

## Chapter 1

# Planning for Enterprise Realignment

---

### *In This Chapter*

- ▶ Defining the enterprise
  - ▶ Defining success
  - ▶ Preventing failure
- 

**I**nformation technology (IT) is everywhere in the business world, and you'd be hard pressed to find a business larger than a sole proprietorship that does not utilize some type of IT. When an IT decision is made, its effect can be felt throughout the organization. Poor decisions, such as those made without consideration of the impact on other elements of the enterprise, can create both immediate and long-term problems.

In this book you focus on enterprise architecture strategies and mechanisms that support both immediate and long-term (three to five years) planning. These strategies are used successfully in all types of enterprises, including small to mid-sized offices, educational institutions, and global commercial enterprises.

## *Defining an Enterprise*



The *enterprise* is a fluid term encompassing all technologies and tech-related policies that relate to services provided to clients, partners, and customers during operation of the organization. The more the enterprise interconnects elements, the more it becomes like a living organism — growing to meet emerging opportunities; consuming resources for sustenance; and generating piles of outdated, outmoded, or outright broken equipment that must be disposed of carefully. The enterprise requires planning to control its growth into useful areas, guidance to maintain its security and integrity during operation, and leadership to face the myriad personal preferences users will bring to their expectations of service value and function.

The strategies you explore in this book enable the enterprise to be stable but agile, which allows for both continuity of operations and the integration of new technologies.

## *Finding the Best Solution*

There's no perfect solution, no one-size-fits all strategy for enterprise architecture. As long as the technology meets the requirements, performs efficiently, supports business processes, is cost-effective, and can be supported and maintained, it's an acceptable solution, perhaps even a good one. There is no "best" technology, only the best technology for your enterprise.

Technology supports business, not the other way around. Technology should support business processes and align with strategic goals of your organization. Your technology choice should not limit your organization's functionality or future goals.

The strategies you look at in later chapters will help you make the right decisions for your organization, minimize cost, foster long-term planning capabilities, and create a stable and agile enterprise.

## *Providing Leadership*

To be an effective enterprise architect, you must provide leadership for the decision-making process; understand the impact generated by each technology selection; and facilitate communication of strategies, policies, and controls to implementation staff and clients.

An enterprise architect must possess both business alignment and broad technological skills in order to filter through user requirements and separate user preferences ("wants") from requirements ("needs,") while also seeing past the technobabble jargon that tech savvy clients and IT staff members often use when dealing with normal mortals.

As an architect, you must identify future technology trends, up-and-coming opportunities, and evolving security requirements to ensure that the current-state enterprise is properly prepared to meet emerging solutions and technologies. If not planned carefully and tested thoroughly, integrating new items like the immensely popular Apple iPad can be catastrophic on enterprise networks.



You must have the strength of vision necessary to stand firm and persuade concerned individuals and key stakeholders that some choices have got to be made from a larger perspective in order to reap the greatest benefits for the organization overall. You must be able to speak comfortably with chief officers and end-users, but also have sufficient technical credentials and understanding to be taken seriously by front-line technical staff members.



The worst thing you can do is present strategies to technical implementers and display a lack of real-world implementation experience, without sufficient updated and personal technical ability to be taken seriously. When lost, respect and support from the IT geeks may be impossible to recover, and the best possible strategies ignored or circumvented as a result. To perform effectively, you are obliged to continually extend your own IT skills through study and training. A purely nontechnical managerial staff member should never attempt to dictate technical policies or strategies because they lack understanding of the complex web of interconnection that forms the modern enterprise network.

The technical lead who fails to keep his skills current rapidly becomes a non-technical lead due to the rapid evolution of both technologies in use and the manner in which they're consumed by clients and knowledge workers. As an example, consider an IT architect whose skills were developed prior to the evolution of service-oriented architectural design, cloud computing, virtualization of storage and hardware, VDI implementation, Green IT initiatives, privacy and encryption regulatory mandates, and a myriad of other emergent options. This architect won't be able to effectively recognize the potential value these technologies can add to the organization's operations — or understand the limitations, cost, and impact of integrating them into the existing enterprise.

We discuss many of the IT leadership roles that may be present in an enterprise architectural project, together with a review of common IT governance and architectural frameworks, in Chapter 2.

## *In the Traditional Enterprise, Everything May Be Independent*



Many organizations still have traditional networks that are structured the same way they were 10 or 20 years ago — often due to a lack of technical knowledge update within the senior technical staff members, leading to a simple repetition of the same outdated functionality simply on updated hardware. Even if your organization isn't that old, chances are that unless modern enterprise architecture principles were involved in its initial design, you will still run into some of these old-school issues:

- ✓ Too many resource silos
- ✓ Too many platforms
- ✓ Too many people with root access

## *Too many resource silos*

In a traditional enterprise, it isn't unusual for each business unit to maintain control over its own information systems, including servers, workstations, data, and even networking hardware. Along with information systems, each unit also has its own technical personnel, makes its own purchases, and is responsible for backing up its own data. In essence, each business unit is its own autonomous network. This autonomy creates difficulties when anyone tries to access resources in another silo or share data between business units. It also leads to excessive duplication of resources and efforts, as each unit may have its own database server, file server, or e-mail server. As an example, I (Kirk) have seen multiple million-dollar-plus document imaging systems implemented by different business units using incompatible technologies, only because there was no enterprise-level coordination of an IT project portfolio. As enterprise architect, one of your tasks will be to consolidate these resource silos into a single, centralized data center.

Because local silos of information resources create inefficiencies and barriers to architectural design, we address the elimination of silos in many chapters throughout the book. Deal with this pervasive problem early in enterprise planning.

## *Too many platforms*

In information technology, a *platform* refers to a hardware or software framework. Examples of platforms include operating systems, hardware, programming environments, database management systems, and desktop or server configurations. In the old-school enterprise, you may find that many different platforms are in use. Administrators all have favorite technologies, and without a directive for standardization, administrators will push management to purchase these favored technologies. You may have to deal with a wide variety of operating systems, both server and workstation; multiple database solutions; or each programming team using a different programming language.

Another task will be to standardize platforms, which requires your vision and understanding of the organization's business requirements in order to keep the realignment process going even through the conflicts that will surely arise.

Chapter 3 includes an examination of technology standardization and its attendant benefits. Standardization is also key to adoption of new technologies enterprise-wide and to disaster recovery procedures, where complexity and customization can extend the recovery window significantly.

## *Too many people with root access*

Often, too many administrators have high levels of administrative access. This type of access is referred to as *root*, *superuser*, *enterprise admin*, *supervisor*, or *admin*, depending on the operating system or application in use. These accounts may even be used as the administrator's normal logon account, in defiance of security best practices. Unfortunately, root accounts are sometimes considered a status symbol and an indicator of the organization's trust. You may even find that nontechnical staff possesses this access. Managers may insist on root access simply because they're managers or because they want to keep an eye on their administrators, even though they don't have the skills or knowledge to actually do so. Yet another of your tasks will be to remove root access from people who don't truly need it.

## *In the Modern Enterprise, Everything Is Connected*

You can't decide on one particular technology without considering how it will affect all other technologies used in your enterprise now and in the future. For example, the selection of a new e-mail platform may seem simple, but it affects more than just how users get their e-mail. It also concerns the following:

- ✓ Directory services and authentication
- ✓ Network fax or voice mail solutions
- ✓ Instant messaging solutions
- ✓ Existing and future e-mail integrated applications
- ✓ Backup and recovery
- ✓ Data storage
- ✓ Security solutions



Selecting a particular application or programming language can affect your enterprise's future agility and impact business operational procedures. You have to base your technology selection on more than just user requirements and cost analysis; it must align with your organization's strategic business plan. Unless you have full understanding of both technical and business requirements, you risk limiting your organization's options. This understanding is necessary for success.

We discuss common collaborative technologies in Chapter 8. However, these technologies are not alone. A central set of standards should drive selection of platforms, standards for interoperability and communication, identify and access management, and all other functions within the enterprise to ensure that you can effectively integrate all existing functionality as well as newly emergent options into the enterprise fabric.

## Defining Success



To be successful, enterprise architecture must provide value to the organization. Change for the sake of change alone is counterproductive.

Although some criteria for success may be specific to your organization, enterprise architecture may generally be considered successful if it does the following:

- ✓ Reduces support and operational costs
- ✓ Defines technical standards
- ✓ Reduces risk
- ✓ Improves continuity of operations
- ✓ Reduces undesirable redundancy while retaining fault tolerance
- ✓ Facilitates business processes
- ✓ Allows for a clear upgrade path to future technologies

These indicators are all fairly straightforward, but another sign of successful enterprise architecture is that the organization sees it as valuable. Because enterprise architecture can have a significant effect on your organization's current and future capabilities and opportunities, your organization needs to be aware of the value provided by the architecture so that costs remain justifiable in the overall business plan.

## Using Maturity Models

*Maturity models* measure how your organization is progressing through an improvement process, and they're used extensively in process improvement, project management, and software development. The models consist of a number of levels, and as your organization matures and improves, it moves up in level. For example, the lowest level of maturity may be None, but when your organization begins to establish processes, even informally, it rises to the next level, which may be Informal or Initial. This process continues until the final level is reached, which is usually Continuously Improving, Audited, Measured, or something similar to indicate that the process is reviewed. Carnegie Mellon's Capability Maturity Model Integration (CMMI) is an example of such a model.

You can also use maturity models for enterprise architecture. Following are some of the more well-recognized enterprise architecture maturity models:

- ✓ Carnegie Mellon - Capability Maturity Model Integration (CMMI) ([www.sei.cmu.edu/cmmi](http://www.sei.cmu.edu/cmmi))
- ✓ National Association of State Chief Information Officers (NASCIO) - Enterprise Architecture Maturity Model v1.3 ([www.nascio.org/publications/documents/NASCIO-EAMM.pdf](http://www.nascio.org/publications/documents/NASCIO-EAMM.pdf))
- ✓ United States Department of Commerce - Enterprise Architecture Capability Maturity Model (ACMM) v1.2 ([ocio.os.doc.gov/ITPolicyandPrograms/Enterprise\\_Architecture/PROD01\\_004935](http://ocio.os.doc.gov/ITPolicyandPrograms/Enterprise_Architecture/PROD01_004935))
- ✓ United States General Accounting Office - Enterprise Architecture Maturity Management Framework (EAMMF) v1.1 ([www.gao.gov/new.items/d03584g.pdf](http://www.gao.gov/new.items/d03584g.pdf))



Maturity models are undoubtedly useful, but you may find that no published maturity models are a perfect fit for your organization. If that's the case, tailor the maturity model to your organization.

## Preventing Failure



Unfortunately, not every enterprise architecture project is successful, but how do you know if you're on the path to failure? Some of the indicators to watch for include

- ✔ **Allotting too much time to respond to problems and too little to planning and actually architecting.** If you're constantly putting out fires, you can't make progress.
- ✔ **Poor leadership skills.** To be an effective enterprise architect, you must be a leader. It isn't enough to have the technical knowledge; you must be able to take charge when necessary, foster open communication, and think strategically.
- ✔ **Neglecting to include business staff.** Remember that information technology supports business processes, and you must include business staff in enterprise architecture decisions in order to ensure that technology is aligned with business goals.
- ✔ **Lack of executive support.** For any enterprise architecture project to succeed, it must have the support of executive staff. Executives have got to understand the value of enterprise architecture so that they can provide proper support. When your executives back the project, corporate culture dictates that the changes to come are not optional.

If you notice any of these problems, it may be time to take a step back and re-evaluate your methods.