ENTERPRISE ARCHITECTURE:

Lessons from Classical Architecture

Journal of Enterprise Architecture 2:2, pp. 20-27, May 2006

ABSTRACT

The main lesson that results from looking at Enterprise Architecture (EA) through the lens of classical architecture is that EA development should be executed in two sequential stages: an architectural stage followed by an engineering stage. The architectural stage begins with a general need and synthesizes distant to-be alternatives, one of which is chosen by the "client." This selected to-be architecture provides a stable specification for engineering development. The engineering stage begins with the chosen to-be architecture and produces an optimized to-be enterprise design along with a staged transition plan. The transition plan migrates between as-is and to-be enterprise design states (not architectures). A robust to-be architecture does not change during the EA life cycle. The enterprise engineering design changes in response to new technology and requirements creep. Staging EA development in this manner enables early management buy-in and reduces risk by providing stable intermediate milestones. While the architecture stage is low cost, its success demands quality execution and is best executed with special expert teams (Pavlak, 2005). – In addition to the main lesson, private sector experience suggests that EA's greatest value to the enterprise will come from Information Technology (IT) enabling the elimination of unnecessary processes. High value process driven architecture should be initiated by the organization's strategic planning function.

INTRODUCTION

Classical architecture reflects thousands of years of history in several mature domains. The definitions, models and processes are well established and reflect a good deal of experience and wisdom.

In contrast, EA is an evolving discipline; today's practice reflects its information systems origins. There is a tendency to look at EA from the perspective of IT. There is confusion over basic questions such as: What is the purpose of EA? What is the difference between architecture and engineering? What comes first, the to-be of the as-is? Is the to-be stable or variable?

The intellectual basis for this paper is traditional systems architecture (Rechtin, 1991). This paper shows that EA closely corresponds to classical architecture as reflected through Rechtin. There is little need for new terms and special processes. This is particularly true if we follow tradition and split EA as currently practiced into an architectural stage and an engineering stage.

The correspondences between EA and system architecture works well because the enterprise itself is a system; by Rechtin's definition a complex assembly that performs a function greater than the simple sum of the parts. Note that this definition is broader than the EA convention of defining a system as an information system.

Other views of classical architecture such as Alexander's patterns (Alexander 1964) are also illuminating. However, at the current state of EA development Rechtin's system perspective is most useful in addressing basic questions.

THE PRIMARY IMPORTANCE OF PROCESSES

We begin with a most important message from private sector case studies: *the big performance gains come from reengineered business processes,* that is, new processes made feasible by Information Technology. Simply automating existing processes is usually disappointing.

For example, in the mid 1980's Ford automated accounts payable and reduced the department size from 500 to 400 people. At the same time Mazda was performing the same task with 5 people. The difference is that Mazda employed new processes that took advantage of IT. Ford was automating processes that evolved under a paper-based system (Hammer & Champy, 2003 p. 43).

Case after case reinforces the same lesson. Wal-Mart spends an extraordinary effort

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

clarifying essential value-added processes and eliminating unnecessary busy work. Over the past 10 years modern corporations have evolved from functional organizational structures into flat process-based structures that rely heavily on process teams.

Automating existing processes with information technology is analogous to "paving old cow paths," improving productivity by 10-30%. On the other hand, IT-enabled reengineering can improve productivity by 500 – 1000% (Hammer & Champy 2003, p. 51). Huge productivity gain comes from eliminating steps that are made unnecessary by information systems.

We conclude that both business process and information systems need to be viewed as having equal importance when creating to-be structure of the enterprise.

EA RELATIONSHIPS

It is appropriate to begin by defining terms. Since terms are most useful if they are defined in the context of their use, every effort has been made to keep these terms consistent with classical architecture. Creating non-traditional definitions causes considerable confusion.

- **$Architecture 1:** the structure of components and their relationships.
- **$Architecture 2:** the practice of creating architecture. Starts with a general need, produces an actionable specification.
- **$Engineering:** starts with a specification, deduces the design of a product/process that optimally satisfies the specification.
- **$Business process:** an activity that creates an output by adding value to an input. The processes can be administrative or mission-specific.
- **$Enterprise:** an organizational unit defined by a clear purpose.
- **$Enterprise Architecture (EA):** the architecture of the enterprise, the structure of the relationship between business processes and information systems.
- **$System:** any organized assembly of resources and procedures united and regulated by interaction or interdependence to accomplish a set of specific functions. (DoDAF, 2003, p. 3-3)

The purpose of an organization is typically expressed in mission/vision statements. The organization's senior managers then create a strategic plan to achieve this purpose. This business strategy defines the scope of the enterprise as illustrated in Figure 1.

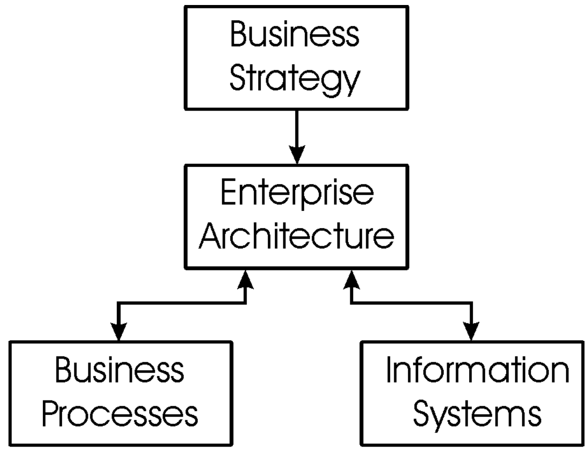

Figure 1. EA relationships

Private sector experience suggests that EA should place processes and information systems at a level of equal importance. Business Strategy requirements can be satisfied in a number of different ways. There is more than one set of business processes that can execute the business strategy, and some of these processes would be easier to automate than others. Likewise, there is more than one information systems configuration that can automate each business process option. This paper posits that a primary purpose of EA is to understand this relationship and develop appropriate combinations of business processes and information systems.

IT ALIGNMENT

Early efforts to control IT cost aligned IT investments with business needs. The resulting relationships are illustrated in figure 2. Existing business processes are presented as requirements for the development of information systems. A good definition of this relationship is Enterprise Wide Information Systems Architecture (EWISA).

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

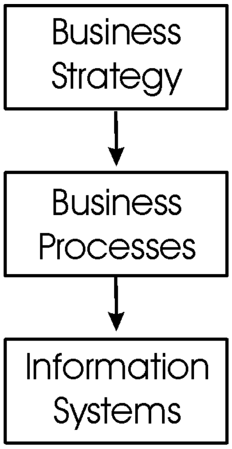

Figure 2. Information systems architecture relationships

EWISA relationships reflect the IT origins of EA. Looking up at the enterprise from the perspective of IT, one expects the organization to provide requirements.

EWISA achieves efficiency gains through commonality and standardization but does not realize the large gains resulting from IT-enabled reengineering of business processes.

While EWISA is a rational intermediate development stage it is necessary for EA to evolve beyond EWISA to the relationships illustrated in Figure 1.

THE BASIC DEVELOPMENT WATERFALL

Classical architecture can be described as a problem solving approach for ill-defined problems; problems that have more than one feasible solution because the goal is defined in general terms. The archetype is a client who wants a “dream house.” As the client is often unclear about exactly what is a dream house, there is more than one feasible solution. Only the client can decide which is satisfactory. The architect’s challenge is to create feasible structures that have the potential to satisfy the needs of a specific client.

The solution approach is hypothesis testing. The architect explores the range of feasible solutions. For the homeowner client, the architect prepares models, sketches, rough cost estimates, and presents alternatives to the client for selection. The architect’s presentation contains no more detail than necessary to clarify the fundamental choice.

The client then chooses a concept. Each alternative concept has a different mix of cost, performance, schedule and risk. Each alternative is feasible. Client satisfaction is a value choice. There is no optimum, no objective basis for choosing one or another. Any one of the alternatives is a feasible solution.

This basic development model is a waterfall (Figure 3), variations of which are presented in most systems engineering textbooks and Rechtin. In military systems client choice is a “concept decision” that kicks off system development (DoDI 5000.2, 2003). Selecting architecture, the client choice, is a major milestone that changes the character of the project.

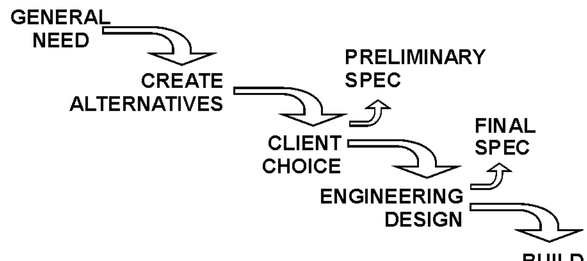

Figure 3. Basic development waterfall

Prior to client choice, the project is all about alternatives, potentials and possibilities, fundamental structure, form, balancing, needs. Client choice, the selection of architecture, results in a unique structure, a preliminary specification that begins an engineering effort. Engineering begins with the preliminary specification and is all about the design and optimization of a product that can be built. The end result of engineering design is a final specification.

Managing the architectural stage is substantially different than managing engineering design. Engineering design tasks can be partitioned and employ thousands of workers. In contrast, creating to-be alternatives requires a unified vision, a holistic view of the whole concept. This is the province of small teams. The best way to create to-be architectures is to construct special expert teams (Pavlak, 2005).

Skill sets are another reason for separating the architectural stage from engineering design. The architectural stage requires inductive reasoning; think outside the box. The engineering design stage requires deductive analysis, attention to detail. At high skill levels, these two skill sets are generally incompatible and the different stages are likely to need different people.

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

EA DEVELOPMENT WATERFALL

The central postulate of this paper is that the EA development waterfall (Figure 4) should follow the same sequence as the basic development waterfall. For EA the general need is the business strategy; there is more than one way to execute a strategic plan. The problem has more than one feasible set of business processes and more than one feasible information system. The architectural challenge is to determine what structure the client finds to be most satisfactory.

As in basic development, the architect explores the range of structural options. Typically there will be a small number of fundamental design drivers that dominate the analysis. The architect characterizes each concept to illuminate the fundamental choices. This is a high level characterization. The level of detail should be sufficient to specify each so there is no confusion between alternatives. This specification includes requirements; rough estimates of cost, performance, schedule and risk. Too much detail during the architectural stage obscures the client's fundamental choices.

As in basic development, client choice is a major project milestone. The selected concept is a "to-be" architecture defined in sufficient detail to initiate engineering design.

Client choice initiates the high manpower engineering design stage. By selecting a to-be architecture, the client is buying into the project. The client is committing to certain levels of cost, performance, schedule and risk.

WHO IS THE CLIENT?

In our archetype, the homeowner, the person with the checkbook, makes the client choice. For EA, the client is the person or the group responsible for achieving the purpose that defines the enterprise.

For certain situations such as a large politically sensitive organization, the client can be constructed for the purpose of selecting the to-be. In this case the client could consist of major stakeholders. Encouraging such a group to reach a consensus requires some attention.

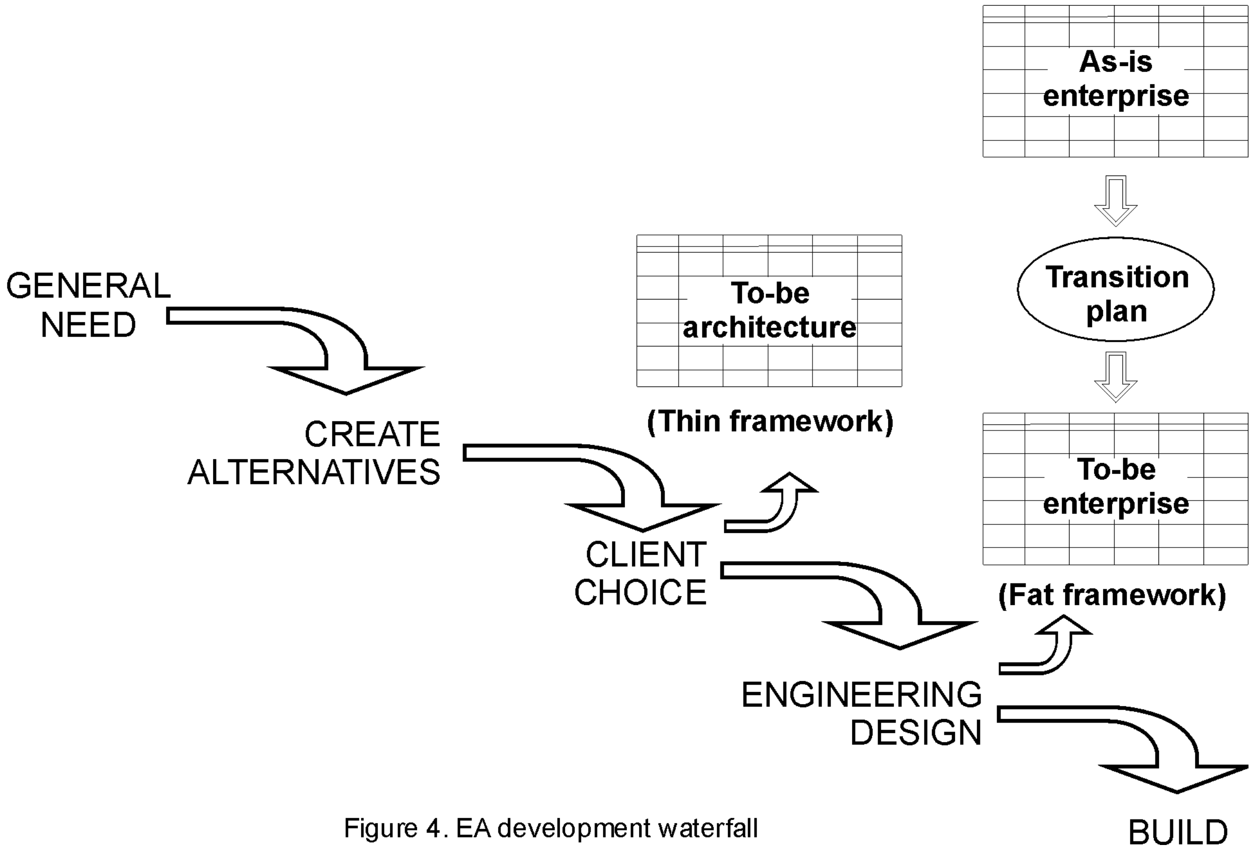

Figure 4. EA development waterfall

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

FRAMEWORKS

Various frameworks have been developed to provide a common basis for describing enterprise architectures. The Zachman framework (O'Rourke, 2003) consists of data artifacts organized in a way that helps the architect think about structure in a disciplined manner. Another well-developed framework is the Department of Defense Architectural Framework (DoDAF 2004).

The view posited in this paper is that frameworks are analogous to "drafting standards" in classical architecture. The framework is not the structure of the enterprise but a method for expressing the structure.

In classical architecture, the client chooses architecture on the basis of sketches and models of the building and its important features. For client choice, the architect will focus on what is important to the client and clarify distinctions between alternative structures. Too much detail is counterproductive as it obscures fundamental distinctions. At this stage there is no need for details such as wiring diagrams.

This paper suggests Thin framework, Figure 4, as the enterprise architecture analog to classic architecture sketches. Thin framework includes only those framework elements necessary to characterize fundamental structure – to distinguish between options and support rough estimates.

Using the Zachman framework, Thin Framework to-be architectures would consist of a complete description of Scope, and partial high-level descriptions of the Business and Information Systems models. The Technology Model, Detailed Representation and Functional System contain more detail than necessary to describe fundamental structure.

This paper suggests Fat framework, Figure 4, as the enterprise architecture analog to detailed engineering design description. The "Fat framework;" is a set of framework elements with sufficient detail to document the design. (Since the design has architecture, documenting the design also documents the architecture.)

ENGINEERING DESIGN STAGE

In classical architecture, client choice produces a specification that initiates engineering design. For a new home, engineering design consists of code compliant detailed drawings and a list of materials for builder quote. For a new bridge or skyscraper, engineering design consists of detailed structural analysis, authorized working drawings, detailed drawings, schedules and materials specification.

As in classical architecture, the enterprise engineering design stage begins with a preliminary specification, the client selected to-be architecture. The purpose of engineering design is to prepare a detailed description of the to-be enterprise and a transition plan that migrates from the as-is to the to-be. Based on this engineering design, the client can purchase information systems and begin to migrate to new business processes.

In Figure 4 the transition plan migrates between an as-is and to-be enterprise states (not architecture). The transition that needs to occur is to migrate from one enterprise design state to another. An enterprise design description contains more information (e.g. interface specs) than an architectural description of fundamental structure. It is not clear how to migrate from one architectural state to another without including more information than necessary to characterize fundamental structure.

The risk associated with an enterprise architecture project is highly contingent on the quality of the transition plan. With a distant to-be architecture as a goal, a simple gap analysis may not be sufficient. The gap may be too large for simple transformations. It is not necessary to proceed in one fell swoop. Indeed, the first stage in most engineering design programs is critical item development. For EA, this could include pilot programs to verify business process models and high-risk aspects of the information system. A transition plan can proceed in a sequence of steps with stable intermediate stages. At these intermediate milestones the client can judge progress and change emphasis or even terminate the project. Intermediate milestones substantially reduce the risk of migrating to a to-be enterprise that is substantially different than the as-is enterprise.

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

TO-BE ARCHITECTURE AS A STABLE GOAL

In most classical architecture domains, once the client chooses an alternative, the selected architecture does not change during the project life cycle. Homes, bridges, buildings, weapon systems, chemical plants - once the client chooses a concept, the development waterfall proceeds into engineering design and there is no feedback. (Spiral development occurs within engineering design.) There are exceptions of course but these exceptions tend to be costly, a failed architecture.

In contrast, engineering design does change and evolve with changing requirements and improved technology. During its life cycle, an aircraft like the air force B-52 bomber will undergo endless engineering design upgrades to engines, avionics, sensors, and weapons. However, its basic form - airframe, payload, range - is stable. This paper posits that this stable underlying form is the architecture of the aircraft. Indeed a “robust” architecture can be defined as one that can tolerate substantial engineering design changes without changing its basic form.

While classical architecture suggests that the to-be enterprise architecture should be defined in a way that it is stable during its life cycle, a more compelling reason to do so is that a stable to-be has great value to the client. A stable to-be architecture provides the basis for long-term investments. The client can make commitments with confidence that the whole structure will not be scrapped in a few years.

Still another reason for a stable to-be is that it provides a target for aligning current decisions, for engineering the enterprise. The to-be design is likely to be implemented in stages, one sub-system now, one later. The stable to-be architecture is the glue that ties it all together.

CREATING TO-BE ARCHITECTURE

Knowing what we know about the business today, and information technology available today, what should the enterprise look like? What is the high value concept, the fundamental structure?

Classical architecture suggests that EA should begin with the to-be architecture. It starts with purpose, the enterprise business strategy (Figure 1). An architecture team explores alternative structures from the perspectives of key stakeholders and information technology. The team focuses on system drivers, those functions that primarily determine performance. Synthesis, the creation process, is accomplished through aggregation, partitioning and balancing. (Rechtin, 1991). Patterns and reference models are useful in organizing this. The product for each alternative is characterized by high level partitioning allocated to business process and information systems. The result is a small set of balanced alternatives available for client choice.

Alternatives result from the fact that there are generally more than one feasible model of the business and more than one feasible model of the Information System. Scope is common across all alternatives. Several salient points:

1. The purpose of to-be alternatives is to provide the client with the basis for a value choice.
2. Client choices should be fundamental. Beware of too many options and too much detail that obscures fundamental structure.
3. Ideally, the to-be time frame should be far enough in the future so that structure is independent of legacy systems. Legacy systems are accommodated in the transition plan.
4. In most cases, the number of fundamental feasible fundamental combinations of business processes and information system options will be small.
5. The to-be design can be a radical departure from the as-is enterprise yet still be low risk. Risk is mainly a function of the transition plan.
6. The IT aspect of the to-be architecture is a logical structure, technology free.
7. Client choice results in a stable to-be architecture that initiates engineering design.
8. Client choice also means client buy-in.

The architecture team must have experience and judgment, the ability to differentiate between what is absolutely essential from what is merely

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*

important. In addition, EA involves the synthesis of two deep disciplines: business process and information systems. Experts in these disciplines have vastly different cultures and values. Communication would benefit from facilitation. The creation of to-be EA is not a high cost effort but is critically dependent on quality of execution.

SUMMARY

The private sector clearly teaches us that the greatest productivity gains comes not from automating existing business processes but through eliminating processes that are made unnecessary by information systems. This paper posits that the primary structural question - the purpose of enterprise architecture - is to determine the relationship between business processes and information systems.

Well-defined problems can be solved by logically deducing a solution. This is engineering, which begins with a specification. With EA, the relationship between business process and information systems is not well defined. There is more than one feasible solution. When the problem is not well defined, the classical approach is to split the project into two sequential phases, an architectural phase followed by an engineering phase. This is the basic development waterfall. The transition between the phases two is a to-be architecture, generally chosen by a client from a finite set of alternatives.

If the project is split into architectural and engineering stages, the concept of EA frameworks needs to be clarified. The suggestion made in this paper is to think about Thin and Fat frameworks. A Fat framework is the usual fully populated description that presents the result of a complete engineering design. The Thin framework contains only enough information to characterize fundamental structure, the to-be architecture.

In the Zachman model, a Thin framework consists only of Scope and high level descriptions of the Business Model and Information Systems Model.

Classical architecture suggests that the to-be enterprise architecture should be stable. Since it represents fundamental structure, it should not change with evolving requirements and new technology. (The design changes, but not the architecture.) A stable to-be has value to the enterprise because it provides a stable basis for long-term investment and a distant target for current decisions.

Creating the to-be architecture involves exploring the relationships between business models and information systems models.

Knowing what we know about the business today, and information technology available today, what should the enterprise look like? What is the high value concept, the fundamental structure?

REFERENCES

Alexander, Christopher, *Notes on the Synthesis of Form,* Harvard University Press, Cambridge, MA, 1964.

DoDI 5000.2, “Operation of the Defense Acquisition System,” 2003, (available online at: http://www.dtic.mil/whs/directives/ corres/pdf2/i50002p.pdf).

DoDAF, “The Department of Defense Architectural Framework,” v1.0 Deskbook, 2004, (available online at http://www.defenselink. mil/nii/doc/DoDAF_v1_Volume_I.pdf).

Hammer, M., Champy, J. (2003), *Reengineering the Corporation; A Manifesto for Business Revolution*, Harper Collins Publishers, Inc., New York, NY, 2003, p. 35.

O’Rourke, C., Fishman, N., Selkow, W., *Enterprise Architecture Using the Zachman Framework*, Course Technology, Boston, MA, 2003.

Pavlak, A., “Simplifying the Creation of EA with Expert Teams,” *Journal of Enterprise Architecture, 1:1,* August 2005, pp. 29-35.

Rechtin, *E., Systems Architecting: Creating and building Complex Systems*, Prentice Hall, Englewood Cliffs, NJ, 1991.

Dr. Alex Pavlak, Thales Research Inc., 315 Dunham Ct., Severna Park, MD 21146; 410.647.7334, *thales@comcast.net*