A Comparison of Frameworks for Enterprise Architecture Modeling

Presented at ER2003 International Conference on Conceptual Modeling

Presentation notes: (⊗ means advance and may occur more than once for animated slides)

Slide 0 – Title

⊗ Slide 1 – Outline

I'll begin with some of the nineteen principles that we have identified relative to modeling and frameworks, then add some observations about the use of frameworks for enterprise architecture followed by a brief overview of 3 frameworks among 20 or so identified, and finish with our take on the complementary nature of the examples.

⊗ Slide 2 – What's a Framework

Our concept of a framework is a container for components. Those components are generally artifacts resulting from a modeling process that occurs internal to the enterprise or externally in which case the artifacts represent the explicit or implicit model acquired. We add implicit because it is rare that we receive the model artifacts upon which an acquired product is constructed. The models contained are interconnected in many different ways that the framework must be able to recognize and for which access, both to model components and their connections, must be provided. By enabling the assessment of artifact fidelity to the enterprise it represents and of consistency in representation across artifacts. We examine frameworks from four aspects.

⊗ Slide 3 – Structure

We characterize structure by arrangement and three distinct scales, distinguishing decomposition from the categorical arrangements for which we use the term ordinate to indicate that they usually have a small number of coordinates, either ordered or unordered. Quite naturally a detail scaling applies to decompositional arrangements.

⊗ Slide 4 – Connections

Among the many kinds of connections, we find ⊗ dependence, equivalence, and transitivity to have particular utility in presenting artifact relationships. For example, ⊗ ordered dependence is indicative of purpose and dependence in decomposition provided by recursion.

⊗ Slide 5 – Views

Viewing mechanisms allow artifacts to be filtered or re-arranged for a particular purpose. Many framework standards include particularly specified views.

⊗ Slide 6 – Constraints

And constraints, by which we evaluate conformance. A deficiency of models is a lack of ability to formalize constraints between models.

⊗ Slide 7 – Artifact Prototypes

Enterprise architecture frameworks are based upon notions of some preexisting set of prototype models that are known to interact. If we did not have so many kinds of model artifacts we would probably not need a framework to organize our access to them. When we only had a few simple models, usually because we severely restricted our model domain, we did not use frameworks. It was only after the complexity of models and their number increased dramatically that we began to see the framework concepts emerge.

⊗ Slide 8 – Entities in Time

The next three slides attempt to convey our feeling that frameworks, of the kind that interest us the most, are of two basic varieties for which the distinguishing characteristic is their expression of time dependency. We'll use the terms continuant and occurrent for their intuitive feel and in a relative sense. We realize that even the current literature is inconsistent in their definition. Recall our claim that purpose is characterized by ordered dependence. When that purposeful order is roughly chronological, the framework is occurrent. Frameworks with a purpose that is not referenced to chronology, or that extracts time from its purposeful dimension is called continuant – it was here yesterday, is here today, and will be here tomorrow.

⊗ Slide 9 – Continuants/Occurrents

⊗ Slide 10 – Enterprise Description

⊗ Slide 11 – Zachman Framework for Enterprise Architecture

This is a somewhat dated by still valid image of John Zachman's framework. You can get a current version off the web.

⊗ Slide 12 – Zachman Framework for Enterprise Architecture (IS version)

The first characteristic of a big "F" framework is the framework or grid structure. In this case it's shown as two dimensional. ⊗ It has a purposeful ordered ordinate dimension R, usually called role or perspective, of 5 or 6 coordinates depending upon version. The first and last coordinates provide an interface to externalities and the middle three abstract a conceptual owner, logical designer, and physical builder partitioning for model artifacts. ⊗ The other dimension is unordered ordinate and consists of coordinates expressing the universal partition of inquiry, what, how, where, who, when, and why. ⊗ The framework is to be populated by a wide variety

of models – among them a logical data model, logistics network, and rule design. ⊗ Of critical importance is the primitive model proto-type or association for each kind of inquiry that serves to distinguish that interrogative from all others. Keeping the models for each column primitive in the associative sense presents the greatest challenge in using this framework.

⊗ Slide 13 – Zachman Recursion

Our formal approach to this framework, and others as well, includes recursion to manage an additional decomposition dimension. ⊗ Something like this as nested frames. ⊗

⊗ Slide 14 – Zachman Properties

⊗ Slide 15 – ISO 15704: Annex A – GERAM

Here we have the overview slide for GERAM, the result of a task force of IFAC/IFIP that pulls together several previous efforts from the manufacturing domains. Notice that in addition to its three dimensional representation, its vertical ordinate dimension is labeled “life-cycle phases” giving it a distinctly occurrent character. GERAM is used in several international efforts including the virtual logistics networks of the Globemen project and is the basis for ISO 19439 that we present and discuss in more detail.

⊗ Slide 16 – ISO/CEN FDIS 19439

Again we have the three dimensional space of ISO 15704 and GERAM. ⊗ The model phases are familiar to anyone building models. Start at the top and work your way down to the bottom – hopefully gaining handsome return during operation. ⊗ 19439 distinguishes four views of the enterprise model that are considered to satisfy completely the need for enterprise description. ⊗ The last dimension makes the notion of proto-type models explicit and distinguishes partial models, say for a business sector or standard, from both the particular models of the enterprise being modeled or the generic constructs from which they are fabricated. Note that generic and partial models form a reference catalog not defined during the operation phase.

⊗ Slide 17 – 19439 – Model Dimension

⊗ Slide 18 – 19439 – View Dimension

⊗ Slide 19 – 19439 – Genericity Dimension

⊗ Slide 20 – 19439 - Recursion

The concept of recursion expressed in 15704’s GERAM and 19439 is somewhat different. Here, the models of the operational phase can be used either in conjunction with a reference catalog or

alone to generate new enterprise models. The Globemen logistics modeling effort uses this kind of recursion along the logistics value chain.

⊗ Slide 21 – 19439- Life History

Consider then the life history of a 19439 framework model accumulating in time by adding different parts until it is complete. ⊗ A life history pictogram is then a linkage of these point-in-time model solutions.

⊗ Slide 22 – ISO/IEC 15288

Now let's consider the recently released ISO 15288 standard where a set of processes that cover the life-cycle of a system are presented. The structure is trivial and only one decompositional dimension, process group, is developed in the normative text.

⊗ Slide 23 – 15288 – Structure

⊗ Slide 24 – 15288 – Dimensions

⊗ Slide 25 – 15288 – Process Groups

⊗ Slide 26 – 15288 – Process Hierarchy

It describes a process hierarchy wherein the ⊗ technical processes, notice embedded Vee model, are nested within the ⊗ project processes nested within the ⊗ enterprise processes all mediated by the agreement processes of acquisition and supply. The 25 process descriptions are terse with 63 purposes, 123 outcomes, and 208 activities identified in 33 pages.

⊗ Slide 27 – 15288 Life Cycle

15288 provides some informative guidance to the coordinates of a life-cycle dimension that actually maps rather well to 19439. For those of you perhaps not familiar with the structure of international standards from organization like ISO, IEEE, and IEC, normative statements, such as the life-cycle phases of 19439, become requirements for conformance to the standard while informative statements, such as the life-cycle stages of 15288, are not requirements for conformance to the standard. In either case, this dimension is ordinate, ordered, and occurrent.

⊗ Slide 28 – 15288 – Recursion

The recursive use of the architectural components is an informative feature of 15288 as well. Here the content at successive stages of the system-of-interest is assembled by utilizing component systems thusly – system-of-interest concepts are provided by utilizing the concept system and similarly content is ⊗ developed into ⊗ production ⊗ supported until ⊗ retirement.

⊗ Slide 29 – Archetype Dimension Summary

In summary, we have two Zachman dimensions, three 19439 dimensions, and one 15288 dimension in the normative sense.

⊗ Slide 30 – Prototype Models

⊗ Slide 31 – Purposive Dimension

The previous slide summarizes proto-type features, but let's focus on the purpose for each kind of architectural representation. The Zachman continuant is always there – implicitly or explicitly – for use in enterprise analysis and model reuse. The 19439 occurrent structures the realization of enterprise models and operation. And lastly, 15288 identifies processes expected to populate the other two.

⊗ Slide 32 – Different Life history

So if the life history of a 19439 occurrent frame is a life-cycle, notice that the appearance of artifacts is not strictly order by phase in time with some iteration between phases but that the operation phase is stable until decommission. Contrast that with a ⊗ Zachman continuant frame that is then characterized as a never-ending saga with artifacts appearing at each role throughout time.

⊗ Slide 33 – Taking a Snapshot

Now consider a connection between these two architectural representations. On the left we have a Zachman framework from which we extract artifacts, say a 15288 process from the 'how' interrogative, for use in articulating 19439 frameworks. Notice, as indicated by the two lower arrows, that the purposeful ordinate ordering does not necessarily correspond. In this case the purposeful dependency is reversed as a result of the mapping. Notice also, that in the ideal case we may be able to completely define a occurent framework from the continuant content.

⊗ Slide 34 – Populating with Artifacts

Having realized an enterprise model with 19439, we can ⊗ place those model artifacts into a Zachman framework for analysis and reuse. ⊗ All manner of correspondence is possible, ⊗ from operational 19439 frameworks to model fragments. And we can begin to ⊗ think formally about a single occurrent framework not completely specifying the continuant expression.

⊗ Slide 35 – Profile of Change

We call this a profile of change and it is a graphic and not a graph. The scale along the bottom is roughly conceived as time in two segments with labels for approximating 19439 phases on the right. The vertical axis is divided into the conceptual distinctions of Zachman's role dimension. Then focusing on the right, realizing a 19439 operation might ⊗ proceed in time from conceptualization through the 19439 phases using 15288 processes along the way. But in most situations, before we realize something new, we assess where we are. On the left, ⊗ we identify the as-is condition by measuring our physical situation (r3), verifying the consistent design (r2), and validating correspondence to a conceptual basis (r1). Then we proceed to realize the new based upon the foundation ⊗ of the Enterprise Book of Knowledge we have available and add to that knowledge resource at each phase so that models can be reused. ⊗

⊗ Slide 36 – Managing Change

So now we can more formally describe the management of change in the enterprise with either scenario. Notice that in either case, both the continuant and occurrent framework are essential. The complementary nature of enterprise architectural frameworks enables more efficient and effective change management. Again, notice that the use of continuant components may provide a complete occurrent, but that even a complete occurrent is likely to provide only partial components to the continuant framework.

⊗ Slide 37 – Comparative Summary

As a continuant, Zachman can be a container for a more comprehensive selection of model artifacts. It also distinguishes primitive models from view oriented composite models. And it provides a conceptual partitioning of the enterprise in strictly human terms – inquiry from essential perspectives.

⊗ Slide 38 – Approaching Frameworks

Our examination of various frameworks results from our goal of guidance in framework construction and use. There are a lot of them out there and more on the way. Frameworks help us to know the model space and thus facilitate model reuse. From our practice we identify principles to build formalisms that support our practice.

⊗ Slide 39 – References

