.BUSINESS PROCESS TRENDS

WHITEPAPER

January 2003

Author:

Paul Harmon
Executive Editor
Business Process Trends

Contents:

The Enterprise Architecture
The Zachman Framework
The Enterprise Architecture Cycle
Maintaining an Enterprise Architecture
How to Create an Enterprise Archiecture
1. Agree on the Need
2. Establish an Organizational Structure
3. Select a Framework
4. Select a Tool and Repository
5. Organize the Existing Material
6. Begin Using the Enterprise Architecture
7. Extend and Maintain the Architecture
Aligning the Organization
Aligning Business Processes with Information Systems
Summary
Notes

Sidebar 1. The U.S. Government's Enterprise Architecture Initiative

Developing an Enterprise Architecture

Overview

This white paper discusses the growing role and importance of enterprise architectures in the management of organizations. I will begin with a definition of an enterprise architecture, then I will examine the Zachman Framework, a typical overview of an enterprise architecture. Subsequently, I will consider how an enterprise architecture might be used in a large organization, and then consider how a company might go about creating an enterprise architecture. Finally, I will consider how an enterprise architecture can be used to align organizational goals and how business processes can be aligned with IS systems. In other words, I will begin by considering strategic issues and gradually drill down into some of the tactical problems involved in the development of an enterprise architecture.

An Enterprise Architecture

The term “architecture” has been used for many years within the IS community to refer to various types of overviews that provide guidance to software systems and applications developers. The term is obviously a metaphor derived from the building trade. Just as builders would not undertake the construction of a house or an office building without an architecture, documented in various blueprints, so software developers should not undertake the development of software systems without a detailed plan, documented with software “blueprints” of various kinds.

In the mid-Nineties, the term “architecture” began to be used by business managers, especially those involved in enterprise planning and in business process reengineering projects, to describe an overview of the business. For example, some managers began to refer to a high-level description of all of the core business processes in an organization as a “business process architecture.”

Today, there is a growing movement among both business managers and IS managers to use the term “enterprise architecture” to refer to a comprehensive description of all of the key elements and relationships that make up an organization. Increasingly, when managers talk about the alignment between business processes and goals and IS applications and middleware systems, they rely on an enterprise architecture to define how the business-IS alignment should be achieved.

There are many different approaches to describing the elements of an enterprise architecture. One approach that has grown in popularity in the past few years is based on a framework developed by John Zachman. Zachman originally proposed his framework in 1987 in an article published in the *IBM Systems Journal*. The article created quite a bit of interest when it was published, but was generally dismissed as too comprehensive for IS developers. Over the years, however, as business managers have become more concerned with business processes and with linking strategic goals to business process goals, and both to IS applications and databases, the Zachman framework has become the most popular approach to describing an enterprise architecture.

I'll consider the Zachman Framework in more detail in a moment. First, however, by way of preparation, I want to establish some expectations. To understand the Zachman Framework it is important for the reader to realize that the framework is not simply a description of a collection of documents and plans. It's a model of how all of the parts of an organization fit together.

A building's architecture is more than a set of blueprints. High-level diagrams explain the concept of the architecture to the owners of the building and allow them to decide if the overall approach will meet their needs. Second-level diagrams lay out the basic units, the foundation, rooms and roofs. These elements must be designed with each of the other units in mind. A foundation must have the strength to support the rooms. If the house has two or three stories, the foundation must be proportionally stronger. More bathrooms require more water heating capacity and larger pipes. More electrical capacity must be routed to rooms that are going to house utilities, like dishwashers or dryers. In other words, there may be different blueprints, one for the foundation, one for the layout of the rooms, one for the electrical systems, and still another for the plumbing, but there must ultimately be relationships among the various blueprints to assure the house, as a whole, works as it should.

Similarly, one group in an organization may survey the competitive environment and recommend changes in strategy and new goals. Another group may define business processes, and still another may create designs for new software applications. The enterprise architecture defines all of these elements, and also defines how they fit together to assure the organization functions as intended.

I recommend that the enterprise architecture function should be a responsibility of the steering committee, perhaps a specific responsibility of the enterprise planning group. Senior management sets strategy, and strategies and their associated goals should drive the development and change of the business process architecture. A properly defined architecture provides an overview of the enterprise and how it functions. Thus, it provides managers with a context for analysis and decisions.

Let me reiterate: A pile of documents does not make an enterprise architecture. To obtain value from the various plans, models and documents that various groups create, they must be integrated. This is normally accomplished by means of a software tool – usually an enterprise or business process modeling tool that relies on an underlying repository. The various plans, models and documents are placed in the repository and the relationships are established by those that place each element into the repository. Different groups using the enterprise modeling tool look at different views of the information in the repository. Business managers, for example, typically look at diagrams of the organization or at specific business processes. IS managers are more likely to check which software applications are associated with specific processes and then go to diagrams describing the applications.

If the enterprise architecture is documented in sufficient detail, managers are in a position to ask about the implications of specific changes. In effect, one can "grab hold" of a specific item on a process diagram, "pull it" and see what's attached to it. Similarly, if the repository contains data on employees and costs, one can determine the specific costs and implications of changing the item and doing it in a different way. In other words, a good enterprise modeling tool will allow managers to run simulations that can show what would happen if different scenarios were to be implemented.

In a similar way, business managers can examine questions about gaps and disconnects. Imagine, for example, that the steering committee is concerned with the fact that customers continue to make the same complaints and the complaints don't seem to result in any changes in the way the product is manufactured. A glance at the appropriate organization or process diagram ought to allow the executives to determine the way information flows from customers back into the company. One diagram shows a line that runs from customers to the customer support group. Another check determines that the customer support group logs complaints. A continued search, however, doesn't turn up any link between customer support and product design or manufacturing. In effect, managers are going to want to create a new, formal information flow from customer support to product design and to manufacturing. At the same time, they are probably going to want to create a formal measurement system and charge specific managers along the new path with monitoring complaints and making changes in products in response to complaints.

To sum up again: An enterprise architecture is a tool to help executives think about the organization as a whole. An enterprise architecture captures a wide variety of information and links it together in a single database or repository, so that managers can then see relationships and ask questions to identify problems or to make decisions about changes they are considering.

This white paper advocates that business managers and IS managers work together to create an enterprise architecture for an organization, and then urges that they maintain the architecture as an ongoing way to assure that business processes and information systems remain aligned. If this approach is embraced, then it means that the group responsible for creating and maintaining the overall enterprise architecture must necessarily have both business and IS representatives and must be a high-level committee, probably reporting to the organization's executive committee. I term this organization the Enterprise Architecture Committee, although it could as easily be called the corporate planning committee or any of a dozen other names. The key, however, is that this is the group that allows business and IS managers to work together to create a common, integrated view of the organization.

The Zachman Framework

In the past few years, the enterprise architecture overview that has been most widely cited as a model is the Zachman Framework. The current version of the Zachman Framework is illustrated in Figure 1.

The color and the labels on the extreme left are ours and we'll explain them shortly. First, notice that the framework is arranged in rows. The rows at the top are the most abstract and are oriented toward very broad goals and plans. If I were building a house, this layer would describe the diagrams, pictures and plans the architect would discuss with the owner. The next level is more specific, but still abstract. These are the diagrams that the architect would discuss with the contractor. In a similar way, the top level of the Zachman framework, labeled “Scope,” is focused on the concerns of senior executives. The second level focuses on the slightly more detailed concerns of business managers. Level three focuses on concerns that business and IS managers often work on together. Levels four through six focus on details that IS managers and software developers are concerned with.

If you look across the top row of the framework, you can see the types of issues Zachman expects managers to consider as they define the organization. He expects that executives will create lists of things important to the business, lists of processes the business performs, lists of locations at which the business operates, and so forth. As you go down the rows, you can see that they are organized as levels. As you work your way down through the framework, you can see how lower level managers focus on the same general topics, but in more specific detail. The top row, labeled SCOPE, focuses on documents that senior managers and planners would normally use. The plans and documents become increasingly specific and detailed as you drop lower. The bottom layer refers to actual data, specific applications, all the physical structures, and the people that comprise the business.

A second glance at the matrix will reveal the various kinds of specific architectures that Zachman includes within his overall framework. The cell that represents the intersection of Function and Systems describes the organization’s application architecture. And the cell that represents the intersection of the Network and the Technological model represents the technology architecture, which describes the hardware used and the links between the platforms.

There are many alternative names being used to describe the cells in the Zachman Framework. Some prefer to call the “technology architecture” the “hardware systems architecture,” for example. Once you start down the path toward creating an enterprise-wide architecture, however, and you want to name and define all of the models and documents used in a large organization, you need a picture that embraces everything, and provides clear distinctions. Zachman’s Framework is popular because it provides a comprehensive overview and assigns a distinct name to each of the cells in his framework. Most prefer to use this comprehensive, established, standard approach instead of trying to recreate the entire approach using slightly different terms.

Even those who rely on the Zachman Framework disagree about whom the architecture is designed to assist. For some, it’s an architecture for Information Systems people. This is probably the most popular approach to the framework, simply because the framework is best known in the IS community. At the same time, however, business managers maintain many of the kinds of documentation described by the framework. For example, lists of business goals and strategies, business plans and the high level models of business processes supported by various divisions and line organizations, are normally created by business managers. If IS were to maintain the framework, they would need to obtain all of these documents from business managers and then maintain them by constantly checking with business managers to determine when things changed.

	The Zachman Framework	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*
Business Managers	SCOPE (Contextual) *Planner*	List of Things Important to the Business	List of Processes the Business Performs	List of Locations in Which the Business Operates	List of Organizations Important to the Business	List of Events Significant to the Business	List of Business Goals/Strategies
	ENTERPRISE MODEL (Conceptual) *Owner*	Semantic Model	Business Process Model	Business Logistics System	Work Flow Model	Master Schedule	Business Plan
IT Managers and Developers	SYSTEM MODEL (Logical) *Designer*	Logical Data Model	Application Architecture	Distributed System Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
	TECHNOLOGICAL MODEL (Physical) *Builder*	Physical Data Model	System Design	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
	DETAILED REPRESENTATIONS (Out-of-Context) *Sub-Contractor*	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Specification
	FUNCTIONING ENTERPRISE	Actual Business Data	Actual Application Code	Actual Physical Networks	Actual Business Organization	Acutal Business Schedule	Actual Business Strategy

Figure 1. The Zachman Framework

In fact, the Zachman Framework was designed to provide an architecture that embraces all of the descriptions used by a large organization. It makes more sense for business managers to maintain those aspects of the architecture that they create and for IS managers to focus on the documentation that they create. To emphasize this, I divided the cells in the Zachman Framework to show that the upper rows are primarily the concern of business managers and a lower area that is primarily the concern of various groups of IS managers.

Once again, I want to emphasize that the framework isn't meant to function simply as a way of classifying the types of documentation a company might create. There should be concrete relationships between cells on the matrix. Applications should be associated with specific business processes. Business processes should have goals and measures that should, in turn, be related upward to corporate goals, and downward to application and system goals. The architecture stresses both information and the relationships between information that the management team must establish and refine.

To make things a bit more concrete, let's consider how an organization might use an enterprise architecture in the daily course of business.

The Enterprise Architecture Cycle

One group must be responsible for maintaining the enterprise architecture. This isn't to suggest that this group must create the architecture, but only that they must pull all the pieces together and maintain them so that anyone else can access the architecture. I suggested earlier that the group to maintain the enterprise architecture was the corporate planning group. Some organizations, of course, place the enterprise architecture function in a different location. To remain neutral, we'll assume that the group that maintains the enterprise architecture is the Enterprise Architecture Committee, a group that reports to the executive steering committee and maintains close relationships with the strategy group and those involved in business process redesign and improvement.

Figure 2 provides an overview of the enterprise architecture cycle that is based on the assumption that an organization has an Enterprise Architecture Committee that is ultimately responsible for the creation and maintenance of the enterprise architecture. As you can see, this committee functions very like the planning committee in many large organizations.

The Enterprise Architecture Committee is responsible for maintaining the organization's architecture and for initiating and prioritizing all changes in the architecture. Thus, they receive inputs from two groups. First, they receive inputs from the strategy committee when they decide that the organization needs to set new strategies or change existing goals. At the same time, the Enterprise Architecture Committee receives inputs from a variety of line managers when they decide that processes they manage aren't performing properly. Different companies allow line managers greater or lesser discretion in initiating their own changes. The key, however, is that all changes that require significant business process redesign or software automation efforts must come through the Enterprise Architecture Committee. It's the only way to assure that IS has a clear set of priorities to guide their efforts.

The Enterprise Architecture Committee initiates all significant business process change efforts. I assume that the Enterprise Architecture Committee is just as concerned with processes that are entirely manual as they are with business processes that include automated subprocesses. Thus, some business process redesign efforts will be primarily driven by business managers, while others will be driven by IS managers or by a mix of business and IS managers.

As redesign efforts are initiated that involve systems, the Enterprise Architecture Committee must also consider the wider implications. A process may be automated by a software application, but that application will probably have to rely on new or existing databases and it may require new infrastructure elements. The Enterprise Architecture Committee, working in conjunction with the IS organization, is responsible for initiating not only new applications to support new business processes, but also initiating data and infrastructure changes to support those applications as they come on-line in the future. This isn't to suggest that the Enterprise Architecture Committee actually designs applications, but only that it sets the high-level goals and requirements for those projects. I assume that some IS executives

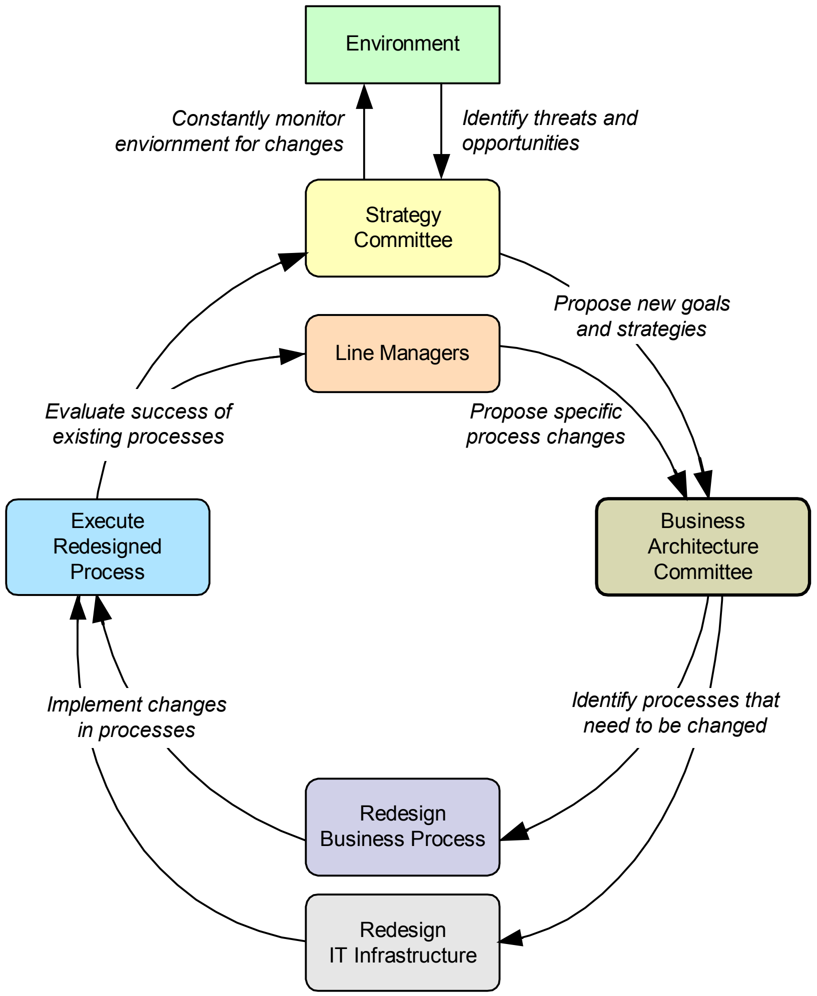
Environment
Constantly monitor enviornment for changes
Identify threats and opportunities
Strategy Committee
Line Managers
Propose new goals and strategies
Evaluate success of existing processes
Propose specific process changes
Execute Redesigned Process
Business Architecture Committee
Implement changes in processes
Identify processes that need to be changed
Redesign Business Process
Redesign IT Infrastructure

Figure 2. The Enterprise Architecture Alignment Cycle.

will be members of the Enterprise Architecture Committee to assure that business and IS executives work together to establish the right overall direction and priorities.

Once a given process or set of processes are redesigned and implemented, the cycle returns to where it began. The Strategy Committee continues to watch for changes in the environment that suggest new strategies, and the line managers continue to look for opportunities to increase the productivity of existing operations.

The Enterprise Architecture Committee will not create most of the documents and diagrams that are stored in the repository it manages. Other groups will create these documents in the course of doing their work. The Strategy Committee will create documents defining new strategies and goals. The IS organization will create designs for applications, databases and infrastructure. The Enterprise Architecture Committee, however, is responsible for integrating all of the information and assuring that each group is aligned with others in a satisfactory way.

Maintaining an Enterprise Architecture

Before going further, let's pause to consider how the Enterprise Architecture Committee maintains and integrates all the information described in the Zachman Framework. In essence, the Enterprise Architecture Committee must rely on a software tool especially designed to maintain an enterprise architecture.

There are several tools available that can maintain an enterprise architecture in a repository. (A repository is simply a database built specifically to store and relate the various kinds of documents and diagrams described in the Zachman Framework.) A good example of a tool that an Enterprise Architecture Committee might use is Popkin Software's System Architect. System Architect has a special interface that pictures an architectural framework. The default framework supported by System Architect 9.0 is the Zachman Framework. When a user sits down to use System Architect, they access the Framework screen and click on a given cell on the Zachman framework to immediately access all of the diagrams and documents that are stored in the specific cell.

The tool allows the person maintaining the repository to store any of a very wide variety of diagrams, as well as popular document formats, like Word and Excel files, in any of the cells of the framework. Moreover, as long as the naming conventions used are consistent, one can search for a given item and access all documents in all cells of the Framework that refer to a given item. In effect, the person entering information into System Architect's repository maintains the relationships by insisting that similar items use the same naming structures. Thus, if a company establishes that an enterprise goal is identified with the number 1, then sub-goals related to that goal might be 1.1, and sub-sub-goals might be 1.1.1. If this approach is maintained, then an executive can ask how a specific goal is being supported, and the Enterprise Architecture Committee can identify business processes, applications, job descriptions, and network systems that are designed to implement or support goal 1 by simply querying System Architect for all items that include goal 1 or extensions of 1.

Figure 3 provides an overview of the relationship between the System Architect framework interface, diagrams and document formats supported by the tool, and the Popkin repository and metamodel.

One important element of any tool is an interface specifically designed to support all of the information required by the framework. Another is support for a wide range of diagrams and document types so that everything can be stored in the repository. Still another is a set of interfaces, including XML and XMI, that assure that documents or diagrams developed in other tools can be easily moved into the common repository. In addition, the tool should allow developers to simulate processes so that existing processes can be compared to possible alternative processes. Simulation makes it possible to compare costs and support requirements for specific changes that are being considered. The hard work of analyzing business processes and designing software systems can't be eliminated, but the work of assembling them and tracking the relationships between all the various models and systems can be significantly eased by the use of a good modeling tool with a powerful repository.

At this point, we've described an enterprise architecture and suggested how it could be used. Some companies already have established enterprise architectures, but

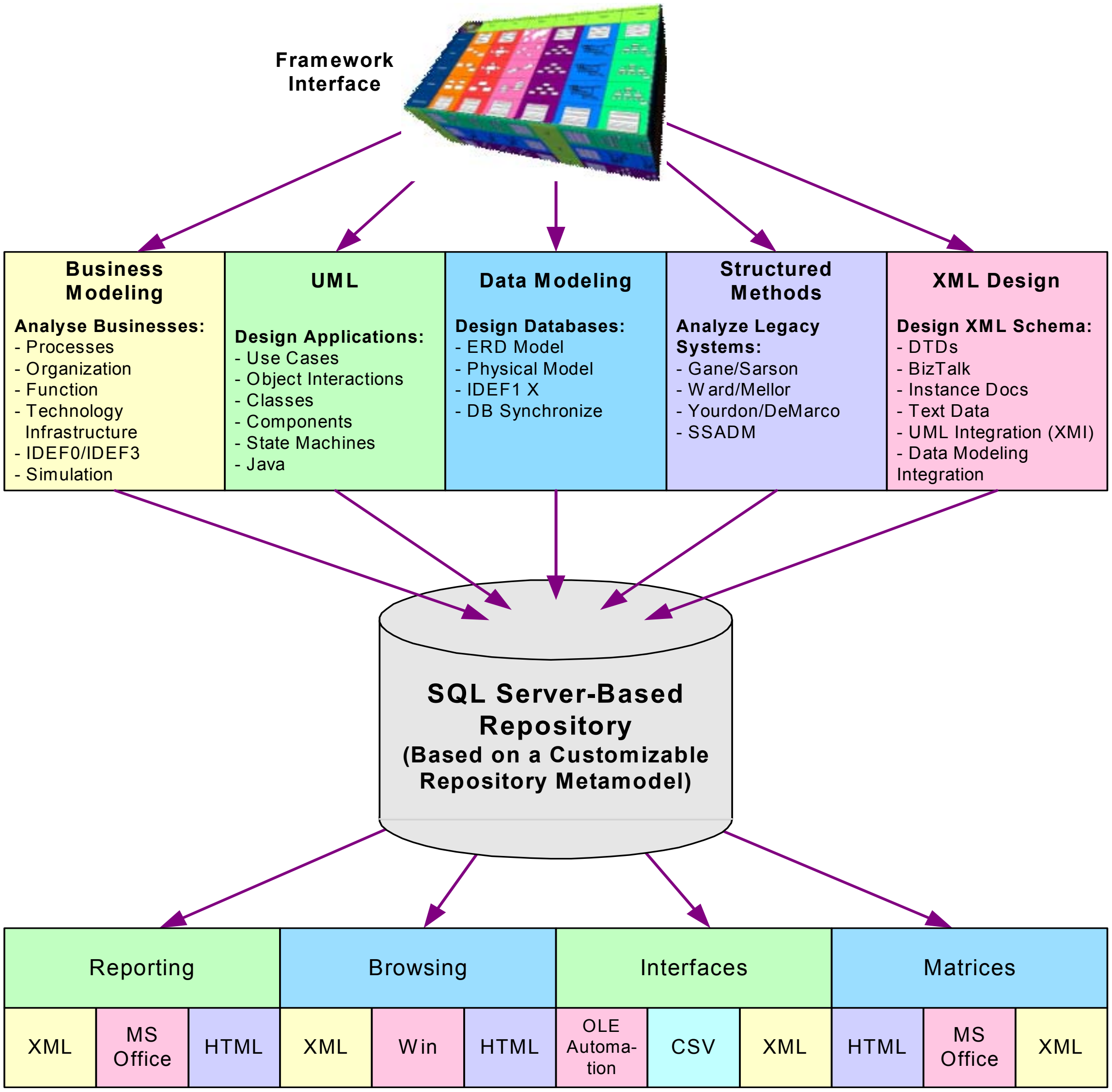

Figure 3. How Popkin's System Architect 9.0 Supports an Enterprise Architecture.

most do not. Thus, I will consider the general steps that a company might take to establish an enterprise architecture and a functioning Enterprise Architecture Committee.

How to Create an Enterprise Architecture

Most organizations go through a series of steps or phases in order to establish a viable enterprise architecture. The differences among company cultures and individual advocates assures that each company follows a more or less unique approach. What I can provide is a general overview of the major steps that most organizations work through.

Most organizations work their way through at least seven steps, as follows:

1. Agree on the Need
2. Establish an Organizational Structure
3. Select a Framework
4. Select a Tool and Repository
5. Organize the Existing Material
6. Begin Using the Enterprise Architecture
7. Extend and Maintain the Architecture

1. Agree on the Need

Nothing happens until senior management agrees that an enterprise architecture is needed. Thus, everything starts with an internal sales effort on the part of those who see the advantages of an architecture. Since the enterprise architecture is often a responsibility of the IS organization, the CIO or some other senior IS manager often leads the effort to sell other senior managers on the advantages of creating and maintaining an enterprise architecture. In other cases, a business manager who is responsible for business process planning takes the lead. The recent support for the enterprise architecture approach provided by analysts like the Gartner Group, and by initiatives like the one being pursued by the U.S. Government, have helped many groups sell the concept to their senior managers. (See the Sidebar, "The U.S. Government's Enterprise Architecture Initiative,")

2. Establish an Organizational Structure

Once there is a general agreement on the need for an enterprise architecture, the next question is who is going to manage the architecture. As I have already suggested, I believe it should be a function of the corporate planning group. I believe each company should establish an Enterprise Architecture Committee that creates and maintains the architecture and that serves as the central coordinating unit for all business process and IS planning.

Once the overall nature of the Enterprise Architecture Committee is determined, it must be assigned a manager and a budget. If the function is assigned to an existing group, like the corporate planning group, then its simply a matter of adjusting their budget and assigning new personnel to support the senior managers already responsible for corporate planning.

As a generalization, organizations with a well-defined process architecture are able to redesign or create new process more quickly. The creation and the maintenance of the architecture pays for itself by means of the savings organizations obtain on subsequent business process efforts.

3. Select a Framework

I described an enterprise architecture as a comprehensive collection of all of the plans, schedules, goals, documents, diagrams and models used to describe the organization, all integrated together and stored in a common repository. The term “framework” is often used as a synonym for an architecture. Others, however, use the term “framework” to describe specific approaches to organizing an enterprise architecture. If I assume that the Zachman Framework describes everything that I might want to maintain, then the Zachman Framework is simply a synonym for a comprehensive enterprise architecture. If an organization decided that it wanted to

The U.S. Government’s Enterprise Architecture Initiative

In 1996 the U.S Congress passed an law widely known as the Clinger-Cohen Act. As a result of hearings on government application development failures, Congress was convinced that U.S government agencies lacked an overview that allowed them to coordinate and management their development efforts both to achieve government mandates and to assure efficient software development. The Clinger-Cohen Act required each U.S government agency to develop an enterprise architecture. The assumption was that these enterprise architectures would assure that agency programs were aligned with agency goals, as assigned by Congress and specified by agency executives, and that software systems were developed to support those same programs. In other words, Congress believes that enterprise architectures will provide an improved basis for organizational decisions.

In 1999, the CIO Council, made up of CIOs of major government departments and agencies, published a benchmark framework, the Federal Enterprise Architecture Framework (FEAF) to provide various agencies with an generic approach. In 2001, the CIO Council published a document, *A Practical Guide to Federal Enterprise Architecture*, to help departments and agencies comply with the Clinger-Cohen Act. During the same period, several Federal departments published their own enterprise architecture guidelines. Among the most widely cited are the Department of the Treasury’s Treasury Enterprise Architecture Framework (TEAF), the National Institute of Standards and Technology’s Enterprise Architecture Model, and the enterprise architecture of the Department of Defense (DOD Architecture Framework).

The work by the various agencies of the U.S. government vary widely, but overall, they have become a major source of information on enterprise architecture development.

assemble some but not all of the documents and diagrams described in the Zachman Framework, then, in effect, it would be creating its own enterprise architecture framework. In essence, it would be defining its enterprise architecture as something less comprehensive than the Zachman Framework.

I mentioned in the sidebar on the U.S. Government's enterprise architecture effort, that each major department had been encouraged to develop its own framework and that each was slightly different. The U.S. Treasury, for example, adopted a framework that it calls the Treasury Enterprise Architecture Framework (TEAF). In this case, the TEAF matrix has four rows and four columns. In essence, the TEAF has omitted the bottom two rows of the Zachman Framework and combined four of the Zachman columns into two.

Each organization needs to determine just what types of documents and models it will include in its enterprise architecture. The result defines that organization's framework. Most organizations start with something like the Zachman Framework, and then tailor it to their specific needs. Each organization that elects to rely on an enterprise architecture must necessarily decide on the specific types of documents and diagrams they intend to keep track of with their architecture. Many begin with a subset simply to limit the time required to establish all the necessary relationships, in order to get an initial enterprise architecture in place quickly. Later, when the initial architecture is established, they expand the framework and commit to the addition of new information.

4. Select a Tool and Repository

An enterprise architecture must be managed and maintained by a committee or group of individuals. The group will necessarily need to keep track of a wide variety of documents, diagrams, models and charts. The only reasonable way to do this is to use a software tool that organizes the information and stores it in a repository, so that any member of the enterprise team can quickly obtain all of the documents, diagrams or models that refer to a given business goal or process, or to a specific activity or application used in a process.

5. Organize the Existing Material

Once a company has established an enterprise architecture group and selected a framework and a software tool, it should begin by collecting everything it can that is specified by the framework. A quick glance at the Zachman Framework illustrated in Figure 1 suggests the range of documentation required for a comprehensive enterprise architecture.

Most companies will find that they don't have much of the documentation in a form suitable for the repository. In many cases, plans and processes are not current, or are incompletely documented. In more cases, existing processes aren't documented at all. Similarly, only a few organizations currently have their software applications organized in a way that makes it clear what process is supported by which applications. Any enterprise architecture team beginning an effort will also face huge taxonomy problems. Activities in business processes are named in different ways than they are in software applications or in databases. Names will have changed without other documents being updated, and so forth.

The key to this effort is usually to start at the top and only go into as much depth as is practical during the period allowed for the creation of the initial enterprise architecture. Most companies can identify their corporate strategies and key plans. Similarly, they can probably identify key goals and measures and the key business processes and the major applications that support them. Start by documenting these items and their relationships. Leave the rest for later. Then, as new projects come up that require more detailed documentation, document them and place the new information in the repository. An enterprise architecture isn't a product, it's a process. Just like the organization it models, the enterprise architecture must be flexible, and designed to grow, just as the organization itself grows and changes.

6. Begin Using the Enterprise Architecture

An enterprise architecture, even an incomplete one, should provide planners with a better way of conceptualizing how a change should be organized and phased. Thus, most organizations begin to use their enterprise architecture even before it is complete. If those involved in planning don't want to use the enterprise architecture, it suggests that you need to reconsider how your enterprise architecture is organized, how it's accessed, or how it has been explained to various business managers. Like any other business tool, its use must be explained and those involved will need time to become familiar with it. As the enterprise architecture is used and familiarity grows, managers will learn the value of the architecture and will identify additional uses for it. In time the enterprise architecture will evolve into a key tool for the management of the organization.

7. Extend and Maintain the Architecture

Once the initial enterprise architecture is established, and the initial documentation is placed in the repository, the enterprise architecture group should proceed to extend and maintain the architecture. Improvement usually occurs in two more or less independent, simultaneous phases. First, the architecture group will identify missing elements that it will want to capture as quickly as possible.

A systematic process should be developed to generate and store additional documentation and diagrams in an orderly way. At the same time, the company should begin to use the architecture to scope and structure new projects. In effect, the enterprise architecture is a tool the executive committee and strategists can use to communicate with those charged with changing specific processes. In addition, the architecture committee will emerge as the group that establishes priorities among projects and monitors organizational changes. Obviously, each new business process change effort should be conducted in such a way that the documentation and diagrams produced, and the IS alignment attained via the effort, is all stored in the enterprise architecture.

Just as every company should maintain a business plan, it should maintain an enterprise architecture. In essence, an enterprise architecture is just a more extensively documented business plan. The more extensive documentation is required because companies are more complex and changes require a better understanding of all of the implications of any change. The payoff is a tool that makes planning and organizational alignment much simpler. The whole effort will come to nothing if it isn't expanded to include all the necessary documents and diagrams, and if it is not maintained once its created.

Aligning the Organization

One test of the usefulness of an enterprise architecture is its ability to assist with alignment, or to spotlight misalignment. In essence, if goals and measures are organized, managers should be able to see exactly how high level goals are implemented at successive levels of detail. In some cases, everyone will be happy with the iterative subdivision of goals, but in other cases, managers will probably find that goals become subtly distorted as they are implemented in lower level systems. While the enterprise architecture provides a tool that managers can use to see how goals and processes are implemented, throughout the organization, the managers themselves must decide if the various implementations actually accomplish what is intended by senior management.

Organizations are best aligned with goals and measures that determine if those goals are being met. Thus, creating a diagram that shows how a given strategy is to be supported by a specific business process, which in turn is implemented, in part, by software applications and databases, doesn't guarantee alignment. It merely guarantees that the person looking at the diagram knows all of the elements that are supposed to be coordinated to achieve the strategic goal. Even that, however, is a major improvement over situations in which managers are unsure what specific processes a given application is designed to support. Once one knows how the elements are intended to be aligned, one is positioned to use specific measures to determine if the processes, subprocesses, applications and databases are actually achieving their purpose. This is done by working top-downward and refining the goals and measures that will determine success. It begins with statements of what will constitute a success for the strategic initiative. That in turn, is used to determine what outcomes the process or processes assigned to that strategic goal must achieve. Subprocesses within processes are assigned their own specific goals. Applications designed to support sub-processes inherit their goals and measures from their processes, and so forth.

I won't go into the technical details of creating, assigning and monitoring measures here, but suffice to say that most organizations with enterprise architectures integrate their measurement systems with their architecture systems to assure a degree of organizational alignment and measurement that is impossible without an enterprise architecture.

Aligning Business Processes with Information Systems

Consider how an enterprise architecture can be used to align business processes with information systems. Figure 4 provides a more detailed and concrete example of how processes are aligned with information systems. In this case, I have taken a slice across the organization and identified four specific business processes: an Order Process, an Assembly Process, a Delivery Process and a Billing Process. Each of these processes is supported, in Figure 4, by a software application, and these, in turn, are supported by databases.

This view of the relationships among processes, applications and databases is a natural outcome of an enterprise architecture. Too many organizations, without a comprehensive architecture, associate applications with departments or specific functions. The keys to a well-organized architecture are (1) goals that are tracked through the layers of detail, and, (2) a set of business processes that provide an overview of the work and the outputs of the organization. Together, this information clarifies how processes link, directly or indirectly to customers, and how they achieve specific organizational goals.

When the Enterprise Architecture Committee receives suggestions for strategic changes, they should immediately translate those changes into changes in specific business processes. If the architecture is well-defined, changes in processes will immediately suggest changes in specific applications and databases.

It's important to stress that there is no one-to-one relationship between business processes and applications, or between applications and databases. Business managers usually think of processes in terms of how work flows through the organization and of how employees are organized to accomplish specific tasks. IS managers usually think of processes only as they relate to applications and databases they must support.

Different ways of organizing tasks at different levels should not represent a problem for the Enterprise Architecture Committee, although it does need to be understood. In essence, it is sometimes necessary to "translate" between the boundaries established for business processes and those established for software applications or databases. The development of an enterprise architecture can help a company understand just where these translations are necessary.

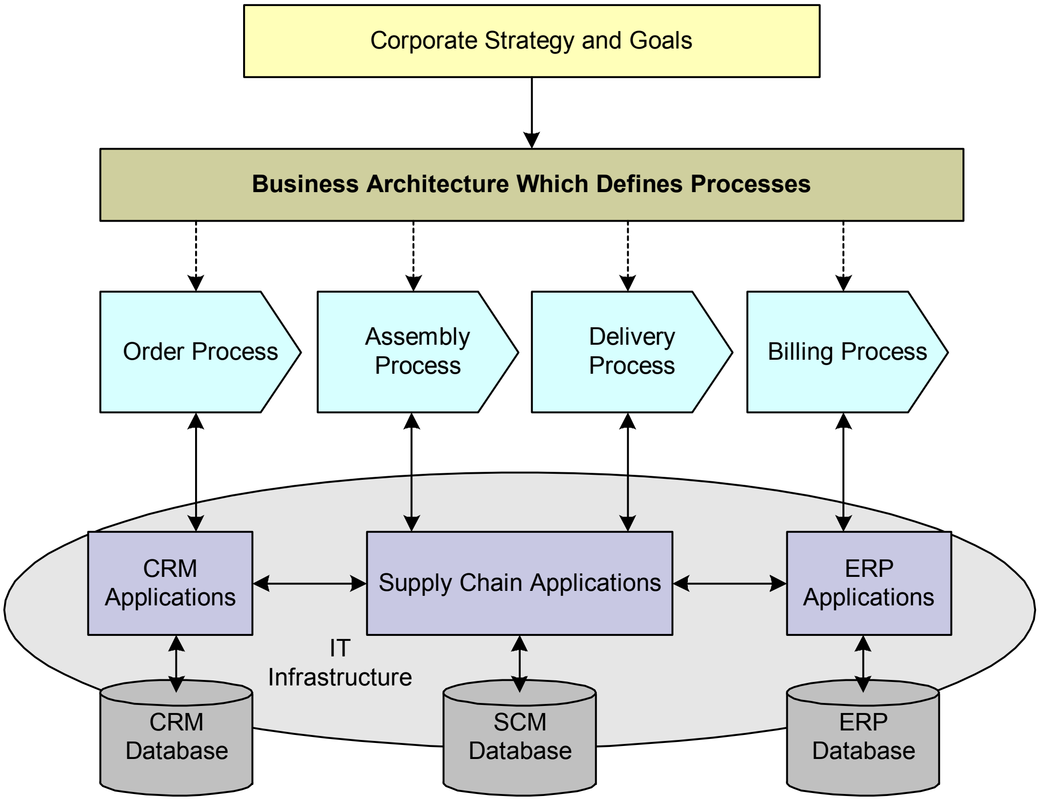

Figure 4. Aligning processes, applications and databases.

Summary

An enterprise architecture is a tool to help executives think about the organization as a whole. An enterprise architecture captures a wide variety of information, establishes relationships among the various documents and diagrams and stores all of the information together in a single repository, so that managers can then see relationships, ask questions, identify problems, or run simulations to help make decisions about changes they are considering.

I have tried to go beyond our general description of an enterprise architecture to suggest that there will be problems along the way and disagreements about how to best represent or align different elements. The organizations that succeed in spite of these problems will be those that clearly understand the ultimate value of the enterprise architecture and persevere.

Change keeps occurring faster all the time. Successful organizations need to be very flexible. Thus, every organization needs tools that help its managers deal with the problems involved in analyzing and implementing potential changes. In the course of the next few years, the enterprise architecture will evolve into one of the major tools of that organizations rely upon to manage change. It will provide the focus around which business and IS managers meet to discuss organizational goals, business processes, and organizational alignment. The enterprise architecture will assure that all necessary information is located in one place, and indexed to business processes. The simulation abilities provided by enterprise architecture software tools will further assure that managers can quickly run simulations to evaluate the implications of any specific change scenario. In other words, the enterprise architecture is the first step toward an enterprise that can respond in real time. It provides managers with the foundation they will need to quickly make the key decisions they will face in the future.■

Notes

For those interested in more information on the Zachman Framework, I recommend the Zachman Web site: www.zifa.com

A popular book that describes the development and use of enterprise architectures is ***Enterprise Architecture Planning: Developing a Blueprint for Data, Applications and Technology*** by Steven H. Spewak (John Wiley & Sons, 1992). This book starts from the Zachman Framework. It puts less emphasis on business processes and more on IS technologies than I prefer, but it contains lots of useful suggestions.

For those interested in a more comprehensive discussion of the steps involved in creating an Enterprise Architecture, I recommend ***A Practical Guide to Federal Enterprise Architecture***, a U.S. Government publication prepared by the U.S. Government CIO Council and released in September 1999. This document can be accessed via the CIO council Web site: www.cio.gov The publication does not consider how to initially establish or justify an enterprise architecture, since all U.S. Government agencies are required by law to establish and maintain enterprise architectures as a result of congressional legislation. Thus, this publication assumes the readers will comply, and focuses instead on the best way to implement the legislative mandate. There are specific documents describing the efforts at particular agencies that are also interesting. Those describing the enterprise architecture frameworks of the US. Treasury, the National Institute of Standards and Technology, and the Department of Defense are especially noteworthy.