Chapter 5
The Analysis and Documentation Framework

Chapter Overview

Chapter 5 defines and describes the purpose of the EA analysis and documentation framework, provides examples of existing frameworks and discusses the EA³ Cube Framework which is a generalized framework that is suitable for use in public and private sector enterprises.

Learning Objectives

- Understand the purpose of an EA framework as part of an EA implementation methodology.
- Understand how an EA framework establishes the scope of an EA.
- Become familiar with the origin of EA frameworks and several examples.
- Understand the design of the EA³ Cube framework.

Introduction

The foundational elements of an EA program are the analysis and documentation framework (EA framework), and the implementation methodology (EA methodology). The EA framework defines what the EA program will document, and the EA methodology defines how that documentation will be developed and used. By defining what parts of the enterprise are included in the EA, the framework defines the scope of the architecture. The design of the framework communicates the relationship of the areas of the EA that are documented.

Home Architecture Analogy: The EA documentation framework is like the structural skeleton of a home. It is the framing that defines the size and relationship of parts of the house and individual rooms.

Discussion

The EA analysis and documentation process is accomplished through an EA implementation methodology that includes (1) the framework, (2) components, (3) current architectural views, (4) future architectural views, (5) a plan for managing the ongoing transition between these views, and (6) vertical threads that effect the architecture at all levels.

Analysis and documentation, as organized through an EA framework, provides standardized, hierarchical views of the enterprise from an integrated strategy, business, and technology perspective, as is shown in Figure 5-1.
The Origin of Frameworks

Information modeling and documentation frameworks emerged during the era of mainframe computing as data, software, and hardware requirements became more complex and multifaceted, and as the types of end-users increased and their locations became more distant. Reflecting the nature of that era, most early architectures were technically-oriented, often vendor and/or product-specific. Vendors of software and hardware products increasingly touted their own proprietary solutions, standards, and product lines under the banner of information or systems architectures. While this vendor-driven approach to architecture did serve to advance the capability of computing in general, it also created significant incompatibility problems for enterprises that operated a collection of IT products from multiple vendors.

In addition to the issue of product incompatibility, there was a focus on developing and operating individual information systems versus the creation of an overall IT capability within an enterprise. Furthermore, systems-level IT planning grew out of an approach to analysis and design that focused on meeting a specific set of requirements within the enterprise. For example, many enterprises introduced IT in response to a perceived need for automated support for accounting, payroll, and administrative business functions. This often grew to include manufacturing, service, and sales support. In most cases, each of these business requirements were met by individual system solutions based on proprietary vendor approaches and products. The result was a heterogeneous collection of IT resources that independently supported business areas, but could not exchange information outside of a particular system or business area. It was this scenario that gave rise to terms like “stovepipe systems” and “islands of computing capability.”

This scenario was increasingly problematic to enterprises that sought to share information between lines of business and support functions. Further, the duplication in systems capability and cost of operating and maintaining a myriad of independent systems became a focal point for improvement. A desire to create interoperability, reduce costs, and increase capability was the organizational driver that changed things.

During the mid-1970’s and 1980’s this change came in two main areas: database and network design. First, an approach to information systems analysis and design that was based on the enterprise’s information requirements came about through the introduction of standardized methods for modeling data, structure, and process. Second, the era of distributed client-server computing came into being as “dumb” mainframe terminals began to be replaced by “smart”
desktop computers that could be networked in a client-server design that reached throughout an enterprise.

In the first area, an approach to database design, now known as the “structured” approach, was developed for modeling the processing and structure of data. Data Flow Diagramming (DFD) techniques allowed enterprises to identify how an information system would process data in support of a business function. The Entity-Relationship Diagram (ERD) technique allowed analysts to identify the types of data items that an enterprise wanted to collect along with the attributes and relationships of those data items. Through these two analysis methods, enterprises could design more efficient and capable “relational” databases that used procedural programming languages (e.g., COBOL, FORTRAN, C) which were capable of serving multiple information systems and business processes. Further, this shifted the analysis and design focus from proprietary solutions to generic information requirements.

In the second area, the movement from mainframe to distributed computing also served to change the way that information systems and networks were designed. While structured information modeling techniques promoted new relational database designs, networked computing promoted the hosting of these databases in multiple locations on smaller computer “servers” that could be located closer to the end-user. Information systems standards based on international and industry agreements emerged, as did new designs for the hosting and transport of the information. Important examples include the Open Systems Interconnection (OSI) Reference Model for information networks that was proposed by J.D. Day and H. Zimmerman in 1983. This model has seven layers that address services, interfaces, and protocols. In 1974 the Transport Communications Protocol/Internet Protocol (TCP/IP) was proposed by Vinton Cerf and Robert Kahn that led to work on a dedicated data network that connected universities and selected military and business enterprises throughout the nation (known as the ARPANET). The acceptance of TCP/IP on a broad scale in the late 1980’s and early 1990’s promoted the integration of telecommunications and data network infrastructures and provided the catalyst for the dramatic growth of the global Internet. Other standards for data transfer emerged in the late 1980’s and early 1990’s including a standard that defined “Ethernet” local area networks and ‘Asynchronous Transport Mode’ (ATM) networks that promoted integrated voice, data, and video transmission.

Data hosting also changed as developments in computer micro processors, hard drive storage, removable disk storage, and telecommunications interfaces all made the desktop computer, also known as the personal computer (PC), a viable candidate to support print, file, and application hosting functions. Since the early 1980’s the performance capability of PCs has risen dramatically each year, while the cost of PCs has dropped. This dynamic boosted the movement away from mainframe computing to networked computing based on ‘client’ and ‘server’ PCs working together to share data and applications. Standardized approaches to application development began to emerge as a result, along with protracted competitive battles among vendors to develop products that would dominate in the new networked computing environment.

The early 1990’s also saw the introduction of a new approach to designing databases. Focusing on the problem of separating structure from process in modeling relational databases, an “object-oriented” approach was developed that took advantage of new programming languages (i.e., Java, C++) that could support data objects that had attributes and behaviors. Additionally, these data objects could encapsulate (prohibit changes to) certain areas of their code to protect them from alteration. This was significant in that objects then represented reusable code whose quality
in key areas was assured. Finally, the non-encapsulated areas of code offered users the ability to customize or add attributes and behaviors such that objects became a reliable and flexible building block for application and database development.

It was in this time that some of the first writing on information architecture frameworks began to emerge. In 1987 Dennis Mulryan and Richard Nolan wrote about “Undertaking and Architecture Program” \(^\text{13}\) and in 1991 Brandt Allen and Andrew Boynton wrote an article entitled “Information Architecture: In Search of Efficient Flexibility.” \(^\text{14}\) In 1989 and 1992, John Zachman published seminal articles in the IBM Systems Journal about an idea for an Information Systems Architecture (ISA) that served to organize the documentation of information hierarchically and by function.\(^\text{15}\) \(^\text{16}\) Zachman’s work served to elevate the discussion of architectures to the level of the enterprise and stimulated additional writing on enterprise-wide information architectures that was to continue throughout the 1990s. In 1992, Steven Spewak built upon Zachman’s work and developed the concept of ‘Enterprise Architecture Planning’ (EAP).\(^\text{17}\) The EAP method represented a distinct departure from the technically-oriented architectures of previous years, as it focused on how IT would be used to support business functions on an enterprise-wide (enterprise) basis. It is the combined work of John Zachman and Steven Spewak that form the basis of most of the enterprise architecture frameworks that are in use today throughout business and government, including the EA\(^3\) Cube framework introduced in this book.

**Examples of Enterprise Architecture Frameworks**


In the mid-1980’s John Zachman observed that the data processing requirements of many of his IBM clients were becoming more complex. There was a need to show information systems from several perspectives that addressed this complexity and promoted planning, design, and configuration management. Zachman drew from both the aircraft and construction industries in developing a highly intuitive and comprehensive schema for documenting information systems architecture (ISA) in the context of several hierarchical perspectives characteristics. Zachman’s ISA framework is a schema with rows and columns that functions much like a relational database in that he calls for the development of basic or “primitive” architectural artifacts for each of the 30 cells in the schema, such that none of these artifacts are repeated in other cells or combined to create what Zachman calls “composite” products. By documenting the ISA (now known as the Zachman EA Framework) in detail at each level of the EA framework, an enterprise develops multiple views of their processes and resources that are useful to senior executives, line managers, and support staff. Further, Zachman’s approach addresses the what, how, where, who, when, and why questions about an enterprise. Figure 5-2 provides the current version of the Zachman Framework for EA (v3.0).\(^\text{18}\)
Since 1992, John Zachman has gone on to influence a number of different EA frameworks and writings on the EA, including the author’s EA³ framework and this textbook. While the basic ISA approach is evident in the current Zachman EA Framework, many new concepts have been addressed such as how IT security is an implicit element in each cell’s artifact(s). Zachman has written a number of papers that are available through his website on how his approach to EA addresses a number of old and new issues and how it is used in current work with organizations worldwide. See Appendix B (Figure B-11) for mappings of other EA approaches to the Zachman Framework.

The Spewak EA Planning Method (1992)

About the time that John Zachman was releasing his second article to expand the original ISA Framework, Steven Spewak was further extending these ideas into a planning-oriented framework that incorporated new features including a focus on business, an implementation approach that includes principles and values, a migration strategy, and ties to project management. Spewak was the Chief Architect for DHL Systems Inc. at the time of developing his “Enterprise Architecture Planning” (EAP) method. He was also the first person to prominently feature the term “enterprise” in his framework as a way to emphasize the need for architecture to move beyond individual systems planning. Spewak’s definition of the term architecture is as follows:

“Since the aim of EAP is to enable an enterprise to share data, the term enterprise should include all areas that need to share substantial amounts of data. A good and proper scope for enterprise often equates to a business unit, division, or subsidiary because such enterprise units include all of the business functions for providing products and services to customers. Also, with responsibility and control of the bottom line, the economic benefits and justification of EAP can more easily be established”. ¹⁹

Spewak states that EAP is a method for developing the top two levels of Zachman’s Framework. The seven phases of EAP are grouped into a four-layer “wedding cake” shaped model that creates an implementation sequence, as is shown in Figure 5-3.

A generalized framework for EA analysis and documentation is introduced in this book, which can be used in both the public and private sectors. The concepts used in the “EA³ Cube” Framework are founded on the works of Talcott Parsons, James Thompson, John Zachman, Steven Spewak, and the creators of the FEAF. The EA³ Framework employs the generic shape of a cube, to show multiple vertical levels that are different EA documentation areas; multiple layers of depth that are distinct activity areas—referred to as lines-of-business; and multiple sub-cubes at each level that represent plug-and-play EA components.

Enterprises can implement the EA³ Framework directly, or can use it as an initial baseline for the development of their own EA management and documentation approach. Many enterprises will most likely need to modify certain elements of the EA³ Framework to fit their particular needs, which is encouraged as it is recognized that business, government, military, non-profit, and academic enterprises have fundamentally different cultures, economic drivers, and critical success factors. These differences may require adjustments in the framework in order to best implement an EA program that captures the current and future business and technology environment.

Common characteristics of most EA frameworks that the EA³ Framework also captures are that they address multiple, often hierarchical views of the enterprise and technology, and that they support integrated systems planning and implementation. The EA³ Framework serves primarily to organize IT resource planning and documentation activities. The framework is hierarchical to distinguish high-level views that are of value to executives and planners from the more detailed views that are of value to line managers and support staff. Figure 5-4 shows the design and key features of the EA³ Framework.
Hierarchical Levels of the EA³ Cube Framework

The five levels of the EA³ Framework are hierarchical and integrated so that separate sub-architectures are not needed to reflect different levels or functional areas of the enterprise. The architectural areas covered at each level are arranged to position high-level strategic goals at the top, general business services and information flows in the middle, and specific support applications and the network infrastructure at the bottom. In this way alignment can be shown between strategy, information, and technology, which aids planning and decision-making.

- **Goals and Initiatives.** This is the driving force behind the architecture. The top level of the EA³ Framework identifies the strategic direction, goals, and initiatives of the enterprise and provides clear descriptions of the contribution that IT will make in achieving these goals. Strategic planning begins with a clear statement of the enterprise’s purpose and/or mission, complimented by a succinct statement of the vision for success. This is followed by descriptions of the strategic direction the enterprise is taking, scenarios that could occur, as well as the competitive strategy that will ensure not only survivability, but success in terms that the enterprise must define. These overarching statements are then supported through the identification of goals and supporting initiatives that include measurable outcomes and performance measures.

- **Products and Services.** This is the architecture’s intended area of primary influence. The second level of the EA³ Framework identifies the business products services of the enterprise and the contribution of technology to support those processes. The term ‘business service’ is used to mean processes and procedures that accomplish the mission and purpose of the enterprise, whether that is to compete in the private sector, provide public services, educate, provide medical services, or provide a defense capability. Strategic planning helps to direct and prioritize the various business services and product delivery activities in an enterprise to ensure that they are collectively moving the enterprise in the strategic direction that is set out in the Strategic Plan. Business services then need to be modeled in their current state and if change is anticipated, also modeled in the envisioned future state. Business services and product delivery processes should be eliminated if they are not adding sufficient value to the enterprise’s strategic goals and initiatives. Business services and product delivery activities should be modified if change can increase value to the enterprise, be it a minor adjustment or a major shift in how that activity is accomplished.

Technology is often a key enabling element in increasing value, but should not be the driving factor in the reengineering or improvement of business services and product delivery processes. It is important to review and adjust the process before IT is applied to ensure that optimal value and efficiency are achieved.

- **Data and Information.** Optimizing data and information exchanges is the secondary purpose of the architecture. The third level of the EA³ Framework is intended to document how information is currently being used by the enterprise and how future information flows
would look. This level can be reflected through an IT Strategy document that ties into the enterprise’s Strategic Plan and/or Business Plan. The purpose of the IT strategy is to establish a high-level approach for gathering, storing, transforming, and disseminating information throughout the enterprise. The use of concepts such as knowledge management, data mining, information warehouses, data marts, and web portals can be organized through the IT strategy. The design and functioning of databases throughout the enterprise are also documented at this level as are standards and formats for data, data dictionaries, and repositories for reusable information objects.

**Systems and Applications.** The fourth level of the EA³ Framework is intended to organize and document the current group of information systems, and applications that the enterprise uses to deliver IT capabilities. Depending on changes at the upper levels of the EA³ framework (Business services or Information Flows) there may be planned changes to systems/applications that must be reflected in the architecture’s future views. This area of the EA³ framework is also where components are a prominent feature in service-oriented architectures, as increasingly interoperable commercial applications are available to enterprises (e.g., J2EE and .NET industry standards). Large, modular applications can handle entire lines of business and/or back office functions (i.e., financial systems, manufacturing control systems, and supply chain management systems). Often referred to as Enterprise Resource Planning (ERP) systems, these commercial applications may offer modules of functionality that can be customized to allow an enterprise to reduce the overall number of applications that they operate and maintain. While ERP systems rarely provide all of the functionality that an enterprise needs for business functions and administrative support, this modular approach is reflective of a “plug-and-play” strategy that enterprises can adopt at this level of the EA³ Framework to increase interoperability and reduce costs.

**Networks and Infrastructure.** This is the connectivity grid of the architecture, the host environment for applications and systems. The fifth and bottom level of the EA³ Framework is intended to organize and document current and future views of the voice, data, and video networks that the enterprise uses to host systems, applications, websites, and databases. This level also documents the infrastructure of the enterprise (e.g. buildings, server rooms, capital equipment). Local Area Networks (LANs), Wide Area Networks (WANs), System Application Networks (SANs), Intranets, Extranets, Wireless Networks, Mobile Networks, and Computing Clouds are documented at this level so that efficient designs can be implemented through the future architecture that reduce duplication, increase cost and performance efficiency, and promote availability and survivability. Often, an enterprise will determine that certain IT capabilities are critical to the success of the enterprise, and in these areas the architecture should reflect redundant resources in different locations such that these capabilities could continue to be available if the primary resource became unavailable.

**Lines of Business within the EA3 Cube Framework**

A Line of Business (LOB) is a distinct area of activity within the enterprise. LOB can also be referred to as ‘vertical’ mission areas. It may involve provision of services, product development/delivery, or internal administrative functions. Each LOB has a complete
architecture that includes all five hierarchical levels of the EA$^3$ framework. The LOB therefore can in some ways stand alone architecturally within the enterprise, except that duplication in data, applications, and network functions would occur if each LOB were truly independent, and crosscutting activities that reduce this duplication would not be represented. There may be cases where an enterprise would want to incrementally develop their EA due to cost or other considerations, and architecting individual LOBs is one way to do this. The LOB architectures then must be tied together so that the EA correctly represents the entire enterprise, which is what is needed for the EA to be of maximum value to executives, management, and staff.

**Crosscutting Components within the EA3 Cube Framework**

To avoid the inefficiencies of duplicative support within LOBs, crosscutting business and technology components are established to provide common service and product delivery capabilities, databases, application suites, and network infrastructures. Crosscutting services are aimed at reducing application hosting costs, increasing the sharing of information, and enabling enterprise-wide infrastructure solutions. Examples of crosscutting initiatives include email service, administrative services, telephone service, video teleconferencing facilities, and computer server rooms.

**Planning Threads within the EA3 Cube Framework**

EA documentation includes “threads” of common activity that pervade all levels of the framework. These threads include security, standards, and workforce considerations.

- **Security.** Security is most effective when it is an integral part of the EA management program and documentation methodology. A comprehensive security and privacy program has several focal areas including: information, personnel, operations, and facilities. To be effective, IT security must work across all levels of the EA framework and within all of the EA components. Chapter 11 provides more.

- **Standards.** One of the most important functions of the EA is that it provides technology-related standards at all levels of the EA framework. The EA should draw on accepted international, National, and industry standards in order to promote the use of non-proprietary commercial solutions in EA components. This in turn enhances the integration of EA components, as well as better supporting the switch-out of components when needed.

- **Skills.** One of the greatest resources that an enterprise has is its people. It is therefore important to ensure that staffing, skill, and training requirements are identified at each level of the EA framework, and appropriate solutions are reflected in the future architecture. A Workforce Plan (Human Capital Plan) is perhaps the best way to articulate how human capital will be employed in enabling technology capabilities, which underlie business services and information flows.

**Summary of Concepts**

This chapter described how an EA framework is one of the foundational elements of an EA program and implementation methodology. The EA framework establishes the scope of the EA
documentation effort, and relates the areas of the architecture together. EA frameworks were first
developed in the 1980’s and have evolved in the public and private sectors, as well as
internationally to provide support for particular approaches to EA. The EA³ Cube Framework
was described in detail, as part of an overall EA methodology. Chapters 6 and 7 will provide
information on how to develop current and future views of EA documentation using this
framework.

Chapter 5 Questions and Exercises

1. Why does an EA implementation methodology begin with the selection of an EA
   framework?

2. What are some of the basic characteristics of an EA framework?

3. Why are hierarchical levels of an EA framework helpful in documenting an enterprise?

4. Why is it necessary to show current and future views of EA documentation?

5. How does the Spewak Enterprise Architecture Planning approach relate to the Zachman EA
   framework?

6. Would the Federal EA Framework be useable in private sector (business) enterprises? If so,
   how? If not, why?

7. Choose a medium or large size enterprise and provide the following regarding the areas of
   the EA³ Cube framework:
   a. List examples of documentation from the enterprise that would be appropriate at each
      of the five functional levels.
   b. List examples of documentation from the enterprise that would be appropriate for the
      three common planning threads.
   c. List examples of documentation from the enterprise that would illustrate Lines of
      Business.
   d. List examples of documentation from the enterprise that would illustrate crosscutting
      and vertical EA components.