

A COMPARISON OF ENTERPRISE ARCHITECTURE FRAMEWORKS

Lise Urbaczewski, Eastern Michigan University, lurbacze@emich.edu
Stevan Mrdalj, Eastern Michigan University, smrdalj@emich.edu

ABSTRACT

An Enterprise Architecture Framework (EAF) maps all of the software development processes within the enterprise and how they relate and interact to fulfill the enterprise's mission. It provides organizations with the ability to understand and analyze weaknesses or inconsistencies to be identified and addressed. There are a number of already established EAF in use today; some of these frameworks were developed for very specific areas, whereas others have broader functionality. This study provides a comparison of several frameworks that can then be used for guidance in the selection of an EAF that meets the needed criteria.

Keywords: Architecture Frameworks, Enterprise Architecture, Comparison

INTRODUCTION

Enterprise architecture serves as the blueprint for the system and the project that develops it. An enterprise architecture framework can describe the underlying infrastructure, thus providing the groundwork for the hardware, software, and networks to work together. According to the Systems & Software Consortium [11], "An Enterprise Architecture relates organizational mission, goals, and objectives to work processes and to the technical or IT infrastructure required to execute them." In addition, a good architecture and its corresponding documentation allow for ease of maintenance in order that the system does not become obsolete before it is even built. There are a number of architectures and architectural frameworks in use today. Though they may overlap or address similar views, frameworks also have been designed to address specific needs or concerns. These frameworks differ by the stakeholders they address and the issues that concern their *world*. These issues or "building blocks" represent methods, common vocabulary, standards [13], and tools that provide a means to implement and integrate the building blocks. In addition, government, commercial, and sub-categories of each of these may require certain protocols to follow. Goethals [6] and Schekkerman [8] provide comprehensive overviews of EAF in the literature. Informal comparisons between specific architectures

have also been made [7]. The Open Group [12] has drawn comparisons between its architectural framework, TOGAF, and existing frameworks. Tang et. al provide an analysis of frameworks at a high level, based on their "goals, inputs and outcomes" [10]. The aim of this paper is to provide a direct comparison of the frameworks, based on their views and aspects. In order to establish a common ground for the framework comparison, we studied several existing enterprise architecture frameworks. Second, we created a method to compare the frameworks based on the perspectives of their stakeholders and abstractions. Third, we then compared the frameworks. From this, we discuss the ways that such comparisons can be used in determining a best-fit of a framework dependent on specific stakeholder needs for a given project.

OVERVIEW OF EXISTING ENTERPRISE ARCHITECTURE FRAMEWORKS

The following are concise descriptions of five EAFs that are used in this comparison.

Zachman Framework for Enterprise Architecture: John Zachman published the Zachman Framework for Enterprise Architecture in 1987 [14] and is considered to be one of the pioneers in this domain [6]. According to Zachman [15], "the increased scope of design and levels of complexity of information systems implementations are forcing the use of some logical construct (or architecture)." The Zachman Framework is based around the principles of classical architecture that establish a common vocabulary and set of perspectives for describing complex enterprise systems. The Zachman Framework has 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 [6]. 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. The Zachman Framework has no explicit compliance rules since it is not a standard written by or for a professional organization. However,

compliance can be assumed if it is used in its entirety and all the relationship rules are followed [9].

Department of Defense Architecture Framework (DoDAF): The Department of Defense Architecture Framework (DoDAF) [2] builds on three sets of “views”: Operational, System, and Technical Standards. A fourth view, ‘All View,’ augments the other views by providing the linkage between the views by means of a dictionary to define terms and by providing context, summary, or overview-level information [3]. This framework provides descriptions of final products as well as guidance and rules for consistency. This ensures a “common denominator for comparing, and integrating Families of Systems, Systems of Systems, and interoperating and interacting architectures” [2].

Federal Enterprise Architecture Framework (FEAF): The Federal Enterprise Architecture Framework was developed and published by the US Federal Chief Information Officers (CIO) Council [5]. Government was following the industry trend of defining architectural frameworks to guide in the development of large, complex systems development. FEAF was in response to the Clinger-Cohen Act, 1996 [1], which required Federal Agency CIOs to develop, maintain, and facilitate integrated systems architectures. The overriding goal of FEAF is to organize and promote sharing of Federal information for the entire Federal Government [6]. The architectural segments are developed individually, within structured guidelines, with each segment considered to be its own enterprise within the Federal Enterprise. We include the *Federal Enterprise Architecture (FEA) - Practical Guide* [4] within our discussion on FEAF because it provides the guidance to U.S. federal agencies for frameworks. FEA allows for flexibility in the use of methods, work products, and tools to be used by the individual federal agencies.

Treasury Enterprise Architecture Framework (TEAF): The Department of the Treasury published the Treasury Enterprise Architecture Framework (TEAF) in July 2000. The Department of the Treasury is comprised of a number of offices that function as individual enterprises. Therefore, its enterprise architecture needs to map the interrelationships among the organizations in order to manage IT resources. The TEAF aims at facilitating “integration, information sharing, and exploitation of common requirements across the department” [6]. Similar to DoDAF, TEAF includes descriptions of work products for documenting and modeling enterprise architectures. TEAF also explicitly states

that these work products align with FEAF models and DoDAF products [6].

The Open Group Architectural Framework (TOGAF): The Open Group Architectural Framework (TOGAF) was first developed in 1995 and was based on the Department of Defense’s Technical Architecture Framework for Information Management [12]. TOGAF focuses on mission-critical business applications that use open systems building blocks. “A key element of TOGAF is Architecture Development Method (ADM) that specifies a process for developing enterprise architecture” [10]. TOGAF explains rules for developing good principles, rather than providing a set of architecture principles. The three levels of principles support decision making across the entire enterprise; provide guidance of IT resources; and support architecture principles for development and implementation.

A PROPOSED METHOD FOR COMPARISON OF EAFs

To compare the existing frameworks, we must first define what an ‘enterprise architecture framework’ is for the sake of our comparison. The definition utilized for this study, as stated earlier, is: A tool that can be used for developing a broad range of architectures for capturing the needed information in an enterprise. The framework also provides a vehicle for accessing, organizing, and displaying that information. We also define two key elements of enterprise architecture as

- A definition of the deliverables that the architecting activity should produce;
- A description of the method by which this is done.

There are many challenges in comparing EAFs. We will list several of them. Some of the frameworks have a very specific scope and therefore are applicable to those applications. There are frameworks that apply to a specific application or development methodology, for example object oriented development or distributed systems. Some frameworks are truly enterprise oriented and some are specific to the development of the IT system only. In order to overcome above challenges, we decided to compare the frameworks based upon their views, their abstractions, and their coverage of the Systems Development Life Cycle.

Comparison by Views and Abstractions

Our study performed both the views and abstractions comparisons, with the views comparison being more quantifiable. We found the Zachman’s views to be the most comprehensive and therefore other EAFs stakeholder perspectives could be represented using the Zachman terminology (Table 1). The planner view includes the concepts for the final product and may encompass items such as the relative size, shape, and basic intent of the final structure. The second view is that of the owner which may represent an architect’s drawings in which the owner agrees that the architect has captured what he has in mind. The designer view is the architect’s final product, whereby he has detailed and described the plans

including materials needed. This represents the plan that will be committed to construction. The fourth view, or that of the ‘builder,’ represents the view in which the architect’s final plans are modified to reflect how construction will proceed. The subcontractor view represents drawings of parts or subsections of the plans. They are considered “an ‘out-of-context’ specification of what actually will be fabricated or assembled. The last view represents the final product, building, or project. It is the physical result. From the standpoint of an information system, this would reflect the users’ view and therefore, the functional enterprise. Though the terminology may differ somewhat amongst frameworks, the views can be represented in this manner.

Table 1. Comparison by Views/Perspectives

Framework	Planner	Owner	Designer	Builder	Subcontractor	User
Zachman	Scope	Business Model	System Model	Technology Model	Detailed Representations	Functioning System
DoDAF	All View	Operational View	Systems View	Technical View		
FEAF	Objectives/Scope Planner’s View	Enterprise Model Owner’s View	Information Systems Model Designer’s View	Technology Model Builder’s View	Detailed Specifications Subcontractor’s View	
TEAF	Planner	Owner	Designer	Builder		
TOGAF		Business Architecture View	Technical Architecture Views			

As noted, the abstractions comparison is less quantifiable (see Table 2). Most of the EAFs studied, provide recommendations for what each of the abstractions represent, but do not provide methods, procedures, or deliverables that are required. All of the EAFs in our study included information in what data was needed and the functionality of the data and system. The frameworks started to differ when comparing the technology and physical aspects of the system, with some providing detailed architectures whereas others were not as specific. In the people abstraction, the frameworks provide the organizational relationships related to implementation of the system. Timelines and justification for the system, as can be represented in the Time/When and Motivation/Why abstractions respectively, were only found in Zachman’s Framework.

DoDAF: The Operations View is the ‘business’ being undertaken by the military. This depicts activities performed as parts of DoD missions, i.e., the exchanges among organizations or personnel, and reveals the requirements for interoperability and capabilities. The Systems View describes the actual existing and future systems that support the DoD and how they are physically interconnected. The Technical View or Technical Standards View adds detail to the Systems View by providing information on standard system parts or components that are currently available off the shelf, either commercially or government, and also provides technical detail and forecasts of standard technology evolution that may apply to the architecture. The DoDAF defines 26 different architecture products that are organized into the four views [3].

Table 2. Comparison by Abstractions

Framework	What	How	Where	Who	When	Why
Zachman	Data	Function	Network	People	Time	Motivation
DoDAF	Data (mission) Logical Data Model	Function / Traceability Functional effectiveness	Physical connectivity plus availability of off- the-shelf solutions	Organizational Relationships		
FEAF	Data Architecture (entities=what)	Applications Architecture (activities = how)	Technology Architecture (locations = where)			
TEAF	Information View	Functional View	Infrastructure View	Organizational View		
TOGAF		Decision-making guidance		IT resource guidance		

FEAF: Currently, the FEAF corresponds to the first three columns of the Zachman Framework and the Spewak Enterprise Architecture Planning (EAP) methodology. The FEAF contains guidance and is oriented toward enterprise architectures as opposed to IT architectures. The rows of the FEAF matrix correspond to the Zachman Framework, but they do not prescribe the approach for developing the products for each of the cells.

TEAF: TEAF can be described in terms relative to the Zachman Framework, i.e., ‘perspectives’ and ‘views.’ The four perspectives are the same as in the Zachman Framework and FEAF matrix with the exception that TEAF combines the ‘Builder’ and ‘Subcontractor’ into one, named ‘Builder.’ Also, the TEAF Information view corresponds to the FEAF Data Architecture; the TEAF Functional view corresponds to the FEAF Applications Architecture; and the TEAF Infrastructure view corresponds to the FEAF Technology Architecture. FEAF does not currently reflect the TEAF Organizational view. This view shows the types of workers and the organizational structures. TEAF allows the flexibility to define additional views and perspectives that focus uniquely on areas important to specific stakeholders.

TOGAF: Strong on the Business Architecture and Technical Architecture aspects. It does not provide as much detail from the planning and maintenance aspects. TOGAF is one of the most comprehensive with regards to the actual process involved. This framework provides guidance towards principles for decision making, guidance of IT resources, and

architecture principles. The framework is gauged towards open systems development.

Comparison with the Systems Development Life Cycle

It is very important to address whether or not the framework encompasses the entire Systems Development Life Cycle (SDLC). The frameworks presented can be compared to the 5 phases commonly used in SDLC models: Planning, Analysis, Design, Implementation, and Maintenance (see Table 3). Overall, the frameworks do not specify HOW the system is to be developed, i.e., tools and work products, and in general are weighted towards planning and analysis, ensuring that all views are addressed. The frameworks provide the guidance that is then implemented in the SDLC.

One might question how the scope/planner and the subcontractor views are incorporated into the SDLC? Those aspects of the framework assist the enterprise in minimizing those risks as outlined earlier, when one may not view the enterprise in its entirety, i.e., designing a system that doesn’t meet the underlying objective. The ‘view’ will determine the requirements necessary in order to be deemed successful. As discussed previously, the frameworks tend to be heavy on the planning, since their objective is more to provide guidance. It was found that most if not all frameworks were weak in addressing the maintenance of an information system.

Table 3. Comparison by SDLC Phases

SDLC Phase/ Framework	Planning	Analysis	Design	Implementation	Maintenance
Zachman	Yes	Yes	Yes	Yes	No
DoDAF	Yes	Yes	Yes	Describes final products	No
FEAF	Yes	Yes	Yes	Yes	Detailed Subcontractor’s View
TEAF	Yes	Owner’s Analysis	Yes	Yes	No
TOGAF		principles that support decision making across enterprise; provide guidance of IT resources; support architecture principles for design and implementation			

CONCLUSIONS

Many of the enterprise architecture frameworks differ in terms of their approach and level of detail. Some are proposed guidelines, whereas others have specific methodologies and aspects to follow. The majority of the frameworks are abstract in that due to their generality of terms, one might then question the validity or the ability to work accurately within that framework. The Zachman framework appears to be the most comprehensive framework of those studied. It uses a number of viewpoints related to the different aspects. Most frameworks only represent a small number of viewpoints and aspects. In this paper we report of an effort to compare EAFs based on their coverage of different viewpoints and aspects. Some of the frameworks do not clearly ‘map’ to the idea of ‘viewpoints’ and “aspects” e.g., the rows and columns of the Zachman framework, therefore making the comparison of the frameworks difficult. The major contribution of this study is a possibility to use it as guidelines in selecting an EAF and determining a best-fit of a framework dependent on specific stakeholder needs for a given project. This is important to minimize risk or failure of an information system development process. Our plan is to expand this research to include more quantifiable measures as well as additional frameworks to better assist in the determination of a framework to meet specific organization needs.

REFERENCES

1. Clinger-Cohen Act of 1996 (1996). Available: www.tricare.osd.mil/imtr/ppm/documents/clingercohen.pdf
2. DoD Architecture Framework (2004). Version 1.0. Volume 1: Definitions and Guidelines.

Available: www.defenselink.mil/nii/doc/DoDAF_v1_Volume_I.pdf

3. DoDAF (2005). DoDAF. Systems & Software Consortium. Available: www.software.org/pub/architecture/dodaf.asp
4. FEA (2001). Federal Enterprise Architecture – Practical Guide, version 1.0, February 2001. Available: https://secure.cio.noaa.gov/hpcc/docita/files/a_practical_guide_to_federal_enterprise_architecture.pdf
5. FEAF (1999). Federal Enterprise Architecture Framework, Version 1.1, September 1999. Available: https://secure.cio.noaa.gov/hpcc/docita/files/federal_enterprise_arch_framework.pdf
6. Goethals, F. (2003). An Overview of Enterprise Architecture Framework Deliverables. May 2003. Available: www.econ.kuleuven.ac.be/leerstoel/sap/FramePage.htm
7. Iyer, B. & Gottlieb, R. (2004). The Four-Domain Architecture: An approach to support enterprise architecture design. *IBM Systems Journal*, 43(3), 587-97.
8. Schekkerman, J. (2003). *How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework*. 2nd edition, ISBN 1-4120-1607-x, 2003, Oxford: Trafford Publishing.
9. Sowa, J. & Zachman, J. (1992). Extending and Formalizing the Framework for Information Systems Architecture, *IBM Systems Journal*, 31(3), 590-616.
10. Tang, A., Han, J., Chen, P. (2004). A Comparative Analysis of Architecture Frameworks, School of Information Technology, Centre for Component Software &

- Enterprise Systems, Swinburne University of Technology, *Technical Report: SUTIT-TR2004.01, CeCSES Centre Report: SUT.CeCSES-TR001, August 25, 2004.*
11. TEAF (2005). TEAF. Systems & Software Consortium. Available: www.software.org/pub/architecture/teaf.asp
 12. The Open Group Architectural Framework (2005). Welcome to TOGAF. Available: www.opengroup.org/architecture/togaf7-doc/arch/
 13. Van den Hoven, J. (2004). Data Architecture Standards for the Effective Enterprise. *Information Systems Management*, 21(3), 61-4.
 14. Zachman, J. (1987). A Framework for Information Systems Architecture, *IBM Systems Journal*, 26(3), IBM Publication G321-5298.
 15. Zachman, J. (1999). A Framework for Information Systems Architecture. *IBM Systems Journal*, 38(2-3), 454-70.