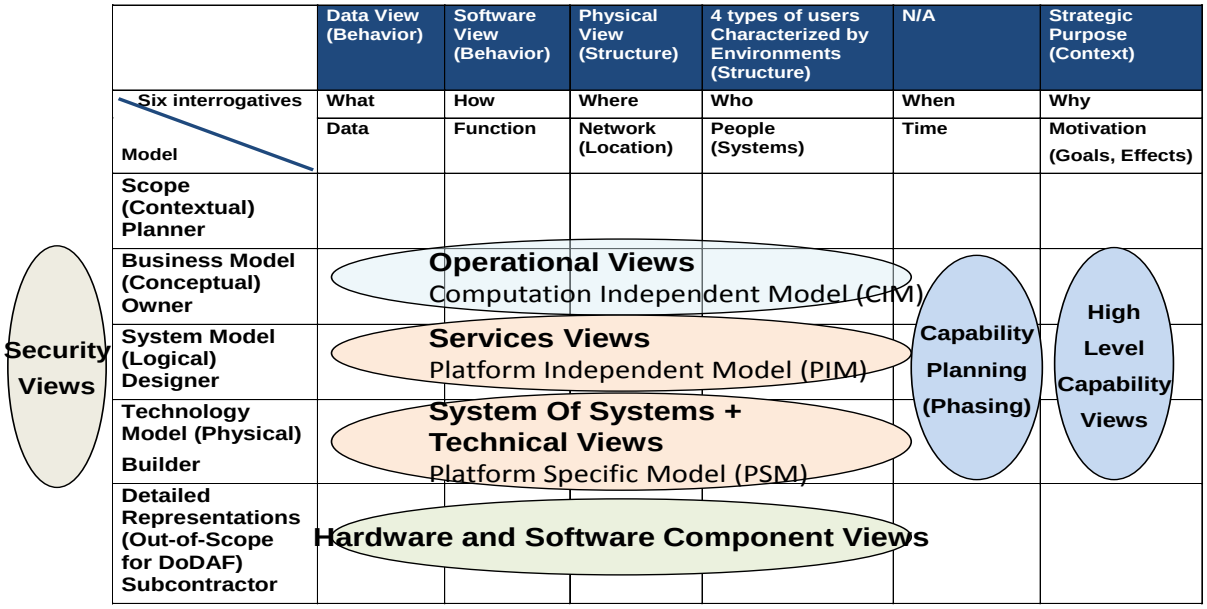

Figure 2-2. DoDAF Views and the Zachman Framework

Currently DoDAF does not have separate security viewpoints, human interface viewpoints, or hardware and software component viewpoints. These do not represent an inadequacy in the DoDAF, as there are as many potential ways to adapt viewpoints for these purposes. Since the goal of DoDAF now is "Fit-for-Purpose" views, then from the information captured by the architecture development process, the user of the information can form as many views as needed to address the concerns of the stakeholders. Continuing to add viewpoints may only add confusion to the vast majority of the stakeholders, many of whom feel overwhelmed by the large number of views already in the DoDAF.

Distinguishing DoDAF from techniques, processes, tools

Over the years, a wide range of techniques, processes, and tools have been developed around the DoDAF as a framework. Each of these approaches have been derived from existing Systems, Software and Hardware engineering techniques, processes, and tools and adapted those to the DoDAF viewpoints. Each follow the guidelines of the DoDAF Viewpoints, but model the structures, relationships, behaviors and constraints in different ways using different ontologies and visualizations. This abundance of approaches to DoDAF has raised some of the key concerns related to sharing and exchanging architecture descriptions, comparing and contrasting architectures and measuring the "goodness" of specific architecture descriptions. Many of the critiques of DoDAF have actually focused more on the tool implementations of DoDAF rather than the current DoDAF specification itself.

Distinguishing DoDAF from tool/language "implementations" of DoDAF e.g. UPDM, IBM SA, etc.

A couple of examples of the tool/language implementation and or standards specifications include the IBM System Architect functional toolset and the OMG based Unified Profile for DoDAF and MODAF (UPDM) standards specification based on a combination of UML (Unified

Modeling Language) and SysML (Systems Modeling Language) modeling language standards. Each of these implementations represents attempts to reduce the DoDAF specification to practice within the context of the systems, software and hardware engineering modeling community standards. Definitely one of the key drivers of the UPDM specification was the need to enable the transition of DoDAF-based architecture descriptions to various engineering language standards to provide a more efficient and reliable translation of DoDAF specifications into systems, software and hardware specification models.

DoDAF was purposely designed to enable views that were developed in these different languages. The original C4ISR Architecture Framework looked at a number of structured analysis techniques (IDEF, Yourdon, etc.) and decided that the information content was most important, not the specific language. With the advent of object-oriented techniques, such as UML and SysML, it became more important that the DoDAF remain language independent. Otherwise, DoDAF would limit its ability to evolve with other emerging standards, such as the Lifecycle Modeling Language (LML) open standard that incorporates both structured analysis and object-oriented analysis visualizations, similar to the emerging software languages that use both types of analysis. DoD should avoid selecting one technique, detailed process, or tool as it will reduce the flexibility of the Government and contractor personnel in providing the best support possible. It would also increase costs to the Department, as the lack of competition always does.

3. What are the perceived benefits from the application of DoDAF?

This section addresses the perceived benefits related to the application of DoDAF in Systems Development projects. The word "perceived" is used only in recognition that the information provided is anecdotal in nature and does not necessarily reflect an unbiased or comprehensive assessment. The context and scope of the use of DoDAF as part of the JCIDS and other DoD core processes, as well as subsequent solution development by the engineering community is established as a baseline. The specific focus of DoDAF is defined and contrasted with broader architecting methodologies that include well defined processes, methods, tools, etc. The distinction between DoDAF as a framework specification and the implementation of the framework in various techniques, languages, tools and environments is also summarized. The specific perceived benefits of DoDAF are itemized and discussed.

The "Context and Scope" for a benefits discussion

The context and scope of DoDAF for the purposes of a benefits discussion should take into account the full lifecycle stages in which various DoDAF viewpoints are used. From a DoD perspective, several processes are relevant including: a) the JCIDS process and the various decision points prior to each of the milestones; and b) the DOTMLPF considerations as part of the acquisition and capability portfolio discussions are directly relevant. JCIDS provides the pre-milestone A, B, C decision making processes focused on the need for specific capabilities given the projected global environment and expected threats. The ability of current defensive and offensive capabilities are assessed and decisions made to proceed to the next level of analysis to

determine if existing or new capabilities are needed to address future operations. Finally, as the DOTMLPF[5] based decisions are made and a materiel solution is deemed necessary the process proceeds to the acquisition and development of specific solution sets through collaboration or government and industry resources. Each of these decision points sets the stage for specific DoDAF benefits discussion.

For example, during pre-Milestone A decision making, what are the benefits of DoDAF Capability and Project Viewpoints; during pre-Milestone B decision making, what are the benefits of operational concepts and the representation of those concepts in the Operational Viewpoints; finally during pre-Milestone C and beyond decision making what are the benefits of the project, systems, services, standards, data and information viewpoints.

DoDAF Perceived Benefits as an Architecture Framework

DoDAF was defined and evolved within in the context of many contemporary architecture frameworks including the RM-ODP (Reference Model of Open Distributed Processing), the Zachman Framework, the FEAF (Federal Enterprise Architecture Framework), the TOGAF (The Open Group Architecture Framework), variants of DoDAF itself including the UK MOD MODAF and the NATO NAF to just name a few. Several successful attempts have been made to map the various DoDAF viewpoints into, for example, the Zachman framework's perspective rows and interrogative columns, the FEAF reference models and the TOGAF Architecture Development Method (see Figure 2-1 above for an example).

A key benefit of the DoDAF is the definition and standardization of key terms, the use of perspectives or viewpoints, the recognition of various architectural stakeholders and their unique interests and views and the recognition of the need to show the linkage or mapping among each of the stakeholder perspectives.

Common Vocabulary, Semantics and Viewpoints

The use of a formal ontology based approach to the DoDAF Meta Model has resulted in a better definition of what had been very amorphous terms, such as Node, Capability, etc. The ontology approach provides a rigorous semantic approach to classifying and defining terms across a relative diverse set of stakeholders. The DM2 also provides mechanisms to extend the scope of the vocabulary using rigorous and consistent methods.

Support for Architecture Model Orientation (Conceptual, Logical, Physical)

As discussed previously, the DM2 defines three levels of abstraction: Conceptual, Logical and Physical). The Conceptual Model is ontology-based and it defines and categorizes each of the DoDAF terms as they are used within the framework context. The Logical model maps these concepts into a form that can be translated into various modeling languages such as UML, SysML, BPMN, IDEF, LML, etc supported by a wide range of toolsets. The Physical Model defines the representation of the Logical Model elements in a form that can be exchanged between architecture modeling tools, registries and repositories.

[5] DOTMLPF stands for Doctrine, Organization, Training, Materiel, Leadership and Education, Personnel and Facilities

Support for DoD Acquisition Phase Decision Making (JCIDS)

Each of the DoDAF Viewpoints and the "fit for purpose" extensibility approach support each of the JCIDS milestone decision making processes through use of the various DoDAF viewpoints: Capability, Operational, Systems, Services, Data & Information, Project and Standards.

Emphasis on Architecture related DOTMLPF Concerns

The DOTMLPF concerns are addressed in each of the DoDAF viewpoints as follows:

- Doctrine – Capability Viewpoint
- Organization – Operational Viewpoint
- Training – Operational, Systems and Services Viewpoints
- Materiel – Systems, Services, Data & Information, Project and Standards Viewpoints
- Leadership – Operational Viewpoint
- Personnel – Operational Viewpoint
- Facilities – Operational and Systems Viewpoints

Emphasis on Operational/Business Concerns

The DoDAF Operational Viewpoint specifically addresses the concerns of the Operational or Business Concerns. Higher level business oriented capabilities and related Information content are addressed in the Capability and Data & Information Viewpoints respectively.

Emphasis on Standards

The DoDAF Standards Viewpoint specifically focuses on the current and forecast technical standards as they apply to each of the other Viewpoint model elements.

Emphasis on Data and Information

The DoDAF Data and Information Viewpoint specifically focuses on the data and information relevant to each of the other viewpoints and provides for definitions, entity/object content and entity/object relationships.

Emphasis on traceability among viewpoint elements (e.g. Capabilities, Operational Activities and System Functions)

Each of the DoDAF Viewpoints includes mapping mechanisms and matrices to ensure completeness in representation across each of the viewpoints. For example, capabilities, operational activities, system functions and services all have mapping matrices associated with them. These mapping tables support the completeness and consistency analysis conducted as part of the architecture description process. The ability to provide traceability is key to using architectures in an MBSE approach.

4. What are the perceived limitations to the application of DoDAF?

The DoDAF application limitations described in this section are derived from a variety of sources including whitepapers, briefings, e-mails, and anecdotal statements at conferences and meetings. Often the issues arise from an incomplete instantiation of DoDAF in a particular tool.

Some of the limitations were identified in a previous 2009 NDIA study "DoDAF Satisfaction of Systems Engineering Needs."[6]

The "Context and Scope" for a limitations discussion

In many cases, managers develop DoDAF products as a result of a directive, instruction or guide that—in effect—mandates DoDAF usage. Thus products are developed to "check off the box." Ideally, we would hope that program or acquisition managers would want to employ architectures as a desired best practice and value-added part of their work to achieve tangible benefits such as improve performance, facilitate interoperability, save resources, and reduce lead times. But, in many instances, DoDAF is not fully used in this manner. Why?

To better understand DoDAF usage, a survey[7] was conducted in 2008 (pre-DODAF 2.0) of 18 organizations, many overseeing multiple projects. The results are shown in Figure 4-1. Albeit a small sample size, the results do indicate that certain DoDAF views have become pervasive such as the OV-1 High-Level Operation Concept graphic, AV-1 Overview and Summary, OV-5 Operational Activity Model (a process flow and hierarchical model), AV-2 Integrated Dictionary (definitions), OV-2/OV-3/SV-6 effectively Information Exchange Requirements (IERs), SV-1/SV-2 system interface/wiring diagrams, and StdV-1 Standards View. Certainly generation of these types of products, within any design framework, make sense. But, not surprisingly, the results indicate that many other types of views are not widely used. It is also not clear how many artifacts were originally developed using non-DoDAF methods that were later tailored to meet a DoDAF "check-off-the-box" requirement. *More recent surveys to provide statistics on DoDAF 2.0, especially related to DM2 usage, have not been conducted, but would be desirable.*

Relationship with Solution Development and System Engineering

There exists a perceived disconnect between enterprise architecture and solution development. The belief is that those who develop architecture frameworks are in one world while system engineers reside in another. There appears to be a lack of understanding in industry of how architecture is an integral part of the systems engineering process to enable model based design. Appropriate application of DoDAF in conjunction with system engineering efforts is often misunderstood, especially in relating DoDAF to widely-used Model-Based System Engineering (MBSE). Historically much work has been done showing the inter-relationships of DoDAF views, but not as much to show how views and data relate to specific systems engineering techniques and acquisition procedures.[8] However, DoDAF is referenced and used as part of the Systems and Software Engineering Defense Acquisition Program Support Methodology, Version 2.0, Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering.

---

[6] "DoDAF Satisfaction of Systems Engineering Needs," Analysis Conclusions, and Recommendations of the Architecture Frameworks Working Group, Systems Engineering Division, National Defense Industrial Association, 6 November 2009

[7] DODAF Product Development Questionnaire Analysis Report and New Product Recommendations Report, Arlington, VA, 5 May 2008 Version 4

[8] For one example of where DoDAF and system engineering have been linked see DoD Architecture Framework – A Guide to Applying System Engineering to Develop Integrated, Executable Architectures, Steven H. Dam, Ph.D., 2006

Development Frequency	DoDAF Product	Project Count
92%	**OV-1**	**35**
84%	**AV-1**	**32**
82%	**OV-5 Node Tree**	**31**
79%	**AV-2**	**30**
76%	**OV-2**	**29**
71%	**OV-5 Activity Model**	**27**
71%	**SV-1**	**27**
68%	**OV-4**	**26**
66%	**OV-3**	**25**
66%	**SV-2**	**25**
63%	**SV-5a**	**24**
58%	**OV-6c**	**22**
58%	**TV1**	**22**
55%	**SV-4a**	**21**
47%	SV-6	18
39%	OV-7	15
37%	TV-2	14
32%	OV-6a	12
29%	Other	11
29%	SV-3	11
29%	SV-5b	11
26%	SV-4b	10
24%	OV-6b	9
24%	SV-10c	9
24%	SV-5c	9
24%	SV-8	9
18%	SV-11	7
11%	SV-10a	4
11%	SV-9	4
8%	SV-10b	3
5%	SV-7	2

Figure 4-1: Survey Results of DODAF Artifact Usage in 2008

The apparent disconnect between architecture and system engineering is evident in examining the front-end of the technical portion of the Defense Acquisition System as shown in Figure 4-2. Connection points between DoDAF and classical system engineering "V" activities are not obvious. This also extends to INCOSE system engineering representations. Figure 4-3 shows an attempt to relate architecture products to the system engineering process, but little documentation exists to describe the details of the relationships. This extends to architectural data that might be useful in a SRR, PDR or CDR. The obvious concern is that architecture and systems engineering can take sometimes duplicative or parallel paths. DoDAF needs to evolve, as does the DoD SE processes, to from tighter linkages between the two; i.e., so that DoDAF architectural activities, views and data constructs can become more relevant to the SE process, and so that the DoD SE

process can better leverage the DoDAF activities, views and data. Additionally, the MBSE industry community needs to define its interrelationship with DoDAF.

In times of shrinking budgets, the challenge is for project and acquisition managers to prudently balance resources between funds spent on architectures models versus solution engineering.

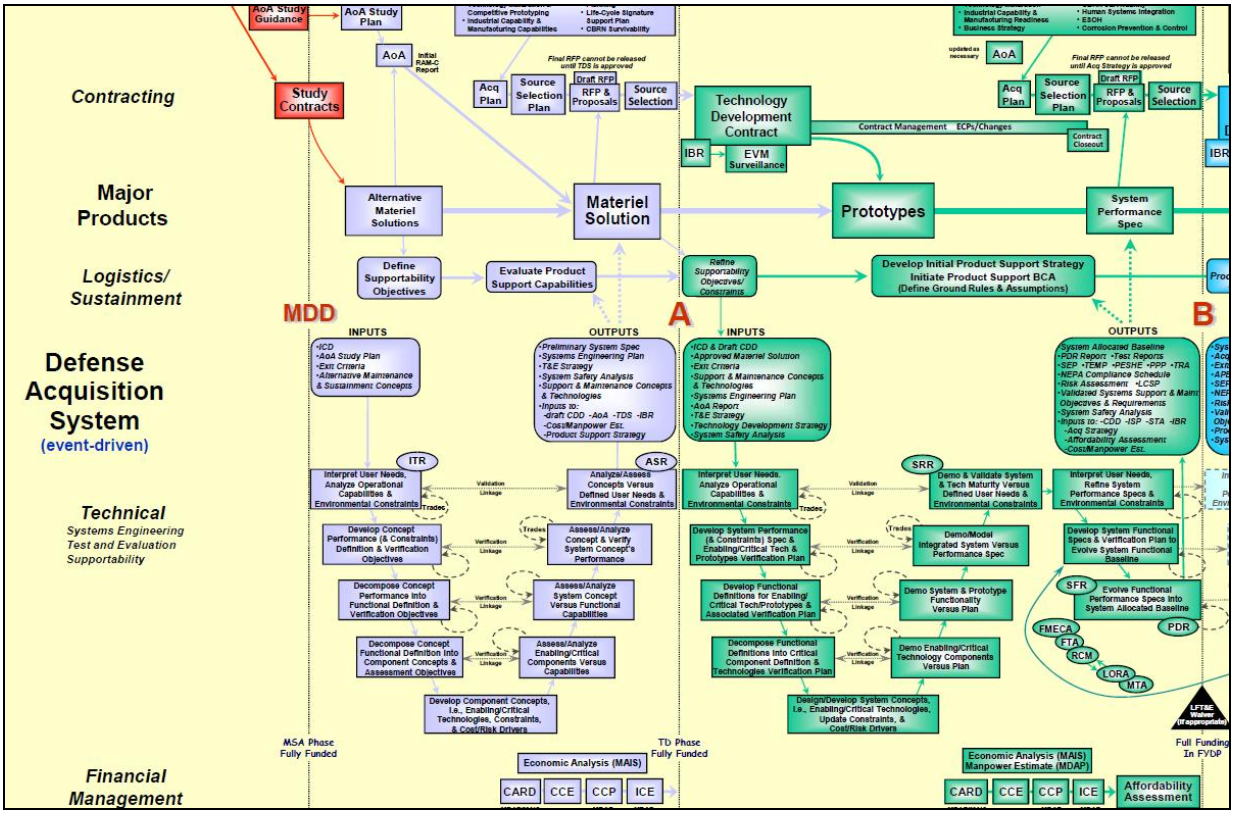

Figure 4-2: Initial Phases of System Engineering Process from the Defense Acquisition University Integrated Defense Acquisition, Technology, & Logistics Life Cycle Management Framework.[9]

[9] DAU "Integrated Defense Acquisition, Technology, & Logistics Life Cycle Management Framework" Version 5.4, 2010.

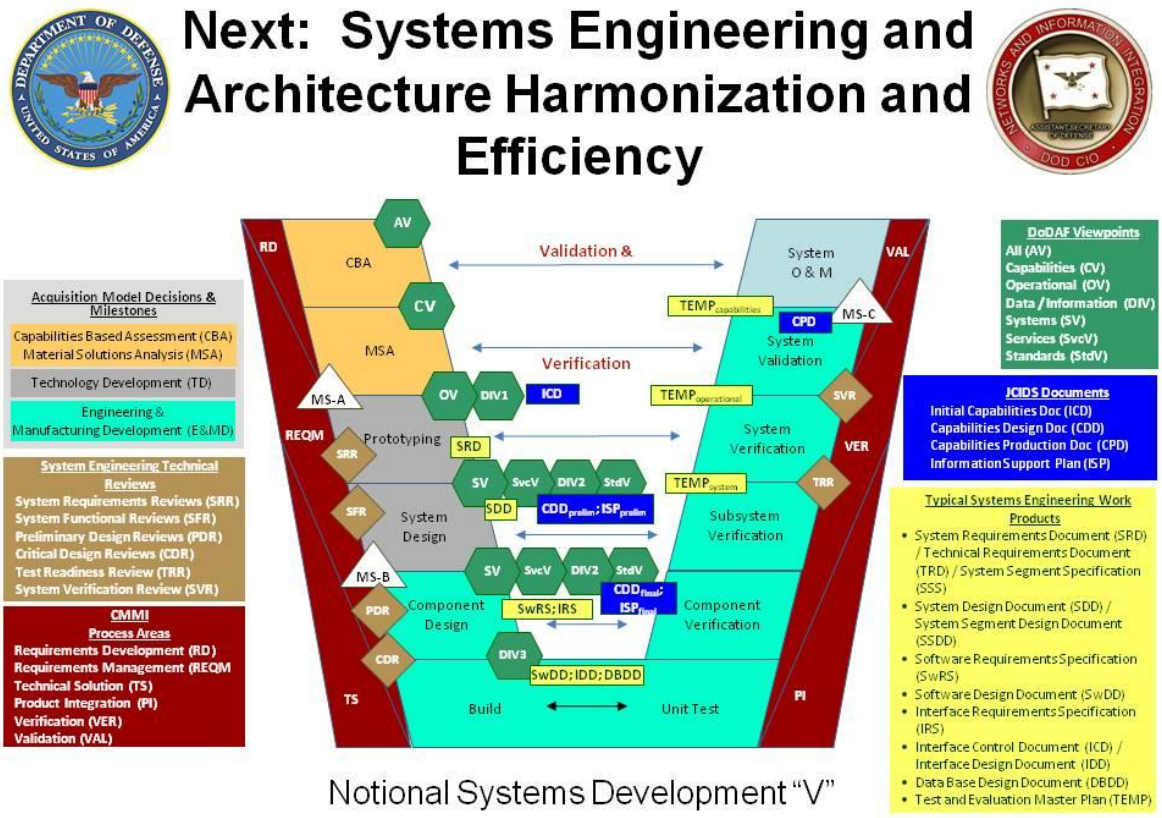

Figure 4-3: Relationship of DoDAF Views to Systems Engineering "V" Process (from January 2012 DoDAF Plenary Conference)

Relationship with JCIDS, Net-Ready KPP, and Requirements Traceability

The CJCSI 6212.01F, "Net Ready Key Performance Parameter (NR KPP)"[10] further tightens the relationship of DoDAF views and data into the JCIDS process with the goal of improved interoperability, supportability, modularity and reuse. Figure 4-4 shows the required, recommended and optional DoDAF products. However, concerns remain:

a. Anywhere from 26 to 33 views are required arguably diminishing "fit for purpose" intent,
b. Unclear as to the specific data being sought from the architecture as required input to the JCIDS process.
c. DoDAF does not have a separate viewpoint for requirements or Use Cases per se.[11]

[10] Chairman Joint Chiefs of Staff Instruction, Net Ready Key Performance Parameter (NR-KPP), CJCSI6212.01F 21 March 2012

[11] Note that "requirements" can exist in a number of the viewpoints. However, if a set of originating requirements in text form is presented there are no specific models and viewpoints for that kind of information. Use cases can be captured in OV-5 and OV-6 views, but neither of these correspond well to the UML Use Case diagram. However, the architect can add them as part of the fit-for purpose tailoring.

Document/ Architecture s	AV-1	AV-2	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	DIV-1	DIV-2 (OV-7)	DIV-3 (SV-11)	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6c	PV-2	SV-1 or SvcV-1	SV-2 or SvcV-2	SV-4 or SvcV-4	SV-5a or SvcV-5	SV-6 or SvcV-6	SV-7 or SVcV-7[6]	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1 [TV-1]	StdV-2 [TV-2]
DCR	R[1]		R	R	R	R						R														R					
CONOPS	R[1]		R	R	R	R		R				R	R		R	R										R					
ICD	X[1]	X	R	R	R	R		R				X	X		X	X	O									R					
CDD	X[1]	X	X	X	X	X	X	X		X		X	X	X	X	X	X		X	X	X	X	X	X	X	X				X[2]	X[2]
CPD	X[1]	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X				X[2]	X[2]
IC[3, 4]	X	X	X	X			X		X	X		X	X	X		X	X	X	X		X	X	X	X	X		X	X	X	X	X
Legend		X – Required O – Optional R- Recommended, PM needs to check with their Component for any additional architectural/regulatory requirements for CDDs, CPDs. (e.g., HQDA requires the SV-10c, USMC requires the SV-3, IC requires the SvcV-10a and SvcV-8)																													
Note 1		The AV-1 must be registered, must be "public" and "released" at the lowest classification level possible in DARS for compliance.																													
Note 2		The technical portion of the StdV-1 and StdV-2 are built using GTG-F DISR standards profiling resources and, within six months of submitting JCIDS documentation, must be current and published for compliance. Use of non-mandated DISR standards in the StdV-1 must be approved by the PM or other duly designated Component cognizant official and documented by a waiver notification provided to the DoD CIO."																													
Note 3		Intelligence Community (IC) requirements IAW the IC Enterprise Architecture Program Architecture Guide and development phase which clarifies the IC Policy Guidance 801.1 Acquisition.																													
Note 4		Service Views (SvcV) only																													
Note 5		1. The Sponsor* and the Program are jointly responsible for the AV-1, AV-2, CV-1, CV-2, CV-3, CV-4, CV-5, CV6, SV-6 or SvcV-7. 2. The Sponsor* is responsible for the development of the architecture data for the OV-1, OV-2, OV-4, OV-5a, OV6c, DIV-2, and the SV-6 or SvcV-6. 3. The Program is responsible for the development of the architecture data for the DIV-1, DIV-3, OV-3, OV-5b, OV-6a, PV-2, SV-1 or SvcV-1, SV-2 or SvcV-2, SV-4 or SvcV-4, SV-5a or SvcV-5, SvcV-10a, SvcV-10b, SvcV-10c, StdV-1, and StdV-2. * Operational user (or representative).																													
Note 6		The NR-KPP Measures data is captured in the SV-7 or the SvcV-7.																													

Figure 2-4: DoDAF Products Required, Recommended or Optional in JCIDS Process (from CJCSI 6212.01F)

Applicability

Despite all the mandates to use DoDAF, one perspective was stated as:

> *"The view that the world needs 'one framework' does not make sense; this is like saying the world (or DoD) needs one programming language. Projects can be equally (or more) successfully using other available approaches"*

An extension to this argument is that having a common language to share information among disparate frameworks and methodologies may be possible.

Methodology

A concern is that governance not emphasized sufficiently in DODAF and that the DoDAF 6-step architecture development methodology may not be sufficient. In earlier versions of DoDAF, this six-step process along with the DM2 constitutes a methodology. While that may be correct by the definition of methodology, these two elements of DoDAF do not provide the in-depth techniques, processes and tools needed to have a complete methodology, nor is it advocated that DoD try to make DoDAF into a complete methodology. Many DoDAF practitioners have incorporated other architecture development methods to complement the DoDAF specification. For example, the TOGAF Architecture Development Method (ADM) is used internationally as a respected architecture standard and complements the DoDAF specification quite well. The interoperability aspect of the DoDAF-based architecture descriptions is independent of the methods used as long as the core baseline set of standard architecture model elements is used within whichever method is chosen.

Data Model & Semantics

Customers, PMs, engineers, and tool vendors have been slow to embrace DM2 and the many additional DODAF 2.0 views, even format though DODAF 2 was released over 3 years ago. Program managers and engineers generally have either not accepted or are slow to accept the new terminology and syntax of DoDAF 2.0. User communities have developed and are using standard architecture elements, many of whose semantics are not apparent in the meta-model. The data-centric focus of DoDAF 2.0 may not be applicable for many programs. Voiced several times at the DoDAF Plenary is a frustration with actually implementing DM2. For some, there is a perceived disconnect between the DM2 and the data needed for system development. The DM2 meta-model semantics differ from the more popular UML/SysML although UPDM is attempting to bridge the gap. The current DM2 v2.02 metamodel has many completeness, consistency, quality issues that are intended to be fixed in v2.03

Updates and Temporal Considerations

Current DoDAF framework methods or tools don't emphasize or easily support the concept of simultaneous existence of "as-is," "to-be," phases in-between, or versioning. A frequent problem is that architectures are not updated after initial development due to maintenance costs. These concerns may be more of a behavior, culture, governance, management or tool limitation issue.

Metrics

There is a lack of measurable benefits for using an architecture framework and related applicability to DoD program and mission success. Metrics do not exist that prove the architecture effort and expense achieved original goals. Metrics are typically not sufficient to perform analysis of alternatives (AoA). *Effective measures for comparing and contrasting architecture descriptions in general and DoDAF description in particular is a critical gap in the architecting community*. Several professional organizations, including INCOSE, NDIA, OMG have taken on the challenge recently and the expectation is that the resulting metrics will be applicable to DoDAF architecture descriptions.

Training

There exists a general disparity between the managers' and users' understanding of DoDAF. This problem is especially true of very high-level DoD officials. Even among architects and engineers there may not be a complete understanding of DoDAF. Certification classes and expanded training will obviously help address this issue, but no certification standard has been developed by OSD/DoD CIO. Certifications are provided by independent organizations, often from Universities, but those courses are not certified by the owners of the DoDAF, so no one knows what these certifications mean.

Executable Architectures

DoDAF 2.0 does not include explicit simulation views. However, the architect can use executable techniques to provide simulation development and execution. DM2 contains time elements, which show at the logical and physical levels of the data model, thus implying that

time is not in DM2. Some of the DoDAF architecture modeling tools include support for the "dynamic" aspects of the DoDAF architecture models. In some cases the dynamic visualization is a simple animation of sequence diagrams and in other cases, the tools provide an export and bridging mechanism into third party simulation tools. This latter feature allows the architecture specification to be translated into a physics based or discrete event simulation of the architecture structure and end to end behaviors. Other tools have simulators built into the tool.

Relationships to other frameworks and methods

In cases where DoD systems interoperate with Federal systems, there is uncertainty about the relationships established between DODAF and the Federal Enterprise Architecture Consolidated Reference Models and Methodologies. *The DoD CIO office is working with the Office of Management and Budget to bring the two frameworks together, an effort that should be encouraged.*

The UPDM specification includes a standardized mapping of the DODAF MetaModel onto both the UML profile and the UML profile extensions defined for SysML. Several tool vendors have implemented the UPDM specification and it is available for use and evaluation.

Tools & Integration

Most enterprise architecture tools don't integrate, which limits reuse of products. The community depends on non-standard, proprietary approaches to link the DoDAF models to more dynamic simulations, animations and 3-D photo realistic renderings of the structure and behavior of the underlying architecture. Certainly UPDM begins to address some of these issues but tool vendors will need financial motivation to fully implement DoDAF and agree on an exchange mechanism.

DODAF Views

The widely used OV-1 view can be too limiting in conveying meaning and understanding of the architecture's purpose, especially showing performance and resource benefits over existing "as-is" systems.

With DoDAF now containing 52 views, it can be daunting for some to understand when, where and how the views are to be applied. Although "fit for purpose" is intended to allow the practitioner to use a fewer number, the JCIDS process seems to mandate up to 26 products. Even given all the views, there may be some that are lacking;

a. Requirement Views
b. Use Case and/or Scenario Views
c. Policy and guidance Views either at a system, enterprise, or governance level.
d. Security Views - Duplication of efforts in areas such as security accreditation and certification which uses separately generated data that could be found in a DoDAF and vice versa.

Obviously a reasonable balance must be struck between addressing architectural content versus creation of an architecture framework so complex that few will want to use it.

Lack of predefined architecture framework templates

The DoDAF 2.02 framework specification does not include domain specific templates for use by the various DoD services or Communities of Interest. However, the framework is flexible enough to accommodate new viewpoints. The use of model libraries, architecture patterns, design templates has been spearheaded by several DoDAF tool vendors. Whether a standard approach for these templates should be developed and included in the DoDAF specification is still under debate and should be coordinated with the OSD offices that own the applicable processes..

Lack of common "sub-domain" viewpoints (e.g. C2, Effects, Communications, Cyber, etc.)

The DoDAF specification was intended as a DoD wide specification and as such was intentionally abstract and general in content. The "fit for purpose" and viewpoint extensibility guidelines, however, should make it relatively easy for specific DoD services and/or Communities of Interest to develop their own standardized viewpoints.

Lack of emphasis on Quality Attributes (e.g. ATAM)

Many DoDAF architecture practitioners incorporate the CMU SEI ATAM (Architecture Tradeoff Analysis Method) as part of their architecture development processes. The Quality Attributes can be mapped as scenarios, tactics or constraints onto the underlying architecture model element within the context of each of the DoDAF viewpoints. However, the DoDAF viewpoints don't readily present the key quality attributes, the trade off, and the tactics used. This information can be embedded and scattered throughout the viewpoints as model elements, but it is hard to get an aggregated view.

5. DoDAF Usage in DoD?

DoDAF belongs to the class of architecture frameworks that provide guidance for developing architecture descriptions, where an architecture description is defined by ISO/IEC/IEEE as: "a work product which models the architecture of a system-of-interest. Architecture descriptions are the "blueprints" used by architects and others for planning, analysis, construction, evolution and coordination throughout the development and operating lifetimes of an enterprise or system." An architecture framework is specified to include: "a) information identifying the architecture framework; b) the identification of one or more concerns (per 5.3); c) the identification of one or more stakeholders having those concerns (per 5.3); d) one or more architecture viewpoints that frame those concerns (per 7); e) any correspondence rules (per 5.7). The verb include when used in this paragraph indicates that either the information is present in the architecture framework or reference to that information is provided therein."[12]

[12] "Systems and software engineering – Architecture Description", ISO/IEC/IEEE 42010:2011, December 2011, http://ieeexplore.ieee.org/xpl/articleDeatilsjsp?stnumnrt=6129467

Similarly, the Zachman Framework[13] defines six perspectives or views: Planner, Owner, Designer, Builder, Subcontractor, and User. The second dimension of Zachman's Framework deals with the six basic questions: what, how, where, who, when and why. The framework does not provide guidance on sequence, process, or implementation, but rather focuses on ensuring that all views are well established, ensuring a complete system regardless of the order in which they were established.

Other established architecture frameworks may describe an architecture methodology such as TOGAF®[14] "A key element of TOGAF is Architecture Development Method (ADM) that specifies a process for developing enterprise architecture". Others still are more concerned with building systems that adhere to a specific system architecture. For example, The Open Distributed Processing – Reference Model (RM-ODP)[15] is based on precise concepts and the use of formal description techniques for specification of the architecture of a distributed system.

Implementation in tools

Proprietary implementations of DODAF abound and cannot be listed here exhaustively. A short list of vendors who offer DoDAF modeling and repository tools (most participated at the last DOD EA conference 2012 include):[16]

Atego, Casewise, Enterprise Elements, Future Tech Systems, Inc., IBM, In2itiv, SPEC Innovations, Metadata Management, No Magic, QualiWare, Software AG, and Sparx.

6. What are the alternatives to DoDAF?

As discussed above, there are a number of other Frameworks that have been developed. Many have detailed methodologies associated with them (e.g., TOGAF). In addition, there are a number of systems engineering techniques (e.g., SysML, LML, etc.) that could form the basis for an alternative to the DoDAF. However, the DoDAF has become an international and USG standard. Its planned transition to the UAF and use in the key DoD policies means that alternatives would be very impractical. Acceptance has grown over time and to adopt some other Framework or technique would be very costly to the Department. The transition from DoDAF to another Framework would also cause great confusion, particularly now amongst allies and other USG Departments and Agencies. Therefore, at this time there does not seem to be a viable

[13] http://www.zachman.com/

[14] http://www.opengroup.org/togaf/

[15] ISO/IEC 10746-1:1998, Information technology – Open Distributed Processing – Reference model: Overview

[16] http://www.dodenterprisearchitecture.org/Pages/default.aspx

alternative to the DoDAF. As such, we need to *recommend better integration at the policy level between the DoDAF and DoD systems engineering*. This recommendation requires greater coordination between the organizations (OSD/DoD CIO and DASD/SE). Mr. Okon, from the CIO's office and who currently manages the DoDAF development, has been participating in the NDIA SE Meetings. Participation by DASD (SE) personnel in the DoDAF Plenary and other meetings would be advisable to enhance this communication.

7. Recommendations

Our recommendations were presented in the previous sections, as identified in italics and summarized here for convenience. The recommendations include:

1. Continue to evolve the DoDAF to support DoD's acquisition policies and enhance systems integration across the Department by using it to communicate between the Uniformed Services, other US Government Departments and Agencies, and coalition partners;

2. Support DoDAF for reuse of architectures and the architectural information for projects throughout the system acquisition process by ensuring that SE policy include model-based systems engineering (MBSE) techniques, processes, and tools to capture the information in a reusable form with DoDAF views as products of MBSE;

3. Do not turn DoDAF into an all-encompassing methodology;

4. Develop effective metrics for architectures and DoDAF; [17]

5. Develop a closer working relationship between the DoD CIO and DASD (SE) offices within OSD to reduce the potential for duplicative policies and procedures, particularly between the System of Systems and early systems engineering efforts with DoDAF;

6. Conduct a survey to quantify the benefits of the DoDAF 2.0 to DoD, in context of providing benefits to the conduct of Systems Engineering.

These recommendations require in-depth analysis and review by the principals of both the CIO and DASD (SE) offices to validate them.

[17] The PSM material entitled, "New Opportunities for Architecture Measurement" written by Ronald S. Carson and Paul Kohl, provides a starting point on this topic, particularly in a table entitled, "Summary of Proposed Architecture Measures and Metrics"

Appendix A. Architecture Working Group Members

The following participants in the working group provided the information for this report.

Barbara Sheeley, The Boeing Company, Architecture Committee Chair

Dr. Steven Dam, SPEC Innovations, Editor and Architecture Committee Co-Chair

Kevin Agee, Army Research Laboratory

Alex Bocast, Silver Bullet Solutions

Bruce Brown, Northrop Grumman

Fatma Dandashi, MITRE

Dave McDaniel, Silver Bullet Solutions

Raschid Muller, DISA

Tom Murphy, Silver Bullet Solutions

John Palmer, The Boeing Company

Ron Williamson, Raytheon

Jack Zavin, OSD/CIO

We also want to thank Mr. Walt Okon for his presentations and support.