

UNCLASSIFIED

2004 Command and Control Research and Technology Symposium

The Power of Information Age Concepts and Technologies

Information Age Architecture Must Be Enterprise Architecture

Thomas H. Augustine

Space and Naval Warfare Systems Center, San Diego

53560 Hull St

Code 24121

San Diego, CA 92152-5001

(619) 553-1412

Thomas.Augustine@navy.mil

UNCLASSIFIED

1/9

Abstract

This paper what characterizes an Information Age architecture and why that architecture must be consistent with John Zachman's Enterprise Architecture Framework.

Problem

The characteristics of Information Age architecture are frequently not found in products developed in accordance with the guidance found in the Department of Defense Architecture Framework, Version 2.

C2 Relevance

C2 will continue to be plagued by projects that overrun budgets, dissatisfy users, return poor performance, or result in project termination until a critical mass of acquisition professionals understand what John Zachman calls "Enterprise Physics". These are fundamental relationships between the essential elements of an enterprise and the different organizational roles that are responsible for them. These relationships are not sufficient to guarantee success but if violated the chance for success is dramatically reduced.

Introduction to the Zachman Enterprise Architecture Framework

John Zachman spent his career at IBM leading large projects. In a bid to understand IT projects, and resolve problems related to coordination and comprehension that seemed endemic, he investigated how other professions build complex things (skyscrapers and ships specifically). From his investigation he was able to generalize the interaction of all these people into a simple schema. The people fit into a few common roles (planner, owner, designer, builder, and subcontractor). The columns are based on the "six primitive interrogatories" (who, what, when, where, why, and how).

The framework is a powerful mechanism for resolving conflicts during project conception because each cell (role, interrogative pair) is unique. Uniqueness is very important precondition to successful requirements definition and requirements management. Uniqueness allows changes to be made without introducing conflicts that would otherwise arise from the same data variable's data value appearing in multiple locations and possibly holding multiple conflicting values. Uniqueness also allows efficient consistency checks which aid developing a complete set of requirements.

The Zachman Framework (ZF) is shown in Figure 1: Zachman Framework. The order of the columns is not important. The level of resolution and detail increases at lower rows.

The ZF has had a significant impact on the Defense Department. The ZF inspired the Department's architecture effort, the basics of which are described in the DoD Architecture Framework, Version 2.

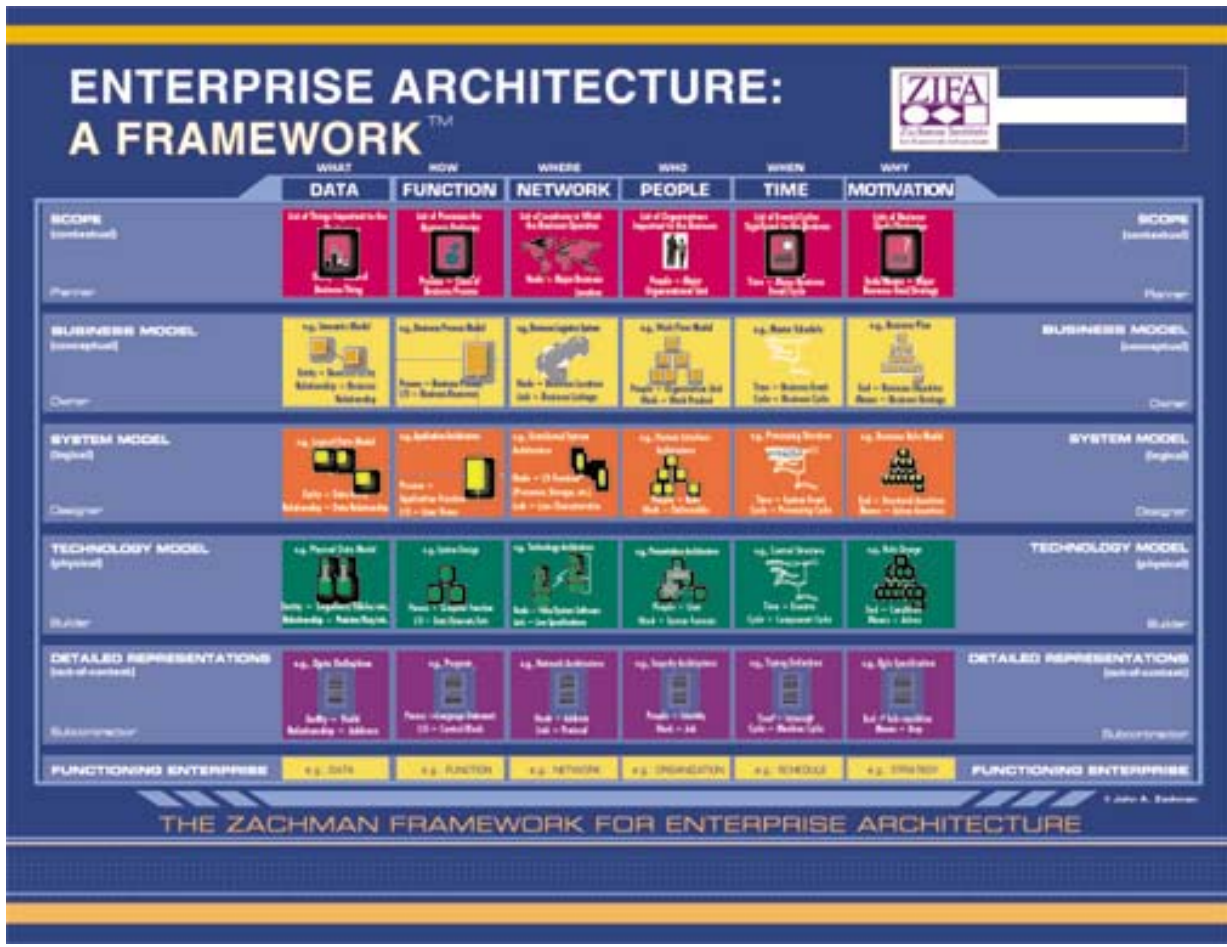


Figure 1: Zachman Framework

Planner, Owner, and Designer in the Defense World

In the ZF the Planner is the individual (or organization) that understands the potential scope of the proposed (or existing) enterprise. The Planner’s purpose is to assist the owner to define his desires so that the Designer can begin the design process.

Within DoD the role of Planner often falls to the designing organization’s system engineers working in conjunction with the project management team of that same organization. This is different than the standard business model in large real-estate development projects where a developer (owner) will hire a large architecture firm (planner) to develop architectural plans which are then used to contract with a general contractor who then develops the design (designer). The owner in the real estate model has a professional firm to assist him in realizing his vision without the conflict of interest that arises when the same group also builds the enterprise.

However the roles are apportioned among the players it is important to realize that they do exist. In defense projects where the planners work for the same program managers as

the builders it is important for program management to understand the difference. In particular, the program management must allow the planners the latitude to develop the scope and conceptual models.

In the defense context program managers frequently achieve their position by virtue of having brought in the work (the golden rule: he with the gold rules!). While bringing in work is obviously a key role in any organization, defense organizations tend to downplay the importance of “salesmen”. Perhaps “salesmen” sound too commercial. But it is nonetheless true that program managers frequently achieve their position not because of professional management experience but rather sales proficiency.

There are two key characteristics of salesmen that make them questionable choices as program managers:

1. Relentless Optimism. No one wants to buy from someone who does not convey confidence and optimism about their product. This is a necessary trait to overcome a potential buyer's fear of failure and to appeal to their desire for gain.
2. Friendship Factor. People buy from people they like.

Program managers need to be steely-eyed pragmatist that understand the technical, cost, and schedule issues so that they can make the appropriate difficult decisions. Program managers also need to maintain a business-like relationship with the customer (owner). If the salesman becomes the PM then it is absolutely crucial that he have a staff that does understand the technical, cost, and schedule issues and can go toe-to-toe with him so that he doesn't roll over whenever the customer sneezes. In particular, it is critical that the PM have a professional relationship with the customer and be able to support the system engineers as they develop the owner's conceptual model.

Why Enterprise Architecture is Fundamental

The idea seems deceptively simple, intuitive even. Use the six primitive interrogatives to describe your existing or proposed enterprise. Of course, why not, who can argue?

1. Who People model.
2. What Product model.
3. How Execution model.
4. Where Distribution model.
5. When Temporal model.
6. Why Motivational model.

Who, the People Model

People are indispensable. They are the workforce, the investors, the owners, and the customers. It is certainly a good idea to know who your “who” will be. In DoD C4ISR parlance this is often equivalent to command nodes.

What, the Product Model

Every enterprise has a product whether it is a physical item or a service. In DoD C4ISR parlance this is data, whether it is intelligence, orders, or other digital flotsam.

How, the Execution Model

The “what” does not just happen to appear. The “who” need a “how” to create the “what”. In DoD C4ISR parlance this is an activity model – a functional decomposition of mission, task, and activities to achieve an objective.

Where, the Distribution Model

The “who”, “what”, and “how” occur at a location, a group of locations, or within a network, all these are aspects of the distribution model. In DoD C4ISR parlance this is captured at a very high level in operational concept graphics and frequently not considered much after that. This may reflect the fact that historically the logistics portion of DoD has not been as closely tied to C4ISR as the operational elements. Distribution is however as critical to DoD as it is to Wal-Mart, the world’s most successful retailer.

When, the Temporal Model

Clearly all things take time, sometimes that is the critical factor between success and failure and other times it is not significant. The Temporal Model captures those instances when it is critical. In DoD C4ISR parlance this is captured in event trace or state transition diagrams.

Why, the Motivational Model

Ah, why do men do what they do? “Why” is perhaps the most crucial, political, difficult, and incendiary part of the primitive interrogatories. In politics differences can be split, shaded over, muddied up, and presented so both sides think they got what they wanted. Politics is all about compromise and frequently a lot to do with spin. The Motivational Model demands clarity. Here is where the system engineer working for clarity will run afoul of the political program manager who wants to remain “friends” with the customer. Clearly, determining why can be a tricky business that may demand internal project clarity and a bit of external obfuscation.

Enterprise Architecture is fundamental because these six interrogatories exist for your enterprise. The question is, “Do you know them?” Just as surely as that anvil will hit Wile E. Coyote on the head the fundamental relations of your enterprise will hit you if they are violated.

A Brief Introduction to “Enterprise Physics”

Just as classical physics deals with the interaction of matter and energy, enterprise physics deals with the interaction of organization and purpose.

The first two levels of the ZF model are particularly important to success. Both can actually be relatively simple, containing just primitive models. A model is called “primitive” if it only contains information pertaining to that primitive interrogatory. In others words a “how” model would only contain functional information.

The scope model, created by the planner, contains the range of possibilities for the enterprise. The concept model contains a subset of those that owner wants to implement.

As an aside, it useful to note, that as in almost all modeling efforts, there will not be the time or money for an exhaustive and complete solution. John Zachman maintains, and it seems entirely sensible, that an 80% solution is vastly preferable to proceeding blindly (watch out for that anvil!).

Since the scope and concept can be developed in simple primitive models (lists or hierarchies in general) they should be, in fact must be, understandable by the owner. They should be developed in terms the owner will understand but also have enough clarity to allow the Logical Model to be developed by the designer.

The “physics” arises from the interrelationship of the components of the primitive models. Potentially correlation matrices could be developed between all models but in keeping with the 80% solution this is not usually done. Here are a few that almost all projects should understand (if not, watch out for that anvil!):

1. Execution and Motivation. Do all the goals in the Execution Model have an implementing Executing function? Do all of the Executing functions have a related Motivation?
2. Execution and Product. Do all the products in the Product Model have an implementing Executing function? Do all of the Executing functions have a related Product?
3. Temporal and Motivation. Do all the temporal requirements in the Temporal Model have a related Motivation? Do all of the pertinent Motivations have related temporal requirements in the Temporal Model?

Identifying the cross-checks that are important for your enterprise is a crucial decision. The purpose is to ensure that a reasonable level of completeness has been achieved.

These cross-checks and the models themselves are open to revision as the enterprise matures or during a spiral development process. The important thing is to understand the changes and control for unintended consequences.

The Problem with the DoD Architecture Framework Products

The Architecture Framework products revolve around the concept of an Information Exchange Requirement (IER). An IER is fundamentally all about “Who” (sending node and receiving node), “What” (data element), and “How” (task) with some conflated “When” information (timeliness). This approach presents some unappreciated risks:

1. The motivational information is often lost or not related. In fact the IER itself is often used as “requirement” itself when the fundamental motivation is not known. The risk here is that the IER may reflect the preference of an individual subject matter expert (SME) and is not actually indicative of a fundamental requirement.

2. Distribution information is absent. Of course, the assumption is that under the network-centric paradigm the network is everywhere and therefore “where” does not matter. But in matters of logistics, ordinance, targets, or time, distribution is critical.
3. The IER method usually does not maintain the inherent relationship between the individual requirements. This information should be contained in the event traces, but frequently this is not the case due to the difficulty of maintaining these complex relationships in typical office productivity software products.
4. The ZF primitive cross-check matrices discussed above are not developed or not maintained. Without ensuring consistency of the basic architectural elements it is doubly difficult to do so when dealing with composite models such as an IER (a composite model contains elements from 2 or more primitive interrogatories).

Information Age

Defense acquisition has changed significantly over the last 20 years, especially with regard to Information Technology. In the latter days of the Cold War DoD was a significant player in IT and developed significant computational hardware to address its needs. Since then the exponential growth in commercially available computing power and the equally significant drop in price has radically changed the landscape. Not only has stand-alone computing power been revolutionized but so has distributed computing due to advances in networking and middleware.

Revolutionary new architectures are available today which are reshaping not just the world of commerce but also government and the military.

This new Information Age world needs a new paradigm to discuss the dimensions of success.

In the Industrial Age the measures of success were Better, Faster, and Cheaper. When deciding whether to invest in a new mill or factory the owner would consider those metrics when determining his return on investment. What goes without saying in this view of the world is that the product is essentially the same, only Better, Faster, and Cheaper.

In the Information Age, where the network has replaced the steam engine as the primary organizing element, producing the same product Better, Faster, and Cheaper will result in a commodity. A commodity is a standardized item which typically does not command a premium but rather trades at a price determined almost solely by supply and demand. Companies can, and do, make money in commodities, but it is generally a low growth, low margin, business where the market relentlessly demands efficiency.

The military equivalent of becoming a commodity in Information Age would be still using carpet bombing to prepare the battlefield. The size of the bombs might have increased; the planes may become more efficient in delivering ordinance; the dynamic of warfare would not have changed. The enemy would be able to adapt, collateral damage

would be severe, and the “yield” of the bombing campaign would only marginally improve.

Precision strike with rapid retargeting is perhaps the most publicized example of Information Age concepts applied to warfare. Is it Better, Faster, and Cheaper? Absolutely, but those three measures are inadequate to measure the value of Precision Strike and are especially inadequate at identifying the contribution of C4ISR to enabling Precision Strike.

To identify the new ROI components for the Information Age we turn to one of the fathers of Enterprise Architecture, John Zachman. In his symposiums he has identified Integration, Alignment, and Flexibility as the Information Age ROI metrics.

The Metrics Defined

- Quality (“Better”)
 - High quality produces display an absence of defects
- Timeliness (“Faster”)
 - The time required to execute a step in a plan.
- Efficiency (“Cheaper”)
 - The ratio of output over input
- Integration
 - Integration in Connectivity:
 - The ability to exchange symbols between nodes (syntax).
 - Integration in Meaning:
 - When symbols are exchanged they convey the same content (semantics).
 - Integration in Rules:
 - When receiving the same content, under the same conditions, all participants will take the same action (cognitive processes).
- Alignment
 - An aligned organization reflects the owner’s intent.
- Flexibility
 - Flexibility is the ability of the enterprise to change with a minimum of disruption in terms of cost, schedule, or function.

Why Information Age Architecture Must Be Enterprise Architecture

To achieve meaningful improvement in the Information Age metrics (Integration, Alignment, and Flexibility) the fundamental elements of the enterprise must be correctly understood, controlled, and arranged.

For an organization to reflect the owner’s intent the owner’s intent must be quantified and related to the specifically affected aspects. For example, by performing a cross-check between the Motivation Model (which captures the owner’s intent) and the Execution Model the enterprise architect can assure that there are the necessary functional elements

to execute the owner's intent and there are not any superfluous or rogue elements. Alignment is achieved through the systematic development and support of the Motivation Model.

To achieve integration it is critically important that the models be developed enterprise wide. If the enterprise model does not have the necessary scope it is unreasonable to expect the enterprise to display integration in connectivity, meaning, or rules.

Flexibility has a long history that originates in the early Industrial Revolution with the development of specifications and the development of replaceable parts made to those specifications in Winchester rifles. In an Information Age context flexibility means that the enterprise can change in minimum time, with minimum disruption, and at minimum cost. Key to achieving this characteristic is the development of normalized models. Normalization is in fact a complex topic but the basic idea is that one fact is found in one place, not two or more. The reason why one place is important is that once the fact may be located in more than one location the possibility of inconsistency arises as well as other anomalies. If a facet of the enterprise needs to change to display a desired capability it is much easier to do so if there will not be unintended or unknowable consequences. The identification and control of these "independent variables" is the key to "interchangeable parts" that underlie flexible systems.

Conclusion

This paper briefly introduced important concepts that have been proven here at SSC-SD to guide the spiral development of successful C4ISR systems. Using the Information Age measures introduced here, along with a coherent Enterprise Architecture approach, a program will be guided to success.