Architecture, Enterprise Architecture, Frameworks, and Processes

Ground Systems Architecture Workshop 2004

Kevin B Kreitman

THE AEROSPACE CORPORATION

What We'll Cover

- Working Definitions
- Enterprise Architecture
- Processes
- Frameworks
- Rough recipe for success
- How the frameworks and approaches measure up

THE AEROSPACE CORPORATION

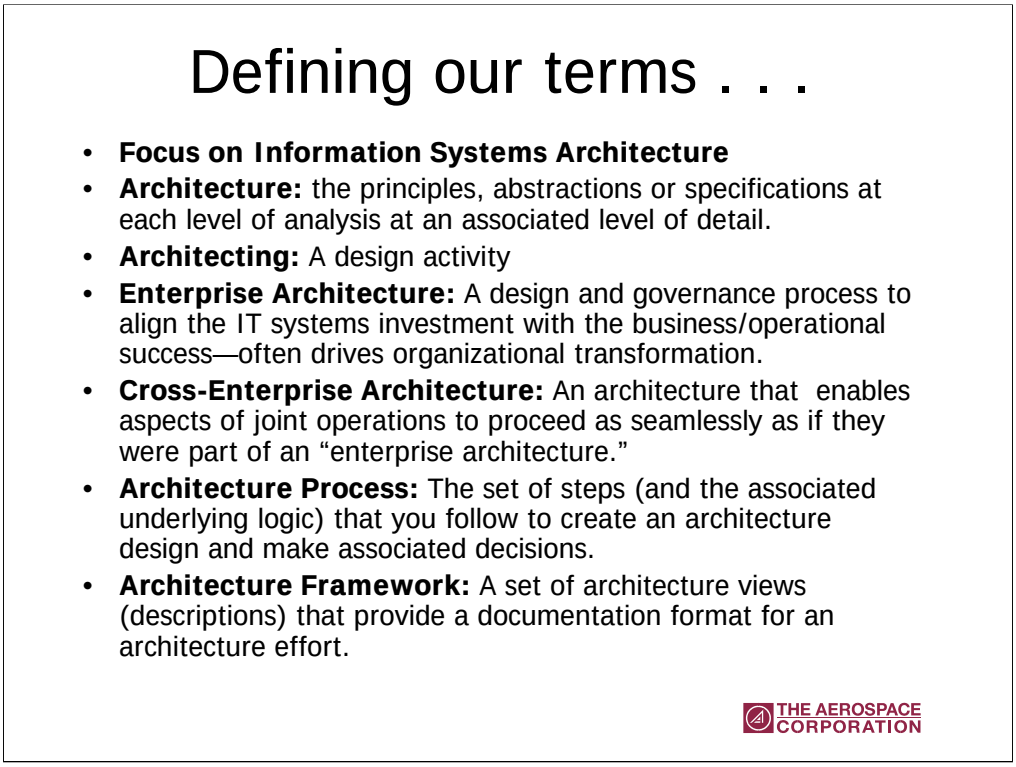
Architecture: The “A” Word

- “Architecture” means different things to different people in different contexts.
  - Documentation format
  - Wiring diagram or schematic
  - Fundamental principles, characteristics and features that give a system its value
  - Required document for funding
  - The structure of components, their relationships, and the principles and guidelines governing their evolution over time.
  - Means of enforcing cost savings, standardization and efficiency improvement practice
  - Design process
  - Etc.
- Be careful of your assumptions about what “it” means.

THE AEROSPACE CORPORATION

The term “architecture” is an overloaded term.

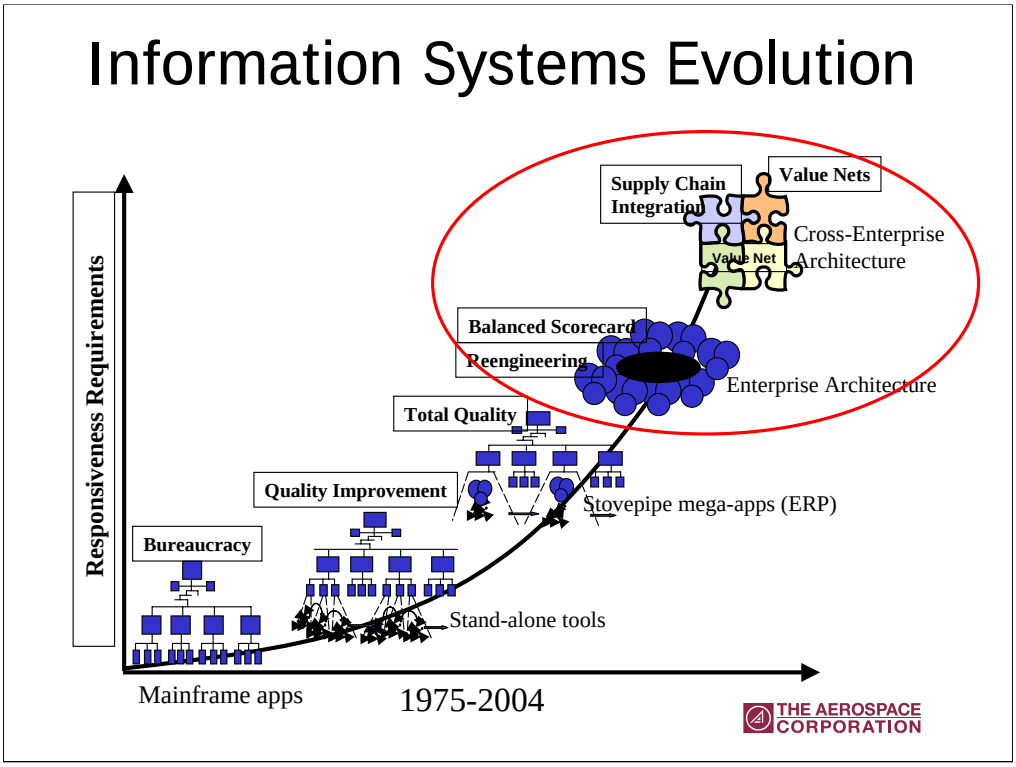
Before we get into arguments about how to do “it” and what makes a good architecture, and whose approach is better . . .we need to acknowledge the diversity of meanings people have, and define what WE (each) mean by architecture (and related terms) for the purpose of this discussion.



This is the frame of reference I will be using for the purposes of this presentation.

Reasons:

1. The focus is on FEAF "vs" DoDAF. . .. Both of these are fundamentally information systems architecture focused. And both focus on Enterprise Architecture and/or cross-enterprise architecture.
2. Enterprise architecture (in common use in industry) means the information systems that function as an organizational "nervous system," enabling and facilitating the business processes that make the organization successful. This also implies that they have a single ultimate owner/decision-maker, regardless of the degrees of local autonomy that might be allowed. There is a single place in the hierarchy where the buck stops.
3. A Cross-enterprise Architecture implies that there are different organizations with separate final nodes of ownership or control.
4. A process is how you carry out an architecture design activitiy, and the underlying logic behind why you do it, and how you manage it.
5. An architecture framework is a (sub)set of architecture description formats. These are usually designed with some underlying assumptions about how and why architecture is carried out, but the description formats are independent of process. Some processes will not use all of the architecture views, and some frameworks will not be sufficient to capture all descriptions or documentations of a process.



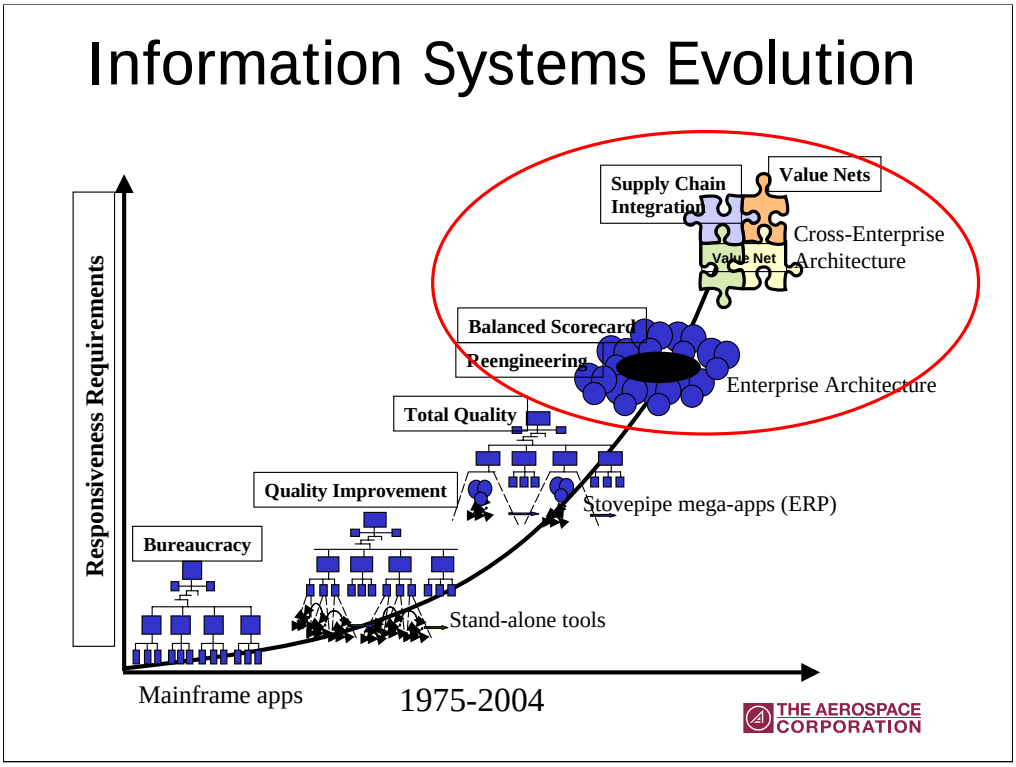
This chart shows the evolution of organizational forms over the past 25 or so years, and the nature of the information systems that have evolved to support them.

Bureaucracy is technically a hierarchical form of organization, where responsibilities and authorities are allocated to each organizational subsystem for maximum independence. The theory is that if each organizational component performs according to the rules, and people obey their roles, the organization as a whole will succeed. Bureaucracies tend to change very slowly. This only works when the environment, threat, opportunities, requirements are slow to change as well, and when there are few dynamic interdependencies among the subsystems. Here, mainframe applications that kept track of standard data processes constituted the information system support.

With the advent of quality improvement, subsystem organizations realized that they could redesign their processes to improve their performance. At the same time, information systems could provide standalone tools to enhance these processes.

With the advent of Total Quality (and practices such as concurrent engineering, IPTs, and cross-functional teams), the focus shifted to improving processes across larger parts of the enterprise—integrating enterprise resource planning, for instance. At the same time, relational databases provided a common data repository for otherwise stovepiped applications.

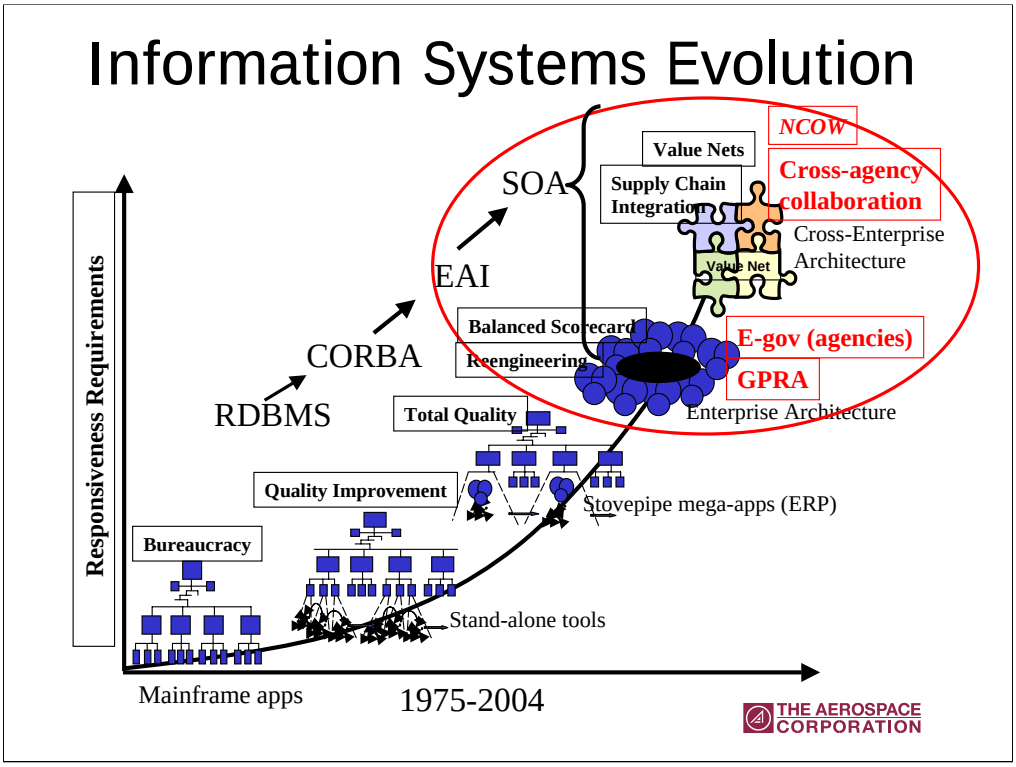
CONTINUED ON NEXT PAGE



CONTINUED

At the point when organizations shifted to looking at optimizing their whole organizations—to streamline operations and cut costs—the information system began to be seen as the nervous system of the organization, rather than a collection of data-processing applications to increase local efficiency. The technology for this step lagged the need by several years—but the interest in aligning and integrating information systems endured. Currently this is the focus of most enterprise architecture efforts. Both cost-efficiency and organizational transformation (or at least fundamental business improvement—and electronic support of customers—e-commerce) are the typical foci of these efforts.

Finally, pushed by the shrinking cycle times of Internet commerce, Cross-enterprise architecture emerged as a serious issue for supply chain integration and distributed operations. The parallels for Federal Government and DoD are clear. FEAF emerges with the constellation of legislation for efficiency and effectiveness in government GPRA, and the alignment of information technology with organizational goals (Clinger-Cohen, and e-gov initiatives) closely parallels the Enterprise Architecture push in industry. DoD has both the internal streamlining of operations (enterprise architecture) and the vision of fully integrated joint operations with the GIG and NCOW (Cross-Enterprise Architecture). Although DoDAF is a general purpose architecture description set, many of the architecture efforts in DoD are focused on Enterprise or Cross-Enterprise issues.



Impact of technology evolution that has co-evolved in support of these changes:

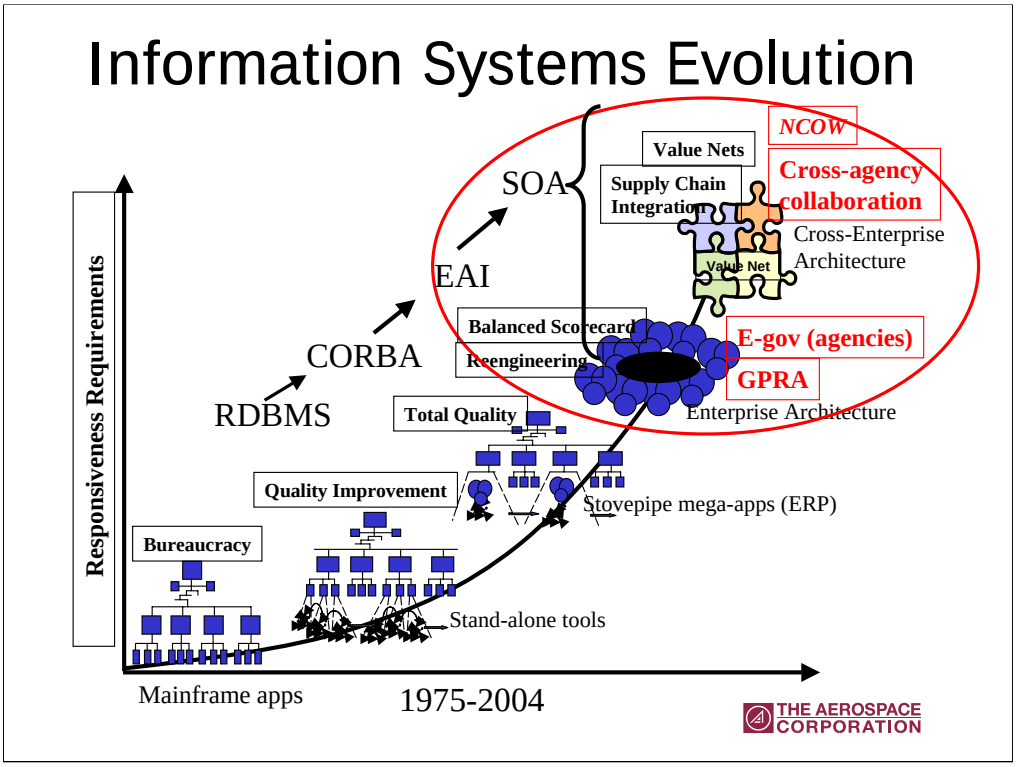
Relational database management systems (RMDBS) were a method of sharing a common data source across multiple applications—the first form of integration commonly seen across the organization.

Common Object Request Broker Architecture (CORBA) was a step forward for integration, in that it integrated application level operations, rather than just sharing data. However, CORBA was a limited, specific, and often point-to-point solution.

Enterprise application integration (EAI), on the other hand, was based on an information bus concept, message-oriented-middleware, message formats in XML, and adapters to create common interfaces. It enabled different stovepipe applications to operate together in an integrated way. More importantly, once the interfaces were adapted, it reduce the complexity of integration from an N*(N-1) dimensional problem(individual point-to-point connections)to a linear problem.

Fundamentally however, there were still problems: lack of semantic interoperability across applications, and flexibility, since the applications were themselves complex and brittle.

CONTINUED ON NEXT PAGE



CONTINUED.

Web services, and the more generic form, service oriented architectures (SOAs), are the most current proposed solutions for these problems. Service oriented architectures are based on breaking down into very small components. The functions that are required to perform tasks across and within organizations. When well-designed, these components can be composed into complex services interface smoothly with each other through designed common interfaces, and swapped out or changed without disrupting the flow of the process that they support. Because each component is designed to provide a very specific and limited service, the system is agile, easy to augment and change by adding components or by reconfiguring components into brand-new processes.

Describing these architectures, however, requires new models, and the ability to express abstractions, rather than concrete pieces like applications and point to point data flows. In fact, the same service-oriented architecture can be implemented over a wide variety of networks, servers,and platforms and operating systems, and is easily deployed across heterogeneous systems.

Architecture Processes:
A Closer Look at EA

- Enterprise Architecture is a way of aligning IT implementations with organizational success.
- Fundamental to EA processes in industry:
  - Start with desired operational future state, purpose;
  - Start from the "top down", even if the focus is only one part of the organization;
  - Focus first on globally desirable outcomes, then on the business/operational processes they are to support, then the data model, and the technical or solutions architecture.
  - Separate the "what" from the "how"
  - "As-is" is captured in parallel and as needed.
  - Include governance process (program management, approval, portfolio management, etc.)
- EA requires a central decision-making authority about the architecture (i.e. constraints on local autonomy, acquisition)

THE AEROSPACE CORPORATION

That said, it is worthwhile to look at commercial best practices for Enterprise Architecture, because the success criteria for EA transcends the commercial/government/military boundaries.

Process Example #1

Enterprise Architecture Planning

(Dr. Stephen Spewak)

Global and comprehensive . . .

- Leadership sets the principles, values, desired future
- Business people define (in plain language) what their business organization needs to do (abstraction)→data model
- Then joint business/IT people explore the options for IT/automation
- As needed, "as-is" systems are documented
- Decision Day comes after the business/IT exploration and conversations; determines what systems will be replaced, by what technology, on what timeline, consistent with the principles and values

THE AEROSPACE CORPORATION

Process Example #2

Enterprise Planning &Architecture Strategy (EPAS) (Meta Group)

Iterative, "Just enough architecture, Just in time"

Define business strategy →EBA (Business Arch)

Define business processes →EIA (Information Arch)

Determine IT attributes →ETA (Technical Arch)

Select/build IT systems → ESA (Solution Arch)

Document existing systems

At each level:

- "Common requirements vision" (simple, broad)
- "Conceptual architecture" (set of relevant principles)
- Flow-down for articulation at next level

Guidance for governance process, program management gates, portfolio management (fixed % for maintenance, improvement, transformation)

THE AEROSPACE CORPORATION

Examples of Frameworks

"What kind of stuff to capture and what format to capture it in . . ."

- ***Zachman***--An extensive and classic capture format—separates the "what" from the "how," "where," "when" . . .(best practice)
- ***DoDAF***--A generic documentation tool, a subset of possible architecture views, oriented toward physical representation rather than abstraction
- ***FEAF***--An Enterprise Architecture approach with an underlying architecture concept (e-gov), and a reference architecture. It has been influenced by successful "EA" practices in commercial industry.

THE AEROSPACE CORPORATION

Zachman framework is perhaps the oldest and most extensive framework in use today. John Stockman defined it broadly, so it would cover all aspects(and separate out into the right abstraction categories) necessary for designing and analysis. Each sale in Zachman's framework is independent and can be captured in a variety of view formats. To whatever level of detail is necessary.

DoDAF was originally developed as the C4ISR architecture framework, when the DOD began to address the problem of interoperability across their joint forces. It proved impossible to align the organizations and their acquisitions to support interoperability directly. It was hoped that the framework would allow acquisition systems to be described in a way that would allow apples to apples comparisons, and evaluation of the system's ability to interoperate with other systems in place or in planning. Unfortunately, it has proved difficult to do this, because the framework only specified the form, and not been level and nature of the contents or internal consistency among the three major views.

FEAF, on the other hand, has been heavily influenced by current commercial architectural thinking and has embedded in the framework and direction on how to use it an underlying model that closely parallels the abstractions needed to represent enterprise application integration and service oriented architecture.

Pitfalls of DoDAF for EA

- Treats cross organizational “integration” as bottom up, or sideways (point-to-point with other systems.)
- Makes it difficult to separate the “what” from the “how” and the “where.”
- Without a fundamental “architecture concept” or reference architecture—or “business-oriented top down” approach—it is hard to see how to design an integrated system. (Think SOA)
- Does not support modern architecture abstractions.
- Historically, DoDAF was the result of the DoD decision not to try to enforce top-down planning. . .and to make it accessible to senior officers.

THE AEROSPACE CORPORATION

Purpose, process, framework

- If you can define your PURPOSE;
- And use an effective PROCESS to align your architecture work to your purpose;
- You can document it in a variety of frameworks—or make up your own.
- Frameworks are not fully equivalent; you may have “more” or “less” required in a framework than is dictated by your purpose.

THE AEROSPACE CORPORATION

On the other hand . . .

- If you start with a FRAMEWORK,
  and create an “as-is” documentation
  or jump to a technology-driven “to-be”
  without a business alignment process,
  you are likely to have a “Learning Experience”

- A “Learning Experience” is what you get
  when you didn’t get what you wanted . . .

THE AEROSPACE CORPORATION

Backup Slides

THE AEROSPACE CORPORATION

Key Take-Aways

- Start from the **top-down operational or business goals**—or the desired result—and architect to achieve the goal.
- If you are required to use a particular framework, use it for documentation—but choose an **appropriate process** for your application. Augment if necessary.
- Cross-enterprise architecture requires a higher level architecture (and compatible semantics) —not just point to point information flow—to be effective.
- Just because you have an “architecture” doesn’t mean you will solve your problem—unless you architect (design) specifically for the problem you wanted to solve.

THE AEROSPACE CORPORATION

Some Past and Current Architecture Experience

- C4ISR Architectures for interoperability and integration
- FEAF Architectures for automation and integration
- Air Force Scientific Advisory Board 1999 on Joint BattleSpace InfoSphere (SOA) and high level architecture for Effects Based Operations
- Air Force Scientific Advisory Board 2000 on C2 Integration
- Composable Services Architecture Research for National Systems
- University of MD, College Park Enterprise Architecture

THE AEROSPACE CORPORATION

DoDAF "Generic Process"

1. Determine the intended use of the architecture description.
2. Determine the architecture description's scope, context, environment and any other assumptions to be considered.
3. Based on the intended use and scope, determine what information the architecture description needs to capture.
4. Determine the products to be built.
5. Gather the architecture data and build the requisite products.
6. Use the architecture description for its intended purpose.

Source: DoD Architecture Framework v. 1.0

THE AEROSPACE CORPORATION