Defense Information Systems Agency
Center for Standards

DEPARTMENT OF DEFENSE TECHNICAL ARCHITECTURE FRAMEWORK FOR INFORMATION MANAGEMENT

Volume 4: DoD Standards-Based Architecture Planning Guide

Version 3.0

30 April 1996

Given the dynamic nature of the DoD environment, this document may contain terminology, diagrams, or policy discussions that are inconsistent with emerging DoD terminology, diagrams, or policy discussions. This document will be updated on a regular basis to incorporate necessary changes.

FOREWORD:
ABOUT THIS DOCUMENT

This edition of the Technical Architecture Framework for Information Management (TAFIM) replaces Version 2.0, dated 30 June 1994. Version 3.0 comprises eight volumes, as listed on the following configuration management page.

TAFIM HARMONIZATION AND ALIGNMENT

This TAFIM version is the result of a review and comment coordination period that began with the release of the 30 September 1995 Version 3.0 Draft. During this coordination period, a number of extremely significant activities were initiated by DoD. As a result, the version of the TAFIM that was valid at the beginning of the coordination period is now “out of step” with the direction and preliminary outcomes of these DoD activities. Work on a complete TAFIM update is underway to reflect the policy, guidance, and recommendations coming from theses activities as they near completion. Each TAFIM volume will be released as it is updated. Specifically, the next TAFIM release will fully reflect decisions stemming from the following:

- The DoD 5000 Series of acquisition policy and procedure documents
- The Joint Technical Architecture (JTA), currently a preliminary draft document under review.
- The C4ISR Integrated Task Force (ITF) recommendations on Operational, Systems, and Technical architectures.

SUMMARY OF MAJOR CHANGES AND EXPECTED UPDATES

This document, Volume 4 of the TAFIM, contains no substantive changes from Volume 4 of Version 2.0. Minor modifications have been made to acknowledge the evolving policies noted above. Substantive revisions to reflect these policy changes fully will be made in the next edition.

A NOTE ON VERSION NUMBERING

A version numbering scheme approved by the Architecture Methodology Working Group (AMWG) will control the version numbers applied to all future editions of TAFIM volumes. Version numbers will be applied and incremented as follows:

- This edition of the TAFIM is the official Version 3.0.

- From this point forward, single volumes will be updated and republished as needed. The second digit in the version number will be incremented each time (e.g., Volume 7 Version 3.1). The new version number will be applied only to the volume(s) that are updated at that time. There is no limit to the number of times the second digit can be changed to account for new editions of particular volumes.

- On an infrequent basis (e.g., every two years or more), the entire TAFIM set will be republished at once. Only when all volumes are released simultaneously will the first digit in the version number be changed. The next complete version will be designated Version 4.0.

- TAFIM volumes bearing a two-digit version number (e.g., Version 3.0, 3.1, etc.) without the DRAFT designation are final, official versions of the TAFIM. Only the TAFIM program manager can change the two-digit version number on a volume.

- A third digit can be added to the version number as needed to control working drafts, proposed volumes, internal review drafts, and other unofficial releases. The sponsoring organization can append and change this digit as desired.

Certain TAFIM volumes developed for purposes outside the TAFIM may appear under a different title and with a different version number from those specified in the configuration management page. These editions are not official releases of TAFIM volumes.

DISTRIBUTION

Version 3.0 is available for download from the DISA Information Technology Standards Information (ITSI) bulletin board system (BBS). Users are welcome to add the TAFIM files to individual organizations' BBSs or file servers to facilitate wider availability.

This final release of Version 3.0 will be made available on the World Wide Web (WWW) shortly after hard-copy publication. The Defense Information Systems Agency (DISA) is also investigating other electronic distribution approaches to facilitate access to the TAFIM and to enhance its usability.

TAFIM Document Configuration Management Page

The latest **authorized versions of the TAFIM** volumes are as follows:

Volume 1: Overview	3.0	30 April 1996
Volume 2: Technical Reference Model	3.0	30 April 1996
Volume 3: Architecture Concepts & Design Guidance	3.0	30 April 1996
Volume 4: DoD SBA Planning Guide	3.0	30 April 1996
Volume 5: Program Manager's Guide for Open Systems	3.0	30 April 1996
Volume 6: DoD Goal Security Architecture	3.0	30 April 1996
Volume 7: Adopted Information Technology Standards	3.0	30 April 1996
Volume 8: HCI Style Guide	3.0	30 April 1996

Working drafts may have been released by volume sponsors for internal coordination purposes. It is not necessary for the general reader to obtain and incorporate these unofficial, working drafts.

Note: Only those versions listed above as authorized versions represent official editions of the TAFIM.

This page intentionally left blank.

Preface

A key element of the United States (U.S.) Department of Defense'S (DoD) Corporate Information Management (CIM) initiatives for the 1990s is the implementation of a computing and communications infrastructure that will support portability, scalability, and interoperability of applications.

Deputy Secretary of Defense William J. Perry's policy memorandum of 13 October 1993 entitled "Accelerated Implementation of Migration Systems, Data Standards, and Process Improvement" reaffirms CIM principles and calls for all DoD components to begin migration from legacy to target systems in such a way "that migrate the system toward an open system environment and a standards-based architecture defined by the DoD Technical Architecture Framework for Information Management" (TAFIM).

In support of this goal, the ***DoD Standards-Based Architecture Planning Guide*** (the SBA Guide) has become Volume 4 of the TAFIM, which defines a common framework and profile of standards for the computing and communications infrastructure. The methodology prescribed in the SBA Guide provides a way of mapping the technology architecture, which is the primary focus of Volumes 1, 2, and 3 of TAFIM, to the three other views of an integrated architecture: work, data or information, and applications.

This version of the SBA Guide is an update of an earlier version that was written from October 1991 through April 1992 under contract #DCA100-91-C-0166. It presents a process for developing a standards-based architecture within the Department of Defense. At the time of this update, two major architecture engagements have been completed based on the use of the planning approach described in the earlier version of the SBA Guide. The goal of the updated SBA Guide is to incorporate recommended changes that effectively echo the lessons learned in the course of these two engagements.

The first major implementations of the SBA Guide were intended to test the methodology in a “small” enterprise (*Office of the Secretary of Defense, Office Automation Standards-Based Architecture*) and in a large-scale enterprise implementation (*Standards-Based Architecture for the U.S. Marine Corps).* The documents created as a result of these initiatives currently constitute the best reference source for the expected output from such an effort.

The planning process itself specifically addresses the Information Technology Policy Board (ITPB) Task 91-01 policy proposal approved 10 April 1991, which states:

> Develop a DoD standards-based open systems information systems architecture development methodology and establish a DoD implementation strategy.

The earlier version of the document was based on DMR Group, Inc.’s *Standards-Based Architectures*, Vol. IV in the STRATEGIES FOR OPEN SYSTEMS research program. This document, and its underlying architecture development process, was unique in its:

- New approach to gaining functional management understanding of, support for, and involvement in the information systems architecture process
- Explicit determination of broad organizational information systems architecture principles
- Explicit approach to creating an architecture based on standards
- Express design to produce “vendor neutral” architectures
- Proven application across a wide range of organizational types
- Immediate availability.

The process described herein is specifically designed so that all target architectures derived through its use meets standards and incorporates the generalized guidance on open systems environments (OSE) found in National Institute of Standards and Technology (NIST) Special Publication 500-187, “Applications Portability Profile

(APP): The U.S. Government's Open Systems Environment Profile OSE/1 Version 2.0."

Corporate Information Management practices and policies are still evolving. As they do, this SBA Guide will also require changes in its diagrammatic representations, terminology, and policy discussions.

This page intentionally left blank.

An Executive Summary

This document was developed to assist users in the United States (U.S.) Department of Defense (DoD) in planning technology architectures based on standards-based platforms. It can be used within a functional unit or department within the DoD (e.g., the Marine Corps). The approach may also be usefully applied at a lower, or sub-department, level to provide a more detailed view of the architecture.

Target audience

Process facilitators constitute the primary audience of interest for whom this document was created. Experience tells us that this planning process is most successful when it is led by someone who can bring an impartial view to bear on the consensus-building process that is central to the success of the effort. An impartial and professional facilitator, experienced in the standards-based architecture (SBA) process, is essential in getting the process off the ground. The facilitator will keep the process on track when local, political, or technical perspectives threaten to get things moving in the wrong direction or risk derailing the process. This is said in recognition of the fact that many of those asked to participate in the process are likely to bring with them parochial views or hidden agendas that might not allow them to work effectively toward the common goal of developing a mission-specific architecture. The best way to address these issues is through reliance on a facilitator who can identify stumbling blocks and move the team around or over them.

The facilitator will have experience in facilitating workshop sessions with key knowledge workers to elicit required architectural content. The facilitator will also possess the ability to tailor the basic methodology as needed to satisfy the unique demands of the enterprise being modeled. Thus, for the facilitator, this SBA Guide becomes a sourcebook for customizing the specific methodology to meet the specific goals of the organization involved.

Other audiences will also be interested in this document. It can be used as a marketing tool to expose prospective participants or sponsors to the process and educate them about what the SBA planning process can achieve. It is also useful to those involved in the process to help them understand the importance of each step they are involved in and how one step serves as a basis for work to be done in successive steps of the process.

The SBA Guide can be used to “hand hold” those involved who may occasionally feel lost or overwhelmed by the task in which they are involved.

This document *is not* a detailed methodology describing all attributes of information modeling, application development, security architecture, or detailed technical implementation project planning, nor does it describe the methodology by which “business process redesign” is accomplished.

While this document discusses such subjects, it is not intended to provide the reader with a detailed understanding of those methodologies and techniques. Furthermore, it was not designed to develop a single monolithic DoD architecture for a single computer and communications solution that will fit all users across the Defense community.

Volume 4 of TAFIM

The SBA methodology that is described in this guide is based on four views of an integrated architecture: work organization, information, applications, and technology. Volumes 1, 2, and 3 of the Technical Architecture for Information Management (TAFIM) focus primarily on the technology architecture. The SBA methodology, which constitutes Volume 4 of the TAFIM, provides a way of mapping the three other views (work, information, and applications) to the technology architecture.

The SBA planning process is an especially important part of the TAFIM because it fleshes out the work, information, and application views of the architecture. It provides a mechanism for translating the functional, or business, needs of the enterprise into the information technology (IT)-based solutions that ultimately flow from implementation of the entire TAFIM process.

The planning process helps align IT with the business needs of the organization. The *DoD Standards-Based Architecture Planning Guide* describes how the overall process of planning for, and implementing, a standards-based architecture is conducted, highlighting some key considerations in the overall effort. Because of the velocity of change in technology, which seems to be increasing, this process may be amended, adopted, and modified to conform to existing IT planning approaches that may already exist in DoD functional areas. Most importantly, it outlines a simple but effective process users may follow to arrive at a technology architecture based on standards.

Reusable building blocks

The ultimate goal of such a process is to yield reusable building blocks that can be used in each additional DoD component as it launches its own SBA planning process for the first time. While this version of the SBA Guide is based on two completed implementations, two other implementations are already under way, with other additional projects expected to follow. Some of the output from the past implementations is beginning to be replicated in the next round. As the DoD develops more understanding of the similarities observed across the entire organization, it can begin to understand how entire business processes may be supported by architecture in an identical way across the various components.

As an example, “work process” may be seen to constitute a reusable building block of the larger enterprise. If the work process “Acquiring Personnel” becomes a standard work process across DoD departments, then the IT architecture that supports this business, or work, process can be borrowed from implementation plans already available rather than having to “reinvent the wheel!”

Figure 1 represents the standards-based architecture planning and implementation cycle outlined in this SBA Guide.

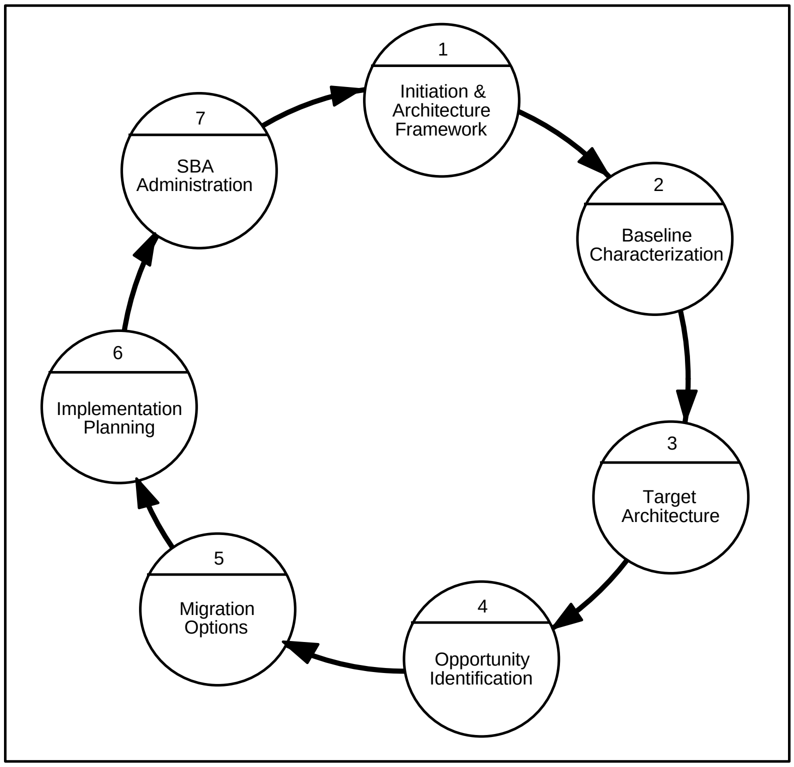
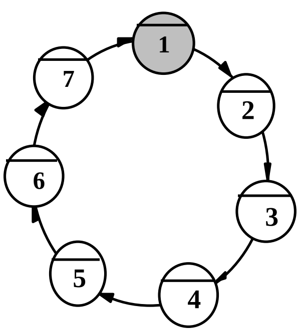


Figure 1. The DoD Standards-Based Architecture (SBA) Planning Process

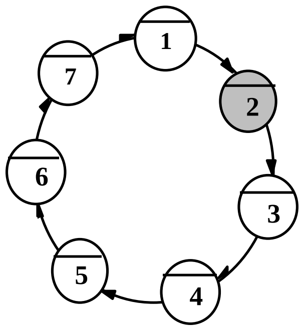
SBA process steps

The SBA planning process consists of seven distinct, but interdependent, phases. Each phase of the SBA process is intended to create specific deliverables which then guide the subsequent step(s). The phases and their deliverables are briefly outlined below:



1. **Initiation and architecture framework.** The methodology begins by properly initiating the process within the host organization. Once the process is properly sponsored and staffed for optimum effectiveness, it is possible to move on to the actual steps necessary to develop the architecture.

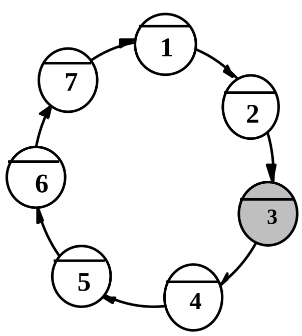
   This orientation phase involves reviewing (or in some cases developing) a set of strategic drivers for the organization. The business model is reviewed (or built) during this project phase to establish a strategic target operational model. Lastly, a set of architecture principles is developed, usually in workshops, to establish what are believed to be good architecture practices for the organization.



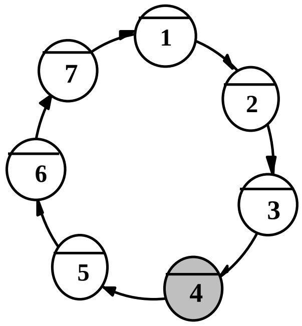
2 . **Baseline characterization.** This is a grounding phase to determine where an organization is currently situated architecturally. It is not an operational review or audit but more an assessment and characterization of the current environment. It is used to establish a baseline or starting point for architecture development. The architecture framework provides an effective means for organizing this review and presenting the current status.

The baseline characterization phase results in a picture of the existing architecture along four key dimensions, or views: work, information, applications, and technology. The term “characterization” is used because the data gathering and analysis are not exhaustive. It is not necessary, nor is it desirable, to expend the time and effort to document every detail of the current architecture. Only enough detail is gathered to allow informed decisions to be made with regard to the desired target architecture (described below).

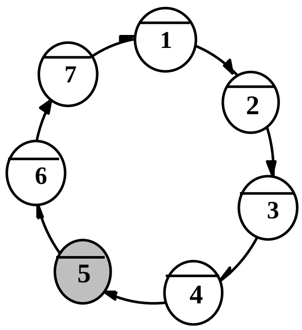
The current situation in each of the four views and their interrelationships will be characterized by completing a series of instruments, or templates. These templates are similar in content and style to the deliverables that will be used to define a target architecture. This will facilitate “gap analysis” for migration and implementation planning in future phases.



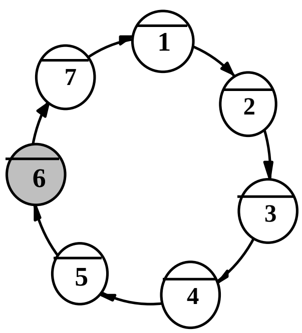
3 . **Target architecture.** This is the heart of the process, where the various views of the framework are modeled in terms of a desirable target architecture, usually 3 to 5 years in the future. The process consists of defining each set of architectural components and its key attributes. The components are then used to define desired relationships using affinity analysis. The result is an organized set of definitions and models from which drawings can be made to reflect the different views of the architecture.



4 . **Opportunity identification.** This phase moves the architecture out of the conceptual world into one where the practical realities govern implementation. In this step, short-term opportunities are identified which, once implemented, can demonstrate the value of the architecture and provide immediate benefits to the organization. In addition, all projects that are necessary to achieve the target architecture are identified and fleshed out in some detail.



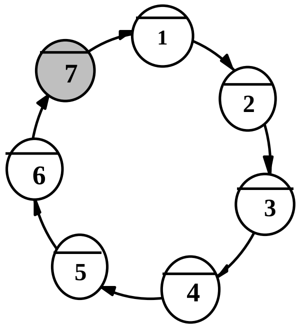
5 . **Migration options.** This phase links the reality of the present with the desirability of the target architecture by establishing one or more plateaus representing practical migration stages. The same types of models, using the common framework, can be used to represent these evolutionary plans. All projects identified in the previous step are prioritized over time based on inter-project dependencies and cost/benefit analyses.



6 . **Implementation planning.** This phase results in a detailed implementation plan for the first plateau of the migration effort. It constitutes the first wave of actionable projects that establish the groundwork for each successive plateau of the target architecture implementation. Plateau 1 projects are generally linked to the next stage in the migration plan. Responsibilities are established to ensure that they are carried out and that the migration plan is properly updated.

The outward manifestation of the architecture is also reflected in a set of standards and guidelines to be used by the organization in acquiring technology and developing applications. They can relate to any or all components in the models. Areas where standards are required most urgently can be identified for quick resolution and others assigned for later investigation.

The activity of identifying standards and guidelines for technology acquisition is informed by Volume 2 of TAFIM and by guidance provided by other Government-sponsored initiatives such as the Application Portability Profile (APP) developed by the National Institute of Standards and Technology (NIST).



7 . SBA administration. This phase is intended to keep the architecture alive and well by continuously improving it. This phase reflects the need to adjust architecture decisions in accordance with unforeseen changes in business directions or advances in technology or its availability. It should also be used to make adjustments based on experience and ensure that modifications in standards and supporting processes reflect a realistic approach. This review process can cause a reentry into the process at any point depending on the area to be adjusted or updated.

Essentially, this management activity ensures that the SBA planning process already is, or is soon to become, well integrated with the mainstream IT planning process within the organization. If it is treated as a special project, or in other ways is not fully institutionalized, the ability of the process to result in funded projects will ultimately suffer. The outcome of this step is a direct reflection of how successful the project initiation was in the first place. We cannot overemphasize the importance of properly positioning this process within the day-to-day operation. High-level sponsorship at the front end will contribute to success at the back end. This is true for a number of reasons that are discussed in Section 8.

Critical success factors

Experience has shown that there are lessons to be learned in how best to conduct architecture planning. The following represents a list of critical success factors that have been established:

Business driven

Wherever possible, use the architecture process to reinforce support of key operational and business drivers.

Participative process

Involve teams of architects, planners, and managers directly in the creation and review of deliverables. Establish corporate “buy-in.”

Fast paced

Set schedules such that deliverables arrive within weeks, not months. Show early results.

Presumptive resolution

Do not get bogged down if facts or information are not available. Be presumptive, make the best guess, and document assumptions.

Architecture, not design

Avoid too much detail. Focus on architecture decisions and save some creative work for the designers to follow.

Minimum set

Do not set out to establish standards for everything in sight. Focus on those where key infrastructure is involved and leave the user departments to sort out the rest.

Key deliverables

It is more important to produce results that everyone can abide by than to follow specific processes or methods. Use the framework but be creative and experimental with methods using standard DoD tools and techniques.

Open, non-secretive

Do not hide the team away and stamp everything "confidential!" Invite participation and circulate drafts for review and discussion. Avoid alarming affected parties.

Ongoing process, not event

This is not intended to produce a shelf document and then allow everyone to get back to their former ways of making IT decisions. Creating ongoing processes for updating and reviewing are critical.

The SBA Guide is organized around the seven phases and associated critical success factors.

Overview of the DoD SBA Guide

This SBA Guide contains eight sections, each dealing with a specific topic:

Section 1
Introduction

Provides a context for this document and describes what the SBA planning process is and why it is important.

Section 2
Initiation and Architecture Framework

Describes Phase 1 of the process whereby an organization develops "architecture principles" and develops a common vision for the development of a standards-based technology architecture.

Section 3
Baseline Characterization

Outlines the overall process that is followed to conduct a high-level inventory of applications, platforms, and standards in place in the function.

Section 4
Target Architecture

Defines the steps and processes involved in developing a target architecture based on standards.

Section 5 *Opportunity Identification*	Illustrates how the Architecture Working Group (AWG) categorizes and identifies opportunities for exploiting the target architecture.
Section 6 *Migration Options*	Provides a framework for developing migration options to the new standards-based architecture.
Section 7 *Implementation Planning*	Defines how implementation project planning occurs and describes the steps by which the near- and mid-term benefits of the architecture are obtained.
Section 8 *SBA Administration*	Looks at the challenge of improving the new architecture over time to assure that incremental improvements are made on a continuous basis.
Appendices	These provide in-depth content and guidance in selected areas outlined by the individual sections.

This page intentionally left blank.

Table of Contents

List of Figures

Section Three: Baseline Characterization

Section Four: Target Architecture

Section Five: Opportunity Identification

Section Six: Migration Options

Section Seven: Implementation Planning

Section Eight: SBA Administration

This page intentionally left blank.

Section One: Introduction

Table of Contents **Page**

Figures

Section description

As the *Foreword: An Executive Summary* stated, this document is not a formal methodology. It is a standards-based approach to standards. Why are standards so important to IT architecture? Simply put:

A new technology paradigm based on the concept of open network computing is emerging. It is driven by advances in technology and a combination of growing interdependence and heightened competition among functional organizations. Standards are “the glue” that enable users to interoperate seamlessly across applications, platforms, and organizations. Today’s reality is that users are confronted with islands of automation—myriad and redundant computer systems that have been used to automate non-standard, and frequently inefficient, functional processes.

Standards-based environments are delivering important benefits to organizations in two main categories: reduced cost of IT and its management, and improved IT effectiveness through the creation of more flexible, modular, and powerful IT infrastructures.

Obstacles to the adoption of open systems include users’ lack of awareness and current investments in proprietary systems, the immaturity of several open systems technologies, and the confusion caused by competing standards efforts. Nevertheless, the open systems “train” has left the station and it will not turn back. Users within DoD need a “standardized” standards planning process for IT. Lack of such a process has resulted in planning and implementation delay. All functions face the challenge of migrating to standards-based technology while prudently managing the installed base of proprietary systems through the interim period towards a standards-based target architecture.

The new IT architecture

IT architecture plays a key role in making IT user requirements work. Traditional computing environments based on proprietary products and isolated data processing systems have resulted in a costly, poorly integrated, and hard-to-change infrastructure in most organizations. IT architecture should provide a coherent blueprint by which systems are integrated into an interoperable whole.

A new, volatile, strategic and operational environment demands new capabilities from IT that traditional computing environments cannot deliver. Rather than upgrading their current environments, leading organizations are setting out on a course of migrating to a new environment based on the new technology paradigm. Research shows that functions that are retooling invariably conclude that a new network architecture can only be achieved through the adoption of standard interfaces and components.

The result is the emergence of the *"standards-based"* architecture. Such a function-owned architecture can include the vendor-independent standards associated with open systems. A standards-based architecture will include a migration strategy from interim proprietary standards to open standards.

The standards-based architecture is based on a number of components that do not appear in traditional technology plans. These include architecture principles, definitions of generic components, and a set of industry standards supported by products and technologies that adhere to those standards. It defines reusable and interchangeable architecture components that promote flexibility and modularity in the architecture.

What is architecture?

An analogy can be useful in understanding what an architecture is and why it is important.

IT architecture is the underlying framework that defines and describes the IT platform required by a function to attain its objectives and achieve a functional vision. It is the structure given to information, applications, and organizational and technological means—the groupings of components, their interrelationships, the principles and guidelines governing their design, and their evolution over time.

Like planning for a building

An IT architecture is analogous to the architecture for a building. The plans for a building include provisions for the various services to be offered in the building, such as electrical power, plumbing, communications wiring, stairwells, and elevators. They must also provide the overall design of the building (i.e., its construction specifications, how many floors there will be, the look of the exterior and interior walls, etc.).

An architecture plan must also consider zoning laws, regulations and standards for building usage, such as set back from the street, orientation on the lot, and blending with the existing environment. It must also consider the ingress and egress, general work patterns of the desired tenants, layout of the equipment that may be housed in the building, and the type of construction material needed to meet the usage requirements of each area of the building.

The architecture must ensure that components of the building fit together to meet the needs of the prospective tenants and the surrounding environment. It must also have the ability to evolve with the changes that time may bring, perhaps the need for expansion or for alternative uses.

The architecture does not, however, concern itself with details such as the specific color of carpet a given tenant may want, or exactly how each person’s desk will be oriented, or even how each individual office space may ultimately be built out to suit the tenants’ cosmetic or work flow needs.

Rather, the architecture concerns itself with providing a flexible, adaptable infrastructure to meet these varying needs without tearing down the building and starting over. This is accomplished by adhering to solid principles of architecture design, by developing a set of blueprints (or frameworks) for the building’s appearance and layout, and by setting some basic standards for the construction teams to follow as they implement the plans.

Typically, the architecture does not specify particular vendors or suppliers for the components of the building. Instead, it provides flexibility by setting standards for the components, which may be met by one or more suppliers. In this way, competition among alternative suppliers allows the architect and construction teams to keep costs in control while minimizing the risk associated with sole source relationships.

Of course, as the construction begins, some specific decisions will have to be made about vendors as well as the details of construction for a given tenant. In the construction planning phase, the architecture still forms the framework for decision making, but more detailed plans will have to be developed for each tenant’s specific requirements. Here, the cost of materials, durability requirements, specific equipment locations, and office layout must be considered. A detailed design must be developed with specific cost estimates, time to complete, and vendors to be used. This goes beyond architecture planning but must remain true to the architecture principles and blueprints for the overall building.

The analogy

There is a direct analogy in the IT area for each of the points discussed above. The architecture principles for the building define the overall style of the building and its general characteristics, given its envisioned usage. Similarly, the IT architecture principles are the foundation for decision making about the general style of computing and technology usage for the company.

For example

"The building will be a skyscraper, no more than 60 floors, envisioned for general office usage, of steel and glass construction with non-opening windows, in the style of a monolith, with integrated underground parking, pre-wired for high-speed telecommunications on every floor, with external elevators facing the bay."

With these principles, one gets a fairly good idea of the kind of building this will be, and some of the constraints that will be placed on vendors who may qualify to work on the project as subcontractors.

In IT, the principles provide a similar mechanism for defining the kind of information systems we will have.

"To the extent possible, similar business functions will be supported by common systems, which will support all physical locations. These systems will be run locally, within each plant location but will be maintained and updated from a central location.

The systems will be developed within an industry standard environment and will be interconnected for data sharing via a series of interconnected telecommunications networks, which will communicate using industry standard protocols. Access to all systems will be via intelligent workstations connected to the network and using a set of common user interface standards."

A starting point for detailed design and system construction

Just as the artist's rendering and a general description of a new building's characteristics are not enough for the construction crews to do their work, the principles of an IT architecture are not sufficient to allow the system designers and implementors to construct appropriate information systems.

In the case of the building, realistic scale models of the structure are developed to aid the architect in envisioning how the various subassemblies of the building will all fit together. Blueprints of the mechanical, electrical, structural, and other aspects of the building will also be developed.

These blueprints and associated specifications define the overall infrastructure of the building, envisioning the needs of the classes of tenants who are likely to occupy the space. The basic services of the building are defined and placed within the infrastructure, usually according to a set of well-defined industry standards and codes.

There is a direct correlation in the development of IT architectures. The principles are used to guide the development of models and associated specifications for the way the organization will use IT.

IT architecture models are like an architect's blueprints

The four views of an IT architecture (the way work activities are organized, the information needed to perform the work, the automated systems that capture and manipulate the information, and the technology environment within which these automated systems run) are analogous to the detailed architecture blueprints and specifications for the subassemblies of a building as described above.

As with the building blueprints, the IT architecture models must anticipate the classes of users, their location within the organization, the type of work they must do, and the anticipated need for automated systems in these locations. It must do so without knowing in advance all the details of each automated system that may be needed by these users in the future.

The bottom line on architectures, for buildings and for IT, is providing a minimum, but rigorous, set of guidelines and standards that will allow the building (or information systems) to be developed in a way that will allow the most flexibility for the tenants (or system users) while constraining the detailed designs enough to ensure that the desired style and characteristics of the building (or the computing environment) are maintained over time.

With these principles, the style of computing and communication is defined in enough depth to allow appropriate detailed design work to begin and vendors to be selected.

What is IT architecture planning?

So, with the prior analogy as a backdrop, we define architecture planning as the art and science of transforming a functional need for computer-based systems into a planned and organized framework that supports integration and enables systems design and delivery.

Architecture planning proceeds on three fronts:

- The definition of a commonly accepted framework around which architecture decisions can be based
- A clear definition of organizational responsibilities and planning procedures is required to ensure architectural integrity
- Each major systems project requires a level of architecture planning based on these guidelines and organization to address specific system requirements.

A new approach to architecture planning

The need and opportunity to create a functional IT architecture based on standards are both new. Similarly, the new functional imperatives and the new technology paradigm demand a new approach to technology planning and migration.

Traditional architecture planning only focused on application and data design to support individual applications. Methods were based on techniques that limited scope and created hard boundaries. Solutions were evaluated and chosen based on specific vendors and products. Criteria emphasized functional fit and cost, not architecture considerations.

The new SBA planning approach is quite a different proposition. The new approach to SBA planning deals with both the structure and style of computer-based systems. It requires the definition of architecture components or "building blocks" and ways to describe the relationship among architectures. IT architecture provides that often elusive link between identifying a strategic opportunity to apply computer solutions and choosing the best available solution. Most importantly, it describes the standards upon which these building blocks are assembled.

Multiple views of the architecture

The IT architect must serve a number of communities of interest. It is therefore necessary that the architecture framework support the communication needs and viewpoints of these various interest groups.

Standards-based architecture is also multifaceted. While constantly relating to strategic functional requirements, architecture must reflect four different views of the transformational change involved in using IT. These four views are:

- **Work organization view.** How will the planned system impact work activities (nature and magnitude), change skill requirements, affect functional operating locations, and eliminate or reduce manual support systems?
- **Information view.** What information bases are required to operate the function? What forms and volumes of information are involved? What relationships between the information bases must be provided? What access and security controls are required?
- **Application function view.** What types of application functions are required to support the transformed organization and associated users? How will functions be grouped and interfaced? What usage levels are anticipated?
- **Technology view.** What types of technology services are required and how should they be distributed to various types of technology platforms? How will these services and platforms be networked, and what standards and guidelines are required to support integration?

The four views of the integrated architecture are shown in Figure 1-1.

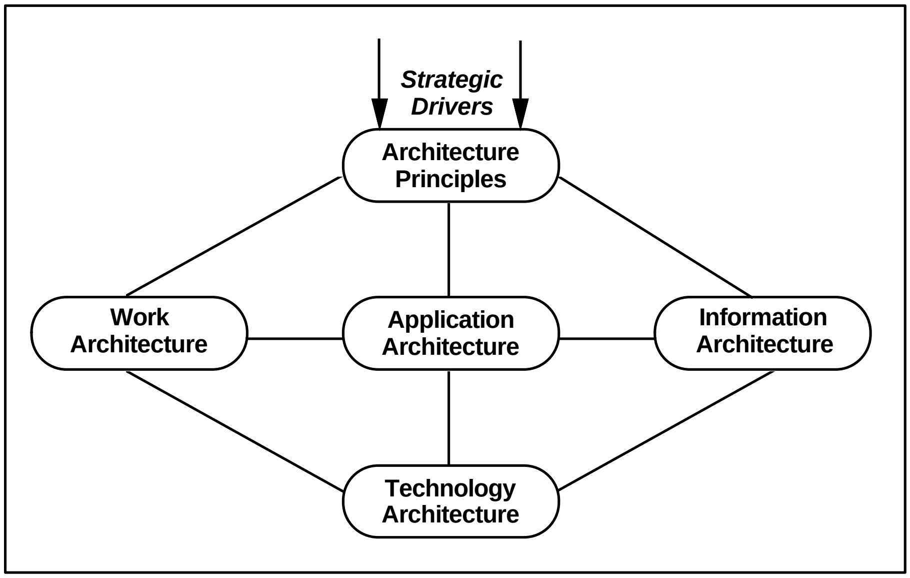


Figure 1-1. Architecture Modeling Framework

The architecture principles, and their upward link to the strategic drivers of the enterprise, provide the basis for reflecting the strategic use of IT—the domain of the executive group and strategic functional planners. They are used to show how the operation of the function will benefit from the transformation changes enabled by IT. They provide the functional strategists' views of the architecture and are used to drive out the predominant architecture principles.

Work organization view

The work organization view describes the major operations that are performed by work groups in support of functions. It defines the types of work (logical working units) in terms of the types of workers (classes of IT users) and types of work locations (places where the functions of the organization are carried out).

The work organization view should be independent of line organization design. Many traditional IT solutions were tailored to specific line organizations, resulting in hard boundaries and inflexibility. Work organization modeling recognizes the realities of "networks" of individuals and their supporting automated and manual systems. It supports the team concept, the multiple roles (or team memberships) that individuals can have, and recognizes that teams can be composed of members who work remotely from each other.

It also should recognize external users and external functional locations. Key external constituencies (e.g., legislative organizations such as Congress) and suppliers are obvious candidates. Employees working from home office locations or while traveling should also be considered for inclusion.

The work organization view helps to describe the before and after impacts of technology on the organization. It becomes the basis for detailed redesign of work processes, communication programs, and user training to address change management requirements.

Information view

The information view describes the information used by the organization and the relationships among collections of information (subject databases).

It is important to include all forms of information and types of media in this view. Again, placement and distribution to working locations in support of user and application access is a key consideration.

Application view

The application view shows which functions of the organization can be supported by IT applications. It provides a high-level description of these application opportunities. It also shows logical dependencies and relationships among application opportunity areas.

This view defines the scope and interfaces of applications and provides the basis for detailed design. It identifies specific work groups and users of applications, their relationships to information, and their placement or possible distribution across types of locations and technology platforms.

The application and information views are used in tandem to define the targeted applications and information that will support the organization. Together they drive the requirements for technology.

Technology view

Technology views are used to describe the enabling infrastructure. To provide the necessary linkage to the work organization, information, and applications architecture views, the technology view can further be described in terms of some generic building blocks. These include: Generic Application Environments (GAEs), Generic Technology Environments (GTEs), and Generic Technology Platforms (GTPs). These are described in Appendix D.

Architecture modeling frameworks and their uses

The architecture modeling framework defined above has been developed to support the IT architecture planning process and related deliverables. The modeling framework has many uses:

- It is used to explain the meaning and concepts of architecture planning, particularly the multiple views and purposes that a complete IT architecture must serve.
- It provides a basis for describing the current IT architecture and assessing its strengths and weaknesses.
- It is used to describe the target IT architecture. It provides all the necessary components to describe the required architecture that best supports the strategic directions of the function. It provides the generic components from which specific target environments and their interrelationships can be modeled. In particular, it can be used to determine common requirements that exist within and across organizational units. These common requirements provide the basis for defining infrastructure. The resulting infrastructure views then provide the basis for defining standards and guidelines for component design and acquisition.
- Finally, the modeling framework is used to guide the major steps in a migration strategy to bridge the current and target architectures. Consequently, it can be used to update the progress toward the target as well as to adjust architecture plans to reflect changes in functional direction or unforeseen technology advances.

In most organizations, IT architecture planning is a relatively new endeavor. Early attempts usually focused on only one or two of these four views, with little regard for the others. It is important that standards-based architectures reflect a balance of these four views of their relationship.

As a result of the newness of architecture planning and the accompanying high rate of change, the “science” component of architecture is incomplete and inconsistent. Businesses typically lack the common language and disciplined approach necessary for architecture planning to serve its practitioners and communities of interest.

Goals of an architecture

Given this, an architecture must address three goals:

- Provide a means of cost effectively organizing information and its technologies to support the organization’s objectives
- Improve the effectiveness of IT in delivering new capabilities to the organization
- Facilitate continual evolution of the IT infrastructure and solutions over time.

The approach outlined herein attempts to do just that—provide a step-by-step process that may be used in a typical function. It may be amended, adopted, and modified to conform to the standard IT planning approaches that may already exist in the enterprise.

The questions it addresses are:

- By what process can we define a standards-based architecture that meets our functional vision?
- How do we get from here to there?

Large enterprises, for example, cannot discard large investments in proprietary mainframe and mid-range applications and hardware. They cannot suddenly switch to an operating system such as UNIX merely because it is more “open.” Likewise, users who have a considerable investment in PC-DOS machines cannot adopt X/Windows overnight if the changeover requires conversion of 10,000-20,000 workstations already field deployed.

A multivendor environment is one characterized by hardware and software diversity. These distinct and unique environments are generally required to work together at the function level. This requires a high degree of technical and operational coordination. In most organizations, this occurs on a “patchwork quilt” basis at best.

The standards-based enterprise focuses on standards-based architecture in a "diverse" technology environment because it enables these diverse environments to interoperate effectively. A key characteristic of an open systems environment is the critical need for "rules of the road" or regulated standards. For open systems to work effectively in an organization, the standards-based organization must have a method for developing a enterprise-wide standards-based architecture.

Traditional IT planning approaches

To understand the new approach to architecture planning let's begin by assessing the inadequacies of existing IT planning methodologies.

Many organizations have tried using a traditional IT planning model. Frequently these IT planning approaches, while interesting exercises, are never implemented in the traditional organization. The reasons for this lack of implementation are organizational, functional, or technology changes that occur before action is taken. These "strategic" plans have typically been built on 3- to 5-year time horizons, with linear project plans that take several years to complete. The fundamental problem is that the planning processes do not reflect the reality of today's operational or functional environment.

Traditional planning approaches, when conducted properly, model a function or organizational entity and outline programs for applications, data, and technology platforms. The output from these planning exercises is a document that often represents the culmination of many person years of planning across a function. In many organizations, such plans are frequently relegated to the filing cabinet and soon become fossilized "shelf documents." The plan's creators are frequently the only personnel that have actually read the detailed plan. Generally, traditional plans include an executive summary that receives wide circulation but, because the larger plan is not read, many unanswered questions are left about what to do next when it comes time for implementation.

Such plans are typically difficult to modify as the function, the organization, or the technology changes. Getting original plan participants to participate on a meaningful but mammoth update effort is difficult. Traditional technology platform programs outlined in the plan become obsolete

12-24 months later as IT vendors introduce new technology or, as is often the case, delay introduction of technology forecasted for adoption in the traditional plan document. The following diagram illustrates this IT planning dilemma:

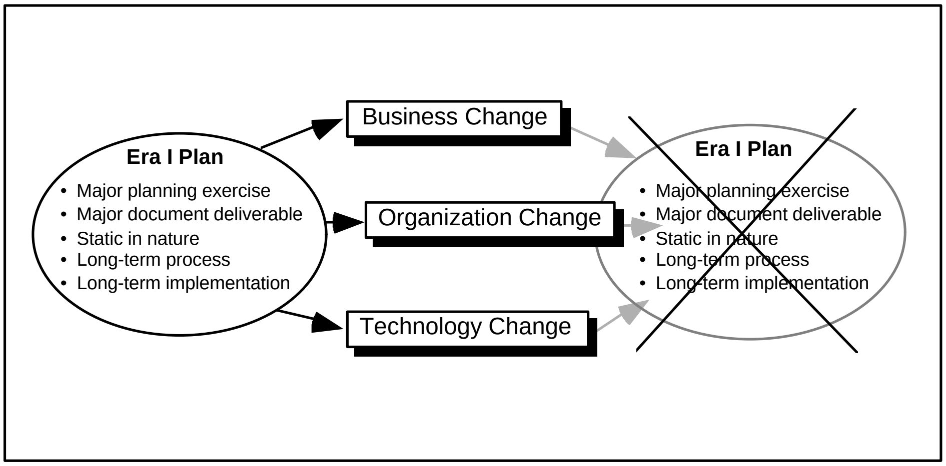


Figure 1-2. Traditional IT Planning Dilemma

Perhaps the weakest link in traditional planning models is implementation. Because of the various functional, environmental, and organizational issues described above, many traditional IT plan efforts are never put in place. These traditional planning approaches typically break down in the manner in which they approach defining technology standards. This activity is simply regarded as an added and unnecessary step in developing architecture. It does not allow for a decoupling of the technology from the "architecture" in the context of standards. By comparison, standards-based infrastructure modeling assumes that the organization and technology will change; indeed, change is the only constant.

Standards-based planning vision

Standards-based organizations place a premium on a flexible, standards-based architecture. They acknowledge today's reality that all business functions are competing in time and that the static, linear planning model that traditional planning methodologies represent is obsolete. Standards-based organizations recognize that relationships between functions, organization, and technology are often not aligned but seemingly discontinuous.

Who "owns" the vision?

With the dispersion of control over IT into the functional units out of the "glass house," the IT planning agenda itself is increasingly driven by the end-user side of the enterprise rather than the traditional IT organization. The "ownership" of the traditional IT plan has changed because the "stakeholders" have changed.

Standards-based organizational stakeholders are operational users, component units, and suppliers. This is a major shift from the traditional IT planning context when IT professionals owned and sponsored the IT agenda. Increasingly, end users are asking their IT professionals to provide value for the investment of the last decade.

In the past, major application projects have been delayed by several months or years, which has resulted in a major negative impact on operations. For better or for worse, end users are demanding results *now*, with no excuses or "technical mumbo-jumbo" for nonperformance.

Operational or functional users are increasingly setting the direction for IT planning. The decentralization of functional units and the parallel and attendant introduction of end-user technologies, such as LANs, personal computers, workstations, and network technology, has only accelerated this trend. The logic is simple: "The IT folks can't deliver, so we functional unit professionals will have to make it happen."

The need for a shared process

Despite the fact that functional users are increasingly taking control of the IT agenda, successful standards-based architectures can only be built when the planning process itself is driven by functional and IT professionals *working together* to integrate the dynamic "counter pulls" of diverse functional initiatives, organizational work flows, applications vehicles, networks, and technology platforms together in an overall strategy with a focused thrust. Any standards-based planning process and effort must take this critical fact into account. Little will be accomplished if standards implementation occurs independently and for its own sake. The key measure of the merits of standards implementation is the degree to which standards cumulatively provide significant functional value to the function.

The following diagram illustrates some of the various tensions at play with IT planning today:

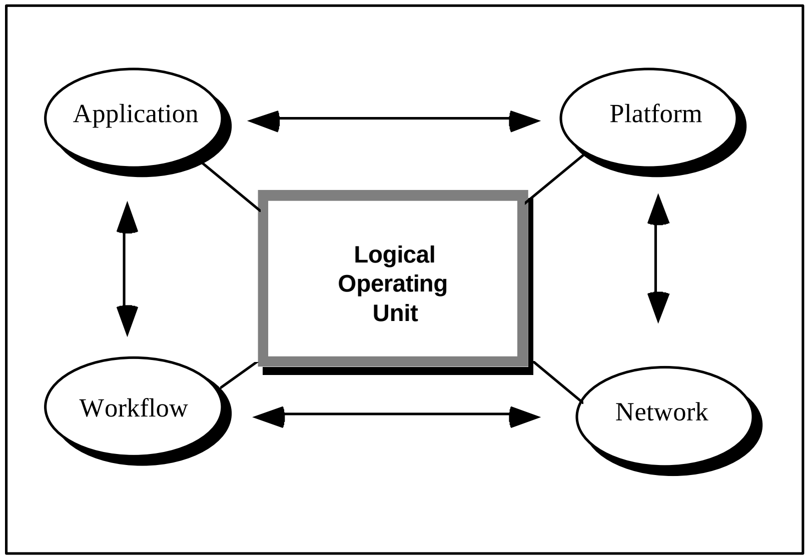


Figure 1-3. IT Planning Tensions

Traditional vs. standards-based planning characteristics

Several key characteristics distinguish standards-based organizations from traditional IT organizations in their functional and IT planning activities:

Traditional IT Planning	Standards-Based Planning
Long-term vision, long-term payoff	Long-term vision, short-term payoff
Major function-wide “data gathering” effort	Function fast-path “process”
Primarily defined and “owned” by the IT organization	Primarily “owned” by the functional unit
Proprietary vendor architecture owned by vendors	Standards-based, open architecture owned by the user
Vendor leverage over user is high	User leverage over vendor is high
Functional unit input limited	Functional unit focus central
Based on coherent “linear” functional strategy	Based on discontinuous, chaotic functional realities of today’s “fast cycle” global marketplace
Static document-oriented deliverable	Project-oriented deliverable payoffs
Obsolete when organization or technology changes	Continuously modified on quarterly basis
Typically defines functional drivers, applications and data and specific proprietary hardware/ software solutions	Defines architecture and standards with room for entrepreneurial improvisation in implementation

Figure 1-4. Traditional Versus SBA Planning Characteristics

The remainder of this SBA Guide explains the steps one should take to develop a standards-based architecture.

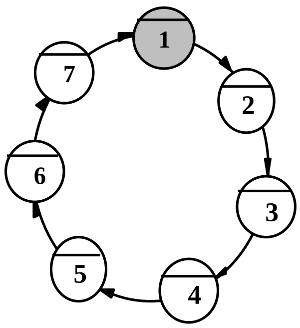
This page intentionally left blank.

Section Two: Initiation and Architecture Framework

Contents **Page**

Figures

Section description



This section describes the overall process that is followed to initiate the SBA planning activity and to develop the first major deliverable—the *Architecture Framework Document*. The following are the key aspects of this phase:

- Project initiation and positioning within the enterprise
- Development of a general definition of the open systems development and architecture environment
- Definition of an architecture vision for the future
- Consideration of a general review of architecture design alternatives
- Identification and documentation of issues underpinning the architecture vision.

Project initiation is a critical key to ultimate project success and, as such, is discussed first.

Project initiation

Project initiation provides for a smooth transition from initial project planning to the architecture framework phase of the project. It is essential that the project initiation step be explicitly defined and executed for, without it, the project will not have the firm foundation needed to withstand the inevitable rough times. Architecture projects, particularly at the enterprise level, uncover all of the basic insecurities of the host enterprise. Sensitivities are revealed, sacred cows are questioned, and political issues are raised. If these foundation issues are not dealt with and clearly agreed upon by all involved parties, the project will falter when these periodic storms hit. The facilitator needs to be aware of all these issues and realize that open lines of communication from the very beginning of the relationship are absolutely essential to the success of the project.

By their nature, all architecture engagements are different. As a result, an explicit project initiation step is a key to success. The phases, tasks, roles, and responsibilities will be affected by the culture of the enterprise, architecture work that may have already been done by the enterprise, the commitment of resources the enterprise is willing to make (or conversely insists on making), the preconceived notions the enterprise has about what an architecture project entails, and a host of other factors too numerous to list. Project initiation allows all involved parties to agree on the customization of the basic SBA planning approach taking all of these factors into consideration. It then allows specific decisions about resourcing and time frames for the agreed-upon tasks. A clear-cut project plan emerges and the first stage of the plan is kicked off.

The project initiation step is not completed until a plan has been laid out in enough detail for the enterprise to know exactly what is expected at all points along the way. Obviously, not every single workshop, interview, or background session will be scheduled to the day and minute, but the necessary events of the early stages of the engagement should be locked in during project initiation. Also, the critical project infrastructure issues (CSFs) must all be resolved.

Almost all of the work of project initiation revolves around the key issues of establishing a mutually agreeable resourcing strategy and allocating those resources to tasks that will result in deliverables and time frames with which all parties can live. Then, of course, the key early tasks in the plan will be kicked off.

Architecture Work Group

The core team that will be involved in the SBA project from beginning to end is the Architecture Work Group (AWG). This is the group of four to six mid-tier managers and IT personnel from the functional areas. This team will be responsible for facilitating the SBA process, for developing the overall project plan, for securing appropriate participation by key knowledge workers, and for ensuring that all documents specified in the project plan are completed.

Architecture Steering Committee

The key to success in this phase depends on the ability of the AWG to help the participants develop a shared understanding of the problems and opportunities related to the existing environment and then to establish a coherent framework for solving these problems over time—*building a shared vision and direction.* While it is the objective of every planning exercise to develop this vision, it is frequently not achieved for a very simple reason—key players were not involved in the process.

Because of this, it is critical that an Architecture Steering Committee (ASC) be formed. This group should be composed of a mix of functional area and IT professionals. Its size and makeup will differ depending on the scope of the SBA effort. If there is a question of team membership balance, it is preferable to err on the side of too many functional area professionals. It is paramount that all stakeholders be involved in the team—this includes any individuals or enterprises with key influence or other "political" power within the functional area.

The *Architecture Framework Document* is developed by the AWG. Together with key knowledge workers (these are the subject matter experts with specialized skills or knowledge that work on an as-needed basis with the AWG), this team becomes the core entity for developing the rest of the SBA project.

The bulk of the research for the *Architecture Framework Document* is conducted by facilitating "fast-path" workshops and interviews with key functional and IT personnel. The team produces evolutionary drafts of the document until all of the stakeholders enthusiastically endorse it.

A multistep process is an effective way not only to identify the central issues underpinning a standards-based architecture but to help develop the architecture principles that will guide the rest of the effort.

It is important to note that this phase of the standards-based implementation cycle is of a direction-setting nature. During this effort, a general understanding of the current environment is developed and a high-level definition of the current architecture direction is rendered. Time should not be spent uncovering minute technical details. That work is better left for subsequent steps of the process.

Objectives

It is important to produce a comprehensive *Architecture Framework Document* that is easy to understand and that engages executive commitment. It is also important that the document be function oriented—addressing issues that are key to the success of the functional area(s) included in the effort.

The AWG should avoid focusing solely on technology and the application development environment. Executive staffs will often dismiss a technical document because they see little benefit in defining technology for technology's sake; however, a document explaining what technology can do to help the enterprise achieve its mission is sure to get executive attention.

Scope

The scope includes all aspects of the enterprise that may have an impact on the future use and deployment of IT—the work of the enterprise and the way IT may be used to support it. Key business drivers are defined as well as the issues surrounding current technology. Workshop and enterprise change-related activities are the primary vehicles by which the *Architecture Framework Document* is produced.

Personnel in each functional area within the enterprise are interviewed by the AWG. The purpose of these interviews is to:

- Discuss the basic mission of the functional areas
- Identify areas for improvement in current practices
- Begin to determine possible ways that information technology can be used to better support the enterprise.

The AWG then synthesizes the findings of the interviews. The results of this synthesis are a set of architecture principles. These principles are then put to the test. They are voted on and discussed with the ASC. This meeting provides a vehicle for key stakeholders to discuss and agree on how the enterprise should proceed with this very important SBA task.

The principles presented in this deliverable will serve as guidelines for developing the plans that will ultimately become the IT architecture for the enterprise.

Deliverables

An *Architecture Framework Document* that contains:

- Enterprise mission/vision
- Strategic drivers
- IT principles
- Key issues that will impact development of the target architecture.

The major deliverable of this phase is the *Architecture Framework Document*. It is recommended that this document be brief in nature, "Executive Summary" in design, and as highly visual as possible. A sample outline for this document is included in Appendix I.

The central objective of this document is to provide a broad understanding of the IT architecture vision. If the document is produced successfully, all key stakeholders will possess an "ownership" of the effort.

Critical success factors

- Identifying shared interests
- Establishing the ASC and chairperson (effectively the "system owner" team)
- Establishing the AWG and primary contact (effectively the "system manager" team)
- Establishing the larger community of knowledge workers who will participate, either in interviews or workshops
- Establishing the mechanism to officially kick off the engagement for all of the participants identified above and for the enterprise as a whole
- Providing initial orientation to the architecture development process for the ASC, the AWG, and the community of knowledge workers who will directly participate
- Supporting the executive level of each functional area within DoD
- Establishing a shared vision
- Providing a communication vehicle for promoting the vision of the architecture design
- Assuring key knowledge worker commitment and participation
- Agreeing on how, when, and to whom project status will be reported
- Procuring and setting up workspace and tools for the facilitator(s) and the AWG.

Constraints

Many enterprises have never formally developed architec-ture principles. The absence of these principles is a definite constraint to the work team, which relies heavily on such documents in defining the mission and vision of the enterprise.

- Commitment and participation of executive staff (ASC)
- Availability of existing source material.

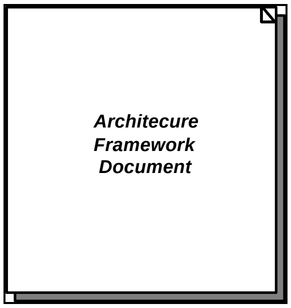
Management must be solicited to dedicate knowledgeable personnel to the effort (at least until the necessary vision statements and principles are created) or the project is doomed to drag on indefinitely, while the AWG attempts to define this starting point.

Task list

- Initiate project and AWG team building
- Form ASC
- Define interview process
- Conduct interviews
- Analyze existing information
- Evaluate existing data-gathering processes
- Optimize those processes to ensure timeliness and accuracy
- Reconcile interview data with existing information
- Draft and circulate principles for principles workshop
- Conduct principles workshop
- Review final principles with ASC
- Create *Architecture Framework Document* outline
- Assign writing, reviewing, and editing tasks
- Draft *Architecture Framework Document*
- Circulate *Architecture Framework Document* for comments and review
- Review *Architecture Framework Document* with ASC

Finalize and publish *Architecture Framework Document.*

Creating and publishing the deliverable



This phase will vary widely in terms of the calendar time required for completion based on culture, individual schedules, etc. Ideally, when conducted on an intensive basis, this phase can be completed in approximately 4 weeks. However, most enterprises require about 2 months to complete the outline, draft, and final document. The document simply goes through several iterations before approval by the ASC. The process is as follows.

With the ASC as a quality check, the AWG can begin to conduct the interviews necessary to gain insight into the business drivers within the function. If done properly, these interviews can also serve the purpose of promoting the architecture project throughout the enterprise.

Senior executives and key "thought leaders" within the enterprise should be interviewed. Because of the high exposure that this activity represents, it is important that the interviewers be well prepared prior to scheduling the first round of interviews.

It is suggested that a set of essential questions be developed jointly across the body of interviewers. This helps the interviewer anticipate underlying issues and problems before actually interviewing key personnel—thus minimizing the potential for failure. Figure 2-1 highlights general questions to be asked. These questions can be more detailed depending on the scope of the SBA endeavor.

Existing models and principles

To expedite building the architecture framework, the team should review any existing business, work organization, application, and information models, as well as current architecture principles for background. There is no need to "reinvent the wheel" if such materials exist. The models provide input and background to the AWG.

Reconciliation and principles workshops

The result of interviews and secondary research of existing material is the development of a set of draft principles. As the effort progresses, principles workshops are held. Each workshop addresses specific topics such as applications, standards issues, database strategies, and communications.

Sample List of Questions

1. What are your responsibilities today?
2. What are your current and long-term priorities? What stands in your way?
3. What are the most critical elements for success in your job?
4. How can technology be used to help you succeed?
5. What has been your experience in technology projects in the past? What has made them successful? Why have they failed?
6. What improvements can be made to make your work environment more productive? Can technology be used?
7. Would you be willing to commit resources to improving the use of technology in your area?
8. Who would you recommend we talk to next regarding the use of technology in your area? Would toy help us schedule a meeting?

Figure 2-1. Interview Questions for Input to Architecture Framework

The purpose of the workshops is to reconcile the views and principles with the information uncovered in the interviews. A group of architecture principles is developed. It is typical for a group to develop 30 to 40 different principles for an enterprise's architecture. A sample principle taken from the USMC project is shown below. In addition, a more complete description of how to develop architecture principles is included in the SBA Guide as Appendix A.

Architecture principles are **statements of preferred architectural direction or practice.** They are simple, direct statements of how an organization wants to use information technology in the long term for five to ten years. They establish a context for architecture design decisions across an organization and help translate business criteria into a language that technology managers can understand. Each principle is accompanied by a statement of the rationale behind stating the principle and a statement of the principle's implications.

Figure 2-2. Definition of an Architecture Principle

Principle

Where feasible, the USMC will use Commercial-Off-The-Shelf (COTS) and Government-Off-The-Shelf (GOTS) application components and systems rather than develop them internally.

Rationale

The use of COTS and GOTS applications and components should lead to an environment of increasingly interchangeable parts. This kind of environment should be more cost effective and efficient than custom development, because multiple "customers" are sharing in the development and maintenance costs. For similar reasons, training and implementation costs should be reduced. The time frame from concept to implementation should be reduced by taking advantage of tested and operationally proven applications and/or application components. Finally, the risks normally associated with custom development (e.g., scope changes, budget overruns, missed target time frames, etc.) are significantly reduced.

Implications

- A process for evaluating and selecting COTS and GOTS applications will be needed. This process must accomplish at least the following tasks:
  - Identify user requirements which can be satisfied by purchasing standard components.
  - Consider if changing the current functions and processes would enable the purchase of standard system components without adverse effect on operational performance.
  - Analyze whether the USMC's customization needs can be accomplished outside the purchased standard component rather than inside it. In so doing, the Marine Corps could subscribe to the vendor's ongoing maintenance releases.
- Some BPR may be needed to align the business process with available COTS or GOTS applications.
- A set of standards and measurements for matching a standard component's functionality with the user requirements should be developed. For example, the standard might state that only systems or components which satisfy 80% of required functionality should be considered for purchase.
- A repository of available COTS and GOTS applications will be needed. This repository will need to accommodate the definitions of the applications and/or application components as well as any predefined interrelationships among the applications.
- Finally, using COTS and GOTS systems and components will make the USMC reliant on those vendors for maintenance and upgrades. Therefore, a vendor qualification process must be undertaken to assess the potential longevity in the marketplace of vendors of prospective packages.

Figure 2-3. Sample USMC Principle

Effectiveness measures

- Degree of consensus achieved with principles

  Acceptance of draft *Architecture Framework Document*
- Amount of rework required
- Management participation
- Awareness of the effort.

The overall objective of this phase is to provide a summary document that is easily understood by business managers and IT personnel alike. It is therefore important that the deliverable be a functionally oriented (rather than technically oriented) document and focus on key issues of importance to the functional area(s).

The work team will be measured against its ability to develop a document that the enterprise "buys into." Granted, this is a very subjective measure. However, it is the only one that really matters at this stage in the SBA project—buy-in is the name of the game.

For this reason, minimal rework alone does not guarantee quality work. Sometimes minimal rework points to a lack of management commitment to the effort.

Therefore, effectiveness can only be measured by the combination of variables listed above. The team will know if the results of its effort are falling on deaf ears, if few people within the function know about the SBA project and even fewer senior managers pay it due.

Technology and tools required

- Dedicated war room for team meetings
- Word processing and graphic presentation packages
- Microcomputer and telecommunications capabilities
- Principles templates (see Appendix A)
- *Architecture Framework Document* outlines (see Appendix I).

To truly expedite the effort, a project "war room" should be established. It should be equipped with a white board and markers for brainstorming, PCs for preparing the document, a table and a set of comfortable chairs for conducting meetings and interviews, and plenty of work space so that the team can get the job done.

The AWG should be equipped with word processing, spreadsheet, and graphics presentation packages so that they can develop the *Architecture Framework Document* easily. If possible, the team should be connected to each other via a network so that the work files can be passed from writer to reviewer more efficiently.

In some of the more sophisticated environments, the work room is staffed with a secretary who can take messages, help with the typing, and assist with the document preparation work; however, this is not a prerequisite.

Staffing skills required

- Group facilitation skills
- Interview skills
- General functional area knowledge and IT technology background
- Project management skills
- Writing and presentation skills.

The key to this effort is the solicitation of management support for the effort. Therefore, it is essential that a good group facilitator is used—one who can manage group dynamics, understands the SBA process, and can keep the work team on track.

This kind of individual is present in most enterprises; however, many firms feel more comfortable getting their facilitation expertise from outside the concern—outsiders tend to be more objective and are less likely to sway the team for personal gain. Figure 2-4 highlights some essential facilitator skills.

Although the facilitator is important to this effort, he/she does not a work team make. The work team must be staffed with people who possess the qualities listed above, or the effort could be in jeopardy. For this reason, work team candidates should be screened prior to project inception—just to make sure the right people are available for the job.

List of Essential Facilitator Skills

- Knowledgeable project manager
- In-depth understanding of SBA process
- In-depth understanding of automated tools used in SBA process
- Expertise in team building
- Expertise in managing group dynamics
- Ability to communicate in both business and technical terms

Figure 2-4. Essential Facilitator Skills

Completion criteria

- Interview schedule completed
- Draft principles document
- *Architecture Framework Document* deliverable
- Management acceptance.

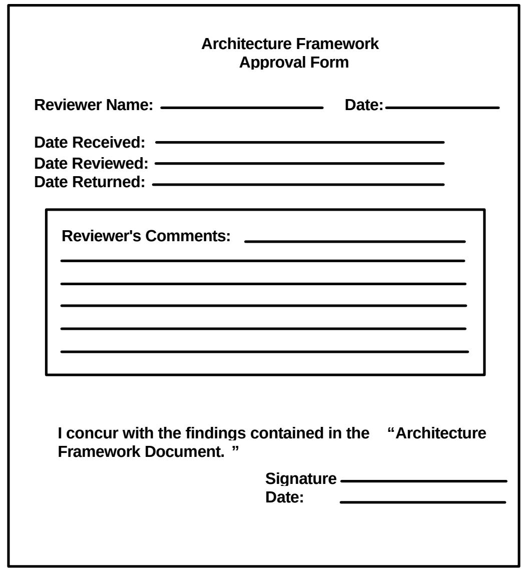
Ultimately, this phase is completed when the ASC accepts and signs off on the *Architecture Framework Document.* While the other items listed above are important milestones, the work is not considered complete until all committee members "own" the deliverable.

For this reason, it is important for the team to establish a sign-off procedure that ensures full committee approval. Many times enterprises will establish a sign-off procedure that assumes acceptance with no formal reply. This should be avoided. Figure 2-5 illustrates a typical Architecture Framework Approval Form for committee sign-off.

A process that requires a written signature has proven to be very effective. ASC members will pay more attention to the effort because they want to understand and be in agreement with what they are signing.

Issues

- Training required
- Executive participation
- Current workload of work team members
- Consulting support required
- Subject matter expert availability.



Architecture Framework
Approval Form

Reviewer Name: ____________ Date: ________

Date Received: ______________________
Date Reviewed: ______________________
Date Returned: ______________________

Reviewer's Comments: ________________

I concur with the findings contained in the "Architecture Framework Document."

Signature ____________
Date: ____________

Figure 2-5. Architecture Framework Approval Form

As mentioned throughout, executive commitment and the availability of key personnel (or key knowledge workers) is essential to the success of this effort. However, there are other issues that an enterprise must face to ensure a quality deliverable from this phase.

The need for training and consulting support is often overlooked by enterprises excited about establishing a standards-based architecture. While every function is different (in the skills and talents that its personnel possess), most require the initiation of training in the planning technique presented here.

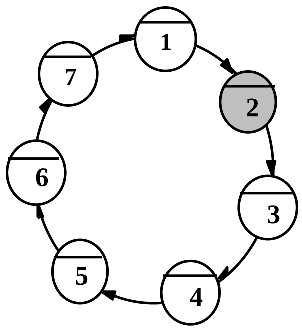
For this reason, most enterprises use consultants to provide the necessary training and to drive the SBA effort—at least until the enterprise becomes self-sufficient (usually after one or two successful SBA pilot projects have been conducted at a functional area level).

Section Three: Baseline Characterization

Contents **Page**

Figures

Section description



This section describes the overall process that is followed to conduct a high-level characterization of existing work organization, information, applications, technology, and standards. This activity includes:

- Reviewing general cost, performance, and security issues related to the baseline architecture
- Developing a framework for characterizing the current environment to help the WAG organize its thinking
- Documenting the characterization of the current environment in a *Baseline Characterization Document.*

Objectives

To create a report that characterizes the existing architecture of the enterprise.

Many organizations have undertaken enormous baseline efforts sometimes requiring many months, if not years, to complete. The detail that would take years to develop is not necessary–characterizing the existing situation in just a few months of elapsed time is the goal.

Without the insight that a baseline characterization provides, it is difficult to develop truly effective implementation plans needed to lead the organization into its chosen target architecture. A clear view of the existing IT architecture allows identification of opportunities for change and a migration plan for implementing those opportunities. Without this view of the existing situation, there is the risk of devising a target environment that is very difficult or impossible to implement.

The SBA process is designed to be “fast path” in nature. That means that traditional long-term inventory efforts will not be appropriate if the task is to proceed quickly and deliver results. While large and timely data collection efforts yield more accurate data, time is sacrificed for accuracy. If a branch of service or entity already possesses much of the baseline data, then most of the work effort should be spent on characterizing the current environment with a high-level description. The difference between a good and bad baseline effort is the degree to which the baseline *characterizes* the current environment accurately. The recommended approach is a generic baseline versus a detailed specific baseline.

Scope

The enterprise that is being modeled (e.g., a branch of service, a subset of a service, the entire DoD):

- Existing views of physical and logical environments can be used if readily available.
- Task teams can be formed to develop information about the current environment, if no formal data exists.
- Matrices for categorizing work, information, application, and technology platforms as well as cost frameworks can be used.
- Descriptive security classification should be applied to each application and the technology environment reviewed.

The AWG should set their sights on conducting a baseline effort that characterizes the current environment rather than conducting the most accurate inventory effort. This is not the same activity as a massive inventory effort! In practice, and as a rule of thumb, 80 percent of the information used in an architecture design activity derives from 20 percent of the data collected. It is therefore inefficient to spend time collecting the last 20 percent of the data when 80 percent is sufficiently accurate in characterizing the current environment. Figure 3-1 illustrates the data collection payoff dilemma all AWGs face.

Fundamentally, all IT architectures are built upon existing technology platforms. In the end, an IT architecture represents how the given sets of existing technology platforms are used and structured and the attendant functionality they deliver for the individual, the work group, the function, or the enterprise.

The task of evaluating and designing a new or alternative architecture requires that the AWG have a convenient method by which it can characterize the current architecture. After the AWG has created a baseline of the existing architecture, its relative merits and shortcomings can be examined. With a baseline in place, assuming the function seeks to improve upon the existing architecture, the team will be able to develop a target architecture and an all-important migration plan to assure its successful implementation.

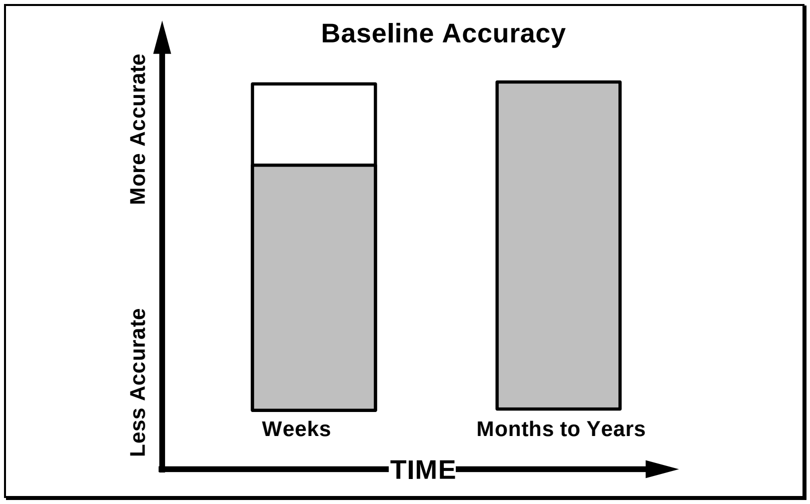


Figure 3-1. The Data Collection Payoff

Baseline elements

A number of elements should be reviewed as inputs to the overall *Baseline Characterization Document.* These include:

- Work organization view
- Information view
- Application view
- Technology view
- U.S. Department of Defense *Technical Reference Model and Standards Profile* (TAFIM, Volume 2) is a framework with which to characterize current profiles in place in different parts of the overall model
- Security design document, which specifies the security plan for the organization. It contains information about such issues as security policy, accountability, security assurance, and security documentation as outlined in the U.S. Department of Defense Trusted Computer System Evaluation Criteria [DoD 5200.28 STD, December 1985].

Deliverables

The major deliverable of this phase is the *Baseline Characterization Document*. The DoD recommends that this document be brief in nature, “Executive Summary” in design, and as highly visual as possible. The idea is that this document will be used by a large number of individuals and organizations as will all deliverables produced during the architecture development activity. Appendices should contain the results of the baseline data gathering, while the body of the document should contain key conclusions and analyses. This document should show readers "the forest" rather than focus on "counting trees." A sample outline for this document is included in Appendix I.

Critical success factors

- Commitment of resources to develop inventory information
- Trained leadership with experience in fast path baseline efforts
- Communication vehicle for reporting inventory information
- Key knowledge worker availability.

A key critical success factor is that the senior management of the function understands, endorses, and enthusiastically champions the SBA project. In a time of shared DoD resources, this means committing DoD personnel to work on the project for dedicated periods of time. Therefore, a premium must be placed on time and doing the baseline effort quickly.

As stated in the previous section, the ASC, composed of representatives from both the business and IT departments, will act as the "project owner." This committee is the conduit between the AWG and the rest of the function. It is key that the ASC makes all concerned organizations aware of the vital nature of the baseline effort and secures cooperation from the same when required.

Constraints

Availability of existing architecture input in readily accessible form.

Many organizations have never formally developed or created baseline models. The absence of these models is a definite constraint to the AWG, which relies heavily on such documents in defining the current environment.

However, these background materials can be developed quickly when the right people are engaged in the effort. There are people within the organization who understand what information exists and the level of effort required to collect data appropriate to the task at hand. Management must be solicited to dedicate such knowledgeable personnel to the effort, at least until the necessary architecture views and principles are created, or the project is doomed to drag on indefinitely.

Task list

The baseline characterization process follows the basic steps listed below. A key step in the process is primary data gathering in the form of workshops with key knowledge workers in various operational areas. Workshops are conducted with one or more representatives from the host organization.

- Initiate baseline task team—identify AWG and task groups
- Define inventory scope, effort, and milestones
- Develop application, technology environments, security, cost and platform classifications, and data collection instruments (templates and tools)
- Assign inventory data-gathering tasks
- Review findings and synthesize results
- Produce first cut *Baseline Characterization Document*
- Conduct management review of *Baseline Characterization Document*
- Refine *Baseline Characterization Document*
- Distribute *Baseline Characterization Document* to ASC for comments and review.

The AWG conducts the overall baseline activity and is responsible for producing the *Baseline Characterization Document.*

Data collection

The AWG should appoint a small subtask group to conduct a baseline effort that characterizes the current computing environment. This task group conducts a technical inventory of the organization’s existing technology infrastructure. Inputs to this process will vary widely from organization to organization based upon the quantity and quality of documentation available. Business, process, and data model documents may also be used as input. Physical diagrams, logical diagrams, tabular inventory, and financial budget data will also be valuable.

One recommended source for baseline data is the *Defense Automation Resources Information Center (*DARIC). DARIC maintains an extensive set of database repositories that inventory installed hardware, software, and data related to management of information technology within the DoD. At the same time that the DARIC resource may be used to provide useful baseline information to AWGs, DARIC may also be used to review technology components that might be valuable for reuse. It is highly recommended that the AWG meet with DARIC personnel to obtain a detailed understanding of DARIC's capabilities and resources.

It may become necessary for the baseline task group to assemble and conduct workshops to derive data from the organization when it is not otherwise readily available from DARIC or other conventional sources.

Overview of the baseline activity

To establish a baseline architecture, an inventory of the existing computer and communications hardware, system software, and application systems must be compiled.

The inventory is not intended to be exhaustive. Do not spend an excessive amount of time and effort on collecting the information. Eighty percent accuracy is sufficient to establish the basic structure of the baseline. The primary goal in collecting this baseline inventory is to establish the overall existing architecture structure and a high-level view of its robustness on a number of levels, including user satisfaction, strategic significance, and technical quality.

Baseline inventory

The baseline inventory will be compiled by completing a series of worksheets or templates. A complete set of templates, used in the baseline assessment, is included in Appendix B. The templates cover all of these categories:

- Existing work functions and processes
- Technology platform inventory
- Applications inventory
- Initial application assessment
- Various affinity (cross-reference) matrices showing the interrelationships of the various components of the baseline architecture.

Work functions and processes

This inventory should include all business functions and the key processes included within the function. For each function, the mission should also be identified. These functions and processes should be cross-referenced to other components of the baseline architecture in the following ways:

- Functions to data groupings
- Functions to applications
- Functions to locations.

Technology platforms

This inventory should include all components of the computer processing and communications environment, including the following information:

- Type of platform (in terms of the generic technology platforms defined in Section 3 of the SBA Guide) and outlined below:
  - Workstation
  - Output/input peripheral
  - Local area network (LAN)
  - LAN server
  - Wide area network (WAN)
  - Network interface device
  - Concentrator/multiplexer/switching device
  - Storage devices
  - Mid-range processor
  - Large processor.
- Vendor name and model (e.g., IBM 3090, IBM 486 PC, Sun Sparcstation). Also include the capacity characteristics in terms of throughput and associated storage (memory and access to separate storage devices).
- Specific technology environments (standards) supported in the following areas:

- User interface
- Operating system
- Communications management
- Database (and/or file) management
- Transaction monitor
- Document management
- Distribution management (e.g., E-mail, electronic data interchange)
- Conferencing management
- Development services (compilers, languages, and tool support)
- Repository services (for systems management and construction, including data dictionary support).

• Platform owner (i.e., who has the budgetary ownership or responsibility for this platform).

• Platform manager (i.e., who has the day-to-day operations responsibility for the platform).

• Platform location (i.e., the physical locations of the platform, address, building number, and/or other designator which will uniquely define the location).

Initial application assessment

As a part of the collection of the existing inventory, an initial assessment of the application systems should be gathered from key application users. System developer/maintainers should also give their assessment of the more widely used applications.

An initial assessment of each application is needed according to the following criteria: user satisfaction, strategic value, and technical quality. As part of the analysis process, after all templates have been returned, these criteria will be mapped in the following pairs on four-quadrant matrices to allow a high-level determination of the recommended disposition of each application:

• User satisfaction versus strategic value

- Technical quality versus strategic value
- Technical quality versus technical evolution.

Mapping attributes to platforms

One of the key activities of this phase is the development of a description of the current environment. This activity must be simple to accomplish. Most organizations have technology platforms in place that handle existing applications. These platforms, more often than not, are supported by proprietary technology.

A range of technology platform categories are provided that will be used in the baseline effort. The criterion for platform definition is that it must be offered in the marketplace as a product. It must be viable, proven technology that is available in the marketplace and one that users can purchase and implement in the present. These technology platforms include:

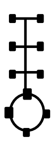
WS

- **Workstations.** Any device ranging from a fixed function or dumb terminal to a high-end workstation capable of complex calculations and graphic requirements (e.g., 3270 terminal, PC, SUN workstation).

O/I Per.

- **Output/input peripherals.** Any device that outputs or inputs electronic data (e.g., laser or line printer, image scanner).

LAN

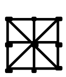


- **Local area networks (LANs).** Operating system protocols associated with local area network solutions (e.g., Ethernet, Token Ring, Starlan).

LAN Server

- **LAN servers.** Network operating system software and hardware attached to LAN networking solutions that allows routing, file storage, and user application services (e.g., LAN Manager, Novell, Banyan, 3Com, Netframe Super-Server).

WAN

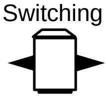


- **Wide area networks (WANs).** All network services offered by public network providers such as public and virtual private switched voice, switched and dedicated data, gateway and enhanced service offerings (e.g., AT&T, MCI, U.S. Sprint, Telenet, Internet, IBM Information Network, Tymnet, Telenet, etc.).

Iterfce.

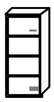
- **Interface devices.** Any device that provides a major bridge or switch between environments (e.g., TCP/IP router, DEC router, LAN bridge).

Con./Mux/
Switching



- **Concentrator/multiplexer/switching devices.** Any device that performs a concentration function, a multiplexing function, or a switching function (e.g., IBM 3705, a NET T-1 multiplexer, an AT&T PBX).

Storage

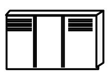


- **Storage devices.** Any traditional magnetic or optical storage device (e.g., floppy disk, magnetic tape, optical disk).

Mid-Ran.
Proc.

- **Mid-range processors.** Historically known as the "mini-computer," this increasingly blurring category includes any processor manufactured for mid-range processing (e.g., IBM AS400, DEC VAX, HP Spectrum).

Large Proc.



- **Large processors.** Traditional mainframe category historically dominated by IBM, UNISYS, and Amdahl. Supercomputers, such as Crays, are included at the high end of this category.

Technology platform attributes

The various generic platform classifications described allow a baseline inventory to be made of the existing architecture. As IT technology changes, so will these categories.

Each platform listed above may be thought of as having various attributes. By categorizing existing platforms and defining their constituent parts, a standards-based current architecture may be defined and examined in a baseline exercise. It may then be used in subsequent steps to define the target architecture. Figure 3-2 illustrates these various platform attributes.

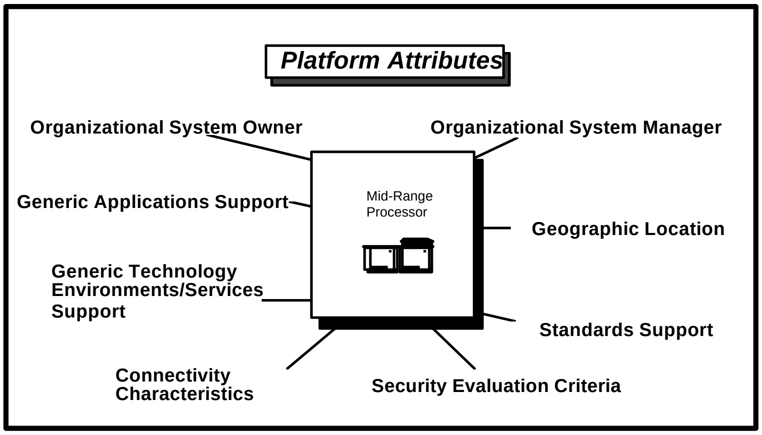


Figure 3-2. Platform Attributes

Each of these platforms:

- Has a specific system owner(s) with a DARIC reference number
- Has a specific organizational system manager
- Supports an application or application suite and thus serves a role as a generic application support environment or “GAE”
- Provides a technology role for an overall architecture through the provision of services as a generic technology environment or “GTE”
- May be classified in terms of its security evaluation criteria as outlined in *Trusted Computer System Evaluation Criteria Summary Chart* (p. 109) of the *U.S. Department of Defense Trusted Computer System Evaluation Criteria* [DoD 5200.28 STD, December 1985]
- Supports various standards, be they proprietary or open in nature, and are built on either de jure or de facto standards
- Has connectivity and interface characteristics with other technology platforms
- Has specific cost performance characteristics associated with its technology life cycle
- Has a specific physical environment.

The following diagram illustrates how the technology platform attribute model may be used as a model for a baseline platform—in this case, a mid-range processor.

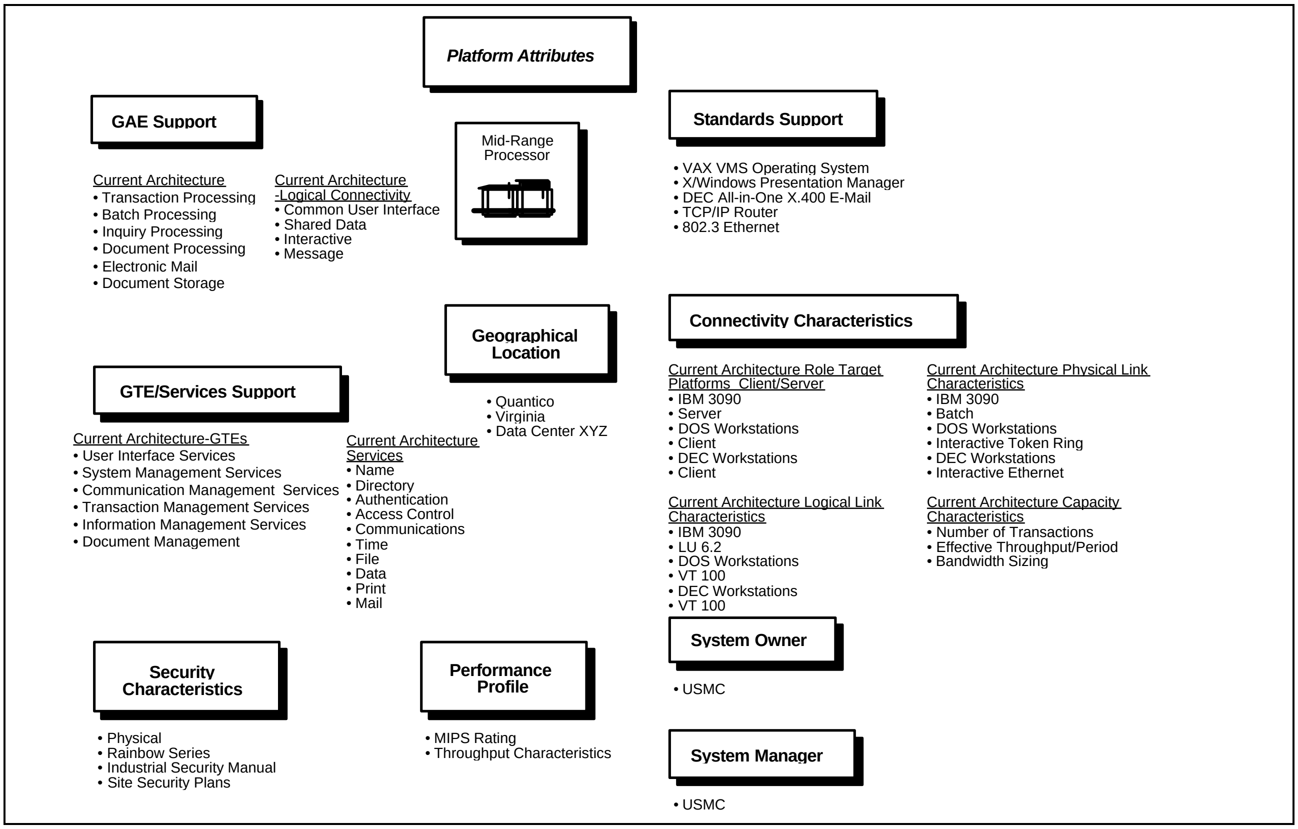
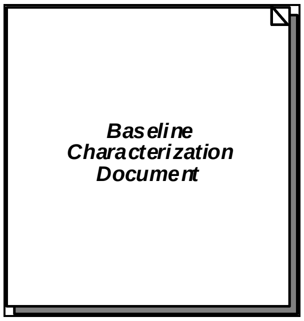


Figure 3-3. Platform Attributes Examples

Creating and publishing the deliverable



The key deliverable out of this phase is the *Baseline Characterization Document*. The sole objective of this document is to characterize the current environment and to highlight systematically the profile and attributes of the current architecture. The baseline will be used as input to the migration options phase where it will be compared to the target architecture. This comparison will be used to identify necessary projects to achieve the vision of the enterprise.

The *Baseline Characterization Document* provides a total picture of the current state of architecture. This phase will vary widely in terms of calendar time required for completion based on enterprise culture, individual schedules, etc. Ideally, when conducted on an intensive basis, this phase may be completed in 8 to 10 weeks.

However, most organizations require about 3 months to complete the outline, draft, and final document. The document should go through several draft iterations before being approved by the ASC.

The overall objective of this phase is to provide a summary document that is easily understood by business managers and IT personnel alike. It is, therefore, important that the deliverable be a business-oriented document and focus on key issues of importance to the function.

Effectiveness measures

- Management acceptance of task deliverable
- Comprehensive global characterization of existing environment
- Amount of existing inventory data that is reused
- Speed of task execution
- Extent that document is accurate as measured by degree of acceptance (and percentage degree of completeness).

Technology and tools required

- Word processing and graphic presentation packages
- Architecture team room for meeting
- Spreadsheet tools and/or user friendly, personal computer-based database packages for inventory logging
- Baseline templates (see Appendix B).

The AWG should be equipped with word processing, spreadsheet, database, and graphics presentation packages so that they can develop the *Baseline Characterization Document* easily. A key aspect of this activity is the development of data collection templates to streamline the project data-gathering exercise. Once these have been created, the rest of the baseline effort is more mechanical than “creative.”

Staffing skills required

- AWG with baseline experience and high familiarity with existing environment to be baselined; for example:
    - An inventory specialist who provides input to Arms database
    - Network administrators
    - System managers

- Data administrators.

- Interview skills
- Writing and presentation skills
- Organizational data collection knowledge
- Familiarity with word processing, presentation, spreadsheet, and database packages that run on most popular personal computers.

The key to this effort is the solicitation of management support for the effort. Therefore, it is essential that an AWG leader is selected to facilitate the baseline effort—one who can manage group dynamics, roll up his or her sleeves with the team and participate, and who understands the SBA process and can keep the work team on track.

Completion criteria

- Inventory scope and deliverable defined
- Inventory completion deadline met on time
- Management acceptance of deliverable
- Completion of *Baseline Characterization Document*.

Ultimately, this phase is completed when the ASC accepts and signs off on the *Baseline Characterization Document*. It is important that all the ASC members as well as the AWG agree that this document is a characterization of the current environment.

The team should obtain a sign-off that ensures full ASC approval. This was described in the Architecture Framework section.

Issues

- Workload of work teams
- Availability of existing inventory data
- Successful amount of data collection in short time frame
- AWG understanding of level of effort and fast path approach
- Core team to remain the same.

The need for resources on this task is crucial to project success. The overall AWG may be at its highest level of headcount during the baseline effort.

Given the severe resource limits that are currently the norm in the DoD, we recommend that the AWG draft members on a temporary duty basis for the baseline effort. The "baseline draftees" may then be demobilized and released or be assigned to the target architecture phase upon completion of the *Baseline Characterization Document.* However, the core AWG members remain the same throughout the overall project period.

The ideal profile for an "enlisted" AWG member drafted to conduct baseline work is an individual who possesses a sense of urgency and the ability to work on a "fast path" basis to ensure project success.

Keep in mind that the baseline effort is not intended to determine an action plan for solving the ills that it uncovers (such plans will be developed during the implementation planning phase of the project). Instead, the intent is to simply define the current environment, which will act as a logical launch point for subsequent phases of the SBA process. What's next, however, is to define the target environment that the organization seeks to embrace over the next few years.

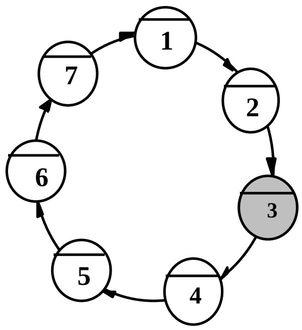
Section Four: Target Architecture

Contents **Page**

Figures

Figures (cont'd)

Section description



This section describes the overall process by which the architecture framework is extended by the AWG. These issues for you to approve includes:

- An extension of the vision defined in the *Architecture Framework Document*
- A description of a desired future architecture
- An identification of what can be extended from the current environment into the target environment.

Objectives

To develop a *Target Architecture Document* that specifies the profile and attributes of the new technology environment and highlights the key opportunities for improvement over the baseline. The new architecture need not be developed based on cost-effective and “business-case-based” criteria. The real world constraints of cost/benefit analysis and cost justification will be introduced in the migration options phase of the SBA process.

At this step in the process, it is desirable to define a target architecture that can be used to achieve the vision of the organization in all of the architecture views and, especially, the work architecture. Ultimately, constraints will come to bear on the funding of each project that is needed to achieve the target but, for now, it is sufficient to flesh out the target to identify the full spectrum of what is needed to achieve the vision of the organization.

Inevitably, the architecture that is implemented will be a blend of the baseline and the target, with architecture principles as the foundation stone. Sometimes, an organization cannot migrate to the target without either disrupting the quality of service provided to the user base or expending an inordinate amount of resources to get there. Therefore, it is important that the team take the time to outline a set of alternative architectures that may become an interim target until the ultimate target can be legitimately reached.

Figure 4-1 depicts an overall framework within which the AWG can operate to develop the target architecture deliverable. Each view of the target architecture has some overlap with aspects of the other views (see Figures 4-2, 4-7, 4-9, and 4-11 below). This overlap supports the argument that we are developing a single, integrated architecture. As we proceed through the remaining discussion of the target architecture development process, we will frequently refer to this meta-model in order to remain focused on the key aspects of the task at hand.

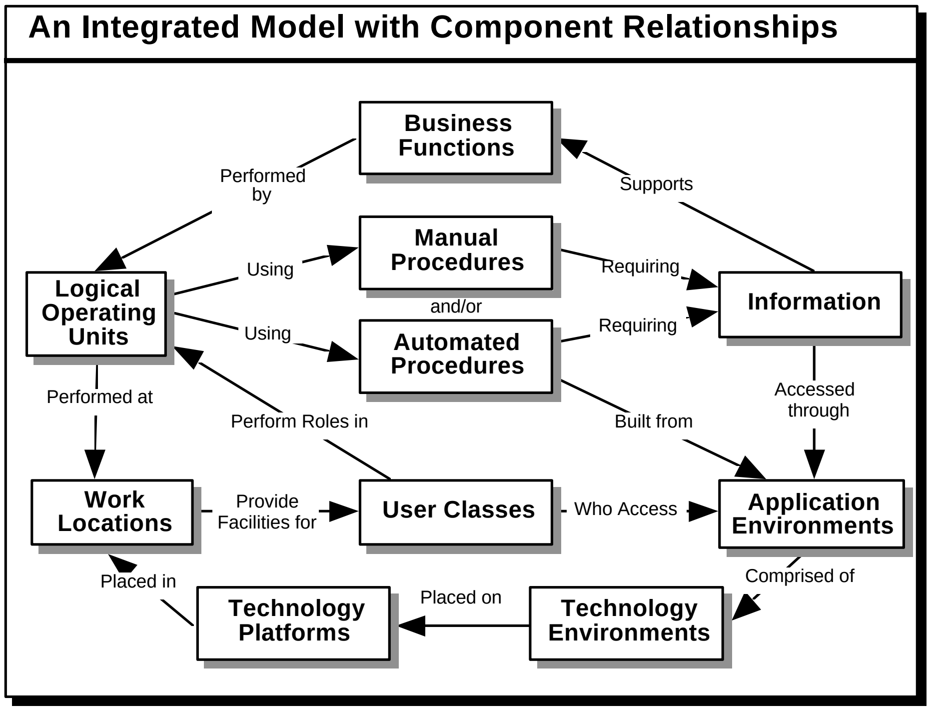


Figure 4-1. Integrated Model of Four Architecture Views

Scope

The entire enterprise, as defined, including:

- Work organization
- Information
- Applications
- Technology.

Many planning methodologies have a process within them that advocates the creation of a target architecture. Frequently, however, the target architecture is too general and is of little value (e.g., "We will use a relational database management system for client files"). At the other extreme, the target definition can be too product specific to be considered truly open (e.g., “We will use IBM’s DB2 for our client files”).

The key to creating a quality blueprint document is defining the target architecture in such a way that it would remain open and flexible over time as technology, products, and infrastructure evolve.

Deliverables

A *Target Architecture Document* that describes:

- Target architecture with the four views defined, as well as the key interrelationships across the views. A sample outline for this document is included in Appendix I.

Critical success factors

An AWG that has:

- A combined general understanding of the current functions and processes of the enterprise
- Experience in long-term functional area and IT planning
- A practical understanding of the tradeoffs between functional issues and technology
- A working knowledge of systems development and maintenance
- An effective communications vehicle between the ASC and the AWG.

It is extremely important to staff the AWG with seasoned professionals. To do otherwise can be disastrous. Team members must come to the planning table with experience in business and IT planning. They must also have the political sensibilities to understand the limitations inherent in their work environment.

Constraints

- Lack of functional area and technology vision in the AWG
- Lack of full-time commitment to the project by management for key knowledge worker participation in workshops
- The team’s inability to comprehend the potential of the SBA process.

Task list

- Initiate task
- Define target architecture environment planning process
- Assign team to review the *Architecture Framework Document*
- Develop the work view of the architecture
- Develop the information view of the architecture
- Develop the applications view of the architecture
- Develop the technology view of the architecture
- Create the draft *Target Architecture Document*
- Conduct review with ASC
- Finalize *Target Architecture Document*
- Distribute *Target Architecture Document.*

Reviewing the principles

In the first phase of developing the SBA framework, the key component of standards were developed—the architecture principles. All target architecture work is based upon these principles. Principles are similar in nature to a federal constitution. They become the central document against which all deliberate and explicit standards-oriented policies and guidelines are developed. In this phase, the target architecture principles are extended into more specific models of the four views of an integrated target architecture.

Detail the target with four views of the architecture

The target architecture defines the IT environment needed to support the organization over the agreed-upon planning interval (usually 5 or more years). Its aim is to achieve the vision for the future outlined in the *Architecture Framework Document* for all four views.

Work architecture

This work view of architecture is developed by identifying specific classes of users within the business environment (e.g., executives, planners, administrators, engineers, recruiters); business locations (e.g., headquarters, sales office, plant, warehouse); and a logical representation of the business functions that are required to deliver products and services. This “logical” unit of work is called a logical operating unit (LOU). These three basic components of the work view will ultimately be mapped to the applications (i.e., automated procedures), manual procedures, and information required to support the work. This linkage helps to integrate the work view with the other views of the target architecture.

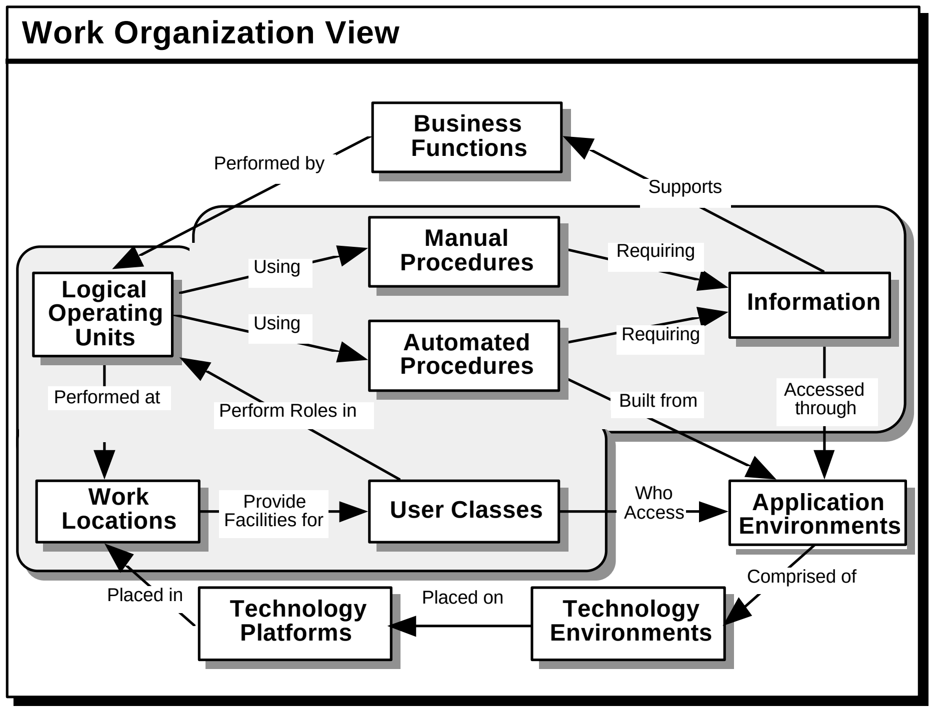


Figure 4-2. A Work View of the Architecture

This “logical view” of work will be independent of today’s line organization and/or physical locations. It will be the “pure” view of the work required to deliver products and services. This pure view can then be mapped to the existing physical organization and locations, allowing opportunities for IT automation, integration (of systems and functions), and/or work redesign to be identified.

Other views of architecture will impact the work view

The other three views of architecture (information, applications, and technology) may have an effect on the work view. As the definition of the future view of work proceeds, the process should include discussions of the information required by each LOU, the kinds of systems (applications) that may be needed, and the kind of technology that might support such systems. Obviously, at this early stage of architecture development, these views of the target architecture do not yet exist, but we can do some early, high-level analysis as a way to help us validate the LOUs in the work view. We want to capture the essence of these discussions to feed into the process of developing the detail of these other views of architecture.

The process facilitator's responsibility is to ensure that the team does not get bogged down in detail during these discussions and, more importantly, to ensure that a broad enough view of the future is taken.

Although there can be multiple ways to legitimately segment an enterprise's business, discussions generally yield 10 or fewer "Major Business Areas." The names for these major areas should not be confused with similar names for existing organizational units since they represent generic business functions, not existing departments or work groups. Start the process of defining these major business areas with a brainstorming session with executives and key knowledge workers from the enterprise. The facilitator should go into these sessions with the following generic major business areas "in their back pocket." These generic areas are used to guide the discussion if it begins to stray or if the teams get stuck and need a little help:

- Planning
- Selling
- Buying (raw materials acquisition)
- Manufacturing (or whatever the "core business" is)
- Delivery (product distribution)
- Collecting
- Support (including such things as finance, human resources, administration).

Each major business area is then broken down into its logical components of work, or LOUs. As with the major business areas, LOUs are not associated with the current organizational structure, its labels, the person performing the work, or any physical location.

Every LOU (see Figure 4-3) must provide a service and may have suppliers of products or services. It must be possible to measure its contribution; if not, it is probably not a LOU but an activity within a LOU. Each LOU is defined by the output (or service) for which it is conceptually responsible and the activities it must perform to achieve this result. A LOU always delivers its product or service to “customers” within the enterprise or within external actors beyond the boundary of the enterprise being modeled. A customer within the enterprise is always another LOU. A customer beyond the boundary of the enterprise is an external actor (e.g., “true” customers, suppliers, other Government agencies, parent organizations). Usually, a LOU will also be supplied with information or materials.

As the work organization view (i.e., a network of LOUs) is being developed, it is important to define the way the work should be partitioned and defined, not necessarily the way it is today. This network of LOUs should reflect the most effective and efficient way for the work to be done in the future. To achieve this, the LOUs themselves and their interrelationships will have to be developed, tested by applying various scenarios to them to see if they hold up, and refined as necessary to optimize the organization of the work within the enterprise. We may think of the major business processes within an enterprise consisting of the execution of one or more LOUs in sequence. In this sense, the LOUs are the major steps along the way in a business process.

A key point to remember is that a LOU may participate in more than one business process at varying points in time. Regardless of how many business processes a LOU participates in, its purpose, and the work activities that are executed to achieve that purpose, remain constant. In this way, the enterprise can develop policies, procedures, and supporting systems and tools for the most stable aspect of the business, the LOUs and, by definition, these policies, procedures, and supporting systems and tools will effectively support all business processes, which are made up of various combinations of LOUs.

The next step in developing the work view of architecture is to map the LOUs to classes of users who will perform the activities of the LOU. These user classes themselves are also logical in nature. As such, a physical employee of the enterprise may belong to one or more user classes.

Characteristics of Logical Operating Units

- The Logical Operating Unit (LOU) is the fundamental building block for defining architecture models.
- The LOU is defined primarily by its role in the production or delivery of one or more products or services within the operation of the enterprise.
- LOUs have distinct roles and responsibilities (no overlaps, redundancy, ambiguity, or gaps).
- It can be related to the overall contribution; the requirement for the LOU is clearly understood.
- Its performance can be measured.
- A LOU must have a customer and provide a service (it also may have a supplier).
- LOUs are independent of:
  - The organizational structure and departmental names
  - The degree of automation
  - Who does the work
  - Where the work is done.

Figure 4-3. Characteristics of Logical Operating Units

One or more user classes can be mapped to a given LOU, signifying that these user classes will perform at least one of the work activities of the LOU. A user class will not be related to the LOU if it only receives or passes information from or to the LOU. The user class must actually be the one performing one or more of the work activities defined within the LOU.

The final piece of the work view of the IT architecture is the concept of logical work locations. All of the “types” of work locations will be defined, regardless of how many physical locations may be involved. For example, “Base” might be a logical work location, while there may be multiple physical locations that contain this logical work location, such as Honolulu, Albany, and New Orleans. Figure 4-4 describes the process of identifying logical work locations.

Logical (and Physical) Work Locations

- Just as we wish to insulate our systems from the effects of organizational changes, we wish to insulate systems as much as possible from the effects of changing physical locations.
- To do this, we identify a set of Logical Work Locations. Similar to the way user classes allow us to categorize employees in terms of the roles they play, in a generic sense, the Logical Work Location concepts allow physical locations to be characterized in terms of the roles they play.
- There can be many Physical Work Locations that contain a given Logical Work Location.
- A given Physical Work Location may contain more than one Logical Work Location.
- In all cases, the Logical Work Locations should be set up to allow a reasonable mapping of Logical Operating Units (LOUs) against these locations.

  This mapping gives the architecture model the necessary linkage back to the user class. It also allows for a forward mapping to Physical Work Locations. These linkages are key tools in determining where application systems and supporting IT platforms will be located within the enterprise.

Figure 4-4. Logical (and Physical) Work Locations

With the logical characterization of work operations, users, and locations, supporting systems can be built that are completely independent of today's physical constraints. This provides the ability to develop the most flexible and adaptable systems.

As the user classes and logical work locations are mapped to the LOUs, additional refinements may be made on the LOUs themselves. Discussing who performs the work and where the work is performed will frequently lead to better ways to partition the work. No part of the work view of architecture is "cast in concrete" until all of the dimensions (LOUs, user classes, logical work locations) and their interrelationships are completely defined.

LOUs and their relationships to the other parts of the architecture and the outside world can be graphically depicted (see Figure 4-5). This is just another view of the basic relationships that were outlined in the target architecture modeling framework earlier in this section as the “Mother of all Models.”

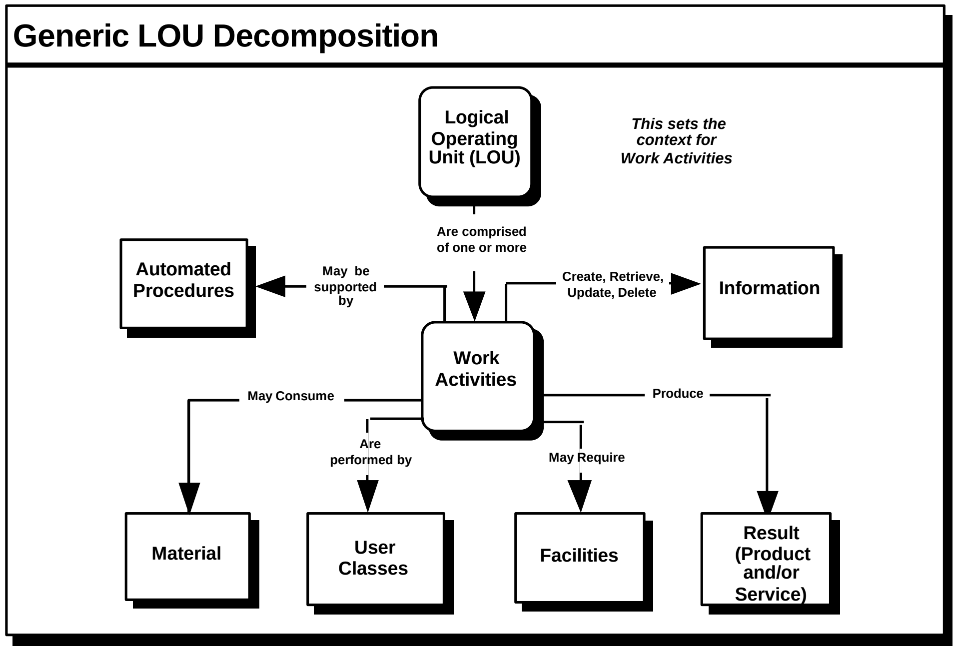


Figure 4-5. Generic LOU Decomposition

As an example of how to use the work view of architecture for analysis, Figure 4-6, the LOU to User Class Affinity Matrix, shows which user classes are likely to perform one or more of the work activities that make up a given LOU. This matrix is a key tool in the analysis of opportunities for automation and the linkage of these automated systems to work locations where these various user classes will perform their work.

Information architecture

The information architecture is composed of high-level subjects that represent all of the information needed to perform the work of the enterprise. The information architecture concentrates on the data being managed in support of the LOUs of work. Each major collection of data needed to support identified functions should be captured in the information architecture.

Major Business Area	Logical Operating Unit	Acquisition Specialist	Administrative Support Specialist	Aircraft Maintainer	Aircrew	Armor Warfare Specialist	Audio Visual Specialist	Civil Affairs Specialist	Communications Electrical Equipment Maintainer	Communications Operator	Embarkation Specialist	Engineer	Engineering / Utilities Equipment Operator	Executive	Facilities Specialist	Financial Specialist	
	User Class																
Plan	Develop and Direct Policy		X											X			
Plan	Develop and Establish Requirements	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Plan	Develop Doctrine and Tactics		X		X	X								X			
Plan	Establish Resource Priorities and Budgets	X	X											X		X	
Plan	Perform Overall Planning	X	X											X			
Train	Train & Educate		X				X										
Man	Acquire Personnel		X														
Man	Manage Personnel		X														
Equip	R&D Equipment and Systems	X	X	X	X	X			X	X			X			X	
Equip	Procure Equipment, Supplies, and Systems	X	X													X	
Equip	Support Equipment, Supplies, and Systems		X	X		X			X	X			X				
Conduct Ops.	Perform Intelligence Process		X		X	X		X				X					
Conduct Ops.	Maneuver Forces & Employ Weapon Systems				X	X						X		X			
Conduct Ops.	Perform Operational Planning		X		X	X		X			X	X		X			
Conduct Ops.	Perform Other Directed Duties				X	X		X				X	X	X			
Mission Support	Administer Distribution of Funds	X	X														
Mission Support	Conduct Ceremonial S																
Mission Supp																	

Figure 4-6. LOU by User Class Affinity Matrix

The information view, illustrated in Figure 4-7 is linked to the LOUs identified earlier, showing where the information is created, used, modified, and/or deleted, over time. The information architecture includes a discussion of the principles of information management as well. The AWG makes decisions that should facilitate this information management process. The models should reflect the workshop participant's best judgment about the future uses and characteristics of information within the enterprise. User access to this information across various business locations is also considered here.

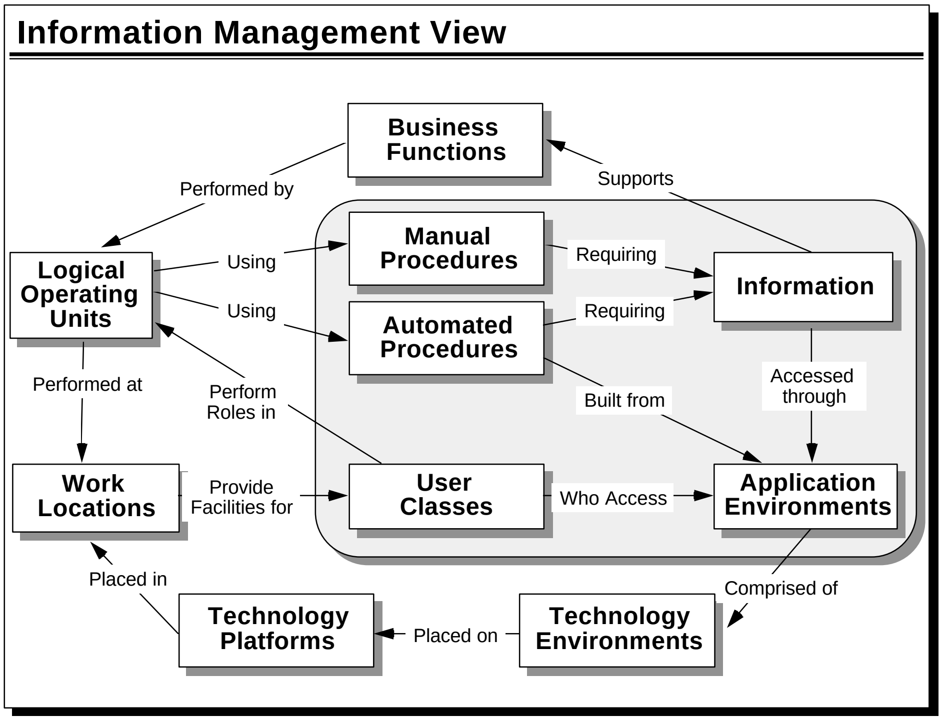


Figure 4-7. Information View of the Architecture

The information architecture for the enterprise will contain three levels of detail, subject areas, data groups, and data attributes.

The LOU to Data Grouping Matrix cross references all of the data groupings defined in the information architecture. This establishes the interrelationships among the data and the LOUs needed to perform the work of the enterprise. It will subsequently be used by systems designers as they develop the projects presented in the applications architecture.

The LOU to Data Matrix, illustrated in Figure 4-8, is used to show which of the LOUs either create, read only, update, or delete data within a given data group. Such a matrix is sometimes referred to as a “CRUD” matrix. This is due to the appearance of the letters C, R, U, and/or D in the cells of the matrix to show respectively Create, Read, Update, and Delete capability by a given LOU. This matrix is used in discussions of opportunities for automation. It is also very useful in decisions regarding the physical location of application systems and the data itself.

Major Business Area	Subject / Data Grouping / Logical Operating Unit	Agreement	Facilities		Financial	
		Contract/ Agreement	Roads	Structure	Budget	Disbursements & Receivables
Plan	Develop & Direct Policy					
Plan	Develop & Establish Requirements	R			R	
Plan	Develop Doctrine & Tactics					
Plan	Establish Resource Priority & Budget				CRUD	R
Plan	Perform Overall Planning	R	R	R	R	
Train	Train & Educate	R				
Man	Acquire Personnel					
Man	Manage Personnel					CRUD
Equip	Procure Equipment, Supplies, and Systems	CRUD		R	R	R
Equip	R & D Equipment and Systems	CRUD	R	R	R	R
Equip	Support Equipment, Supplies, and Systems			R	R	R
Conduct Operations	Perform Intelligence Process	R	CRUD	CRUD		
Conduct Operations	Maneuver Forces & Employ Weapons Systems		R	R		
Conduct Operations	Perform Operational Planning	R	R	R		
Conduct Operations	Perform Other Directed Duties					
Mission Support						
Mi						

Figure 4-8. LOU by Data Matrix

Applications architecture

This view of architecture focuses on the opportunities to automate aspects of work and/or the access to information needed to perform work (i.e., the target application systems to support the business). (See Figure 4-9.) Using the work view and the information required by each unit of work within this view, the team identifies application system opportunities, or clusters of functionality, required to support specific business needs. The application view of architecture shows the information usage and flow. The architecture defines the high-level scope and interfaces among applications, not the detailed requirements of each.

The team should identify all future applications that will be needed to manipulate the information and support the work being performed. In the process, the AWG should develop a set of high-level application descriptions. These descriptions are intended to serve as a first-cut view of the major applications.

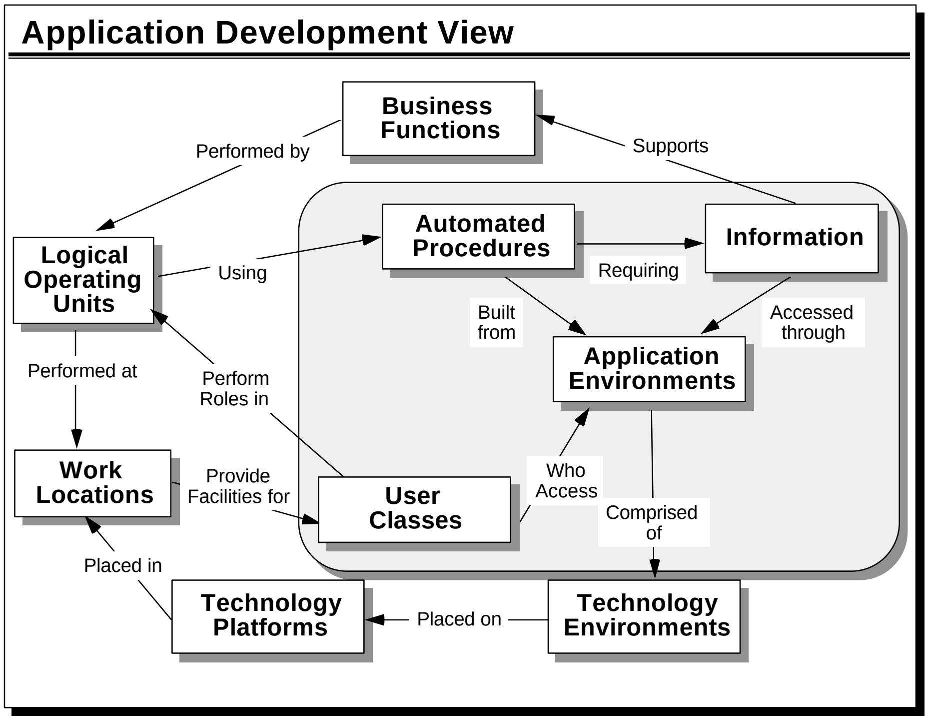


Figure 4-9. An Applications View of the Architecture

A matrix should be developed that shows which applications require read-only access to specific data and which applications may both read and update specific data. (See Figure 4-10 for an example.) Such a matrix is sometimes referred to as an “I/O” matrix. This is due to the appearance of the letters I or I/O in the cells of the matrix to show respectively Input only or both Input and Output capability against particular data. This mapping will be useful in decisions regarding the physical location of the application systems and the information itself.

Technology architecture

This part of architecture development typically requires a reversal of the workshop backroom sessions approach used in developing other views of architecture. (See Figure 4-11.) It is in this phase where, as the old joke goes, “a miracle happens.” Usually, the technology architecture models begin to emerge in the mind of a single technology architect who has some quiet time to mull over all of the deliverables of all prior phases and the three views of the target architecture that have already been developed in this phase. This person will have some rules of thumb and years of experience to guide him or her, but it is still somewhat more art than science. This section gives an overview of the thought process that such a technology architect might follow.

Subject / Data Grouping / Application System	Agreement	Facilities		Financial	
	Contract/ Agreement	Roads	Structure	Budget	Disbursements & Receivables
Aircraft Control System		I	I		
Automated Intelligence Analysis System	I	I/O	I/O		
Automated Software Catalog System			I	I	I
Career Management System					
Computer Services Chargeback System				I	I/O
Construction Estimating System	I	I/O	I/O		
Deficiency Identification System	I	I/O	I/O	I	I/O
Doctrine Data Base System					
Fire Support Control System		I	I		
Force Automated Routing and Travel System					
Force Management System	I	I	I	I	
Ground Position/Location System					
Imagery Dissemination System		I	I		
Incident Reporting System					
Information Technology Capacity Management			I	I	I
Integrated Accounting System					
Inventory Distribution Managem					
Joint Task Fo					

Figure 4-10. Information by Application Matrix

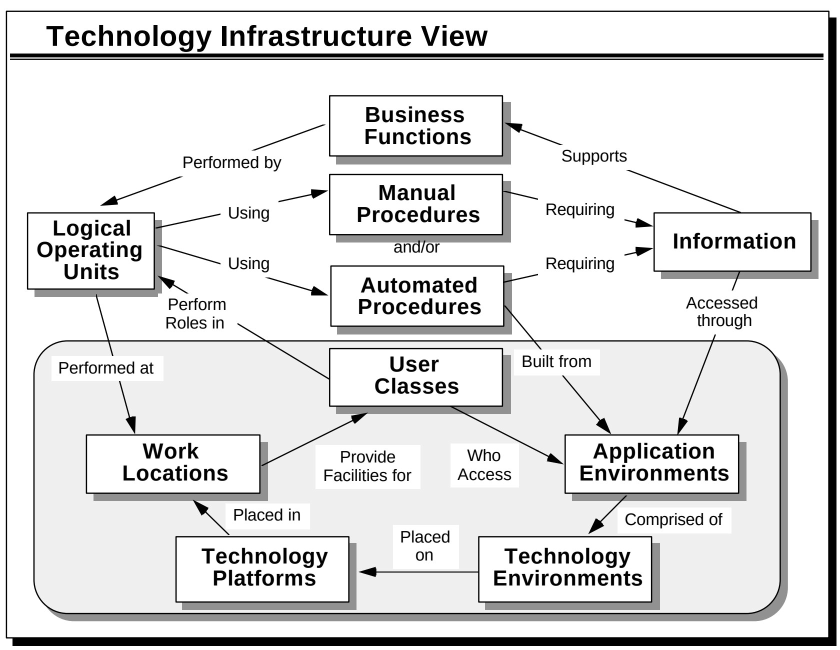


Figure 4-11. A Technology View of the Architecture

This area of architecture uses specific component-level models to provide the basis for linking the technology view of architecture to the work, information, and application views. The linchpin to the other views of architecture is the generic application environment.

With each application area characterized in terms of its generic application environment(s), the other components of the technology architecture can be defined precisely. Using additional component models and generic terminology, the technology architecture will describe the IT infrastructure (framework) required to support the enterprise's objectives as characterized by the other three views of architecture discussed earlier.

Technology architecture building blocks

Three types of building block models (sometimes referred to as constructs) are used in building the overall Technology Architecture Model. They are described below.

<u>Generic application environments</u>

Generic application environments (GAEs) describe types of IT applications and tools needed to support specific application systems. This is the primary building block in linking application systems back to the technology environment.

<u>Generic technology environments</u>

Generic technology environments (GTEs) describe types of services required to support GAEs (i.e., system software). GTEs provide a means of defining a technology environment that has a standard set of characteristics and attributes. Each GTE uses a set of "servers" that provide specific technical capabilities for the GTE. Like the GTEs, the servers are generic components with standard interfaces to the "clients." They are built on, but independent of, specific technology implementations. The result is a layered technology that, if implemented through a rigorously defined set of interfaces, can isolate applications and major technology components from differences in the underlying technology implementation.

GTEs provide the SBA link from GAEs to the technology components and technology implementations within an organization's infrastructure. Each GAE is supported by one or more GTEs. The combination of the GAEs and GTEs provides the infrastructure components for delivering systems and services to the organization.

Generic technology platforms

Generic technology platforms (GTPs) describe the delivery components required to run the applications that ride on the GAEs (i.e., “system hardware”).

These generic modeling constructs are planning tools which provide a framework for comparing current and target environments. They also support standards-based architecture planning. They are not, in and of themselves, the final deliverable but are used as a tool to aid in developing the specific technology architecture.

Six technology constructs or GTPs provide fundamental building blocks in a standards-based architecture. Each GTP can function as a fully independent “architecture” in that each has an interface along with processing, storage, and communications capabilities. As such, each GTP may offer alternative choices in delivery of the same GAE. For example, all six constructs are capable of supporting some form of electronic mail, with different associated strengths and weaknesses. These six GTPs include:

- Intelligent WAN systems
- Establishment-based switching systems
- LAN systems
- Enterprise or corporate processing systems
- Divisional or departmental processing systems
- Desktop or portable intelligent workstations.

It is important to note that the GTPs do not connote a particular size/capacity. The names for the GTPs connote the usage of the processor, not size. In fact, departmental processors may be larger or smaller than enterprise processors. Some processors acting as LAN servers could well be larger than departmental or enterprise processors depending on the way a given organization wishes to organize its work.

How to use the building blocks

The generic building blocks just described are useful in the process of developing the target technology architecture. The end result of such a process is best shown by the use of an example. The target technology architecture developed for the U.S. Marine Corps provides an excellent example of the output from this process, and the reader is referred to the target architecture deliverable from that project.

The thought process that was used to produce the USMC technology architecture is guided by technology “rules of thumb” based on experience and informed by the other views of the architecture. Specific characteristics of work, information, and applications enter into the interpretation of these rules. Some of these rules are:

- Keep the processor as close as possible to the users of systems residing on the processor
- Maximize independence between major application groupings (stepwise escalation from loose coupling to tighter coupling)
- Within major groups of applications, look for ways to gain tighter coupling (such as shared databases)
- Establish the smallest practical set of standards as possible
- Maintain vendor independence in standards for as long as possible
- Take locations into account but do not “agonize.” (Follow accepted rules of the road and the effect of being “off” on locations will be minimized.)
- Be pragmatic—do not wait for the ultimate environment. Build up to it by accepting some short-term compromises while keeping as many options open as possible.

In addition to this guidance, there are other practical issues to consider about the placement of applications on technology platforms. The support requirements of the applications can be used to assess which platform is a best candidate for placement. For example, highly individualistic applications and tools (e.g., text processors, CAD/CAM, CASE) have a high affinity for the desktop. Applications requiring the need for terminal support or which act as the server side of client/server applications have a high affinity for departmental or enterprise processors. Finally, infrastructure applications such as E-mail, EDI, and other common services have a high affinity for LAN or other network servers.

Techniques to arrive at the target technology architecture

There are recommended steps to follow to analyze the other architecture views that will facilitate the process of defining the target technology architecture.

First, begin by reviewing the characteristics of information. Produce a first cut map of the technology platform using rules of thumb. Then, review the characteristics of applications. This should result in a first cut map of each application to technology platform where the bulk of the most demanding data resides.

The CRUD matrix should next be reviewed to gain insights about potential data sharing and the effect this will have on data distribution. Also, the application to information (I/O) matrix should be reviewed for similar insights (and potential adjustments).

Each of these steps is performed in an iterative fashion until all applications, data, and associated platforms are mapped to logical work locations. By now, a reasonable model should begin to emerge that can be tweaked by looking at the form of information and the potential impact on network traffic. Finally, with all of these steps complete, some judgment calls can be made about the style of computing:

- Distributed presentation
- Remote presentation
- Distributed function
- Remote data management
- Distributed data management.

Capacity requirements should be considered as well to finalize the model. This last step represents the final "proof" of the model. The information volume, timeliness, and currency requirements, along with application availability and reliability, can be used to make a guess at the scale of the processing platform required at each location. The volume, timeliness, and currency requirements can be used to categorize the network transmission capacity needed between locations. The result of this examination of capacity issues may cause some final adjustments in application and information distribution across the network.

Standards model

To implement the standards-based infrastructure, it is important to consider the scope and depth of the standards to be adopted. Fundamentally, all cases of standards adoption require answering three questions:

- What standards should I adopt?
- Where in my architecture should I adopt them?
- When should I adopt them?

Both TAFIM Volumes 2 and 7 should be used in this phase. Volume 2 suggests a standards-based model for user interface, database, applications, operating system, communications, languages, management, and other services. Volume 7 identifies the standards and specifications approved by DoD as the method for satisfying those service areas. Architects are encouraged to select appropriate standards and specifications from Volume 7 to form a standards profile. Profiles vary as functional requirements vary. The AWG must be prepared to define the details that underpin each section of the diagram for their functional area's particular implementation. Appendix C on detailing the target architecture can also be a good reference point for teams attempting to define the details of the standards model.

De jure vs. de facto standards

A target architecture must be developed such that it will permit implementation migration towards full standards compliance — described as either de jure or de facto.

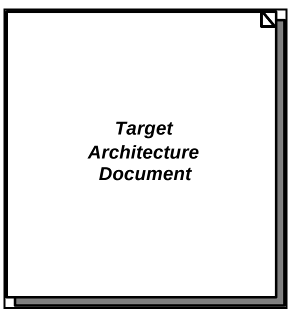
Business requirements should not be compromised strictly for the sake of "open systems." However, whenever a de jure standard is available in effective price/performance product form, it should be implemented as quickly as possible. Specifically, the de jure standards should be:

- Specified in policies, guidelines, and architecture

- Products and services based on standards-based policies and guidelines selected whenever viable competitive cost/performance alternatives to proprietary solutions exist in the marketplace.

Product implementations today tend to be based more on de facto standards than de jure standards. The target architecture effort should take this reality into account. Products based on de facto standards are more available today in the marketplace. A standards-based architecture based solely on de jure standards may be elegant and pristine in concept but can also be essentially sterile because so few of the adopted standards are actually in the marketplace via vendor implementations. All effective standards-based architectures must acknowledge the hybrid nature of this reality.

Creating and publishing the deliverable



The target architecture is one of the more creative aspects of the SBA process. The deliverable is arrived at only after significant thought has been invested in an iterative review of the baseline material. The architecture principles should be clearly reflected in the target architecture, and the technology view should be capable of supporting the new work processes envisioned in the target.

- Clarity of Target Architecture Document
- Management acceptance of Target Architecture Document.

Effectiveness measures

- Ability to map from current embedded base to target architecture
- Inherent flexibility in the SBA action plans.

The effectiveness of the *Target Architecture Document* will ultimately be determined by the degree to which it is used by the DoD. As discussed earlier, the document must be easy to understand and must set a reasonable target, otherwise no one will use it.

Technology and tools required

- Word processing tools (with graphics capabilities)
- Spreadsheet tools
- Business graphics and drawing tools
- Work room for AWG meetings.

It’s important that the blueprint document be highly visual (i.e., contain many diagrams and checklists). The easier the document is to understand, the more likely it is to be used and referenced.

Appropriate resources should be dedicated to creating a user-friendly blueprint. Some organizations have even gone so far as to hire layout artists to streamline the document. While this kind of zeal is not required, too much emphasis cannot be placed on making the document easy to use (i.e., technology should be available to facilitate the creation of a quality deliverable).

Staffing skills required

- Experienced planners
- Business professionals
- Acquisition experts
- Information technologists
- Writing and presentation skills.

The ASC will provide guidance, direction, and high-level review for the work of the AWG. The AWG will be responsible for assisting in facilitating working sessions and for producing the deliverables in the planning process. This team will have broad, non-overlapping backgrounds in the business to be modeled.

Key executive and knowledge workers need to be available for interviews and/or workshop sessions according to a schedule to be developed within the initial weeks of the project.

The AWG will develop a working *Target Architecture Document*. This document will then be reviewed with the ASC and other key stakeholders within the enterprise (see Figure 4-12).

The committee then sponsors a draft document that is reviewed, amended, and approved by the appropriate players within the enterprise. This is typically a management group composed of functional area heads and the Chief Information Officer or his/her equivalent. In some organizations, the chief executive will review the SBA document.

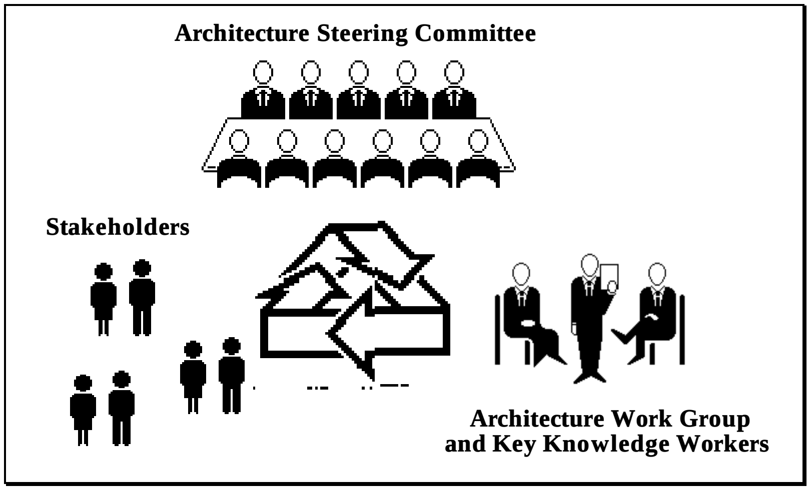


Figure 4-12. The Review and Approval Cycle

Completion criteria

- Creation of reviewed and reconciled models
- Creation of target standards (if part of agreed-upon scope)
- Management acceptance of target definition
- Acceptance of *Target Architecture Document.*

Standards, as articulated in the policies and guidelines section of this document, will be the core to enabling construction of the standards-based infrastructure. This document will be a key input document to the remaining steps in the implementation cycle, particularly in identifying opportunities and migration options.

Issues

- Workload of architecture work team(s)
- Target architecture scope management
- Key knowledge workers' availability for workshops
- Trained, experienced standards-based architects
- Correct understanding and anticipation of the future.

It is essential that the AWG be properly trained in SBA planning practices and that members be full-time participants in the effort. This implies that management eliminate the pro forma activities that team members are typically required to perform.

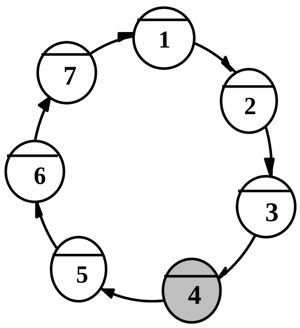
Failure to make a commitment to this effort seriously can result in the execution of another tired planning exercise that carries little or no weight within the function after its completion. The old adage “you get out what you put in” truly applies to SBA planning projects.

Section Five: Opportunity Identification

Contents **Page**

Figures

Section description



This section describes the overall process by which the AWG categorizes and identifies opportunities for exploiting the target architecture. Opportunity identification is the phase dedicated to identifying the projects needed to move the organization from the present to the future (the target architecture). This phase defines the parameters of change, the major steps along the way, and the major activities to be undertaken.

Objectives

To identify key opportunities for implementing the target architecture environment on a "fast path" basis while also developing a context for development of migration options and detailed implementation plans.

In the opportunity identification phase, projects necessary to move the organization from its current environment (as defined in the baseline deliverable) to its target environment are identified. This includes a detailed description of the automated and non-automated initiatives that will be necessary to reach the target architecture. This phase will flesh out the application, non-application (technology infrastructure), and non-technology initiatives that should be implemented to achieve the vision of the organization.

At this stage in the project, it is sufficient to provide documentation of the essential steps needed to achieve the target and not to provide a cost/benefit justification for these projects. This will be done in the migration options phase as projects are justified and ordered into plateaus, and dependencies between projects are identified.

The AWG identifies various opportunities through workshops and work group analysis. These opportunities are tested and filtered by the business and IT functions. Once finalized, the opportunities are documented in the *Opportunity Identification Document.*

Scope

Throughout the AWG process, numerous opportunities are identified for introducing standards-based architectures and harvesting benefits associated with the proposed architecture solutions. During this phase, the identified opportunities are classified with regard to a number of criteria. This classification scheme becomes the foundation for migration planning and implementation.

Experience shows that if the project cannot deliver fundamental opportunities on a short-term payoff basis within 3 to 6 months, the rest of the standards-based architecture will probably never be implemented. Results are critical to success. This places a premium upon identifying opportunities that are:

- Short and medium term in nature
- Low risk, high payoff in implementation
- Offer a high degree of freedom within the existing architecture so that they may be implemented easily and migrated to as quickly as possible.

As is customary, many opportunities are identified at the same time that the application component of the target architecture is being developed. Therefore, the systems introduced there appear here as project opportunities. Also included is the definition of infrastructure projects (i.e., technology features that must exist in order for the applications to run) and non-technology projects (i.e., non-systems projects that are necessary to achieve the vision of the future presented in the target architecture).

Figure 5-1 illustrates the contrast between the traditional information plan and the standards-based fast path implementation focus:

Deliverables

An *Opportunity Identification Document* that contains:

- Description of project opportunities
- Dependencies
- Issues to be considered.

In this phase, the team describes the opportunities in general terms, the size and scope of opportunities, as well as the dependencies that need to be considered when the time comes to deliver the project. A sample outline for this document is included in Appendix I.

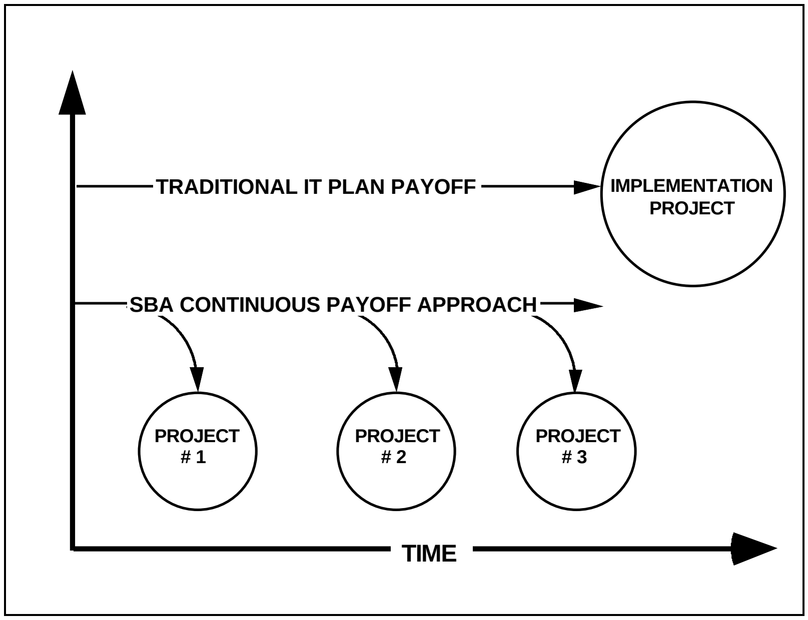


Figure 5-1. Implementation Payoff Approaches

Critical success factors

- Understanding of the implementation challenges and payoffs at a high level
- Understanding of the Baseline Characterization Document, Target Architecture Document, and other source data
- Experience in business and IT planning
- Practical understanding of the tradeoffs between business issues, technology, tactical, and operations settings
- Understanding of Federal procurement guidelines and issues
- Working knowledge of systems development and maintenance
- Familiarity with IT security planning
- A systems migration planning background
- An effective communications vehicle between team members and from the ASC and the AWG.

In this phase, it is important that the AWG balance the strategic long-term objectives of the target architecture with a reality-based tactical view of what may be accomplished in the near- to mid-term time frame. Grand plans are indeed grand; in most cases, they either fail or never see the light of day. Unfortunately, most implementation efforts are judged by the first projects delivered rather than on the merits of the overall design rationale. Thus, the demonstrable practicality, efficiency, and effectiveness of the proposed projects will be used to assess the success of this effort.

Constraints

- Working within the current architecture paradigm may limit the team's ability to "see" new opportunities
- Vision (or lack thereof) may limit successful execution
- Lack of a coherent business case.

Many times, implementation efforts focus only on tactical programs. The ability to discern opportunities is only increased when team members have a structured approach and are able to see beyond the constraints of the current environment.

Task list

- Initiate task
- Identify gaps between baseline and target architectures
- Identify payoff categories
- Identify key payoff projects
- Draft *Opportunity Identification Document*
- Conduct review with ASC
- Finalize *Opportunity Identification Document*
- Distribute *Opportunity Identification Document.*

Gap analysis

Determine the "gaps" between the baseline and the target in all four views of the architecture. Spreadsheets are a good tool for this. One approach might be: across the top, list all of the "target" components of a given view (e.g., future business processes, future information facets, future applications, or future technology components). Along the left-hand column, list the current components. In each cell of the spreadsheet, account for all current components. Some current components may be eliminated. For example, an "auditing" work process may be "non-value added" for the future; therefore, it is eliminated. For cases such as this, create another column in the spreadsheet entitled “eliminated.” On the other hand, new components may be added. For example, a new service process may result in higher satisfaction for the user of the service. For cases such as this, create another row entitled “new.” All eliminated components and new components create gaps. The identification of opportunities must fill these gaps. Figure 5-2 below illustrates this technique for determining gaps.

Target / Current	Solicit Business	Fill Order	Provide Customer Service	(Eliminated)
Take Order				GAP
Fill Order		Okay		
Audit				GAP
(New)	GAP		GAP	

Figure 5-2. Gaps Between Baseline and Target Architectures

Payoff categories: the opportunity context

A number of benefits are associated with open systems and standards-based architectures. The TAFIM series highlights the implementation opportunity initiatives that support portability, scalability, and interoperability of applications and systems. As such, it defines an “opportunity vision” for the future. It was devised to permit the DoD to take advantage of the benefits of open systems and new standards-based technologies available in the commercial market.

Specific objectives for the DoD TAFIM include:

- Improving user productivity
- Improving development efficiency
- Improving portability and scalability
- Improving interoperability
- Promoting vendor independence
- Reducing life-cycle costs.

These objectives may be used as categories for evaluating implementation “payoff” opportunities (see the TAFIM, Volume 2 for more detail.)

Creating and publishing the deliverable

The key deliverable in this phase is the *Opportunity Identification Document.* It should focus on providing the ASC with a high-level understanding of the opportunities at hand. As described in the opening of this section, the document should focus on highly visible short-term payoff projects with a “continuous payoff” approach to implementation opportunity identification. The document’s entire objective is to describe the nature of the target architecture opportunities and the role they will play in closing the gap between the baseline environment and the target architecture.

Effectiveness measures

- Degree to which implementation plans can be developed
- Management enthusiasm regarding opportunities identified.

This phase will vary widely in terms of calendar time required for completion based on organizational culture, individual schedules, and the formats that organizations are accustomed to using. Ideally, when conducted on an intensive basis, this phase may be completed in 6 to 10 weeks. The draft and final iterations of this document should be reviewed with the ASC before any action is taken and changes made accordingly. As with other deliverables, the document should go through several draft iterations before being approved by the ASC.

Tools required

- Word processing and graphic presentation packages
- Architecture team room for meeting
- Spreadsheet tools and/or user-friendly personal computer-based database packages for inventory logging.

It is key that the AWG put together a high-level presentation for the ASC that highlights the features and logic of the implementation opportunities it has identified. “Selling” the architecture to the ASC must be done on this basis.

Staffing skills required

- Migration planning skills
- Software modeling skills
- Writing and presentation skills
- Organizational data collection knowledge
- Familiarity with word processing, presentation, spreadsheet, and database packages that run on most popular personal computers.

This phase requires individuals who are familiar with project definition and who understand the requirements of the next phase in the process, which will assess the benefits and risks associated with such projects as well as the priority which should be placed on each. Ultimately, each of the projects must be justified in terms of its contribution to the target architecture or as a stand-alone project. The goal of this step in the process is not to encourage the creation of an undisciplined wish list. Rather, there is every expectation that the minimum set of projects (automated and non-automated) necessary to achieve the vision will have been identified.

Completion criteria

- *Opportunity Identification Document* completed
- Management acceptance of Opportunity Identification Document.

This phase is completed when the ASC accepts and signs off on the *Opportunity Identification Document*. It is important that all the ASC members, as well as the AWG, have a shared understanding of its content since it will become the basis for developing migration options and for implementation planning.

The AWG should obtain a sign-off that ensures full ASC approval as with all other steps in the process.

Issues

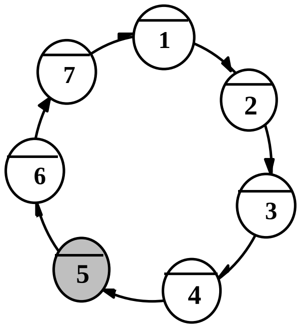
- Executive "buy-in"
- Workload of work team(s)
- Consulting required
- Training required
- Subject matter expert availability.

Section Six: Migration Options

Contents **Page**

Figures

Section description



This section describes the overall process by which the AWG identifies and develops migration options for moving to the new target architecture. This section also describes the overall process by which the AWG categorizes and identifies opportunities for exploiting the target architecture and shows how such opportunities can be justified in areas such as their cost-to-benefit ratios or the role they play in providing support for future projects to be implemented as part of the target architecture. Included in this activity are descriptions of how the *Migration Options Document* is developed.

Migration planning is the phase in the process when all essential projects are sorted into plateaus for implementation planning. The sort process is based on the interdependencies between projects. In addition, projects are sorted by strategic value. Those with greatest payoff or strategic significance should be implemented as early as possible to take maximum advantage of the value they represent. Finally, cost is considered in developing the implementation plan. Cost is an important consideration in recognition of the fact that budgets are limited and most, if not all, expenditures must be justified in terms of the benefits they will provide or in terms of the essential infrastructure support they represent. The following provides a feel for the content of this phase:

- Estimates of the work and resources required to migrate from the current environment to the target environment are developed with resource estimates and responsibility assignment.
- Comparison of target to baseline architecture is performed to identify areas where the current situation satisfies the target requirements and where gaps exist.
- High-level plans for migrating from the current to the target architecture are described and dimensioned.
- The migration plan must account for organizational change and must also be flexible enough to accommodate changes in the architecture itself as the migration plan is being implemented. We refer to this last step as “innovation-proofing” the architecture. The output from this phase is similar in nature to the

document that is produced after the architect has blueprinted the architecture of a building project—a “construction plan” that tells the builder how to actually erect the building.

Objectives

To develop a comprehensive, prioritized set of project initiatives, which, when completed, will move the enterprise from the current state to the target architecture.

The AWG identifies alternative construction options. Major critical implementation steps are developed by the AWG. The detailed implementation plan is then reviewed, not only with IT, but with functional area personnel to assure that time frames are realistic and goals achievable. Project implementation responsibilities are assigned, as well as implementation dates, based entirely on functional area requirements. This entire phase is documented by the AWG in the *Migration Options Document*.

Scope

This phase will identify all projects required to fully implement the target architecture.

The AWG must determine how many areas of the target architecture to tackle at one time as well as the interdependencies between the components. Theoretically, all four views of the target could be pursued simultaneously. However, practically speaking, they will be easier to manage if they are handled in an independent but related manner. These two conceptual approaches are shown in Figure 6-1.

After a high-level determination is made on which of these dimensions of the architecture are to be addressed, and a high level description of the necessary projects has been created, the scope of each project is defined. This should include a project statement, a scope definition, the major components of the project, and major steps to be covered during the project’s life cycle.

A project scope statement addresses and delimits a project that is as small as putting a standard user interface across a group of applications. Alternatively, the project could be on a much larger scale wherein all major work processes in a customer service environment, as well as the standards-based technology to support them, are reengineered.

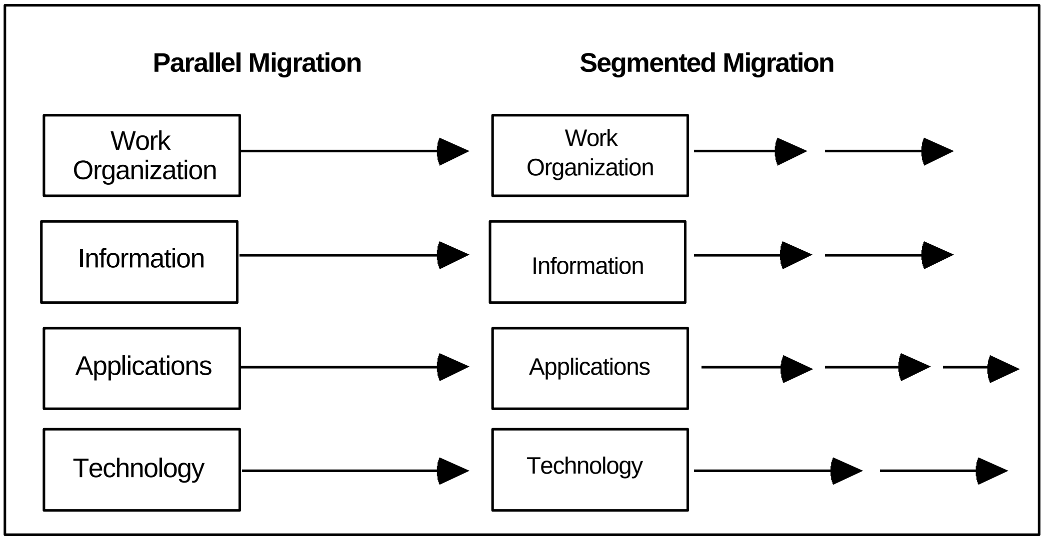


Figure 6-1. Migration Approaches

Deliverables

The *Migration Options Document* provides specific recommendations for the priority of the project initiatives that must be performed to move the enterprise toward the target architecture. This document should include a thorough discussion of the migration plateaus. A sample outline for this document is included in Appendix I.

Plateaus are fashioned to deliver "clusters" of business benefit. There are usually three plateaus, with the first plateau containing some "quick hit" projects as well as the highest priority major projects:

- Plateau 1 — projects beginning in years one and two
- Plateau 2 — projects beginning in years three and four
- Plateau 3 — projects beginning in years five and beyond.

This document should indicate the priority order of the projects and ballpark costs associated with each plateau. The following are the key sources of information (from prior phases of the SBA) that are used in the migration options phase.

Baseline application assessment charts

These deliverables classify all existing applications as to recommended disposition based on target architecture requirements and the rating of the existing applications against standard criteria. The result is that the applications are placed in one of the following four categories:

- Renovate/reengineer
- Replace/discard
- Keep/tune
- Asset/build upon.

Target application characteristics

This deliverable provides a number of characteristics of envisioned application systems for use in prioritizing these applications. The most important characteristic is the application's perceived contribution to strategic drivers (i.e., a measure of the strategic significance of the target application). This allows the target applications to be sorted in order of highest strategic significance.

Target application to existing application matrix

This deliverable provides the connection between identified future application functionality and existing applications that may currently supply some (or possibly all) of this functionality. It combines this mapping with the assessment of the existing application and the target application's strategic significance.

These source deliverables provide much of the rationale for the prioritization. They are also valuable in arriving at the ballpark cost estimates.

Critical success factors

- Understanding of implementation challenges and payoffs
- Experience in business and IT planning
- General cost/benefit orientation towards technology planning
- A team that has experience in implementing one or more of the target areas (i.e., work flow, application, etc.)
- Migration options that avoid full conversions.

Conversions tend to conflict with functional area priorities. Migration to open systems will take many different paths for users. It will depend upon the embedded base of existing systems and the rate and speed the enterprise seeks to move into target systems over time.

If open systems standards are specified in the target architecture, other considerations must be reviewed. For most organizations, the move into open systems will mean maintaining separate environments over some period of time and running parallel environments. This should be factored into the business case for open systems. When the overall case is examined in terms of long-term benefit, the parallel environment will be most cost effective. This is typically the case in spite of the fact that the initial and additional cost of running parallel environments may skew the cost case against parallel facility-based migration.

Delays in implementing a migration strategy to a standards-based architecture may ultimately increase the number and effort of conversions required.

Constraints

- Inexperience in migration planning may limit the team's ability to develop a realistic and acceptable set of migration options.
- The existing work organization may be unable to adjust to the options defined.

As part of the *Migration Options Document,* it is important that the AWG consider issues surrounding organizational change processes. These include, but are not be limited to, the establishment of an ongoing architecture review board and process. The architecture management function itself needs to be authorized to specify architecture standards, administer implementation of the additional strategy, roll out standard tool sets used in the SBA process, and audit compliance with those standards. Thought should be given to establishing a system architect role or function, if the function does not presently exist.

In addition, it would be helpful to describe the various work flow and organizational change processes associated with implementation of the new architecture. This should be an integral part of the overall planning process, because this is where the synergy of organization and standards working together will be most powerful. These and other concepts are covered in more detail in the final phase of the SBA process, SBA administration.

Some of the issues to be faced in this phase are listed below. The AWG should review, modify, and extend this list to make it more meaningful to its specific DoD functional area. This can help ensure success in the SBA process.

- Embedded legacy systems must remain in place for some time for investment or work force resource reasons.

- Open system products which implement de jure standards simply do not exist for many requirements.
- Proprietary solutions can be very effective price/performance solutions if the larger cost savings associated with implementation of open systems are not well understood.
- Organizational inertia—implementing technological change is as much a cultural, organizational, and political challenge as it is a technical process.
- Lack of cohesion between the IT technical community and the function-oriented players.
- Lack of an organizational strategic vision can lead to squandered resources as funds are spent on insignificant or inappropriate efforts.

Lack of a planning and implementation process with which to identify common requirements for standards-based systems.

It is important for the team to remember that with standards-based planning it is possible to eliminate entire classes of technology and replace them with new technology platforms. For instance, an organization can:

- Move applications from mainframes to mid-range platforms
- Move applications from mid-range platforms to high-power networked workstations
- Move applications from master/slave implementations to cooperative processing implementations within an existing proprietary architecture
- Migrate connectivity services (such as E-mail) from proprietary mid-range platforms into a diverse, multiplatform standards environment (X.400) with a parallel strategy for directory services (transition to X.500)
- Implement UNIX-based workstations and servers and replace an entire existing application and platform portfolio.

Task list

This phase will determine the migration plateaus needed to reach the vision by the target date. It is improbable (and probably not recommended) that the organization will want to implement the vision all at once. Usually the vision is attained (or the architecture implemented) by achieving a series of objectives, each of which builds upon the prior, until the vision is attained. The migration plan includes the tasks, timing, dependencies, and resources needed to achieve all the plateaus described in the migration strategy.

- Determine the gaps
- Use any available examples of applicable work
- Determine pace of change desired by the enterprise
- Determine the migration plateaus needed to reach the vision by the target date
- Determine components (work, information, applications, and technology) required to achieve the vision
- Produce migration plan implementation alternatives
- Include security planning migration considerations
- Draft *Migration Options Document*
- Conduct review with ASC
- Finalize and distribute document.

Gap analysis

This process phase is based on the gap analysis between baseline and target architectures. (See Figure 6-2.) The pace of change (i.e., how soon the enterprise wants to complete the implementation of the architecture), along with the priority and interdependence of the projects, will contribute to defining the plateaus needed to accomplish this vision.

Once the target architectures have been developed, the AWG should determine the degree to which the existing technology environments, applications, and platforms support the target environment(s). The data collected during the baseline characterization phase should be useful in this effort.

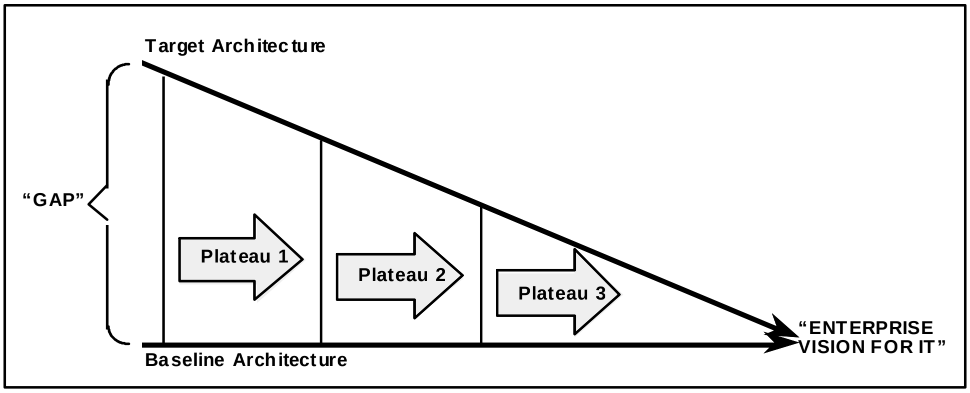


Figure 6-2. Closing the "Gap" Between Baseline and Target Architectures

Opportunity categorization

To initiate this activity, the AWG begins by categorizing each of the opportunities identified during development of the target architecture into three categories:

- Magnitude classification
- Risk classification
- Degrees of freedom classification.

After the opportunity is reviewed in terms of these considerations, the details of these classifications are put into the business cases for implementation consideration.

Magnitude classification

Primarily, the AWG seeks to determine whether or not the opportunities represent major architecture shifts from existing legacy systems in place or an incremental move towards standards over time. The team seeks to classify opportunities in terms of "moves" that may be made in standards-implementation over time.

In Figure 6-3, a user has decided to replace an entire proprietary system with a POSIX-compliant architecture implemented under an X/Windows user interface within a short time interval. Based on the architecture framework, baseline characterization, and target architecture work conducted by the AWG, this solution appears to be quite attractive from every dimension but must be characterized as a "radical" move. Every aspect of the "old system" architecture will be changed in quickly moving to the "new system" architecture.

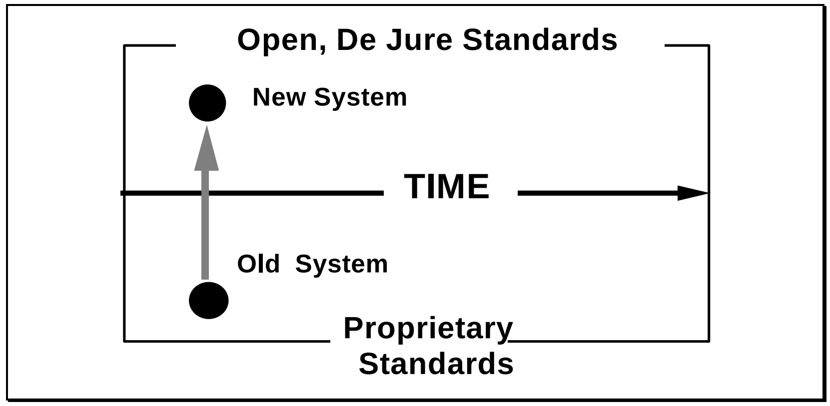


Figure 6-3. Radical Move to Open Standards

In Figure 6-4, the AWG has gone through the same planning process as the one previously described. However, it has decided to implement only OSI connectivity solutions within its proprietary "old system" architecture over the next 3 years. It will adopt SQL whenever possible in its database design activities, but only for new systems. Old databases will remain non-SQL compliant. Other than these two standards-related activities it will remain, for all intents and purposes, proprietary in its "new system" architecture, evolving towards "openness" over time. These moves may be characterized as *incremental*.

Risk classification

In addition to characterizing opportunities as incremental or radical in nature, they may be characterized in terms of risk as shown on the following two matrices. In Figure 6-5 the ideal "low risk, high payoff" opportunity is described in terms of migration.

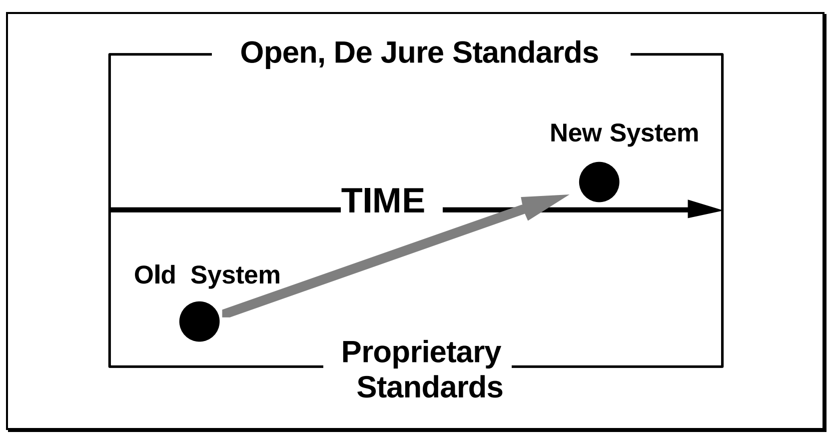


Figure 6-4. Incremental Move to Open Standards

Thus, we may use this example to classify an opportunity where the user is moving from a proprietary SDLC communications protocol to an open protocol such as X.25. In this case, the user is attempting to connect diverse functions via standards internationally. The new system is based on X.25 OSI packet switching protocol. Because X.25 is an established international standard and is widely available in products, it is therefore a low risk move. As a result of its implementation, the two hypothetical international functions will be able to connect their networks together quickly. The opportunity is high payoff in nature.

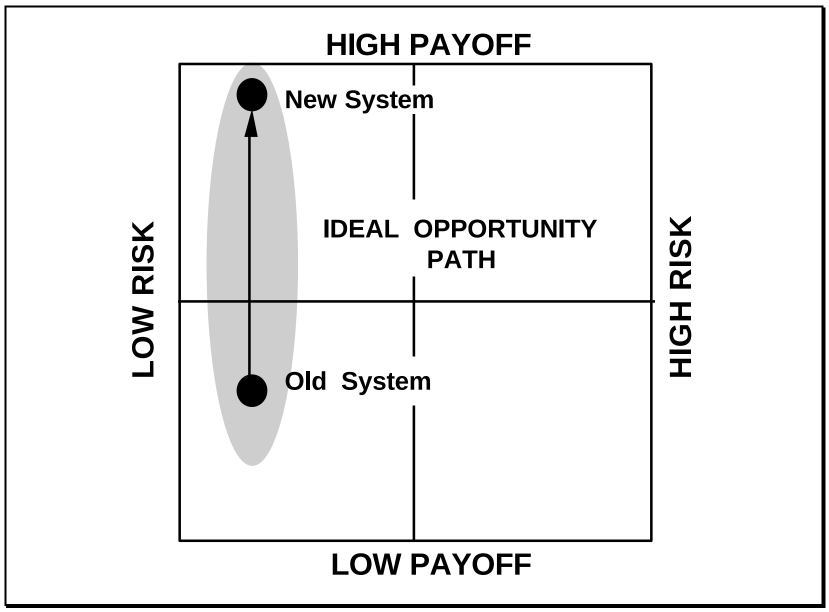


Figure 6-5. Risk: Ideal Migration Path

More often than not, however, the typical IT manager sets out to deliver a “low risk, high payoff” opportunity only to find himself or herself implementing a “high risk, low payoff” solution. Figure 6-6 shows this situation, as contrasted with Figure 6-5 above:

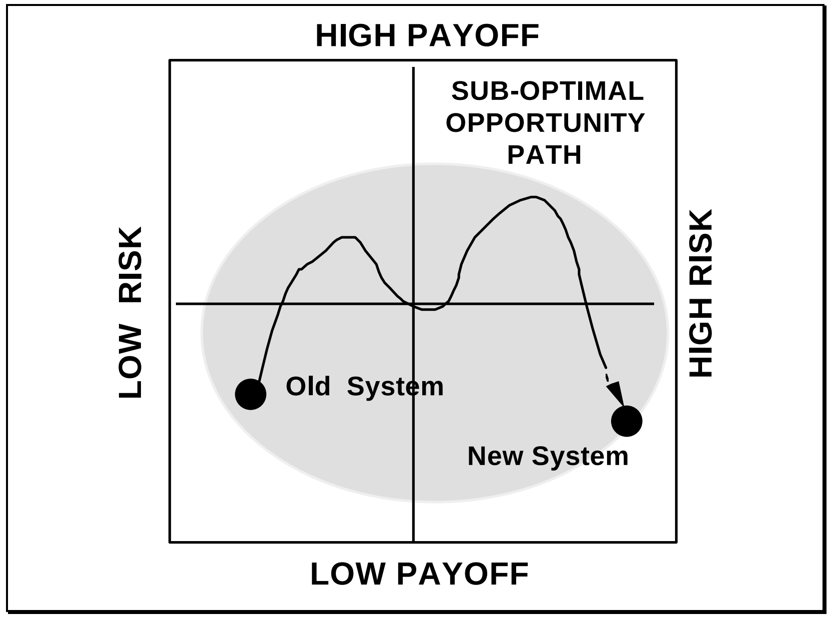


Figure 6-6. Risk: Typical Migration Path

An example of how an IT manager might set out to implement standards and end up with a “high risk, low payoff solution” opportunity may be illustrated with X.500 directory standards.

In this example, a user decides to implement X.500 in a new target system for directory management for the evolving electronic mail application based on diverse LAN environments. Since X.500 standards are not complete, the user assumes the gamble that the X.500 standard will be completed within 48 months and will be widely available in products. In fact, *the standard is fully specified and completed in the user’s hypothetical 48-month time frame but is not implemented in products* as quickly as the user requires.

In this imaginary instance, the LAN-based electronic mail users cannot find other electronic mail users on distant LANs throughout the function, because the system was implemented with a key standard architecture component missing. The result is chaos.

Degrees of freedom classification

A third way to conceptualize standards and their implementation and categorization is to describe the opportunity in terms of “degrees of freedom.” Degrees of freedom describe the degree to which, given the current architecture, you are free to adopt open-system-based standards and technology and achieve significant benefits in a new architecture in relatively short order.

If the current architecture does not allow you to implement open standards quickly, then you will be consigned to a slow migration (low payoff). On the other hand, if your current architecture permits you to implement open standards quickly, you have a high degree of freedom within your existing architecture, and you will be able to migrate to your new architecture quickly (high payoff). This concept is illustrated in Figure 6-7.

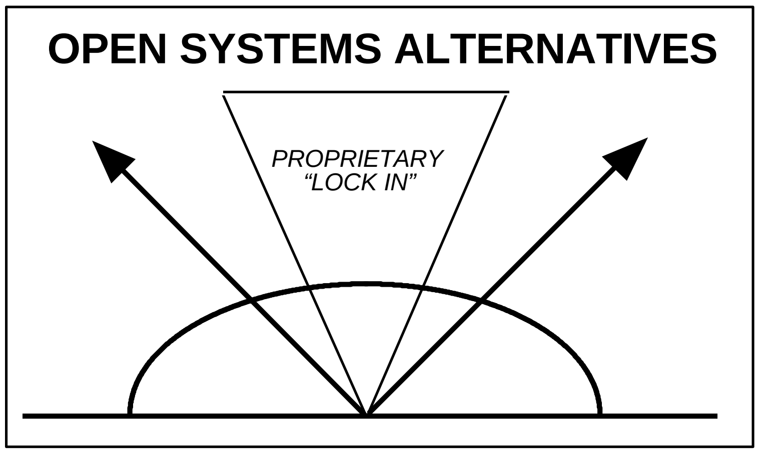


Figure 6-7. Standards: Degrees of Freedom

Overall benefit classification

Finally, an opportunity may be classified in terms of its overall benefits. These include the degree to which the opportunity provides possibilities for cost reduction and various categories of improved IT effectiveness. The following diagram describes this matrix classification.

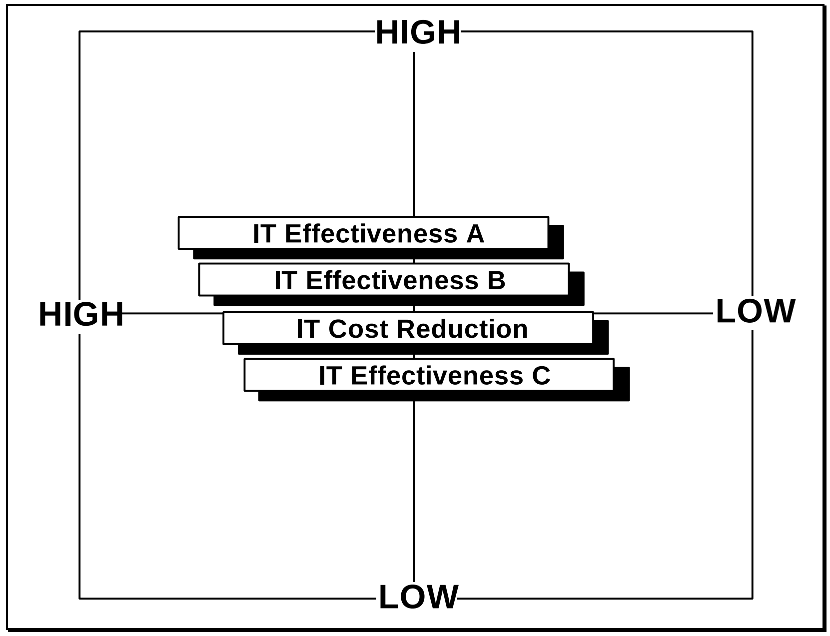


Figure 6-8. Benefit Matrix

The business case cost/benefit analysis process

Once opportunities have been categorized and classified, the business case cost/benefit analysis may be conducted.

Appendix F describes how the business case and the cost/benefit analysis could be constructed. A sample business case is provided, as well as the steps involved in building the case. These should be taken as only one way to perform this task. If the enterprise has other preferred approaches to developing cost/benefit analyses, they can be substituted.

Once the task is initiated, the AWG must review the baseline and target architecture documents developed in previous phases.

Migration planning

Upon review, the team selects a component(s) of the target architecture to consider for implementation and creates the action plans to implement that selected piece. In doing so, the work group must be careful not to lose track of the installed base of applications and technology. Few organizations can afford to scrap this investment and embrace open systems in a “flash-cut” fashion.

Instead, migration from old to new must be a gradual process. As the samples provided in Figures 6-3 and 6-4 suggest, these timeline issues must be considered as the team prioritizes its migration plans. Often, one project must be completed before another can begin. For instance, an organization may want to determine its DBMS technology before it identifies design generators or CASE tools. More examples of migration paths are included in Appendix E. While not a complete view of all types of projects that will be included in the migration options, Figure 6-9 depicts potential plateaus to migrate from an existing environment to one characterized by technology standards in the target environment.

Plateau costs

To assist in planning for the implementation of an IT architecture, it is useful to have a feel for the size of the effort in terms of staffing and costs. Unfortunately, at the architecture level, it is not possible to derive these estimates with a high degree of accuracy. It is possible, however, to apply past experiences in the form of "rules of thumb" and standard application development estimates.

Costs will crop up in a number of areas as a result of a series of projects. However, to arrive at a reasonable order of magnitude cost, we will focus on the following areas:

- The incremental computer processing and network hardware and system software needed to support the projects that will move the organization to the desired target architecture
- The application development and/or package procurement/modifications required to move to the target architecture
- The non-application initiatives needed to move to the target architecture.

Figure 6-10 is a sample summary of these cost projections by plateau and type of project as derived from the USMC SBA development project.

These ballpark estimates are intended to help strategic decision makers understand the resources required to properly evolve into the next generation of computing and reap all of the benefits that a strong IT environment brings.

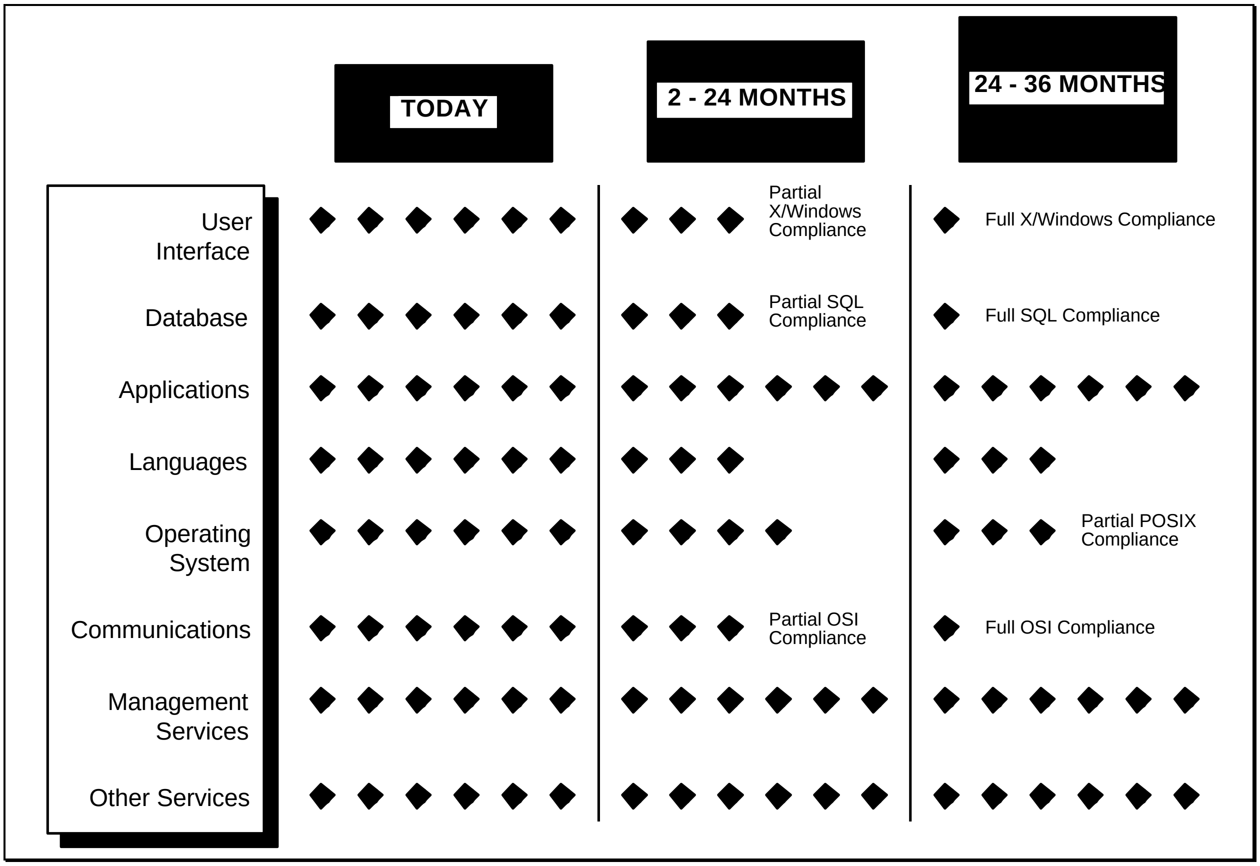


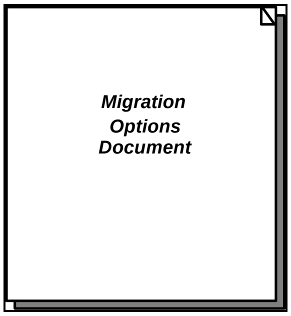
Figure 6-9. Standards Migration

Project Classification	Plateau 1 Estimated Cost	Plateau 2 Estimated Cost	Plateau 3 Estimated Cost	Total Estimated Cost
Application Development/Procurement	$9M	$7M	$9M	$25M
Non-Application Initiatives	$2M	$0M	$0M	$2M
Computing and Network Facilities	$28M	$21M	$21M	$70M
Totals	$39M	$28M	$30M	$97M

Figure 6-10. Summary Ballpark Cost Estimates by Plateau

Creating and publishing the deliverable

The key deliverable in this phase is the *Migration Options Document* along with the high-level cost estimates. The migration plan will probably consist of three separate plateaus. Because of the unique time horizons in the Federal Government, it may be desirable to link the plateaus with the 2-year POM process.



It should focus on providing the ASC with a high-level understanding of the opportunities at hand while also providing business case backup information that justifies the proposed implementation opportunities and schedules. This document should also focus on highly visible short-term “payoff” projects to demonstrate the utility of this process along the way to the target.

After finalization and approval, the document is then delivered to the rest of the organization. The options document is extremely valuable to stakeholders who must prepare for the challenges that SBA implementation brings.

Effectiveness measures

- Organization’s ability to accept and execute migration plans
- Rework required of the *Migration Options Document*
- Management’s general acceptance of the plans.

In order to achieve management acceptance, the *Migration Options Document* must describe the basic elements of the undertaking (i.e., the major program components and initiatives).

The components should be such that they are easy to read and understand by functional area managers as well as upper management. They should not dwell excessively on the technical dimensions of the architecture, elements that should be included in a detailed implementation plan. For example, if a communications project is undertaken as part of the larger project, it would be appropriate to state that all buildings would be wired with token ring or Ethernet wiring, but it would be inappropriate to go into the details surrounding wiring closet issues and the link, or the time and dates they will be installed and which project team members would accomplish the task.

Technology and tools required

- Workstation and connectivity technology
- Word processing and graphics capabilities
- Dedicated workspace with clerical support.

Staffing skills required

- Migration planning expertise
- Writing and presentation skills

- Project planning skills and experience in assigning larger efforts into implementable "chunks."

Completion criteria

- Creation of high-level plans for each component of the target architecture
- *Migration Options Document* deliverable
- Management sign-off.

One of the hallmarks of information technology is that it constantly changes. IT managers are always confronted with one of two phenomena: The technology they have installed is made obsolete very quickly, or the technology they had forecasted never materializes. For this reason, we recommend that the *Migration Options Document* contain a contingency section to address these two dilemmas.

In essence, we recommend that each major architecture project contain an assessment of the technology and standard directions possible in the future. With that forecast, we recommend that users develop alternative scenarios for implementation should the technology or standards set forecasted for project implementation never materialize. We refer to this part of the process as "innovation-proofing."

In the DoD, the other volumes of the TAFIM series—which deal explicitly with technologies, standards, styles of computing, etc.—are already in place and should evolve over time to provide a large measure of this innovation-proofing input.

No person or organization is entirely successful at predicting the future, but successful organizations will do it well most of the time by dedicating resources to technology forecasting and SBA administration.

Issues

- Consulting support needed
- Executive "buy-in"
- Workload of work team(s)
- Inventory scope management.

When plotting standards, there are other concerns to be addressed in the architecture. For instance, users may not want to "turn on" the proprietary extensions to open system products, such as relational database packages, because that single action moves them away from being open. While attractive functionality may be sacrificed, a passport to openness has been maintained. The team must keep this in mind as it consider its migration options.

- Consulting required
- Training required
- Key knowledge worker availability
- Existence and maturity of “open” technologies and standards.

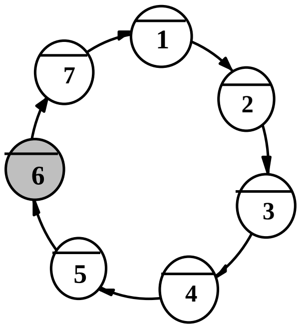
In many instances, one might find architectures based on evolving but currently incomplete standards. This requires that “workaround” strategies be developed. If the AWG regards standards on a continuum as we have recommended, this will not be as large a problem as it would appear at first inspection.

Section Seven: Implementation Planning

Contents **Page**

Figures

Section description



This section describes the overall process by which the AWG identifies and develops specific implementation plans for moving to the new target architecture. Included in this activity are descriptions of how the *Implementation Plan Document* is developed:

- Implementation project plans for Plateau 1 are developed.
- "Quick hits" for fast payoff projects are identified and pursued.
- Organizational communication mechanisms for promoting success are put in place as part of the architecture project effort and in anticipation of the SBA administration phase.

Objectives

To develop additional planning detail for the project initiatives identified as Plateau 1 of the *Migration Options Document*

- To define projects that can be completed quickly
- To create effective communication mechanisms for promoting success.

With the completion of the *Migration Options Document*, the SBA project is nearly complete. This section of the SBA Guide contains the process for developing the implementation plans for all Plateau 1 efforts.

The *Migration Options Document*, along with the other deliverables from prior phases of the SBA project, should be used to guide a detailed project scheduling process for the Plateau 1 initiatives, including specific delivery time frames and clear assignment of roles and responsibilities for each project.

At this point, the enterprise is well positioned to begin its transition towards the target IT architecture defined earlier in the SBA process. Enterprise project managers will be able to use these project plans as guides to development. The plans contain information about such issues as what is to be included in the project, the type of talent needed for the implementation team, and the infrastructure issues that may impact the success of the effort.

The plans do not define the total amount of resources required nor the project schedule because that level of detail can only be defined when each project is sanctioned by senior management. However, the details needed to get a project successfully off and running are certainly available within the plans.

As described in the previous section of this SBA Guide, the implementation projects have, by now, been organized into plateaus. Each plateau contains a set of interrelated projects in priority order. The plans that follow are for those to be tackled in Plateau 1.

Also included in this document are the project plans for a set of quick hits that the enterprise should strongly consider completing within the first year of its SBA implementation effort. The quick hit projects offer a good deal of benefit in a relatively short delivery time as well as providing a foundation for other Plateau 1 projects.

Individual project initiatives should then be kicked off with a preliminary analysis phase. In the initial design phase, more detailed deliverables will be developed showing a refined view of the information and system functionality through conceptual models and supporting documentation. Also, a refined cost and benefit estimate should be made at this time for each project allowing a “go/no-go” decision to be made on a project-by-project basis, considering all of the interrelationships defined in the architecture deliverables.

This phase is based upon the very simple notion that if an architecture does not begin to deliver concrete benefits in under 12 months, it has a low probability of being implemented overall. As a test of its real world viability in today’s world of results-oriented management and reward, a program must be able to deliver a concrete payoff project to ensure that a manager’s year-end personal objectives are met or the program will not be implemented. For this reason, it is key that short-term payoffs are identified and implemented early on in the architecture process. Once these “small wins” have been put into place, this phase focuses on broadening awareness throughout the organization to induce “culture change.” Mid-term benefits are then harvested and a benefits measurement program is put in place for the duration of the program.

Scope

To define the plans necessary for migration, with an emphasis on quick hits, while the longer-term strategic standards-based architecture is developed and implemented. The document has a short-term payoff orientation.

It is recommended that, if the AWG wants to deliver a detailed technical implementation plan, technical and operational professionals be introduced as key players during this phase.

A natural question arises from this approach: To what degree should the AWG be involved in detailed project management? The answer depends upon the size and scope of the implementation project. It is recommended that the "Level 1" high-level project plan be developed by the AWG, and that more detailed project implementation plans be managed within the operational or business units in which they logically reside. Progress updates may then be delivered to the AWG and ASC. Figure 7-1 illustrates this relationship.

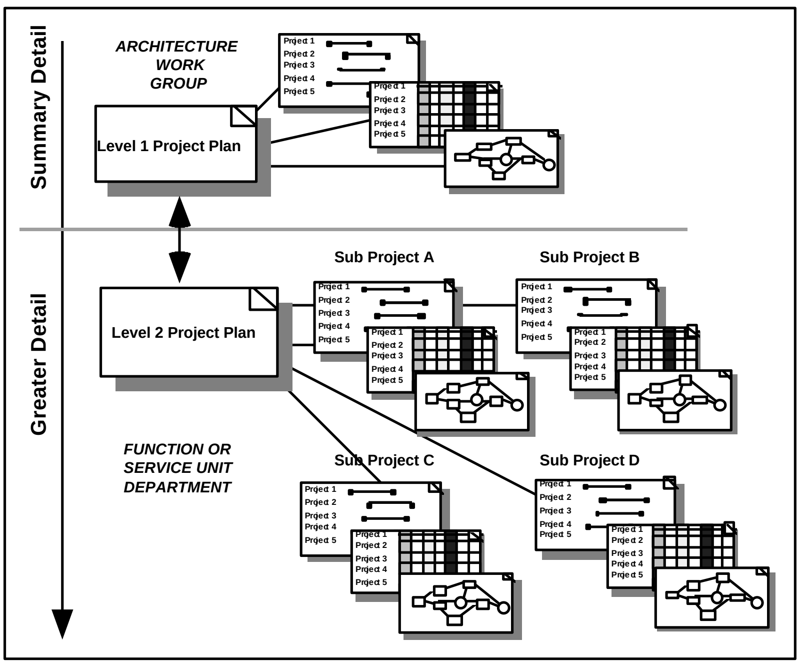


Figure 7-1. Levels of Implementation Planning

Most organizations have key technical leaders on the AWG, and detailed implementation plans are most successfully developed within the discrete operational units in which they are being implemented.

Some AWGs may elect to split implementation planning into two levels of activity: a high-level architecture implementation plan and a secondary technical implementation plan.

Deliverables

Implementation project plan documents that contain the detailed road map for migration to the *Implementation Plan Document*. A sample outline for this document is included in Appendix I.

During the migration options phase, a series of migration steps were outlined. In this phase, the team characterizes the size and scope of implementation plans and the timing of the projects, as well as developing alternative contingency plans.

Critical success factors

- Project management and estimating skills
- Detailed planning talent on the team
- Team that is comfortable in working with a short-term focus.

Standard implementation planning techniques that should be used during this phase have not been discussed. It is assumed that the reader will be familiar with these techniques in the same manner in which he/she understands other processes such as data modeling (which is likewise outside the scope of this document, although examples are provided).

Throughout this document, the focus has been on the need to identify opportunities that provide concrete payoffs in implementation. If an architecture does not provide initial payoffs, there is a high probability that the entire architecture will never "see the light of day." The following needs have been described:

- A short-term focus combined with a "fast path process"
- An architecture and attendant implementation based on discontinuous, chaotic business realities of today's "fast cycle" organization

- Implementation projects that provide project-oriented deliverable payoffs rather than “grand strategic” payoffs some time in the distant future
- An ongoing process that defines architecture and standards with room for entrepreneurial improvisation and implementation.

SBA implementation must possess all of these qualities. For this reason, it is recommended that, in addition to standard project planning techniques, the AWG focus on several other aspects of implementation to ensure successful implementation.

“Quick hits”: Implementation of short-term payoffs

There is more than a grain of truth in the saying *“in the long run, we’re all dead.”* Nowhere is this more true than in implementation planning. In today’s typical organizational culture, short-term (3 to 6 months) payoffs are required as a condition of employment and advancement. If the entire implementation program is to be a success, it must contain a minimum of one major implementation activity that is an integral part of the SBA plan and may be capable of being implemented in a short time frame. It must be of sufficient significance that its implementation will assure the AWG members (or their management) of attaining their annual program goals and objectives.

When implementation activities are linked to the enterprise’s reward system, things get done and heretofore non-cooperative organizational task force members begin to make things happen.

The other central objective of providing a short-term payoff is that the successful implementation may then be used as a pilot case example for the rest of the organization of how a standards-based architecture can provide immediate benefits, and that truly major benefits will accrue to the program if it is followed over time.

Communication: Organizational awareness programs

Upon identification and implementation of a major short-term payoff opportunity, the AWG should spend a significant amount of time conducting a “public relations advertising program.” Figure 7-2 illustrates the recommended process that AWGs should follow to ensure that the organization is behind the implementation effort throughout the SBA life cycle.

It is recommended that workshops or presentations be continuously conducted throughout the organization after the AWG has a solid implementation success on its hands. People or processes that actually “get something done” are rare in most organizations. If projects are successful, there is a great likelihood that the architecture planning documents will be read and implemented throughout the organization.

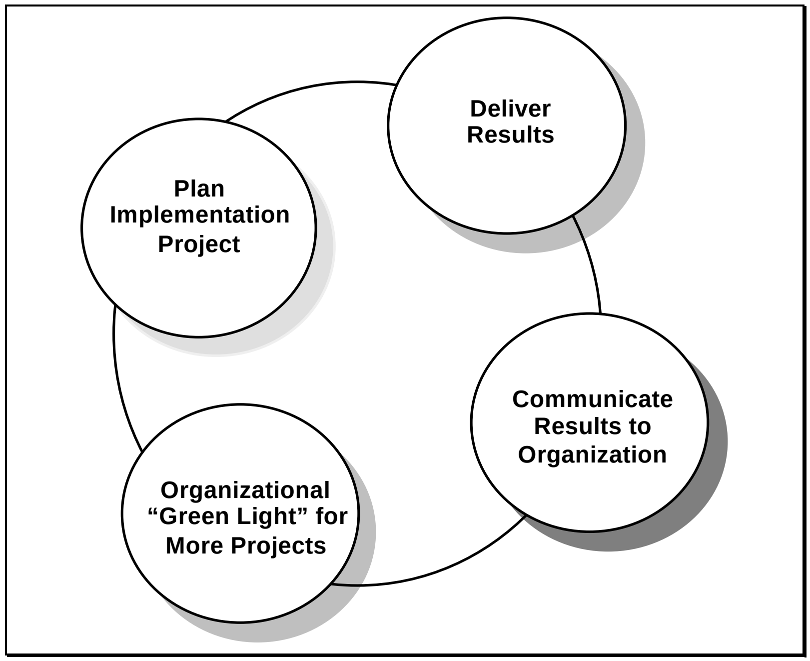


Figure 7-2. The “Results Communication” Cycle

Architecture plan modifications

The AWG’s designated implementation team will make ongoing modifications to the overall process as it progresses in implementation over time. There are times when individual implementation projects blow up or need to be terminated. Sometimes these projects are outright failures due to poor management or resource constraints and the like. At such times, it is sometimes convenient for management to conclude that the “architecture is fundamentally flawed.” Thus, important projects are sometimes eliminated because of subproject deliverable failures. The overall architecture becomes, as it were, the fall guy for a poorly implemented project.

It is recommended that such failures be carefully evaluated in the context of the overall architecture project implementation cycle before changes are made to the overall architecture. In nine cases out of ten, implementation strategies and tactics will require adjustment, rather than the overall architecture. However, sometimes failed projects do show opportunities to improve the overall architecture.

Because the architecture is developed on a group consensus basis, making significant changes requires ASC sign-off. In theory, one aspect that will not change is the architecture principles. *These provide the "constitutional" backdrop to the overall standards-based architecture.* If the organization does discover that some principles must be changed, then the equivalent of a "constitutional amendment" process must be developed by the AWG and approved by the ASC. Figure 7-3 illustrates this process.

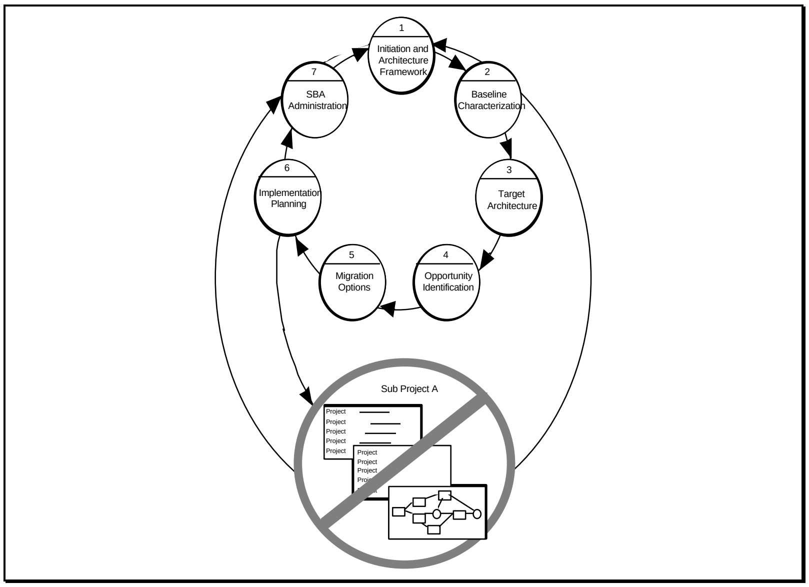


Figure 7-3 Project Impact on the Architecture

Constraints

An inexperienced implementation planning background will limit the team's ability to develop effective plans.

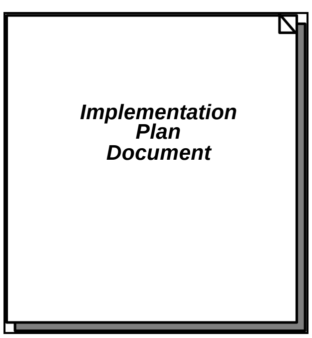
The degree to which highly granular implementation plans are developed will depend upon the skill set and experience of the

AWG. If the AWG is more highly skilled at planning versus implementation, it might be logical to identify business or service unit department-level personnel to actually carry out the detailed implementation planning discussed in this phase.

Task list

- Initiate task
- Assign team to build detailed implementation plans for Plateau 1 projects
- Develop cost/benefit case by project
- Produce implementation plans by project
- Develop security implementation plans by project as necessary
- Identify standards implementation strategy by project
- Identify key interrelationships and dependencies among projects
- Establish timeline for each project
- Draft *Implementation Plan Document*
- Conduct review with ASC
- Finalize *Implementation Plan Document*
- Distribute *Implementation Plan Document.*

Creating and publishing the deliverable



The key deliverable out of this phase is the *Implementation Plan Document.* It should focus on providing the ASC with a detailed understanding of the projects being developed as well as all traditional project management reporting techniques. It should include:

- Major project descriptions
- Milestones and project interrelationships
- Resource requirement definitions
- Project deliverable definitions
- Key responsibilities and accountabilities by project and program.

Effectiveness measures

This phase will vary widely in terms of calendar time required for completion based on project size, scope, organizational culture, individual schedules, and the resources required to perform the project. We recommend that the implementation project teams constantly remind management of the need for a "fast path" implementation to ensure rapid deployment of project implementation efforts. Effectiveness measures include:

- The ability of the plan to show continuous improvement and results
- The degree to which implementation plans can be developed
- Management enthusiasm regarding opportunities identified
- Timeliness of project implementation.

The ASC should be kept informed of all status activities as mentioned previously in this section. It is this group that will keep pressure on their management groups to ensure that projects are implemented successfully.

Tools required

- Word processing and graphic presentation packages
- Project planning software tools
- Spreadsheet tools and/or user-friendly personal computer-based database packages for inventory logging.

The key deliverables out of this phase are the individual implementation plans themselves. Therefore, project planning tools, as well as those described above, will be required for the task at hand.

Staffing skills required

- Migration planning skills
- Project management skills
- Writing and presentation skills
- Familiarity with word processing, presentation, spreadsheet, and database packages that run on most popular personal computers.

This phase requires individuals who are well-seasoned individuals in the art and science of migration planning and project management. If the AWG does not have members with these traditional skills, the team may be augmented on a temporary basis with personnel outside the team.

Completion criteria

- The development of all short-term and mid-term project plans
- Management review and acceptance.

Successful, on-time implementation of projects identified during the implementation planning effort is the sole measure of how well the completion criteria have been met.

In addition, the degree to which middle- and long-term opportunity projects are pursued is key to the successful implementation of the overall architecture. Frequently, such initiatives get dropped before "the war is won." With the focus on short-term payoffs, it is critical that the ASC not abandon its efforts after early "successes."

Issues

- Project management skill capabilities
- Workload of work team(s)
- Business case criteria acceptability
- Consulting required
- Training required
- Subject matter expert availability.

Resource constraints may make project implementation a challenge for both the ASC and the AWG. It is very important that all of the issues outlined on the list above be addressed in reviewing all implementation plans.

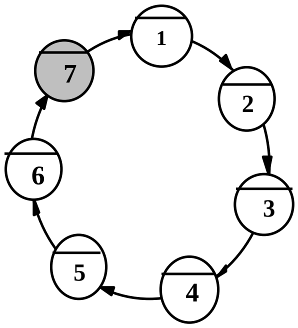
This page intentionally left blank.

Section Eight: SBA Administration

Contents **Page**

Figures

Section description



In the SBA administration phase, the process by which the organization maintains its new IT architecture is identified. The SBA administration process defines the procedures, human resources, and communication devices needed to keep the plan current with the organization's mission and priorities. Because this process is an integral part of the IT planning effort, it is essential that personnel be dedicated full-time to architecture administration.

This section describes the overall process by which the AWG monitors and checks the success of the new target architecture. This is a key activity, as the team seeks to continuously improve the development and implementation of the IT architecture. Included in this activity are descriptions of the need for an ongoing architecture administration process and of how an *SBA Assessment Document* is developed:

- An SBA management team (SBAMT) is recommended to maintain the SBA.
- An SBA development project review process is developed.
- An ongoing process is developed for the measurement and monitoring of project problems and architecture compliance.
- An ongoing process is developed for keeping the SBA document alive.

The output of ongoing architecture reviews is a self-critical document that is used to modify the architecture documents produced in prior phases to "keep them alive." As such, this phase is the last in a continuous cyclical improvement process. As Figure 8-1 suggests, it provides the organization with a way to learn from past mistakes and make adjustments to future plans to ensure its ongoing success.

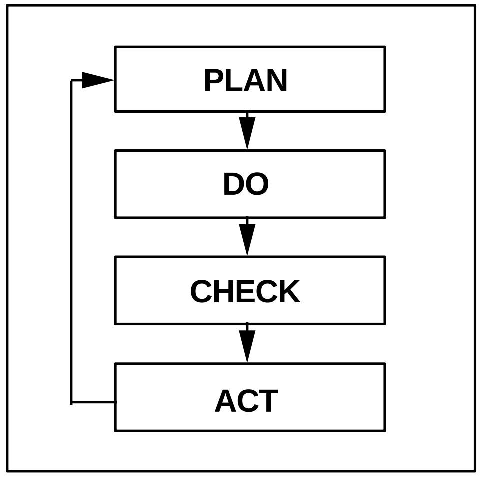


Figure 8-1. The Continuous Process Improvement Cycle

Objectives

- To create a measurement process for the remainder of the SBA implementation
- To review the results of project implementation
- To modify current plans based on actual experience
- To integrate the SBA process into the mainstream planning and management activity.

Now, more than ever, it is important that IT professionals have plans that work when implemented. The plain fact is that some plans *do not work.* A number of contributing factors result in the half-implementation or failure of architecture plans. The most common one is that management changes direction, and the attendant technology priorities change as well. Other times, plans are not implemented because of flaws in the planning process itself—some of which have been discussed in this SBA Guide. Architectures are frequently not implemented because either the recommended technology does not deliver the solution or the technology *"never shows up"* (also known as technology lag). In the latter case, a vendor's technology promises never materialize in the marketplace. This happens with both technology and standards themselves.

The final phase in the SBA planning process is to “reality-check” the architecture to ensure that the original design criteria are bearing results. The best way to accomplish this on an ongoing basis is to fully integrate the SBA planning process into the mainstream management practices within the enterprise.

Scope

- All projects defined in the implementation plan are within scope of the evaluation.

This step is executed once the implementation plans for Plateau 1 projects have been approved. This is an optimal time for effecting the transition of the SBA process from the “experimental” arena into the mainstream management function. By establishing a credible position within the management function, the projects coming out of the SBA process will have greater likelihood of funding and implementation. It may be a significant challenge to become a full-fledged component of the general business planning process, but anything short of this status is associated with risks to the projects and to the SBA process itself. The ability of the SBA process to achieve such status will, in many cases, reflect the success of the initiation phase that launched the SBA in the first place.

After a reasonable period has elapsed in the implementation process (or, alternatively, as a direct follow-on to the delivery of approved plans), the AWG should conduct a brief review of the projects defined in the *Implementation Plan Document* to ensure that those projects’ objectives are being met and that payoffs are being obtained through the implementation process (see Figure 8-2.). We refer to this as a “process check” and, as such, it will provide a *quality assurance* dimension to the overall planning process.

This process check of the architecture should occur on a cyclical basis throughout the IT planning process. This check should focus on the deliverables of the architecture, as well as on the architecture process itself, and be modified accordingly.

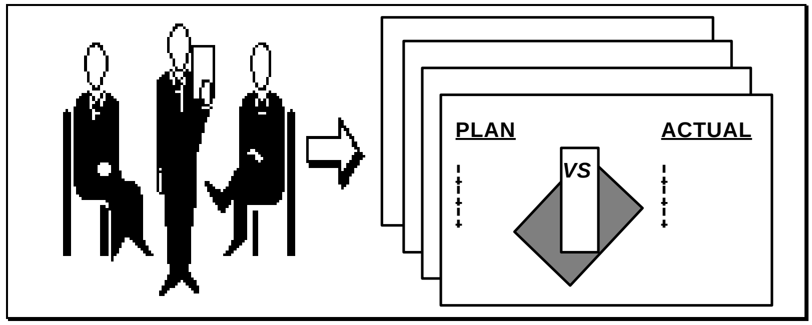


Figure 8-2. The Team Reviews Each SBA Project Plan

Deliverables

- Establishment of the SBAMT and recommended processes to keep the SBA "alive"

SBA Assessment Document (at a later date in the implementation cycle, but after the SBA development project concludes)

The architecture is subject to regular assessment and an *SBA Assessment Document* is produced at each review. The *SBA Assessment Document* may be developed by the SBAMT. The process of SBA implementation, however, is ongoing and subject to the organization's commitment to continuous process improvement. (Quarterly reviews are recommended in the first year, semi-annual reviews thereafter.) A sample outline for this document is included in Appendix I.

Critical success factors

- The organization must be willing to sponsor the SBA process as an ongoing management activity.
- A team review of the process that solicits organizational buy-in must be used.
- Time must be dedicated to this effort.
- Key knowledge workers must participate as required.
- Results must be communicated.

Modifications to existing plans must be made.

It is essential that the organization establish a review process and dedicate resources to the effort. Perhaps the greatest reason for implementation failure is the simple, but often overlooked, requirement to obtain organizational buy-in and make the architecture implementation process a team-based effort.

Much of this activity is organizational and political. In the end, politics is the art of inclusion. Any success enjoyed early on in the initiation phase will contribute to continuity of the process now that the process deliverables have all been approved. The real challenge starts now as implementation plans are to be put in place. If initiation was not successful, there remains much to do in positioning the SBA process within the organization. Any organization pursuing standards on a managerial dictatorship model will run a much higher probability of failed implementations than the more team-oriented process that has been outlined throughout this SBA Guide.

In the area of standards-based architecture, it is paramount that the AWG build into the overall process a review system to ensure compliance with the objectives set out by the *Architecture Framework Document, Baseline Characterization Document, Target Architecture Document, Opportunity Identification Document, Migration Options Document,* and *Implementation Plan Document.*

Constraints

- Fear of being labeled a failure can undermine this effort.
- Other priorities can also limit the effort that participants can dedicate to this project.

As Figure 8-3 suggests, the key to success in establishing an assessment process is to have the plans owned by as wide a team as possible across the enterprise (rather than a set of individuals with "agendas" ready to assign blame for failure).

Perhaps the largest constraint is management's unwillingness to dedicate the resources needed to keep the SBA in the forefront of activities in the systems development and/or work redesign arena. The ASC must be ready to address this issue in order to create the team-oriented environment necessary to make SBA a success.

In this new kind of environment, assessment and review become less personally and politically charged. The result is that the assessment process becomes easier to successfully conduct. This form of organizational behavior also encourages successful implementation in the first place.

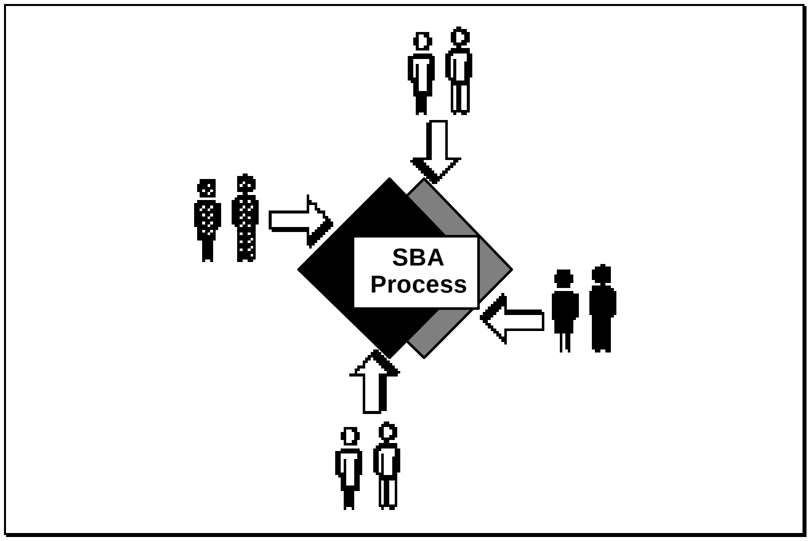


Figure 8-3. The Entire Organization Should Be Included in the SBA Process

Task list

- Launch the implementation plan, staff the SBAMT, and establish the ongoing process to be followed.

The following tasks can only be done after some progress has been made on the project initiatives defined in the SBA implementation plans (i.e., after the SBA development "project" has concluded).

- The SBAMT maps results against the *Architecture Framework Document*, the *Implementation Plan Document*, and their measurement criteria.
- Current project and future plans are reviewed.
- Appropriate modifications are made and distributed to the review committee and the appropriate project managers.
- "Lessons Learned" are developed and included in the *SBA Assessment Document* for distribution.

The first step in the assessment process is to establish an SBAMT. This team should be staffed with experienced planners and technologists who have a deep-rooted understanding of the implementation projects.

Once established, the team must conduct a general assessment of the projects to see if, in fact, the projects are being implemented. This is done by mapping the results against the implementation plans and asking some hard questions, such as:

- Is the architecture framework still valid? Should any of the architecture principles be modified? Which ones and why? What has changed?
- What were the benefits of the identified projects? Cost savings, value-added benefits, or softer long-term intangible benefits?
- Have adopted standards been materially implemented in the organization? How far along has the standards road been traveled? How far, given this process check, do we have yet to go? Have we gleaned 80 percent of the benefit already or is there still significant payoff down the road?
- Does the organization recognize the payoff that has been achieved?
- Given the current state of implementation, have any other payoffs been obtained that may not have been originally predicted (the *Opportunity Identification Document* should be reviewed in this context)?
- In general, do the plan's standards appear to be changing?
- Have any standards, targeted as important, not yet matured as much as was originally anticipated by this point in time?
- What is the status of the technology that was selected for implementation? Has it "shown up on time" in the marketplace?

After these questions have been answered, adjustments to the original plans should be made (i.e., if implementation is not working for tactical reasons, specific steps will have to be developed to produce "workarounds") and reviewed with the ASC.

After review of the plans, the team should step back from the assessment and begin to analyze the exact cause of the shifts of emphasis. These "lessons learned," together with the modified plans, become the *SBA Assessment Document* (see Figure 8-4.).

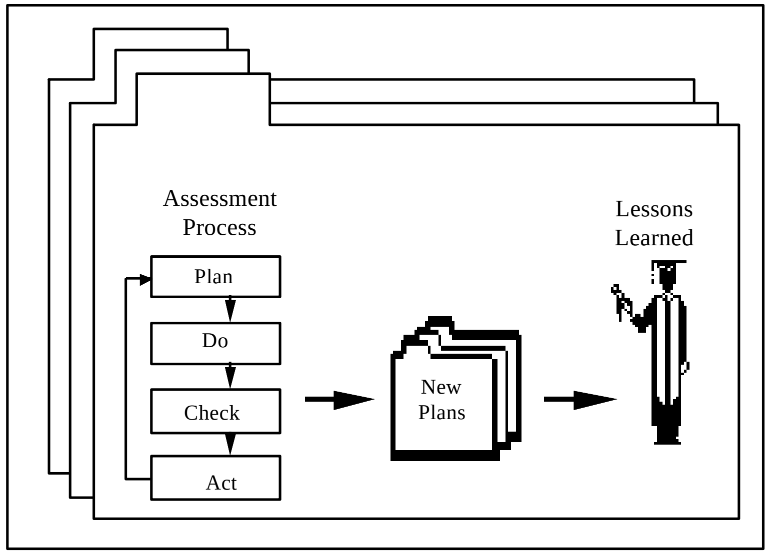


Figure 8-4. The *SBA Assessment Document* Includes New Plans, Revisions to Old Plans, and Lessons Learned

Often overlooked, documenting the lessons learned becomes very valuable to the review team when defining the modifications to future plans, and it helps future implementation teams to "not make the same mistake twice."

Effectiveness measures

- Organizational buy-in to the process measured by active and enthusiastic involvement
- Implemented architecture attributes are measured and assessed
- Ease of plan modification
- Communication mechanism.

The assessment effort can be judged by the degree to which the assessment team can examine SBA results to date and determine the appropriate actions to take to keep the architecture process on track. Ultimately, the effectiveness of a given assessment can only be measured by the success of future implementations.

Technology and tools required

- Workstation and connectivity technology
- Word processing and graphics capabilities
- Dedicated workspace with clerical support.

Staffing skills required

The assessment team is typically composed of members from the original SBA AWG and a few implementation project managers. While it is true that this team meets only a few times a year, management must be prepared to reassign workloads and the like because sometimes the effort needed to complete the assessment can be quite extensive. The staffing skills required include:

- General knowledge of SBA
- Planning skills
- Subject matter experts (as needed).

Occasionally, the assessment team will need to rework existing plans. They will need to call upon key knowledge workers who have working knowledge of specific projects or technologies. Management must be willing to commit what it takes to keep the SBA process alive and on track.

Completion criteria

While architecture assessment is an ongoing effort, a particular review cycle can be considered complete when:

- Each SBA project status and plan has been reviewed and compared against the architecture principles and target architecture plans.
- Lessons learned have been documented.
- The completed assessment document has been reviewed and approved by the ASC.

Ultimately, it is the ASC's decision as to when a given assessment effort is complete (i.e., the committee is responsible for the success of the SBA effort as a whole).

Issues

- Training needed
- Consulting needed
- Remodeling the core architecture may become necessary
- Time must be spent changing the culture such that reviews are seen as a process improvement vehicle and not as an exercise in "pointing the finger."

Besides the training and consulting support needed for proper architecture assessment, there are two major issues that can impact this phase: The extent of *architecture remodeling* needed and the management of cultural *change*.

Architecture remodeling

When should you remodel? When one of the architecture principles has changed. Another reason for remodeling could be that a major change in technology took place that was so significant that your architecture plans did not anticipate it; however, this will become increasingly rare. One of the major benefits of standards planning is that standards, unlike the underlying technology itself, change far less frequently.

In theory, you should never have to change your architecture framework if the architecture principles never change; however, they do change from time to time. When this happens, the review team should discuss and confirm the perceived changes with the ASC.

If necessary, the committee can sanction a task force to do necessary rework of the affected SBA documents. However, this step is usually not required, because the assessment team is more than likely composed of the same personnel who developed the original plans.

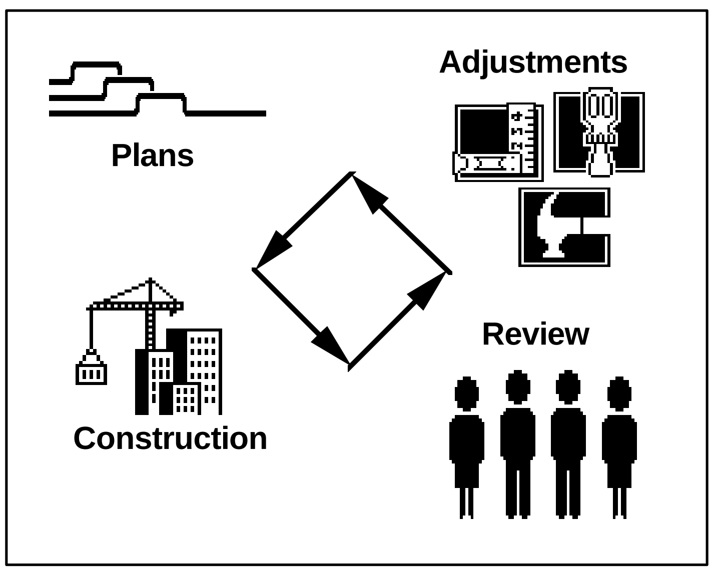


Figure 8-5. The Organization Must Gain a Working Understanding of SBA and Learn to Appreciate Its Value

Cultural change

As a final note, the review process described in this section should not be taken lightly. It is central to building and maintaining a solid SBA. Because of its importance, the DoD community should dedicate resources to the promotion of, and education in, standards-based architecture.

The goals of the promotion and training program should be to expose the entire organization to the change process and familiarize personnel with the benefits inherent to open systems. In so doing, the “gut-level” values of the organization will change and SBA management will become everyone’s business.

Appendix H contains a more detailed example of the kinds of processes which may be recommended for SBA administration at the end of the SBA development project.

Appendix A: How To Do Architecture Principles

Foundation of a standards-based architecture

Architecture principles are statements of preferred architecture direction or practice. They are simple, direct statements of how an organization wants to use information technology in the long term for 5 to 10 years. They establish a context for architecture design decisions across an organization and help translate business criteria into a language that technology managers can understand. Each principle is accompanied by a statement of the rationale for the principle and a statement of the principle's implications.

Many organizations skip the principles definition process and jump right to modeling their architectures and setting standards. The result has often been a technical myopia–organization focus on technology selection issues and never deals with how they are going to manage the technology until a selection of an unpopular vendor or technology raises the issue to a head.

The "IT constitution"

Principles allow for diverse business, operational, and technology personnel in the enterprise or work group to develop a common language and shared understanding of the challenges facing the organization. Architecture principles become the "constitution" by which the overall architecture is designed and implemented. In theory, principles change unless, like the U.S. Constitution, they are amended through a formal amendment process. This process was described in the previous sections.

Architecture principles are the foundation of a standards-based architecture and are necessary to achieve the degree of organizational consensus and understanding required to move ahead with an integrated, standards-based architecture. Experience with architecture principles has shown that a more open, standards-based environment is often the result of a principles definition process. Principles also provide organizations with a stable base from which to make decisions. Principles change as the organization's mission or business changes—often relatively slowly. They provide a framework against which to test later decisions and guide subsequent procurement and implementation decisions.

For some time now, many leading practitioners and academics have been arguing for a generic approach to principles. Principles can be especially powerful in helping an organization move to a new technology architecture; for example, the benefits achievable through a network computing environment enable the adoption of new classes of principles. Additionally, an appreciation of the case for standards-based architectures enables the “driving down” of principles to standards and guidelines, which can enable the actual implementation of systems. Consequently, the reader will note that the following discussion of principles has a unique thrust.

Establishing a coherent set of architecture principles is therefore critical to forging a standards-based architecture. Principles force enterprises away from individual discussions of vendor products to focus on the desired behavior of the architecture. Principles provide a vehicle for key stockholders to discuss and agree upon how they will organize and implement information technology.

A principle may deal with any aspect of architecture; for example, a principle that deals with information architecture may be:

“Business terms and associated data element definitions should be defined consistently and be readily available to users throughout the organization.”

A technology principle might be:

“All computing and communicating devices should interconnect through a common networking environment that is based on industry standards. It should support interconnection among internal units and with users, suppliers, and other business partners.”

The definition of principles can be influenced by a number of factors: current policies, business drivers, strategic business decisions, IT trends, existing architectures, and organizational practices. We have found that principles generally fall into five categories:

- Principles that affect all aspects of IT (meat-principles)
- Work organization
- Information
- Applications
- Technology.

Although principles are the foundation of an architecture, they are not a complete architecture as illustrated below. A thorough analysis of how technology will be deployed and what viable vendor products and industry standards are available must be performed before technology can be procured or systems can be delivered.

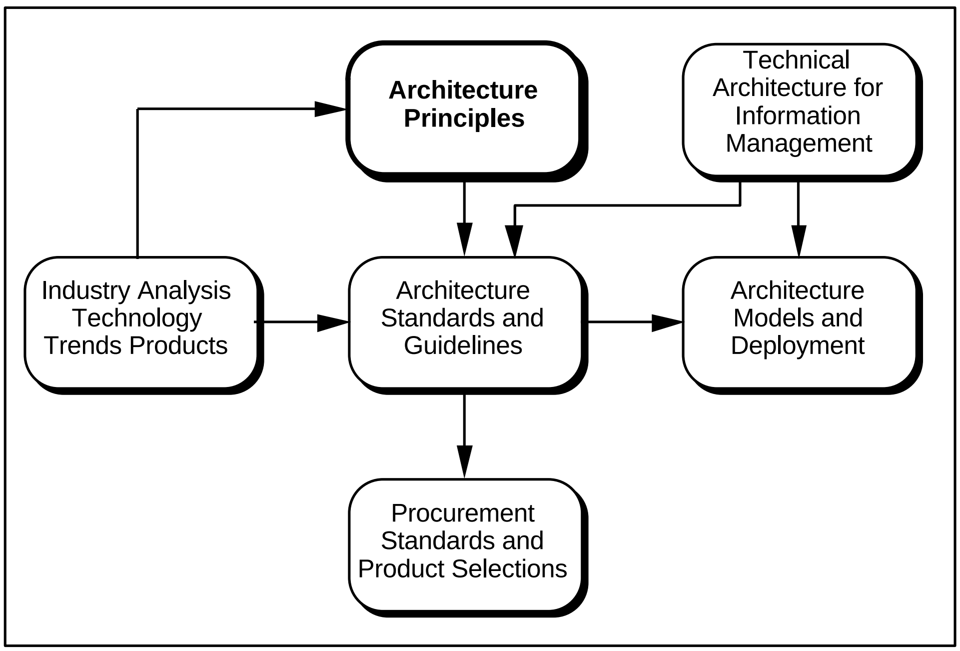


Figure A-1. Relationship of Architecture Principles to Standards

The remainder of this appendix discusses how principles begin to define a style of computing, how principles are defined, and provides a generic list of principles that can be used as the basis for defining a customized set of principles for an organization. It is important that the organization develop its own principles and not simply duplicate those listed here, since the value of the exercise is the group consensus and discussion around these key issues.

Styles of computing

Principles are analogous to zoning laws. Zoning laws establish a set of rules for the usage of land (setbacks, building size, etc.) and for the type of building that will be put on the property. Like zoning laws, principles tend to change relatively infrequently. Likewise, architecture principles set rules for how IT will be used, guide implementation of systems, and begin to define a “style of computing” that an organization will undertake.

An organization’s computing style has a number of dimensions:

- **Dispersion–**To what degree will control over IT be dispersed to business units and departments within the organization? How much autonomy do business units have about decisions on applications, data, and technology?
- **Distribution of applications and data–**Will applications and/or data be centralized or will they be placed close to the user?
- **Decentralization of technology–**Will the technology environment be mainframe-based? Will it be highly decentralized and integrated around a network? What is the role of intelligent workstations?
- **Proprietary or open.** Will the architecture be based on a vendor’s product approach (e.g., AS)? Will it be based on industry standards? To what degree?

The principles should articulate the organization’s view on each of the dimensions. If successfully articulated, the principles can simplify many subsequent modeling and standards decisions.

Principles and their relationship to open systems

Principles often promote a shift to a standards-based architecture. First, when organizations go through the principles definition process, they begin to articulate the valuable characteristics of their desired architecture. Characteristics such as reusability, common components, interchangeable parts, and increased modularity of the architecture are often stated in principles.

When discussing the implications of principles, many organizations begin to see open systems and industry standards as at least partial solutions. Articulating the architecture principles provides a way to discuss "openness" as a desired attribute without getting into a battle between the proprietary and open camps that exist in many organizations.

The process for creating principles

Creating principles is inherently a dynamic, consensus building process. One senior IT executive characterized it as "social engineering" by providing a forum for a diverse group of IT and business unit managers to gain consensus regarding what is to be done and how it will be done. Most organizations find that they can adequately articulate their architecture direction in thirty to forty well-thought-out principles.

Creating principles is a five-step process to be conducted within the first phase of the SBA planning process, architecture framework:

1. ***Establish a principles task force within the ASC***

The first step is to create a task force within the architecture framework phase that includes a mix of both IT and business unit personnel that represents the organization as a whole. This group functions as a subcommittee of the overall ASC. Development, operations, data management, and planning functions from the IT community should be represented. Business and operational unit representatives should be chosen who can speak for operational units. If there are tactical considerations, such as boundary interface definitions and the like, they should be an integral part of the unit as well.

It is important to have decentralized (dispersed) IT and business units represented as well as operational and tactical constituencies. While this may result in a large task force, the value of getting broad buy-in to the result is critical. A good task force size in a large organization is about ten people; however, task forces as large as *sixty* people have successfully defined principles. The process must be kept moving. If it bogs down, the commitment of task force members will disappear.

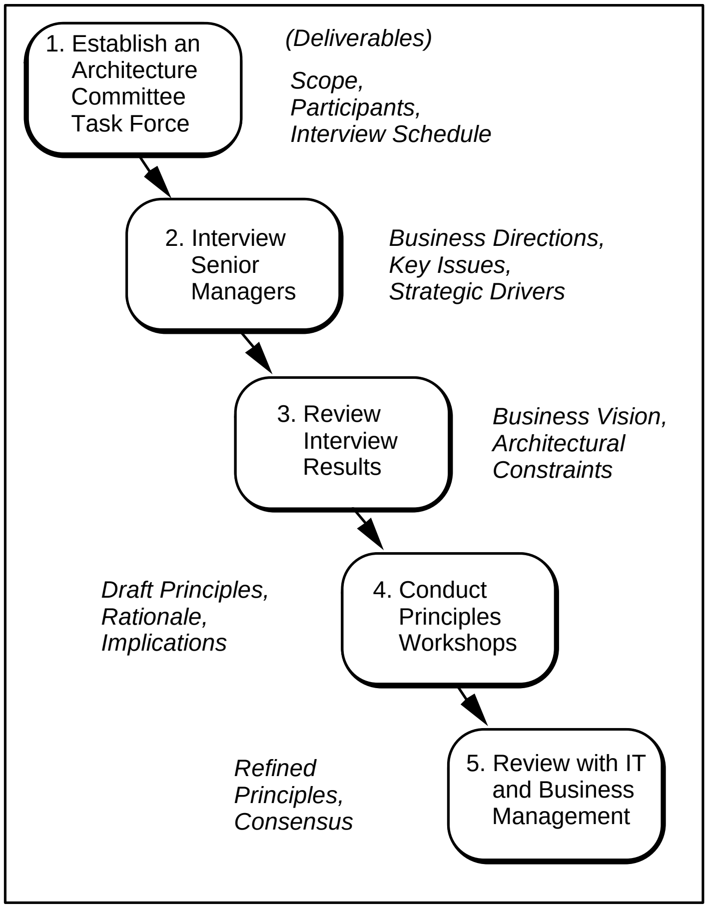


Figure A-2. Architecture Principles Process and Deliverables

Once the task force participants are defined, the next step is to hold a workshop to introduce examples of architecture principles, the process that the task force will be going through, and how the task force will be organized. If the task force is large, it may be broken down into different topic areas such as the ones identified above (overall principles, IT organization, information management, application management, and technology management). The examples of principles discussed in this appendix can be used as "straw man" examples.

Next, the task force needs to identify the senior business managers to be interviewed. These are managers who can discuss the key business initiatives of the organization and the major directions that the organization will be taking in the next few years.

2. *Interview senior IT and business and operations managers*

Interviews are conducted with the business or operational unit managers. The objective is to understand the key business issues, directions, and constraints that the organization is dealing with and the organization's view of IT. For example, what is their view of the role of IT? Strategic or purely tactical support? How much risk are they willing to take with IT? Do they view IT as providing value, or as an additional cost of operating? How much control and autonomy do they want to exercise over IT decisions? What kind of time frames are they planning within (one year, five years, longer)?

3. *Review interview results with ASC*

The next step is to use the input from the interviews to identify the overall role of IT and to define the business and organizational constraints on IT. Information on the exist-ing IT environment is valuable here, as it may constrain the principles or make some principles unrealistic.

4. *Conduct principles workshops*

Once the task force understands the constraints and plans, it can begin to work on the principles themselves. The topic areas discussed throughout the rest of this appendix are a good starting point, but the principles need to be stated in the organization's own words, discussed, and agreed upon by the participants. Some characteristics of good principles are:

Principle Characteristics

1. They clearly state a fundamental belief of the organization.
2. No motherhood! Each principle should have a counterargument; for example, "information is an asset" is not a good principle, because it is hard to disagree with it.
3. They should be simply stated and understandable to both business and IT managers.
4. They need to have rationale. Why did this principle get stated this way? What alternatives were discussed?
5. The implications need to be discussed and documented; for example, what impact does this principle have on the IT organization? On management processes? On technology?
6. They conform to Federal mandates.

Figure A-3. Characteristics of Good Principles

It is also important to keep principles at the correct level. Too often organizations get into too much detail and actually end up defining standards and technology choices. That comes later, when input on the installed base and target architecture is available.

The following is an example of a principle, its rationale, and implications:

Principle

Our systems should utilize standard, shareable, reusable components across the enterprise.

Rationale

It is critical that the IT organization improve its response time to business needs and delivery systems faster and with better quality. Our organization is going through substantial change and IT must be better able to build flexibility into its systems and allow them to adapt to changing business requirements.

Using standard components as the basis for defining and building the architecture and delivered systems can improve our productivity by using previously defined and built components. Rather than build new components each time, developers can concentrate on new business requirements, rather than redoing existing work. We believe that the ability of our systems to adapt to changing requirements can be improved by using standard components.

Implications

There are a number of management and organizational implications from this principle:

- A means of coordinating, defining, and communicating the available standard components will need to be developed.
- Areas where definitions of standard components will be required include business processes, applications (at all levels), and technology components (processors, system software, network components, languages and development tools, and data, such as subject databases, conceptual designs, physical implementations, etc.).

A management process will be required to track the generation and usage of these shareable components and to standardize them where needed.

- A standard definition of each component type will also need to be defined. This could be facilitated through a well-implemented common system delivery methodology.
- A library of definitions, terms, access rules, characteristics, and interrelationships of each of the application, information, technology and, potentially, organizational and business components needs to be implemented corporate wide.

Figure A-4. Sample Principle

Meat-principles

Meat-principles are principles that apply to the IT environment as a whole. They address the organization's position on architecture, migration, and risk management, as well as its orientation to open or proprietary systems.

Architecture focus and compliance

Organizations have different views regarding how much they are willing to spend for an architecturally compliant environment. Some organizations believe that the potential additional cost of architectural compliance outweighs increased short-term costs. In other cases, cost pressures or a shorter-term view of benefits will reduce the impact of an "architected" environment.

Systems and technology infrastructure implemented by our organization will be compliant with our architecture even though there may be an additional cost for architectural compliance.

Implications

Agree	*Disagree*
• Faster migration to new infrastructure	• Slow migration—probably will not implement architecture
• Longer-term view of benefits	• Shorter-term view
• Lessened dependence upon existing installed base of hardware/software	• Probably stay with existing vendor/product set—more oriented toward existing proprietary systems

Cross-functionality

Several organizations have seen the opportunity to reuse applications, data, and related infrastructure in similar type functions across the organization. This requires a broader view of the business and an understanding of how to identify similar functions across the enterprise.

An orientation toward identifying and implementing cross-functional systems creates an opportunity for standards, standard components, and open systems. Technical integration opportunities are identified later on when developing architectures based on such principles. Portability of applications and data become more important so that similar systems and data can be implemented on different platforms that may exist across the organization. A standard means of identifying, classifying, and specifying system components is also required. Interface standards embodied in frameworks such as those outlined in the CIM Technical Reference Model can form the basis for interface and component specifications.

We will identify opportunities for cross-functional systems and implement systems in such a way that we can take advantage of standard components throughout the organization.

Implications

Agree	*Disagree*
• Need to identify generic components and how they are implemented	• Organization likely has a strong line-of-business orientation with significant business unit autonomy
• Standard application and data definitions are critical	• Potential problems with consolidation of information across organization
• Planning, architecture, and development process needs to incorporate cross-functional review	• View that similar functions contain enough differences that reuse of standard components would not be beneficial (too much modification)
• Significant reuse of design, code possible, but significant change in IT process, incentives and culture required	
• Role for a "repository" of standard elements and their definition	

Industry standards

An organization's position with regard to the source and use of standards is a critical factor in its position with regard to open and proprietary systems.

Organizations that have completed a principles definition process typically become favorably disposed toward using industry standards, especially if they have an orientation toward reuse of system components and cross-functional systems. The perceived risk of continuing to be vendor dependent is too high making the shift toward more open, industry standards appear less risky in the long run.

Our standards and technology choices will be based on vendor neutral standards where available and implementable.	
Agree ← **Implications** → *Disagree*	
• Role for open systems standards and technologies	• Perception that industry standards are not mature enough for use
• Development of standards and their implementation by vendors needs to be carefully tracked	• Focus more on a vendor's product architecture—vendor-dependent such as AS, NAS, etc.
• Migration strategies need to be developed for utilizing industry standards	• Organization unable to deal with migration issue at present time or does not see benefit of migration
• Focus on standards selection, then technology selection to support standards	• Limited set of vendors
• Need a mechanism for evaluating products in terms of compliance to standards and how to select products where standards have not been defined	
• Evaluate organization's industry standards as well as IT industry standards needed	

Measurement

While at first the issue of measurement would appear to be self-evident, the organization's attitude and investment in measurement and metrics vary dramatically. Some organizations view IT as delivering substantial business value—the actual IT measurements may not be critical. Other organizations, especially ones focused on IT efficiency, may want to have explicit metrics of many facets of the IT environment and its impact on the organization.

There are many measurement areas revolving around IT productivity, efficiency, and quality. Some statement of the organization's belief about measurement needs to be stated, as it will affect management processes around justification and direct investment in measurement programs.

Applications and technology components (processors, network, etc.) need to be implemented in such a way that measurement data can be captured for analysis and management of the IT environment.

Agree ← **Implications** →	*Disagree*
• Appropriate metrics and explicit indicators need to be established as well as a management process for collecting and managing the measurement information	• View that IT provides intrinsic business benefits and the added cost for implementing measurement capabilities are not justified
• Technology components need to provide data about their operation and performance	

Efficiency vs. effectiveness

Whether to focus on IT effectiveness or IT efficiency is closely tied with the organization's view of measurement. IT effectiveness tends to focus on external measures, ones that are often hard to quantify; these include evaluating the business value of implemented systems, the impact IT has on the organization's market share, etc. Efficiency, on the other hand, focuses more on internal measures such as productivity, cost control, processing efficiency, and transaction costs. The organization's belief on this issue can indicate their view of IT. Is IT a needed but unwanted expense, or is IT critical to the organization's success in its mission?

Information technology has a critical impact on our organization's business success. We must focus on improving IT's impact on operations.

Agree ← **Implications** →	*Disagree*
• IT viewed as a strategic asset	• IT is viewed as a support organization and not necessarily critical to the organization's mission
• External effectiveness measures critical	• Internal efficiency and cost control measures key
• Value focus—emphasis on increasing business benefits of IT	• Cost focus—increased cost and downsizing pressure on IT
• Probably more willing to take risks on IT investments	• Risk adverse

Security A statement on security and contingency planning is needed if roles and responsibilities are not defined or the policy is unclear. While the need for secure systems can be considered "motherhood," the organization's view of where and how security is implemented is important to state.

The following principle is one example of a security principle:

Implementation of security measures and contingency plans are the responsibility of the business unit manager where the system is implemented and must be "orange book" compliant.	
Agree ← **Implications**	→ *Disagree*
• Decentralized approach to security	• More centralization
• User managed security—IT plays an advisory role	• IT is responsible for security
• Process needed to ensure adequate security and contingency plans are implemented by the business units	• Security can only be achieved through central control

IT organization The IT organization principles deal with the organization's view of how IT is organized and how it interacts with the business. These principles will have an impact on the degree of dispersion of IT and the role of a centralized (if any) IT organization.

Dispersion Dispersion deals with the degree of control and autonomy that business units have over IT decisions. In a highly dispersed organization, business units make essentially all the IT decisions and may implement systems. In a non-dispersed organization, most IT-related decisions are made within the IT function.

Dispersion is different from centralization/decentralization. It is possible to have a decentralized IT function that is not dispersed. In this case, individual units may have their own IT functions. On the other hand, in a centralized IT function with dispersed control, the IT function may provide resources or advice to the business units.

Our IT organization will become increasingly dispersed into the operational units but policy will be made centrally.

Implications

Agree	*Disagree*
• Increasing control will migrate to the business units	• Centralized management of IT
• Diminishing role and control for central IT	• Linkages with business units have to be defined and managed
• Standards become critical to ensure that data can be shared and to limit potential duplication of effort across the organization	• Standards can be managed centrally
• Focus on “strategic” systems that directly support the business unit	
• Technology decentralization needs to be addressed—likely increased pressure for distributed computing	

System ownership

Management and ownership of implemented systems and technologies must be addressed to clarify roles and responsibilities. This principle has a direct impact on the rights and obligations of the business unit and IT managers.

Successful implementation and operation of information systems and technology is the responsibility of the business and operational unit(s) that the system supports.

Implications

Agree	*Disagree*
• Responsibility for systems implementation is the business unit manager’s	• IT likely responsible for successful implementation and realization of benefits
• Business unit managers must understand how to successfully utilize IT and manage implementation projects	

Role of centralized organization

The role of the centralized organization is closely related to the organization’s view of IT dispersion. In a highly dispersed environment, centralized IT may be responsible only for corporate financial and reporting systems and some standards setting (as in the following example). In other cases, the centralized IT organization may be a source of resources for business unit projects or, in the centralized case, be responsible for all IT functions.

The centralized IT organization will ensure that systems comply with our organization's standards and will assist organizations in IT.

Agree ← **Implications** → *Disagree*

Agree	Disagree
• Standards setting, advisory, and compliance verification role	• Implies either a limited role for centralized IT or a highly centralized IT function
• Management processes needed to develop and promulgate standards	• Standards and standards compliance performed by centralized IT organization

Life-cycle management Development, change management, and retirement of systems and technology infrastructure need to be managed on an ongoing basis. Architecture, and the systems developed from the architecture, must take into account constantly changing business needs and evolve with the business.

Our systems should be developed in such a way that they recognize the need for future changes to functional and technology requirements even if the development cost is increased.

Agree ← **Implications** → *Disagree*

Agree	Disagree
• Willing to accept additional cost to build and achieve easier maintenance and change management	• Limited change to technology
• Acceptance of changing technology may imply need for portability and portable environments	• Perceived to have stable technology base
• Increasingly shared and integrated systems will require adequate change-management facilities	• Standards less important
• Desire reduced maintenance cost and time	
• Increasingly modular design to facilitate change	

Application management

Application management principles deal with the organization's stated directions for managing applications and application components.

Depending upon the context for the architecture, these principles can focus on structural system issues (portability, modularity, etc.), management issues (methods, techniques, distribution), or some combination of the two.

Taken as a whole, the application management principles have to state the organization's beliefs on "How will we distribute and manage applications to get the maximum benefit for the organization?"

Development process and methods

The role of a development methodology and associated techniques across an organization should be addressed.

Systems should be implemented using a consistent methodology across the organization.	
Agree ← **Implications** →	*Disagree*
• Common development methodology • Promotes sharing of techniques and language	• Methodology managed by individual business units or departments

Reusability

The issue of reusability of applications and application components is analogous to many of the data standardization efforts under way. Standard definitions of business functions and application components are addressed in the following principle. This can be used to expand the focus of reusability beyond sharing code to sharing business designs, documentation, etc. Potentially, investment could focus more on an expanded system repository or I-CASE tools.

Applications should be developed using standard system components that are shared across the organization

Implications

Agree	Disagree
• Standard definitions	• High degree of customization required (possibly by business)
• Library of shared components needed	• Limited cross-functional systems
• Components may be source code, application designs, documentation, etc.	
• Migration to this environment needs to be planned—promote activities that provide shareability	

Build or purchase

The make-versus-buy issue needs to be resolved. Organizations can swing either way on this principle, depending on their view of the uniqueness of their business or applications.

Where possible we will purchase systems and components of systems by using commercial off-the-shelf (COTS) products

Implications

Agree	Disagree
• Buy orientation	• Tendency to build applications
• Procurement standards and process need to be defined	• Development resources and standards critical
• Cost of modification needs to be incorporated into package cost	

Cross-functional opportunities

Many organizations today have missed the opportunity to reuse portions of applications (see standard components principle above) by not identifying and architecting systems that support similar functions in multiple areas.

This represents a significant opportunity to improve productivity and obtain some economies of scale through functional or technical integration.

Applications should be developed so that they can be reused in similar business functions across the organization	
Agree ← **Implications** →	*Disagree*
• Process needed to determine what portions of an application are generic versus specific to the particular business function	• Traditional design methods adequate
• Standard interfaces and standard components needed	

Distribution of application functions

Distributing application functions away from centralized data centers will have a significant impact on the resulting architectures and management processes required to manage a distributed applications environment.

Application systems should be distributed and executed as closely as possible to the users of the application.	
Agree ← **Implications** →	*Disagree*
• Decentralized, workstation focus	• Centralized, host-processor focused
• Architecture levels (personal, work group, enterprise) need to be defined consistent with the organizational levels	• Traditional application design and development environment
• Potential cooperative processing implementation	• Economy of scale
• Potential multivendor implementations of similar software	• Likely single or limited set of vendor platforms
• Critical need for application environment standards	

Common application environments

A principle such as the one stated below supports a vendor-independent, portable environment. The result is a strong focus toward open systems.

Applications should be developed in a common environment that is independent of the underlying technology.

Agree ← **Implications** →	*Disagree*
• Open systems needs to play a key role	• Common environment may be single vendor
• Multivendor, standards based—standards are key	• Open standards less important than vendor products and standards
• Network computing standards required	
• Architecture models must deal with creating an “opaque” layer between application and technology	

Common user interface

The need for a common user interface (one with the same behavior) has emerged as an important requirement in many organizations. Common user interfaces (CUIs) can potentially provide significant improvements in productivity and training for users.

In the context of this principle, a common user interface does not necessarily imply a *graphic* user interface though the two seem to becoming synonymous with each other. The ability to customize the CUI for a particular need is often important as a generic CUI may not provide the best solution in all cases. The issue of migration from existing character terminals will need to be addressed, especially if graphic user interfaces are the chosen direction.

Applications should present a common user interface that is adaptable and extendible to particular user requirements.

Implications: Agree	Disagree
• Selection of an extendible common user interface toolkit critical	• Application-specific interfaces
• Migration from existing workstation and terminal technologies need to be addressed	• Easier migration—can utilize existing installed base
• Application design standards and procurement standards need to incorporate CUI standard	• Purchasing software and hardware not restricted by CUI
• Multivendor workstations and PCs may need to be accommodated	• Able to customize user interface to specific applications
• Can integrate applications from various sources more easily (from user perspective)	

Information management

The organization's approach to managing information is addressed in the information management principles.

Multiform vs. single form

The scope of the information managed and the degree of integration of different forms of information are addressed in the following sample principle.

Our architecture and implemented systems must address the management of all forms of information (data, text, voice, image) in an integrated manner.

Implications: Agree	Disagree
• Compound document standards required	• Easier to implement today
• Information architecture needs to address all forms	• Oriented toward data and possibly text management
• Tools and standards needed	• Standards available

Data standardization

The organization's view on data standardization needs to be articulated. Is there a need for standard definitions? Is the expense and effort justified? Is the organization so decentralized that standardization efforts are not really worthwhile?

Standardization of data definitions and their implementation, access, interoperability, and communication is needed across the organization to provide improved quality and consistency of data and improve overall effectiveness of implemented systems.	
Agree ← **Implications**	→ *Disagree*
• Standard definitions required	• Limited communication across organization
• Significant effort and cost associated with standardization effort	• Limited need or ability to consolidate data
• Improvement in sharing and consolidation of data	

Ownership and stewardship

Ownership and stewardship of data needs to be addressed and agreed upon. This principle has a number of implications on how roles and responsibilities get defined.

Data is a corporate asset and does not belong to a particular business unit or individual.	
Agree ← **Implications**	→ *Disagree*
• "Corporate" ownership and management of data	• Business unit management and control of data
• Management of data may be delegated—process is required to ensure it is managed correctly	• Fragmentation of data a possibility—standards for consolidation and interchange required

Data access and distribution

Like distribution of application function, distribution of data should be resolved. The organization's view on application and data distribution helps determine its "style" of computing—centralized, decentralized, or some combination of the two.

Data will be managed and stored as closely as possible to the person/organizational unit who uses it.

Implications

Agree	*Disagree*
• Distributed data management required	• Data can be managed using existing technologies
• Degree of sharing needs to be established for data	• Accessibility of data by distributed users needs to be addressed

Technology management

A wide variety of technology management principles can be stated by an organization. Organizations often break such principles down into key topic areas that deal with the major components of the technology architecture (hardware, system software, communications, etc.).

It is important to articulate the role of each technology component and the organization's attitude toward managing vendors and technology. With the input from the application and information management principles, the technology management principles are where the organization's position toward open systems and standards often gets stated in black and white.

Interchangeable components

The first principle addresses interchangeability of vendor products and services and, by implication, the organization's view toward open systems and standards.

Some organizations view this principle (as stated) as unachievable in a reasonable time frame and choose a more conservative approach.

We will implement technology components so that we have the option of exchanging vendor products with minimal disruption to the environment.

Implications

Agree	*Disagree*
• Standards-based, open-systems approach	• Lock-in to a vendor
• Interfaces and environments need to be standardized	• Can stay with existing technology base or do a selective migration.

Vendor management

An alternative statement of the following principle could be:

"We will limit the number of alternative vendors to a limited, manageable set" or *"We are committed to a single-vendor environment."*

We will utilize any vendor who provides us with the best technology for a business need.

Implications

Agree	*Disagree*
• Need for standard environments to support multivendor	• Allowable vendor set needs to be established that can meet most needs
• Standard connectivity approaches needed	• Limited set of vendors can be managed—build stronger relationships
• Portability of applications and data must be addressed	

Distribution of processing capability

We will decentralize our processing environment so that individual business units control their own computing resources.	
Agree ← **Implications** → *Disagree*	
• Promotes highly distributed environment • Remote management and standardization critical	• Promotes more centralized environment

Role of intelligent workstations

The following principle brings intelligent workstations (PCs, workstations) to the forefront as a platform for delivering applications in the architecture.

The result is a highly distributed, processing environment. Applications and access to data would be provided through the workstation, supplemented by servers and hosts (minicomputers and/or mainframes) providing processing and data services to the workstations.

Intelligent workstations will be the primary access and delivery vehicles for applications and data.	
Agree ← **Implications** → *Disagree*	
• Workstation, LAN orientation • Workstation standards and connectivity critical—network computing needed to integrate with other components • Strong network computing role	• Delivery through minis, mainframes • Standards critical, but applications and data less distributed

Network connectivity

The following three principles address networking issues. The first establishes the role of the common network utility.

We will use a common network environment using industry standards to interconnect all workstations, computers, and communicating devices.

Agree ← **Implications** → Disagree	
• Connectivity standards need to be defined	• Autonomous computing/network environments
• Full set of communications and transport facilities will be required	• Tend to stay with host-based communications—terminal to host
• Vendors need to support interconnectivity	• Can implement different networking approaches in different areas of organization
• Integrated LAN/WAN/external network design required	
• Connectivity with customers/suppliers needs to be addressed	
• No system is an "island"	

Network interfaces

The following principle supports the premise that the "network is the computer" by placing the common network environment as the core through which all devices communicate. Standard protocols and interfaces begin to establish the need for a common interface standard.

The OSI model is often used to describe the various interface layers and as a framework for identifying standards.

All communicating devices must interface to the network through a standard set of protocols and interfaces.

Agree ← **Implications** → Disagree	
• Limit direct connection of devices to computers—connect through common network	• Point-to-point links (e.g., computer to workstation) allowed
• Migration from existing installed base needs to be addressed	

Network services

The network's role as a value-added service provider is established in the following principle. This represents a belief that value-added services can be delivered by the network separately from the processors attached to the network.

Common services such as file transfer, electronic mail, directory management, and network management should be provided through a common networking environment.

Implications

Agree	*Disagree*
• Value-added applications need to be provided	• Common services provided by host processor
• Security of directories and services need to be addressed	• May need to integrate different services at the workstation instead of through the network
• Additional system management services should be examined	
• Network-based processors for network services need to be defined	

Conclusion

The above-listed principles are but a few of the many that an organization may seek to develop. We recommend that all the existent CIM principles be incorporated into each architecture effort.

Appendix B: How To Do A Baseline Characterization

General approach

The baseline data collection effort is the first step in developing a useful baseline characterization of the current architecture. Standard templates were developed over the course of the first SBA projects that were completed (or are under way) at the time this update to the SBA Guide was produced.

The first section of this appendix presents these templates, along with the instructions that accompany them, and includes a sample of a completed template from the USMC project.

The second major section of this appendix provides guidance for the analysis of the information generated from the completed templates. General questions of interest and "rules of thumb" for analysts are provided.

This page intentionally left blank.

Baseline Data Collection

Work Organization Templates

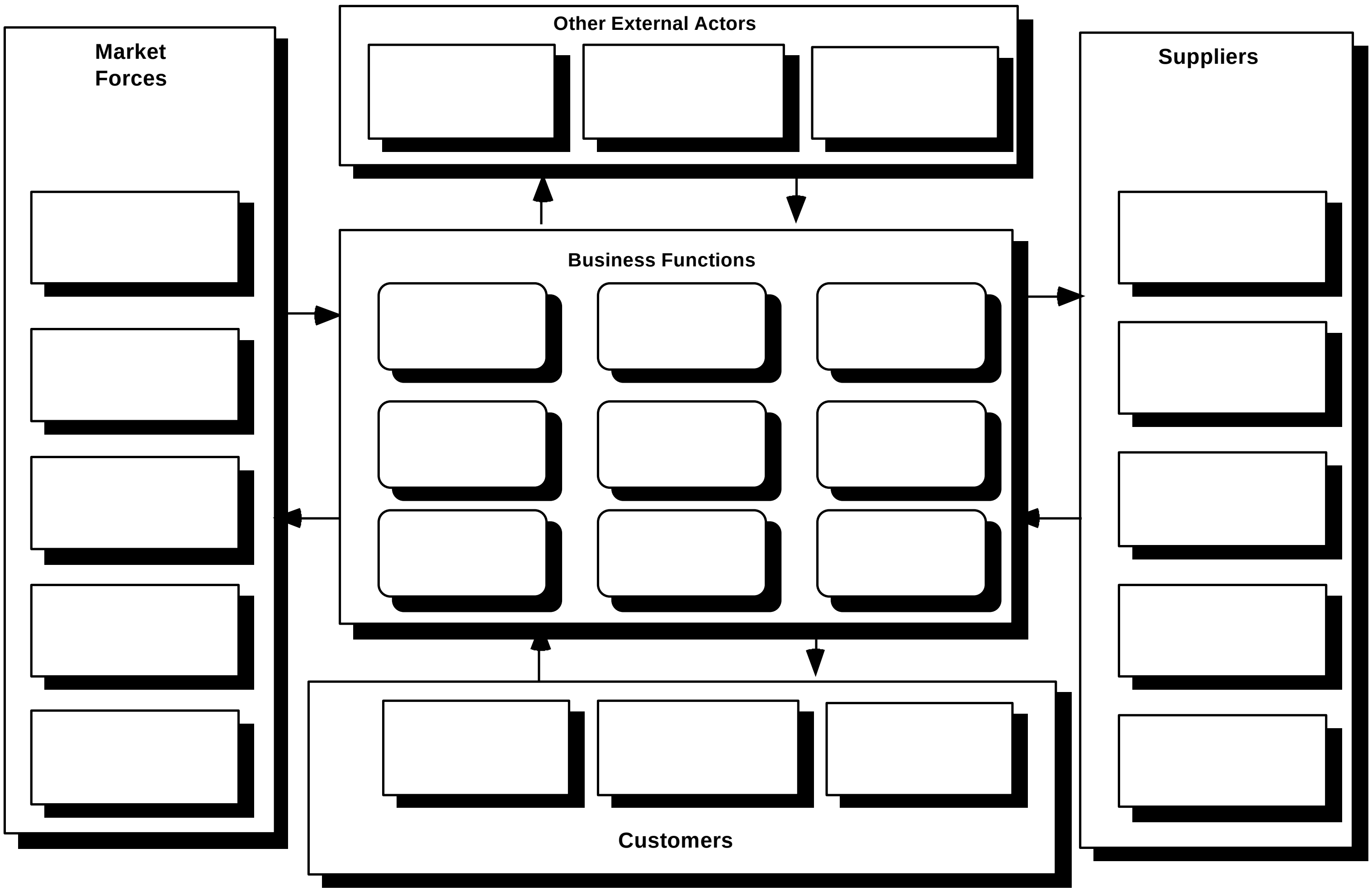
This page intentionally left blank.

Business and Work Models Template • Fill out one of these templates for each major business function in the enterprise • See the Baseline Assessment Glossary of Terms for definitions	
Mission	• This is the organization's mission:
Function	• A major grouping of work for the enterprise:
Processes	• Activities or job steps leading to a desired result within the function:
Location	• Physical location(s) where work is performed:
Headcount	This should also be entered on Baseline Template for Function Costs.
Budget	This should also be entered on Baseline Template for Function Costs.

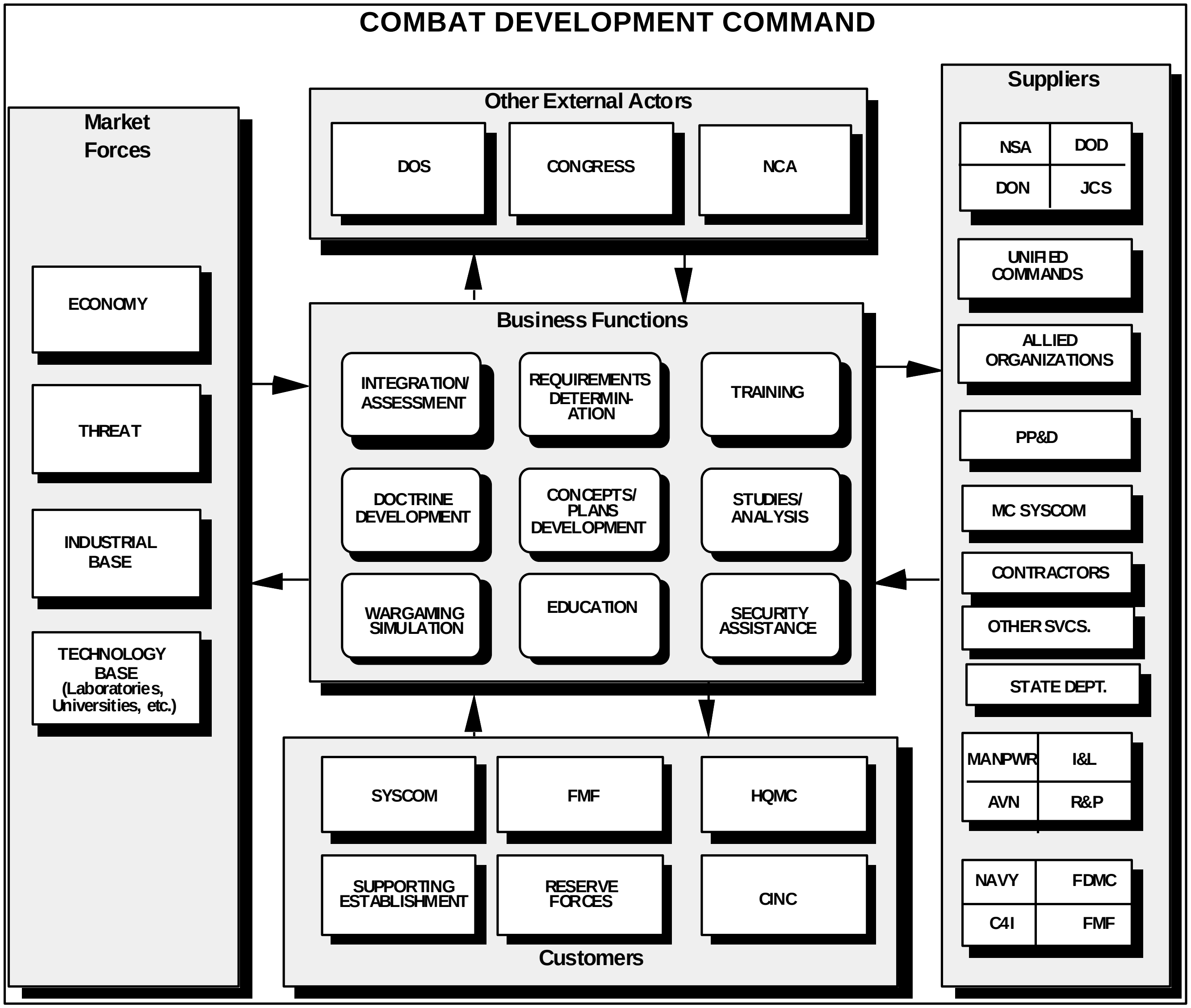
Business Context Template

Competed by: ______________________

- ***Fill in name of organization or business Unit:*** ______________________
- ***Then fill in the boxes with appropriate information about your organization or business unit***



Business Context - Sample



This page intentionally left blank.

Information Templates

This page intentionally left blank.

Existing Application(s) Linked to Data Groups:

- ***List all known existing applications which support business functions within your organization in the left-most column of this matrix. (give common application abrreviation and full name if known)***
- ***List all known data groups across the top of the matrix, one per column. Attempt to list closely related data groups in adjoining columns under a larger heading called "Subject", i.e. the Subject will span one or more data group columns.***
- ***If more columns are needed repeat the applications on a second sheet and continue listing the domains and information subjects.***
- ***Place a C, R, U, and/or D in the intersection of the application and information subject to signify whether the application Creates, Reads, Updates, and/or Deletes the information subject as part of its functionality. Combinations are Possible.***

<u>This template should contain all data groupings and applications of the entire enterprise.</u>

Subject													
Data Group Application													

Baseline Template – Application to Information – Sample

Subject / Data Grouping / Application	Customers			Finished Products & Services			Supplier			Transport		Plant Equipment & Facilities	
	Customer Information	Business Agreement	Orders	Products	Inventory	Services	Supplier Information	Business Agreement	Purchase Order	Transport Acquisition	Transport Utilization	Equipment	Facilities
Planning Applications													
New Product Concept and Planning System	R			CRUD		R						CRUD	R
Executive Information System	R	R	R										
Profit Planning System	R	R	R	R	R	R	R	R	R	R	R	R	R
Stategic Planning System	R	R	R	R	R	R	R	R	R	R	R	R	R
Profitability Analysis and Reporting System	R	R	R	R	R	R	R	R	R	R	R	R	R
Market Analysis and Trending System	R	R		R		R							
Capital Planning and Tracking System	R	R	R	R	R	R	R	R	R	R	R	R	R
Patent & License System				CRUD		CRUD	R	R				CRUD	CRUD
Selling Applications													
Customer Business Agreement System	CRUD	CRUD	R	R		R				R		R	R
Sales Demand Forecasting System	CRUD	R	R	R		R							
Call Reporting System	CRUD	CRUD	R	UD		UD							
Contract Versus Actual Reporting System	CRUD	CRUD	R	R		R							
Product Applications Technology System	CRUD	R		R		CRUD						R	R
Consulting Service Tracking and Problem Resolution System	CRUD	R	R	R		CRUD						R	R
Advertising and Promotion Scheduling and Information System	R			R		R							
Credit Management System	CRUD	R	R				R	R	CRUD				
Product and Services Delivery Applications													
Order Management System	CRUD	R	CRUD	R	R						R		
Complaint Tracking and Resolution System	CRUD	R	CRUD	R	R						R		
Customer Information System	CRUD	R	R	R		R				R		R	R
Finished Product Inventory System			R	R	CRUD							R	CRUD
Production Scheduling System			R	R	R							R	CRUD
Automated Load Out System	R		CRUD	R	R						CRUD		
Transport Scheduling and Optimization System	R		R	R							CRUD		
Fleet Maintenance and Inventory System	R	R		R			R	R	R	CRUD	CRUD	CRUD	
Fleet Acquisition System	R	R		R			R	R	R	CRUD	R	CRUD	

Database Inventory – Template

Instructions:

Fill in the requested information for databases and/or electronic files which support your area of operation. The following definitions of the columns apply:

Database Name	The commonly used name or acronym for the database or file (i.e. SASSY, MIMMS, TMS, etc.)
Description	A brief description of the contents of the database or file (i.e. personnel records, patient records, requisitions, etc.)
Geographic Location	The location where the database or file physically resides (i.e. the commonly accepted designation, such as city, base name, command name, etc.)
Type	The type of database or file system used (i.e. DOS file, dBase, Adabas, Oracle, Paradox, DB2, etc.)
Platform	The type of computer platform (i.e. PC, Workstation, Midrange, or Mainframe; also state specific kind and model number if known, such as IBM PS/2, Sun Sparc Station, AS/400, DEC/VAX, IBM 3090, etc.)
Additional Notes	Any other clarifying notes which your feel will be helpful in characterizing the Database or File

Database Name	**Description**	**Geographic Location(s)**	**Type**	**Platform**	**Additional Notes**

Database Inventory – Sample

Database Name	Description	Geographic Location(s)	Type	Platform	Additional Notes
3M	Maintenance, Scheduling, and Parts	Philadelphia ASO		Mainframe NAVY	
BNA	Training/Assignment	Quantico CDPA	Adabas	Mainframe	
CAEMS	Embarkation Data (size, weight, bal, etc.)	Numerous Commands	Paradox	PC	
CAIMS	Ammunition (AVN)	Philadelphia ASO		Secure Dial-in to Mainframe	
Central DB System	MTF Formats, Data Codes, Syntax Checks (also TADIL formats)	Reston JIEO		Sun Sparcstation and 7	
Control Master File	Personnel and Pay	Kansas City CDPA	Adabas	Mainframe	
Defense Intelligence Data Sys	Counter Terrorist/Counter Intelligence	Bowling AFB DIA	RDB	DEC VAX	
Emerald	Counter Narcotics	Bowling AFB DIA	Sybase	Sun (client-server)	
HAS	Accounting	Quantico	Adabas	Mainframe	
IAS Database	Portion of MIIDS/IDB plus Tactical Update	MEF, DIV, REGT(?)	Sybase	Sun Sparcstation 2 or 1.0	BN version working at Army LCU Laptop
LFADS	Supply and Equipment Data	Numerous Commands	Clipper/ADA	PC	
MAGTF Data Library	Reformatted Data of all types	Major Commands	Clipper	PC	
MAGTF II	Org and Transport Data	Numerous Commands	Clipper	PC	
MCAIMS	Student Information/Course Data (Central File at Quantico planned)	Each School	Adabas	PC	
MCCRES	Unit Scores, Mission and Performance Standards	Quantico	Adasage	PC	
MCLLS	Lessons Learned Notes (distributed by CD to Major Commands)	Quantico	dBase/Clipper	PC	
Met Table	Meteorological Data	Each BCS		PC	
MIIDS/IDB	All Non-US Military Info, Electronic Order of Battle, Airfields/Facilities, General Military Intelligence	Bowling AFB DIA	Model 204	Mainframe	
MIMMS	Maintenance Data	Numerous Commands		Mainframe	
NALCOMIS	Maintenance/Supply Data	Numerous Commands	Cobol	Mainframe and PC	
NALISS	Supply Parts	Philadelphia ASO		Mainframe	
NAVFLIRS	Extract of Flight Info/Pilot Info	Norfolk NAVNASSO		PC to Mainframe Tape	
Ord Table	Ordnance Characteristics	Each BCS		PC	
Org, Equip, and Supply Data	Org, Equipment and Supply Data	Numerous Commands	Paradox	PC	
POM	Fiscal Data				
Resource Allocation Display	Navy/Marine Corps Shared Financial Data	Washington Navy HQ Op-80		Mainframe	
SABRS	Accounting and Budget	Quantico CDPD	Adabas	Mainframe	
SASSY	Parts Data	Albany/Barstow		Mainframe	
SUADPS	Financial and Inventory Data	Numerous Commands	Cobol	Mainframe	
TCAMS	Travel Data	Numerous Commands	Clipper	PC	
TDMS	Parts Technical Data		Adabas	Mainframe	
TERPES	MIIS/IDB EDB Plus MISSLE OB, GOLDDB and Tactical copy of updates (send tactical version back for updates to MIIDS/IDB)	VMAQ2 (four locations)	Sybase	Sun 620 Fileserver	
TMR Data Base	Organization Structure and History	Quantico CDPA	Adabas (w/CICS)	Mainframe	

Application Templates

This page intentionally left blank.

Application Inventory – Template

Instructions:

Fill in the requested information for all applications which support your area of operation. The acronyms used on this template should also be used on any other templates which refer to application systems. The following definition of columns apply:

Application Acronym	The commonly used abbreviation for the application
Application Name	The full English name of the application
Type	Denote whether On-line, Batch, or Both On-line and Batch (“O”, “B”, or “OB”)
Number Users	The number of users
Language	The language(s) the application is written in
Operating System	The Operating System and On-line Transaction Processing Monitor (if applicable) under which the application runs
Where Run: Specific Technology Platform	The commonly used designation for the technology platform upon which this application runs (i.e. the unique name for a particular processor and the basic vendor model information, such as “DAISY2, IBM3090 Model 600E”)
Where Run: Location of Platform	The actual locations where the application runs on the platform identified in the prior column
Age	The age of the application in years
Number Programs	Number of executable programs in the application
Changes Requested	Number of change requests in past year; over entire history
Changes Implemented	Number of change requests implemented in past year; over entire history
Failures	Number of application failures in past year; over entire history
Developer	What organization developed the application
Support	What organization supports the application

						Where Run									
Application Acronym	***Application Name***	***Type***	***Number Users***	***Language***	***Operating System***	***Specific Technology Platform***	***Location of Platform***	***Age***	***Number Programs***	***Changes Requested***	***Changes Implemented***	***Failures***	***Developer***	***Support***	

Application Inventory – Sample

Application Acronym	Application Name	Type	Number Users	Language	Operating System	Where Run: Specific Technology Platform	Where Run: Location of Platform	Age	Number Programs	Changes Requested	Changes Implemented	Failures	Developer	Support
AB	Automated Budget													
ABA	Bonds & Allotments		All MC	COBOL II	MVS-XA	IBM, AMDAHL, BURROUGHS, UNISYS, TANDEM	LEJEUNE, PENDLETON, OKINAWA, EL TORO, CHERRY PT., KANSAS CITY	22	159	27	27		KANSAS	KANSAS
ABA	Bonds & Allotments		All MC	COBOL II	MVS-XA	IBM, AMDAHL, BURROUGHS, UNISYS, TANDEM	LEJEUNE, PENDLETON, OKINAWA, EL TORO, CHERRY PT., KANSAS CITY	22	159	27	27		KANSAS	KANSAS
ABE	Application B Enhanced			TACL	GUARDIAN	TANDEM	CHERRY PT.	5	68				FMSO	FMSO
ACIS	Automated Claims Information System		MH Claims	DBASE III	MS-DOS	PC	HQMC	5	21					CONTRACT
AFRS	Automated Fitness Report System		MMPE	NATURAL	MVS-XA	IBM	HQMC	20	216	270	270	0	QUANTICO	QUANTICO
AIMS	Awards information Management System		MH	FORTRAN	MS-DOS	PC	HQMC	4	200					CONTRACT
ALPS	Automated Leave and Pay System		CIV PAY OFFICE, NAVCOMP, IRS	COBOL	MVS-XA	IBM, AMDAHL, BURROUGHS, UNISYS, TANDEM	QUANTICO, ALBANY, KANSAS, LEJEUNE, PENDLETON, OKINAWA, EL TORO, CHERRY PT.	23	85	92	89	1	QUANTICO	QUANTICO
AMHS	Automated Message Handling System			C/TOPIC	UNIX	DEC	MEF, Garrison, CPs	NEW					DIA	
AMMOLOGS	Ammunitions Logistics System			COBOL	MVS-XA	IBM, AMDAHL, BURROUGHS, UNISYS, TANDEM	LEJEUNE, PENDLETON, OKINAWA, EL TORO, CHERRY PT., KINSER PARIS ISLAND	4	12	1	1	6	ALBANY	ALBANY
AN/TPS-59	Radar Set 3D	OL	15 Units	CMS-2, ULTRA-32		AN/UYK-7	AIRWING	7						
AOWP	Automated Orders Writing Process		MMEA, MMOA, MMSR	COBOL II	MVS-XA	IBM	QUANTICO, KANSASNCITY	13	55	33	31	3	KANSAS	KANSAS
APADE	Auto procurement			TACL	GUARDIAN	TANDEM	CHERRY PT.	3	1458				FMSO	FMSO
APCS	Automated Production Control System							NEW						
ARMS	Automated Recruit Management System		Recruiting	COBOL II	MVS-XA	AMDAHL	KANSAS CITY	15	675	11	11	0	KANSAS	KANSAS
ASMS	Advanced Shipping and Monitoring System			TACL	GUARDIAN	TANDEM	CHERRY PT.	3	10				FMSO	FMSO
ATAC+							(Maintained by the Navy)							
ATRIMS	Aviation Training & Readiness Information Management System		ALL FMF	ADA	MVS-XA	IBM	MC WIDE	8	1				QUANTICO	QUANTICO
AV-3M/NAVFLIRS	Aviation Maintenance Material Management System Naval Flight Record		Aviation Units	COBOL II, ADA	MVS-XA, MS-DOS	IBM, AMDAHL, BURROUGHS, UNISYS, TANDEM, PC	LEJEUNE, PENDLETON, OKINAWA, EL TORO, CHERRY PT.	6	72	41	40	2	QUANTICO	QUANTICO

Application to Function – Template

Instructions:

List all applications identified on the Application Inventory template in the left-most column of this template (give the common application abbreviation and full name if known).

Major functions as identified in MHSS Technical Management Plan document are listed in the column headings.

If there are other functions that your area performs which are not covered by these columns, list them in the blank columns.

Place an "X" at the intersection of the row and column if the application in that row provides some automated information systems support to the business function represented by the column.

Function / *Existing Application*	Health Care Delivery Clinical	Health Care Delivery Preventative	Health Care Management	Financial	Logistics	Human Resources	Information	Research	Education	Liaison		

Application to Function – Sample

Existing Application \ Function	Command & Control	Doctrine	Finance	Human Services	Intelligence	Logistics	Mainten-ance	Manpower	Plans/ Concepts	Policy	Procure-ment	Requirements	Supply	Training/ Education	Transport	Warfare/ Operations/ Combat Simulation
ABA			X													
ABA				X												
ABE						X										
ACIS																
AFRS																
ALPS																
AMHS					X											
AMMOLOGS											X		X			
AN/TPS-59 RADAR																X
AOWP																
APADE						X										
APCS							X									
ARMS									X							
ATAC+																
ATRIMS												X		X		
AV-3M/NAVFLIRS						X		X								
AWIS						X										
AWN																
BAS																
BCS																X
BCS																X
BNA								X	X	X						
BNA								X	X	X						
BNA					X			X	X	X						
BNA								X	X	X						
BREES						X										
BUDGET																
BUDS CLASS II																
CAEMS									X				X		X	
CAEMS									X				X		X	
CAIMS																
CAIS				X												
CASPRO								X								
CATS										X						
CCS													X			
CDCS											X		X			
CMIS									X				X			
CPMS				X												
DAIS				X												
DASC																
DCERPS																
DCIP			X													
DDA						X										
DE													X			
DEERS IIB				X												
DEERS IIIB				X												
DEPMEDS													X			

Application to Physical Location – Template

Instructions:

List all applications identified on the Application Inventory template in the left-most column of this template (give common application abbreviation and full name if known).
List all known physical locations where these applications may be accessed across the top of the matrix, one per column.
If more columns are needed repeat the applications on a additional sheet(s) and continue listing the locations on those sheets.
Place an "X" at the intersection of the row and column if the application in that row is accessed by at least one user at the location represented by the column.
NOTE: Access to the application at a location does not necessarily mean the application is running on a computer at that location. It could be remote.

Physical Location / *Existing Application*										

Application to Physical Location – Sample

Physical Location / *Existing Application*	1st MAW	29 Palms	2nd MAW	3rd MAW	4th MAW	Albany	BN Level	Cherry Point	El Toro	Parris Island	Pendleton	Quantico
ABA												
ABA											x	
ABE								x	x			
ACIS												x
AFRS												x
AIMS												x
ALPS						x					x	x
AMHS												
AMMOLOGS						x				x	x	
AN/TPS-59	x		x	x	x							
AOWP												x
APADE								x	x			
APCS						x						
ARMS											x	
ATAC+												
ATRIMS											x	x
AV-3M/NAVFLIRS								x	x		x	
AWIS												
AWN												
BCS		x										
BCS												
BNA												x
BNA												x
BNA												
BNA												x
BREES								x	x			
BUDGET						x					x	x
BUDGET CLASS II						x					x	x
CAEMS												
CAIMS												
CAIS												x
CASPRO												
CATS								x	x			
CCS						x					x	x
CDCS												
CMIS											x	
CPMS						x						
DAIS												x
DASC	x		x	x	x							
DCIP												
DDA								x	x			
DE						x					x	x
DEERS IIB						x					x	x
DEERS IIIB						x					x	x
DEPMEDS						x						

Application to Standard – Template

Instructions:
For all applications identified on the Application Inventory Template fill in the standards which are in use currently.

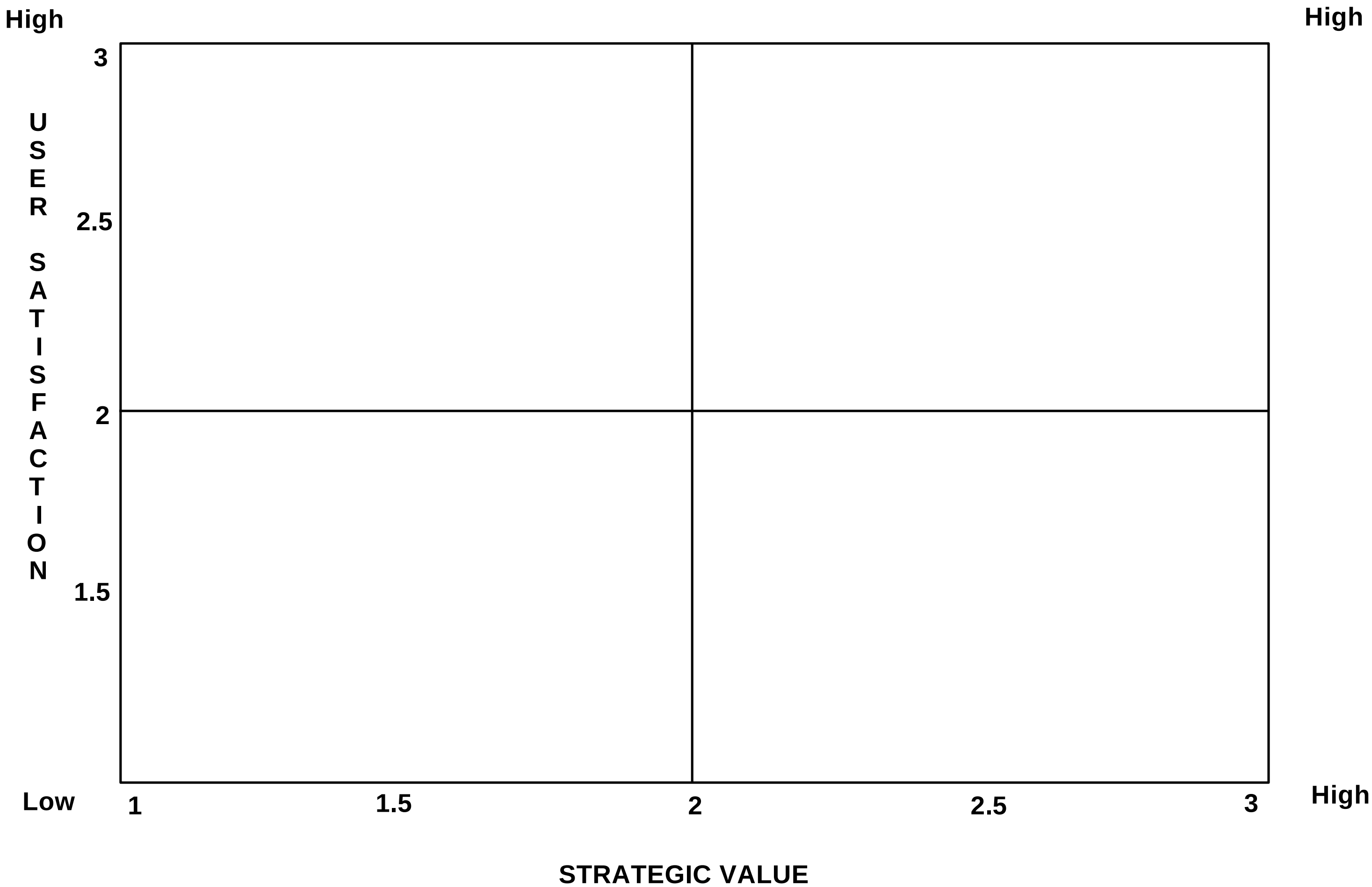
User Interface	The software (or hardware) which "presents" the results of the application system's processing to the user (i.e. MS-Windows, 3270, Character Based, etc.) Include any online monitor software such as CICS or IMS/DC in this column as well.
Database	The data base or file management software used by the application (i.e. Oracle, DB2, dBase IV, Adabas, VSAM, ISAM, etc.)
Application	Any self-contained "package" software which has been made an integral part of the application (i.e. Commander EIS, MS-Excel, Harvard Graphics, etc.)
Language	What the application code is written in (i.e. Cobol, Cobol II, C, C++, Pascal, ADA, etc.)
Operating System	The system software under which the application runs (i.e. MS-DOS, MVS-XA, DOS/VSE, OS/400, OS/2, UNIX System V, Xenix, etc.)
Communications Media	The transmission path for data communications (i.e. Marine Corps Data Network, Defense Data Network , Autodin, Token Ring, Arcnet, LocalTalk, etc.)
Communications Protocol	The protocol(s) used to control data communications related to this application (i.e. Banyan Vines, Novell Netware, SNP, SDLC, SNA, LU6.2, etc.)
Other Services	Any other aspects of the application which do not fall under any standards listed in prior columns

Standard / *Existing Application*	User Interface	Database	Application	Language	Operating System	Commun. Media	Commun. Protocol	Other Services

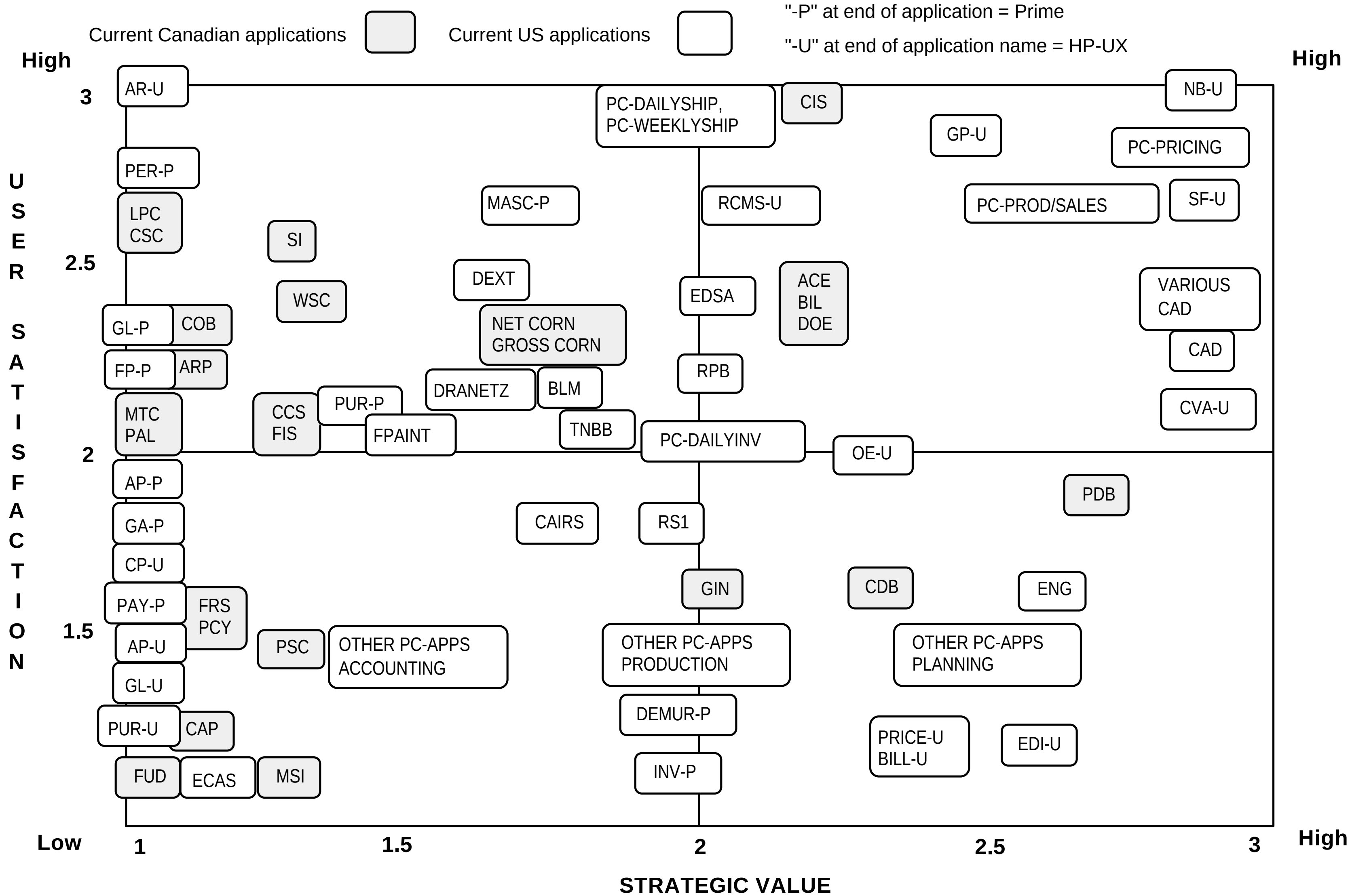
Application to Standard – Sample

Standard / *Existing Application*	User Interface	Database	Application	Language	Operating System	Commun. Media	Commun. Protocol	Other Services
UADPS-WCS	UTS-30 (Sperry Terminals)	DMS 1100		Cobol	EXEC (Univac)	NAVNET	SDLC	
CATS	DPS (Display	DMS 1100		Mapper	EXEC (Univac)		SDLC	
DDA	PC / 6530 (Tandem			TACL	Guardian		SDLC / 3270	
ABE	TDI			TACL	Guardian		Async / 6530	
G-MAN	PC / 3270			TACL	Guardian			
MDAS	PC / 6530 (Tandem			TACL	Guardian	NLN (Navy Logisitics Net)	SDLC	
RODS				TACL	Guardian			
UADPS-SP	PC / 3270			Cobol	MCP (Burroughs)			
MCCRES	CICS	DBaseIII		ADA	MS-DOS	MCDN		
SWIFTHAWK				C	MS-DOS			
SORTS	3270	Adabase		Cobol	MS-DOS			
MAGTF II				Cobol	MS-DOS	WWMCCS		
SASSY IA				Cobol	MS-DOS			
MCLLS		DBaseIII		DBase / Clipper	MS-DOS	MCDN		
TCAC				Fortran	MS-DOS	VAF/UHF		
EPOS	3270 Emulation			PC Focus	MS-DOS		SNA SYS	Bar Code
OCIS				PC Focus	MS-DOS			
RPM/FHS		Oracle		SQL	MS-DOS			
MTF EDITOR				Turbo Pascal	MS-DOS			
MIDAS (EIS)			Redimaster		MS-DOS		Async	
CMIS			Harvard Graphics		MS-DOS			
AV-3M/NAVFLIRS	CICS			ADA / Cobol II	MS-DOS / MVS-XA		TDI	
ATRIMS	CICS	Adabase		Ada	MVS-XA	MCDN		
MCAIMS	CICS	Adabase		C / Adabase	MVS-XA	MCDN		
AMMOLOGS	CICS/Natural	Adabase		Cobol	MVS-XA	MCDN		
MEDLOGS	CICS/Natural	Adabase		Cobol	MVS-XA	MCDN / Autodin		
MIMMS IB	CICS/Natural	Adabase		Cobol	MVS-XA	MCDN		
JUMPS/MMS	CICS / Natural	Adabase		Cobol	MVS-XA	MCDN	SNA LU6.2	
BUDGET				Cobol II	MVS-XA	Tape / Disk Transfer		
ATAC +					MVS-XA	Token Ring	VINES	
TMS	CICS/Complete/ Natural	Adabase		Cobol / Natural	MVS-XA / MS-DOS	MCDN		
APADE	MCR (Borroughs Master Control program)			TACL	Pathway	Token Ring	VINES	
PLRS				CMS2Y	SDEX / M	UHF	PLRS Interface Controller	
PLRS	PLRS Unique	PLRS Unique	Navigator CMD & Control	CMS-2	SDEX/M, SDEX	PLRS Unique	PLRS Unique	
NALISS	CICS			Natural			SDLC / 3270	
BCS				TACFIRE S/W		Comm. Wire	TACFIRE S/W	

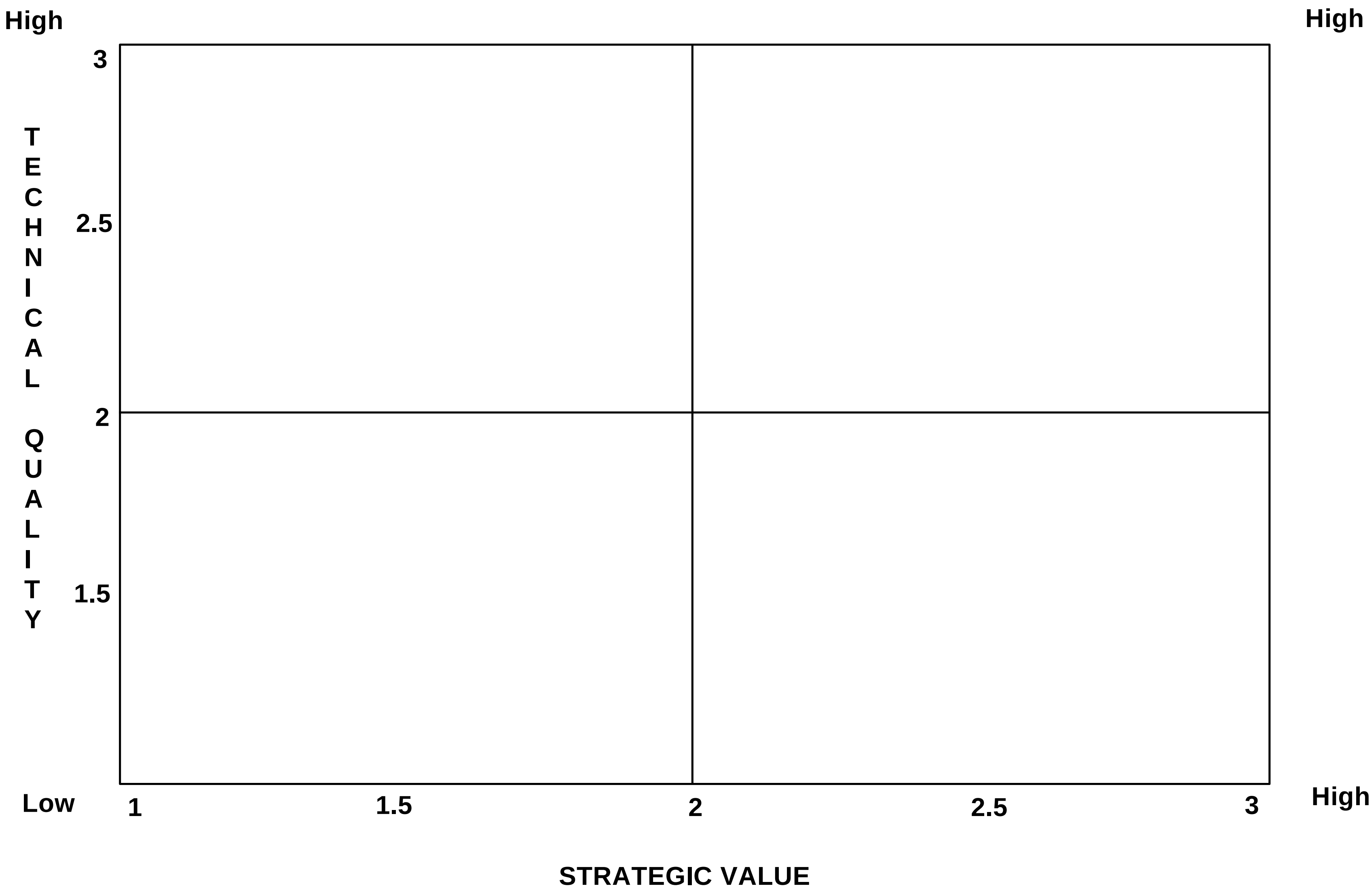
User Satisfaction versus Strategic Value



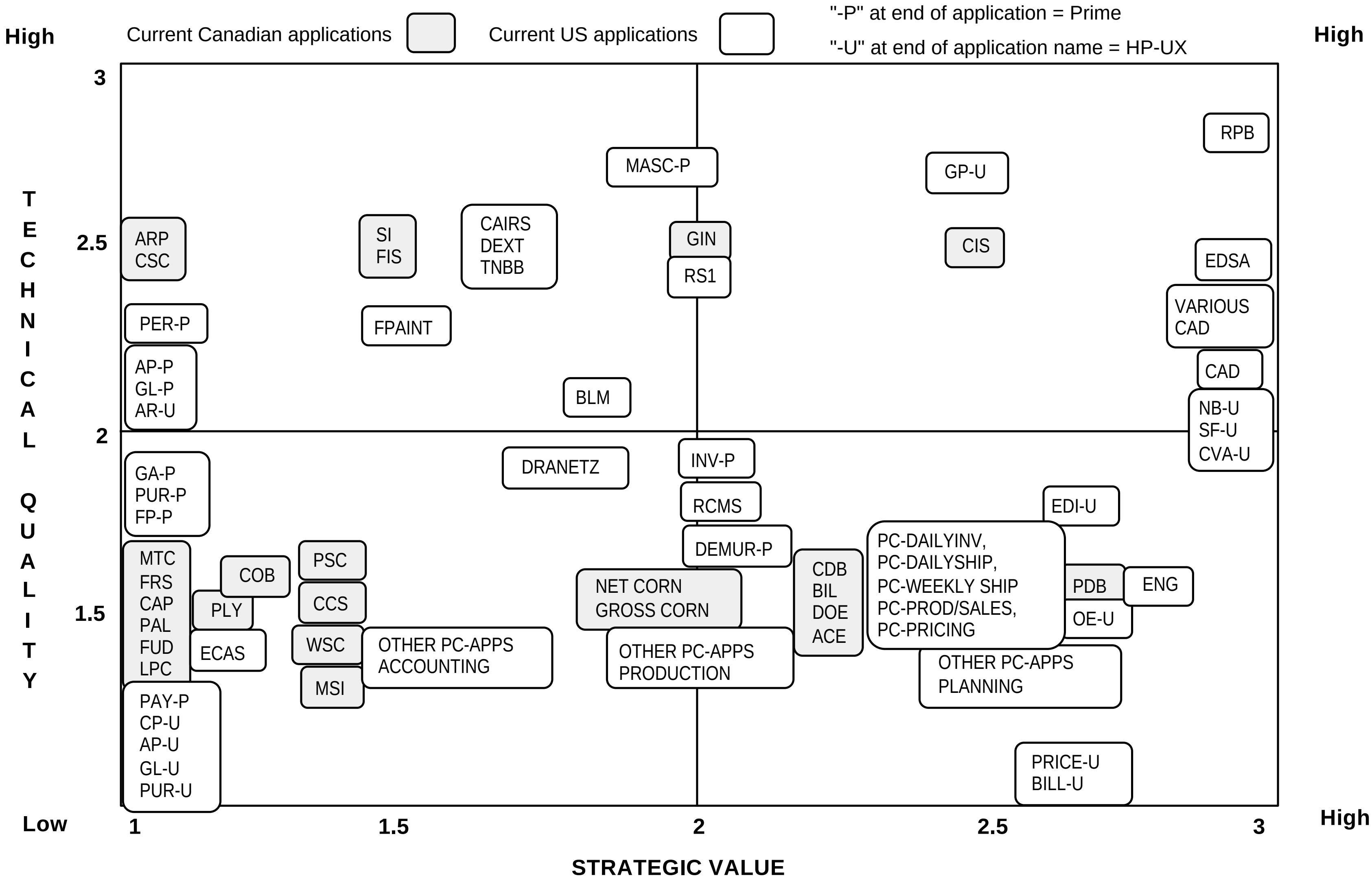
User Satisfaction versus Strategic Value



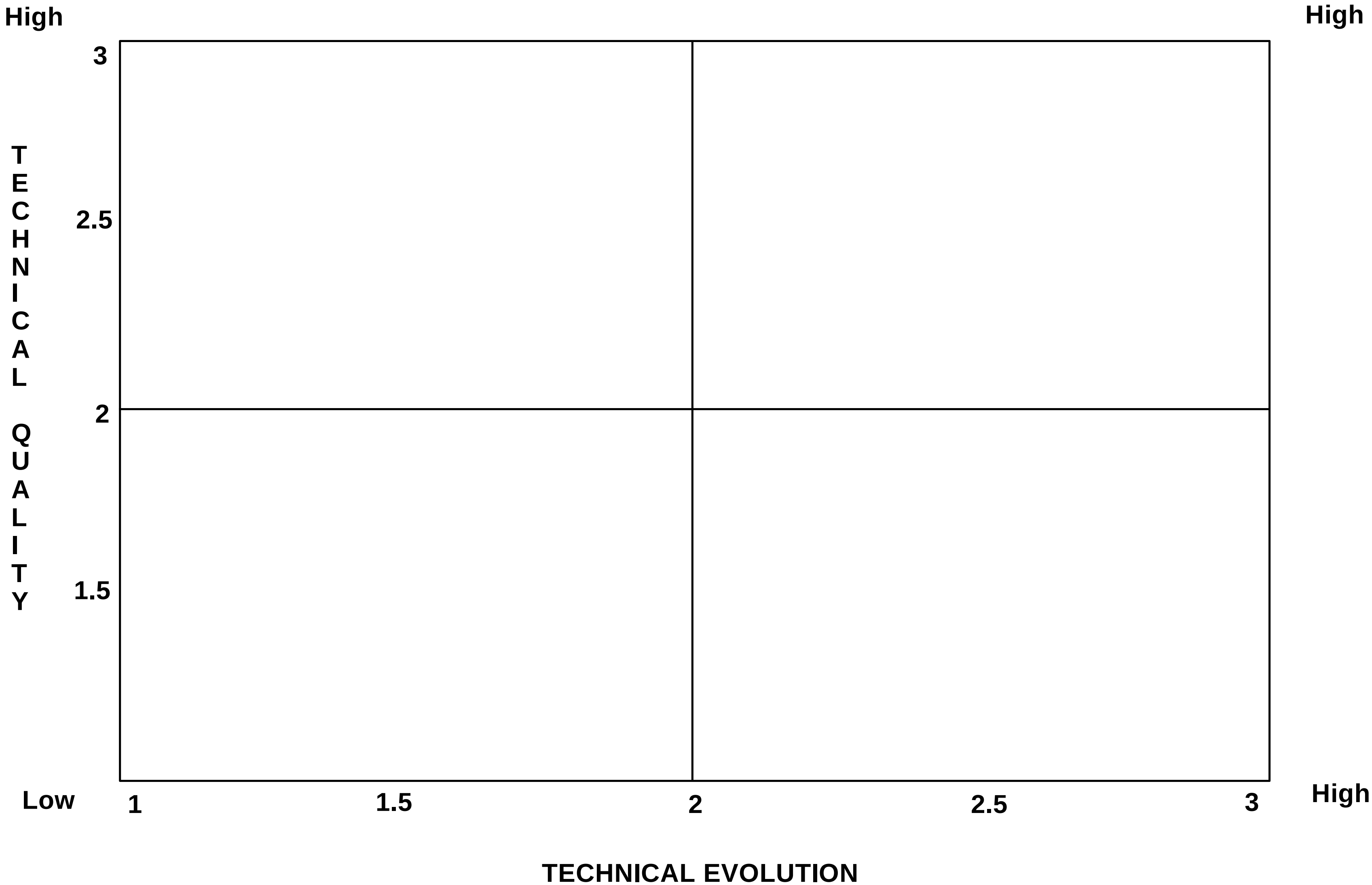
Technical Quality versus Strategic Value



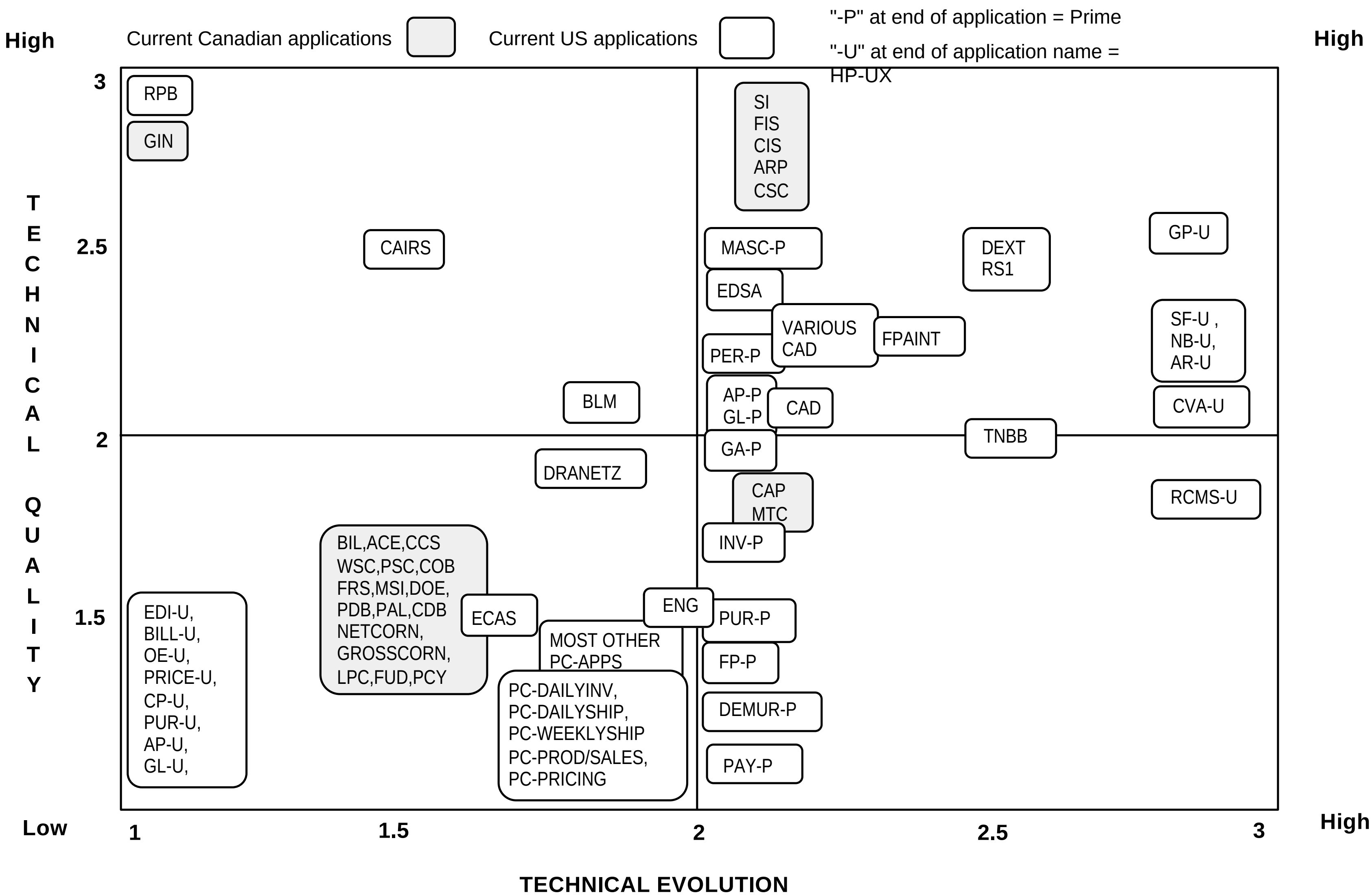
Technical Quality versus Strategic Value



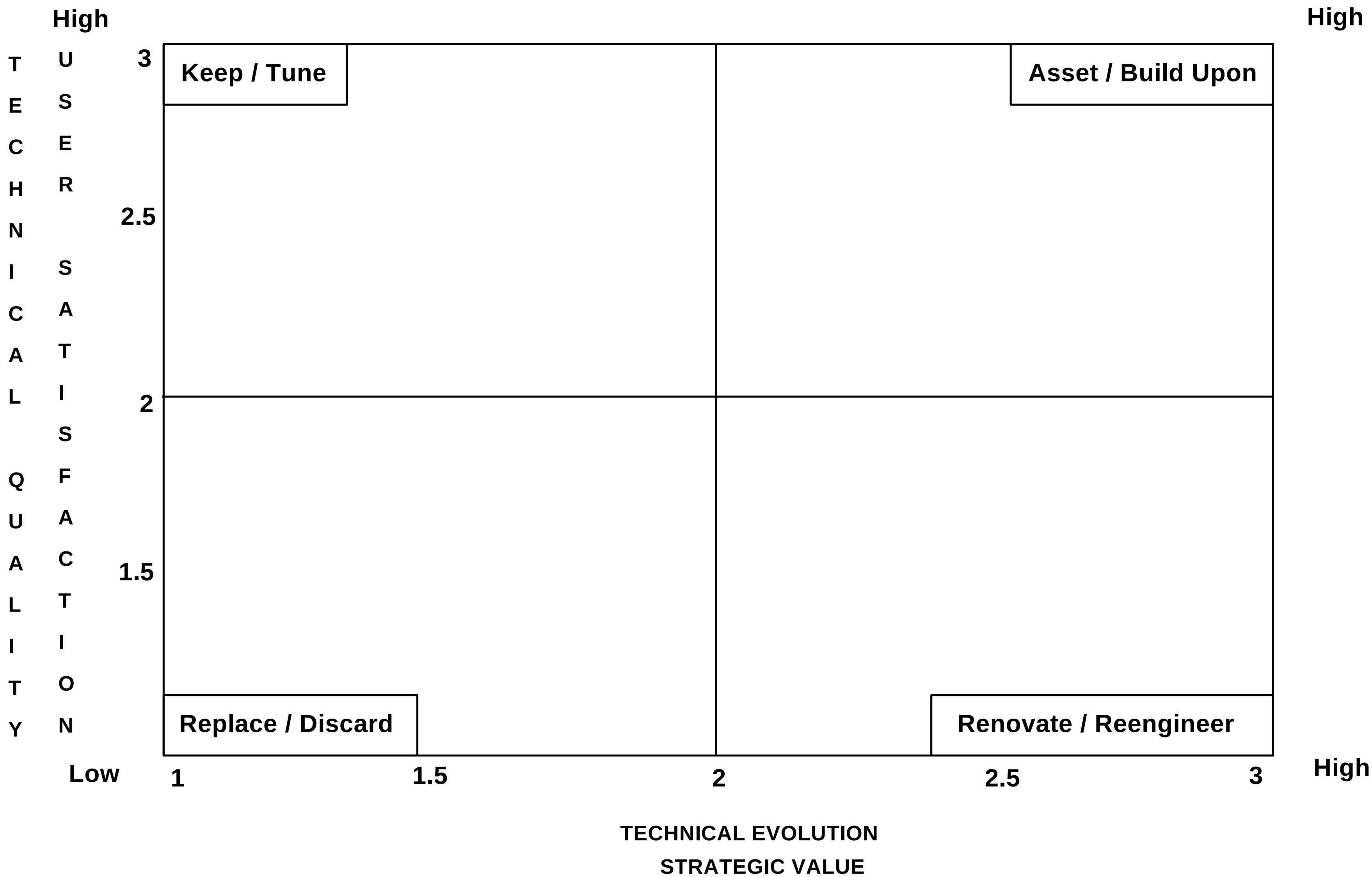
Technical Quality versus Technical Evolution



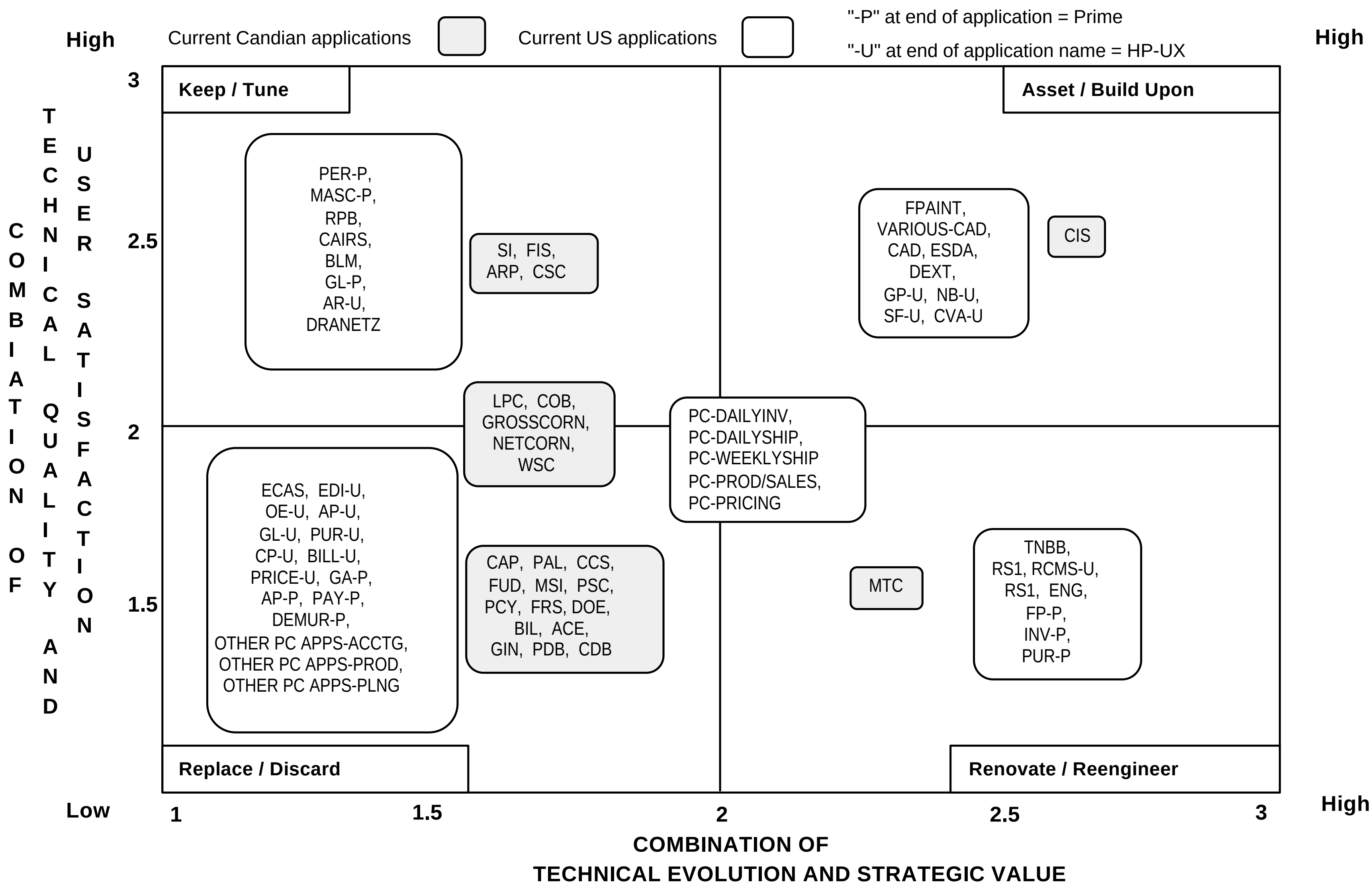
Technical Quality versus Technical Evolution



Summarized Application Assessment



Summarized Application Assessment



Technology Templates

This page intentionally left blank.

Personal Computer Workstation Inventory – Template

Instructions:

Fill in the column headings with all known physical locations where Personal Computers and/or Workstations are in use.
Place a quantity at the intersection of the row and column to depict the number of items in use at the physical location represented by the column.
The following definition of rows apply:

Number of Users	The number of users who use or potentially could use the PCs and/or workstations as this location
IBM Compatibles	The number of IBM PCs or Compatibles by CPU type (i.e. XT,286,386,486,586) in use at this location
Other	The number of any other type of PC or Workstation (one row for each specific Vendor/Model) in use at this location
Total PC's and/or Workstations	The sum of the preceding rows, providing the total number of PCs and/or Workstations in use at this location
Number Connected to LANs	The number of PCs and/or Workstations which are connected to a Local Area Network at this location
Number of LANs	The number of discrete Local Area Networks in use at this location, without regard to whether these LANs are interconnected
Number of LANs Connected to WANs	The number of Local Area Networks at this location which have connectivity to other remote locations via a Wide Area Network
PC and/or Workstation Owner	The Person and/or Organization with the budgetary ownership or responsibility for the PC's and/or Workstations at this location
PC and/or Workstation Manager	The Person and/or Organization with the day-to-day operations responsibility for the PC's and/or Workstations at this location

	Physical Locations											
Inventory Item:												
Number of Users												
IBM Compatibles												
XT												
286												
386												
486												
586												
Others (list Vendor/Models below)												
Total PCs and/or Workstations												
Number Connected to LANs												
Number of LANs												
Number of LANs Connected to WANs												
PC and/or Workstation Owner												
PC and/or Workstation Manager												

Personal Computer and Workstation – Sample

	Physical Locations									
Inventory Item	29 Palms MCAGCC	Camp Lejeune 2nd Div	Camp Lejeune 2nd FSSG	Camp Lejeune CG,MCB	Camp Lejeune II MEF	Camp Pendleton MCTSSA	Camp Pendleton 1st Div	Camp Pendleton 1st FSSG	Camp Pendleton CG,MCB	Camp Pendleton I MEF
Number of Users										
IBM Compatibles										
XT	16	11	40	128		20	95	71	31	14
286	116	626	608	967	216	344	872	718	722	159
386	89	157	118	223	33	204	211	322	238	80
486	8	17	61		4	59	18	70		19
586										
Others (List Vendor/Models Below)										
Apple Macintosh	2			4		10			2	1
Total PCs and/or Workstations	231	811	827	1322	253	637	1196	1181	993	273
Number Connected to LANs	173	608	620	991	189	477	897	885	744	204
Number of LANs	22	17	20	32	12	10	33	52	70	25
Number of LANs Connected to WANs	22	14	16	32	8	10	29	44	70	20
PC and/or Workstation Owner	RJE	G6/ISMO	G6/ISMO	RASC	G6/ISMO	MARCOR	G6/ISMO	G6/ISMO	RASC	G6/ISMO
PC and/or Workstation Manager	RJE	G6/ISMO	G6/ISMO	RASC	G6/ISMO	MARCOR	G6/ISMO	G6/ISMO	RASC	G6/ISMO

Standards to Platform – Template

Instructions:

Describe the specific standards in place for your Information Technology Platforms in the categories depicted by the Generic Technology Platform names in each row.
Fill in the column headings with the same Platform names used on the Technology Platform Inventory Template. This should be a unique identifier for the specific technology platform.
At the intersection of the Generic Technology Platform and the Platform Name enter the specific standard which is in place on that platform.
Where multiple platforms of the same type are used in a similar fashion, list the Platform Name once and the approximate number of units in parentheses (i.e. IBM-PC486 (150 units)).
The following row definitions apply:

User Interface	The standard(s) which controls the presentation of the results of computer system processing to the user (i.e. MS-Windows, 3270, Character-based, etc.
Operating System	The standard(s) which controls the basic operation of the computing platform (i.e. MS-DOS, MVS-XA, DOS/VSE, OS/400, OS/2, UNIX System V, Xenix, etc.)
Communications Management	The standard(s) which controls the connectivity of this platform to others (i.e. Banyan Vines, Novell Netware, SNA, TCP/IP, SMP, SDLC, etc.)
Data Base (and/or file) Management	The standard(s) which controls how the data is managed on the platform (i.e. DOS File System, dBase, Adabas, Oracle, Paradox, DB2, etc.)
Transaction Monitor	The standard(s) which controls the processing of online transactions (i.e. CICS, IMS-DC, TSO, CMS, etc.)
Document Management	The standard(s) which controls document creation, storage and retrieval for the platform (i.e. DISOSS, MS-Word, Word Perfect, Office Vision, etc.)
Distribution Management	The standard(s) which controls the distribution of user messages and/or files within this platform and to other interconnected platforms (i.e. Vines Mail, Quickmail, Office Vision, DISOSS, etc.)
Conferencing Management	The standard(s) which controls the resources of the platform to allow computer conferencing, shared screen conferencing, etc.
Development Services	The standard(s) which controls the environment under which computer programs are developed and tested (i.e. compilers, toolkits, CASE tools, debugging aids, etc.)
Repository Services	The standard(s) which controls the metadata, and data which describes the overall information management environment (i.e. IBM's MVS/Repository Manager, IMS-Data Dictionary, etc.)
Notes	Any additional notes which will help clarify the specific platform standards in place or planned for the near term

	Platform Names					
Standard						
User Interface						
Operating System						
Communications Management						
Data Base (and/or file) Management						
Transaction Monitor						
Document Management						
Distribution Management						
Conferencing Management						
Development Services						
Repository Services						

Notes: ______________________________

Standards to Platform – Sample

	Platform Names					
Standard	286, 386, 486 Micro/PCs	Network Servers (LAN)	AS/400 (mini)	Hewlett Packard 3000 (mini)	Mid-Size Mainframe	Large Mainframe
User Interface	Various Proprietary S/W	Various Proprietary S/W	Proprietary S/W	Proprietary S/W	TSO	TSO/ROSCOE/ CICS
Operating System	MS-DOS	Banyan Vines	OS/400	MPEV - VDelta9	MVS/SP	MVS-XA
Communications Management	Enable/Pro Comm	Vines NOS	IBM 3270 Emulation	NS-3000		VTAM
Data Base (and/or file) Managemen	Enable/dBase IV		SQL Compliant	Turbo Image 3000 (now SQL)		ADABASE
Transaction Monitor						CICS
Document Management	Enable		Office Vision			DISOSS
Distribution Management		Vines Mail Program	Office Vision			DISOSS
Conferencing Management						
Development Services	ADA/Clipper	Vines ToolKit	RPG/Cobol	HP-TRANSACT/Cobol		ADA/Cobol/ Assembler
Repository Services				Dictionary 3000		ADABASE

Notes: Within 6 months all "Mid-Sized Mainframe" systems will be replaced by "IDNX Boxes" & channel extended off the Marine Corps Large Mainframe systems.

Exceptions: RJE Iwakuni JA
RJE Kaneohe HI
RJE CCDH, HQMC Washington DC

Technology Inventory – Template

Instructions:
Generic Technology Platforms are listed in the leftmost column as row identifiers. There are likely to be more than one specific instance of each of these Generic Technology Platforms, i.e. within workstation, we may have Various kinds of terminals (NOTE: PCs and other Intelligent Workstations should not be included on this template. They are covered on the Personal Computer and Workstation Inventory Template). We may have multiple Vendors (i.e. IBM, Teletype Corp, Hitachi, HP, etc.). Within each vendor there could be multiple models (i.e. IBM 3278, IBM 3279, TLX 178-2, etc.) And each of these unique non-intelligent workstations would have an "owner", i.e. the person or organization which has budgetary ownership or responsibility for these workstations.
Enter one line for each unique combination of Platform Name, Vendor Name, Platform Model, and Platform Owner. Use additional pages if you exhaust the blank rows within a Generic Technology Platform section.
Enter each physical location which can contain one or more of these technology platforms as column headings under the broad heading "Quantities by Physical Location".
Enter the quantity of each specific Platform (as depicted by a unique combination of Platform Name, Vendor Name, Platform Model, and Platform Owner) in the intersection with the column depicting the physical location which houses the platform.
Use additional pages if more columns are needed to depict physical locations.
The Following definition of the Generic Technology Platforms apply:

Workstation	Any terminal device which is not a PC or Intelligent Workstation (i.e. "dumb" or single-purpose terminals which have not been included on the Personal Computer and Workstation Template)
Output/Input Peripheral	Platforms such as Laser Printers, Line Printers, Scanners, Card Readers, etc.
Local Area Network (LAN)	The hardware aspects of Local Area Networks such as Token Ring, Ethernet, LocalTalk, etc.
LAN Server	A platform which is used to run the LAN operating system and services. This could be a PC or a specialized server platform of any size.
Wide Area Network (WAN)	A device which provides long haul communications services, such as DISN, DDN, Telenet, Tymnet, etc.
Network Interface Device	A device which provide an interface between the network and the computer processor, such as Front-end processors like IBM 3705 and NCR Comten.
Concentrator, Multiplexer, Switching Device	Devices such as routers, gateways, cluster controllers, etc., which allow the basic network resources to be effectively used for multiple purposes and devices at a various points in time.
Transmission Device	Specialized devices which provide the transmission medium for information transfer, beyond the facilities identified earlier in the WAN category, such as VHF and SHF Radio and/or Satellite.
Storage Device	Devices which allow the storage and retrieval of information, such as Disk Drives, Tape Drives, Microfilm Processors, etc.
Mid-Range Processor	Processors which are larger than workstation or terminal devices both physically and in processing and peripheral controlling capability, but not of the large, mainframe class, i.e. AS/400, 4381, HP9000, etc.
Large Processor	The largest mainframe processing platforms such as IBM's 3090 Model 400E and Hitachi's 5890-600E. This would also include large massively parallel processors and/or vector based supercomputers.
Image Processor	Devices which are specifically devoted to the processing of digital images, such as Kodak's Komstar system.

Identifying Information					**Quantities by Physical Location**												
Technology Platform	**Platform Name**	**Vendor Name**	**Platform Model**	**Platform Owner**													
Workstation																	
Output/Input Peripheral																	
Local Area Network																	
LAN Server																	
Wide Area Network																	
Network Interface																	
Concentrator/ Multiplexer/ Switching																	
Transmission Device																	
Storage Devices																	
Mid-Range Processor																	
Large Processor																	
Image Processor																	

Technology Inventory – Sample

Identifying Information					Quantities by Physical Location										
Technology Platform	**Platform Name**	**Vendor Name**	**Platform Model**	**Platform Owner**	29 Palms MCAGCC	Albany IRMD	Barstow	Blount Island	Camp Foster	Camp Lejeune FASC	Camp Lejeune RASC	Camp Pendleton FASC	Camp Pendleton RASC	Cherry Point RASC	El Toro RASC
Workstation	Terminal	HDS	3472-FC1	C4I, HQMC											
Workstation	Terminal	HP	2392-A	C4I, HQMC											
Output/Input Peripheral	Laser Printer	IBM	3812-2	C4I, HQMC											
Output/Input Peripheral	Laser Printer	Xerox	9700	C4I, HQMC							2				
Local Area Network (LAN)	10 Base-T	10 Base-T		RJE 29 Palms	22										
Local Area Network (LAN)	Ethernet	3Com		1st. FSSG								*N/A			
LAN Server	Banyan FS	IBM	386/486	1st. FSSG								*N/A			
LAN Server	Banyan FS	IBM	386/486	2nd. FSSG						*N/A					
Wide Area Network (WAN)	MCDN	USMC		MCCTA, Quantico	1	1	1	1	1	1	1	1	1	1	1
Network Interface Device	FEP	Comten	NCR 3690-DS	C4I, HQMC			1								
Network Interface Device	FEP	Comten	NCR5675	C4I, HQMC		1					1		1		
Concentrator/ Multiplexer/ Switching Device	Digital Switch (Tactical)	Tactical	AN/TTC-42	MEF											
Concentrator/ Multiplexer/ Switching Device	Digital Switch (Tactical)	Tactical	SB-3865	MEF											
Transmission Device	HF MUX- Radio Central (semi-mobile)	Tactical	AN/TSC-120 (proposed)	MEF											
Transmission Device	VHF-Single Channel-Manpack Radio	Tactical	AN/PRC-68	MEF											
Storage Devices	DASD	Amdahl	62/6880	C4I, HQMC		7									
Storage Devices	DASD	HP	3937	C4I, HQMC											
Mid-Range Processor	Midrange	HDS	EX33	C4I, HQMC											
Mid-Range Processor	Midrange	HP	3000-58	C4I, HQMC											
Large Processor	Mainframe	Amdahl	5890-300E	C4I, HQMC											
Large Processor	Mainframe	Amdahl	5890-600E	C4I, HQMC		1							1		
Image System	Komstar	Kodak	Model IV	C4I, HQMC		1					2		1		
Image System	Komstar	Kodak	Model VI	C4I, HQMC											

Cost Templates

This page intentionally left blank.

Application System Costs – Template

Instructions:

Fill in the column headings with the application acronyms which are represented on the Application Inventory Template. Use additional pages as needed.
The leftmost column provides the cost categories for which we need actual costs or headcount related to each application system.
At the intersection of a cost category and application system enter the costs (or headcount if the category relates to FTEs).
If you share any of these categories with other organizations, allocate your proportionate share of support and place a note at the bottom of the template, explaining the rationale for your allocation.
For example, if an outside contractor supplies a Help Desk with 25 Full Time Equivalents (FTEs) but you only use 4 of the total FTEs, record the 4 FTEs in the Support Service Contractors FTE Headcount.
The following definitions apply to the cost categories:

Hardware	Processors, terminals, PCs, Workstations, Frontend Processors, Disk/Tape Drives, etc.
Software	Online Monitors, DBMS's, Compilers, Report Writers, Operating Systems, etc.
Applications	Commercial packages and custom-developed application systems which provide end-user functionality
Maintenance (Hardware)	Maintenance costs on all of the above
Internal Direct Support	Staff costs directly associated with developing, maintaining and operating the above items
Support Service Contractors	Supplemental staff beyond those listed for in-house staff shown above
Network	Includes owned/leased equipment, Public Switched Network, VAN's and other network services
Internal FTE Headcount	Internal full time equivalent headcount
Support Service Contractors (FTE Headcount)	Full time equivalent headcount for supplemental staff beyond internal staff identified above
Other	Any other item not covered in above categories (please explain with a note)

	Applications				
Cost Categories					
Hardware					
Software					
Applications					
Maintenance (Hardware)					
Internal Direct Support					
Support Service Contractors (Cost)					
Network					
Internal FTE Headcount					
Support Service Contractors (FTE Headcount)					
Other					
Total Costs					

Notes: ______________________________

Application System Costs – Sample

	Applications				
Cost Categories	*SASSY*	*APCS*	*MCFMIS*	*RPM/FHS*	*TCAIMS*
Hardware			1,340K	4,450K	17K
Software		40K	100K	1,800K	
Applications					
Maintenance (Hardware)				170K	
Internal Direct Support	1,650K		760K	70K	1,640K
Support Service Contractors (Cost)	2,400K	40K		1,260K	2,760K
Network					
Internal FTE Headcount					
Support Service Contractors (FTE Headcount)					
Other					
Total Costs	4,050K	80K	2,200K	7,750K	4,417K

Notes: ______________________________

Security Templates

This page intentionally left blank.

Application to Security Level:

- ***Enter all Applications as identified on the Baseline Template for Application Inventory***
- ***Selected Security Classifications are across the top. If these need modification, please mark the headings approprpriately.***
- ***If other security classifications should be tracked, use blank column.***
- ***Enter an "x" in intersection of platform type and column to denote the presence of security at this level for this application.***
- ***If more columns are needed for additional security classifications or rows for additional applications, use a second sheet.***
- ***If there are specific exceptions or notes for clarification, enter the information in the space below the matrix.***

<u>This template should apply to all applications in the entire enterprise</u>.

Security Classification / ***Application***	Top Secret	Secret	Confidential	Unclassified Sensistive	Unclassified		

Additional Notes and Clarifications:

Location to Security Level:

- ***Enter all Physical Locations as identified on the Baseline Template for Business Functions and Work Organization***
- ***Selected Security Classifications are across the top. If these need modification, please mark the headings approprpriately.***
- ***If other security classifications should be tracked, use blank column.***
- ***Enter an "x" in intersection of platform type and column to denote the presence of security at this level for this work location.***
- ***If more columns are needed for additional security classifications or rows for additional locations, use a second sheet.***
- ***If there are specific exceptions or notes for clarification, enter the information in the space below the matrix.***

This template should apply to all locations in the entire enterprise.

Security Classification / *Location*	Top Secret	Secret	Confidential	Unclassified Sensistive	Unclassified		

Additional Notes and Clarifications: ______________________________

Generic Technology Platforms to Security Level:

- ***Locations are listed in the left column, one per row.***
- ***Generic Technology Platforms are across the top***
- ***If another security classification should be noted, add it to the key and to the appropriate column(s)***
- ***Enter all applicable security classifications at the intersection of each row and column***
- ***If more rows are needed for additional locations, use a second sheet.***
- ***If there are specific exceptions or notes for clarification, enter the information in the space below the matrix.***

<u>This template should apply to all platforms in the entire enterprise.</u>

Technology Platforms / *Locations*	Workstation	Output/Input Device	Local Area Network (LAN)	LAN Server	Wide Area Network	Concentrator /Multiplexer /Switching	Storage Devices	Mid Range Processors	Large Processors

Additional Notes and Clarifications:

Security Classification Key:

TS = Top Secret
S = Secret
C = Confidential
US = Unclassified Sensitve
U = Unclassified

Process to Security Level:

- ***Enter all work processes as identified on the Baseline Template for Business Functions and Work Organization***
- ***Selected Security Classifications are across the top. If these need modification, please mark the headings approprpriately.***
- ***If other security classifications should be tracked, use blank column.***
- ***Enter an "x" in intersection of platform type and column to denote the presence of security at this level for the activities and outputs of this work process.***
- ***If more columns are needed for additional security classifications or rows for additional processes, use a second sheet.***
- ***If there are specific exceptions or notes for clarification, enter the information in the space below the matrix.***

<u>This template should apply to all processes in the entire enterprise.</u>

Security Classification / *Work Process*	Top Secret	Secret	Confidential	Unclassified Sensistive	Unclassified	Location	Misc.

Additional Notes and Clarifications:

Questions of Interest and Rules of Thumb for Baseline Analysis

This page intentionally left blank.

Connectivity characteristics

- What is the relationship of the current platform to other target platforms? Is there a client/server relationship in place? If so, detail the associated platform environments and describe the client/server relationship.
- What are the characteristics of the logical links that the platform under review has to other platforms with which it is linked. Are the links based on peer-to-peer relationships such as LU 6.2? From a terminal interface perspective, how does the terminal view the platform linkage?
- What are the characteristics of the physical links that the platform under review has to other platforms with which it is linked? Are the links batch or interactive in nature? Interactive token ring or dial-up?
- What are the current characteristics of the existing platform with regard to the capacity of the platform under review—the number of transactions supported, effective throughput per period of time, bandwidth required, etc.?
- What are the problems and opportunities related to current connectivity attributes? Do they enable or inhibit platform performance? How do they relate to application, technology, or cost performance?

Standards support

What standards does the existing platform support and which of the following services?

- User interface services
- Database services
- Operating system services
- Communication services
- Management services
- Languages
- Applications.

- What interface standards does this platform support for all of the above (e.g. UNIX operating system support for POSIX P 1003.1 for operating system interface standard)?

- What effect does the current suite of standards have upon IT objectives? Are they enabling or hindering growth and attainment of IT objectives?

- What are the costs of using these standards? To what degree are our current standards "open"?

- To what degree are our standards vendor independent?

- What degrees of freedom do we have within our current standards as implemented in existing products or services?

- What is the nature of the standards supported? Are they proprietary or open?

- Are they de facto or de jure standards?

- How stable are they? Have they been in place for 6 to 24 months?

- Are they "developing" standards? If so, is the future standards path for this platform clear?

- To what degree does the existing platform either promote or inhibit portability, scalability, or interoperability?

Generic application environment support

- What generic application environments are currently supported on the existing platform?

- What are the logical linkages between existing GAEs?

- What are the problems and opportunities related to current application environment attributes—application delivery, technology, or cost-related issues?

Generic technology environment

- What are the existing generic technology components that make up the existing environment?

- What kinds of services are being supported by GTE?

- What classes of users are using which types of GTEs?

- What set of services is supported by the existing technology platform under baseline review?

- What are the problems and opportunities related to current generic technology attributes—application, technology, or cost-related? How mature are these environments? Where will these new applications go in the future?

GAE and GTE relationship: logical and physical connectivity

One of the key aspects of platform attributes is how individual platforms work together to “plug and play” in an overall architecture. In a function, every platform has a relationship with all other platforms in the function, even if they are standalone by nature. A method is needed to characterize these logical and physical relationships as well as their attendant costs in a simple visual manner. This is the essence of an architecture.

The problem in most functions is that the various platform attributes have not been decoupled from the technology itself and therefore do not lend themselves to a logical characterization for architecture planning and analysis. By examining each of the platform attributes on a logical as well as physical basis, we may develop an overall picture of how various platforms relate to one another across a function.

The following matrix demonstrates a typical three-tiered architecture in a hypothetical function composed of LAN-based work group computing, mid-range computers, and traditional mainframe access. Each one of the points on the matrix (dark dots) may be thought of as the logical connectivity point between the two platforms. By examining the individual attributes of each of the two platforms connected by these two points, one may examine the nature of platform connectivity. Indeed, using such a matrix, one may characterize a number of attributes in a baseline architecture:

- How GAEs are related between business units across a function.

- How GTEs are related between business units across a function.
- What standards are in place across a function or department.
- The physical or logical connectivity characteristics across a function as well as the relationship one processing environment has with another. For example, the “client/server” model may be illustrated in this manner.
- The relative cost of a platform or set of platforms in a function or department.

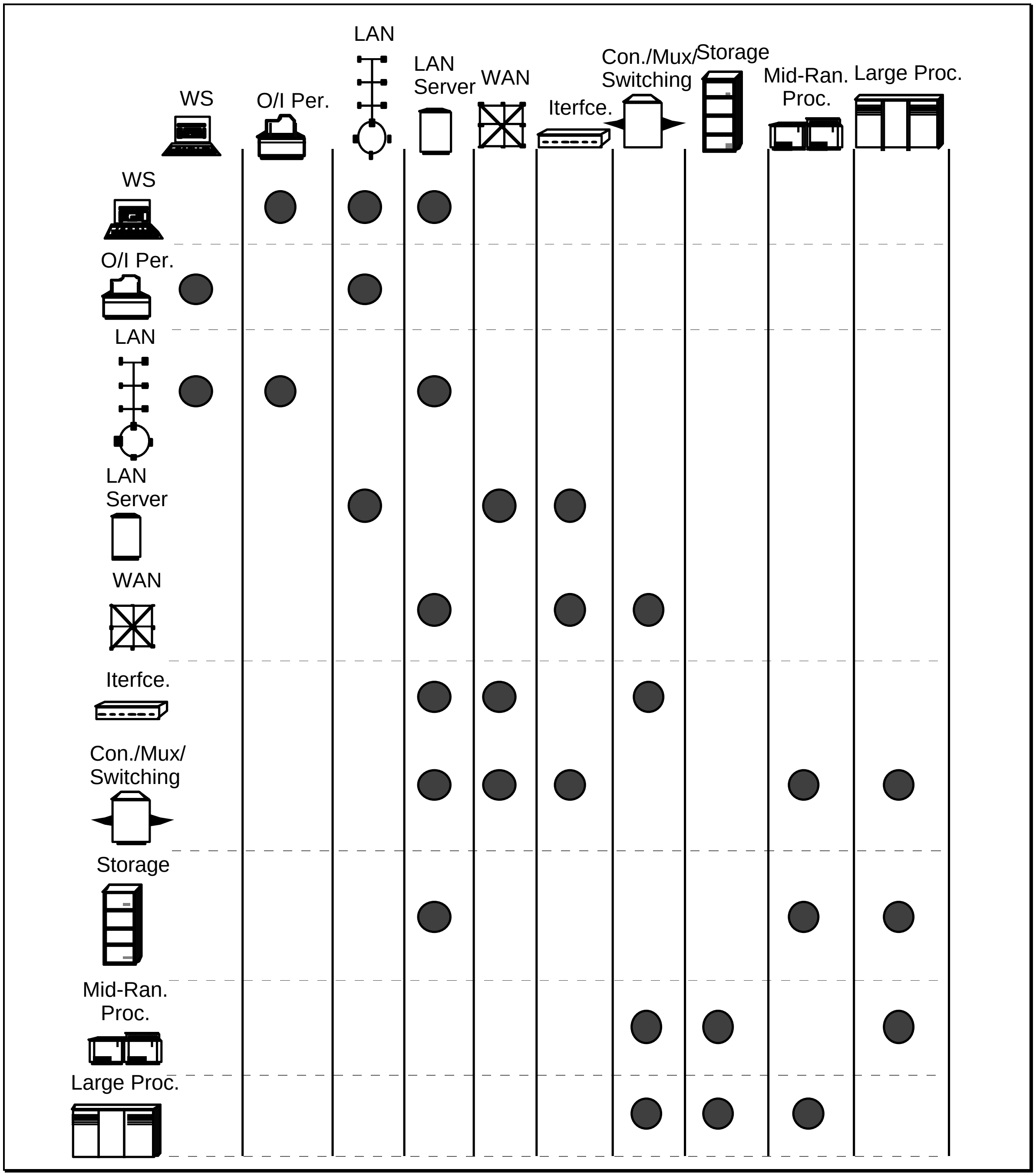


Figure B-1
Physical or Logical Connectivity

Platform cost data

- **Direct hardware costs**—purchase cost, depreciation or lease.
- **Direct operating system software costs**—purchase, cost, depreciation, or lease.

- **Maintenance and service costs**—recurring operational cases.
- **Personnel costs—**direct and indirect for both hardware and software.
- **Training costs—**direct and indirect for hardware and software.
- **Application costs—**direct and indirect including software licensing and maintenance as well as other “intangibles” related to work flow, business procedures, inventory turn rates, management “time value,” and general productivity. (This last category will vary enormously depending upon how the application costs are quantified.)

Typically, cost data are the most difficult types of information to collect in an architecture assignment. There are many reasons for this including the fact that little useful cost information is kept in the first place. Good cost data can be very helpful, especially in justifying the migration plan later on. When cost data are available, they should be collected by the team to incorporate in the baseline characterization and for use in migration planning in later stages of the project. (see Appendix F for a full discussion of the business case cost analysis.)

Security evaluation criteria

A key aspect of the baseline includes providing a classification of the application and technology platforms using the criteria and classification scheme described in the *U.S. Department of Defense Trusted Computer System Evaluation Criteria* [DOD 5200.28 STD, December 1985]. The appendix includes an entire section of security planning considerations.

Rules of thumb for the baseline characterization

Once the matrices have all been completed, the analysis phase can begin in earnest. A substantial amount of effort must go into the analysis and interpretation of the baseline characterization to provide a solid platform for identifying the projects that will be required to move the enterprise from the baseline to the target architecture. The assessment of applications is of particular interest, since this provides the guidance that will be instrumental in improving the effectiveness of the enterprise. An example of application assessment is provided below in light of the significant role applications play in any architecture.

Baseline application assessment rules of thumb

To prioritize future application opportunities, an assessment of existing application systems is needed. In this way, the existing applications and their associated assessment can be mapped to the target application opportunities. For example, if an envisioned target application is of high strategic significance and the existing applications which provide equivalent functionality are assessed as being in need of replacement, the target application would be a high priority initiative in the migration and implementation planning phases. If there is no existing application and the other conditions described above for the target application were the same, the target initiative would be at an even higher priority.

This kind of analysis must be done for each target and existing application in the enterprise. To make this work, a high-level assessment of the existing applications is needed. The following provides some criteria that can be used for this process.

Templates should be provided to representatives of each functional area affected by the architecture and supporting IT staff, and should be completed with their assessment of any existing application systems with which they were familiar. Emphasis should be placed on doing them relatively quickly, with a reasonable subset of the user/developer community providing input. The intention is to perform the assessment only at a macro level for overall trends and conclusions. An example of such a template is shown below.

Categorizations

The recipients should be asked to categorize (high, medium, and low within each category) each application according to the following criteria:

- User satisfaction
- Technical quality
- Strategic value
- Technical evolution.

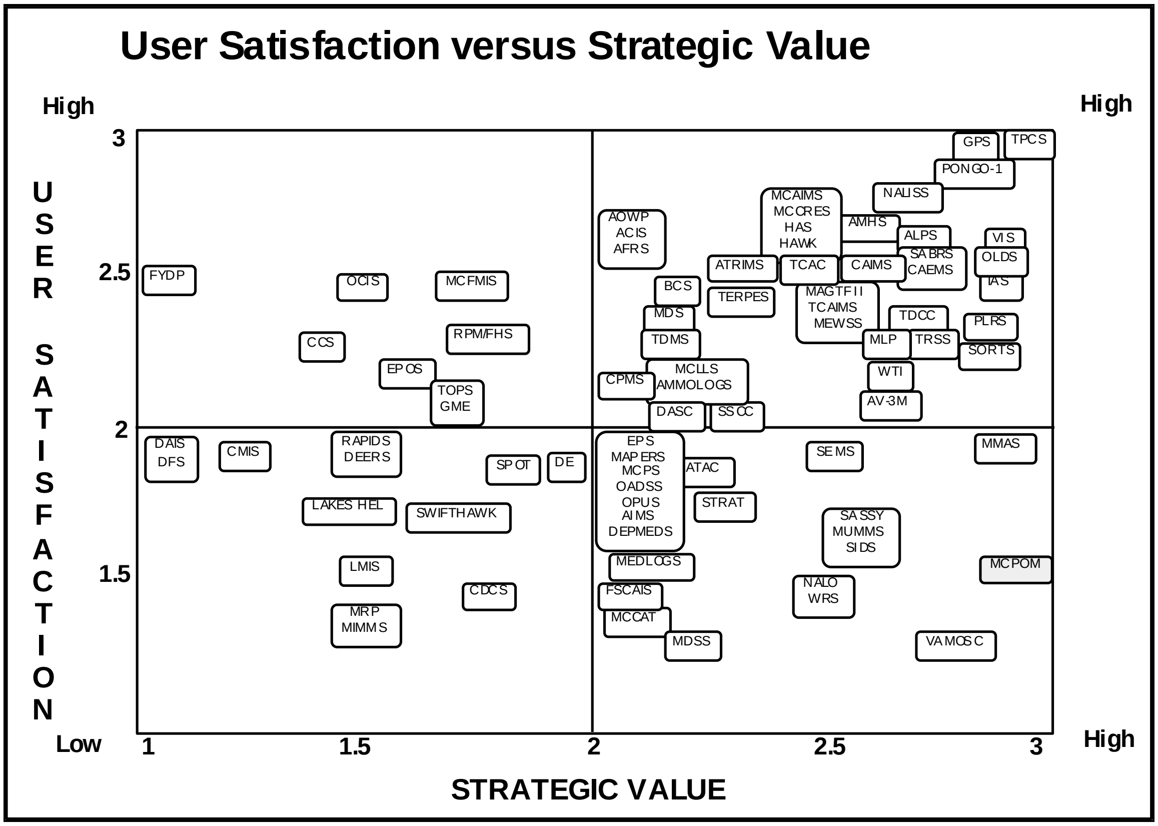


Figure B-2
Template for User Satisfaction Versus Strategic Value of Existing Applications

User satisfaction — User satisfaction is self explanatory. For the other categories, the following definitions should be provided:

Technical quality — This assessment criterion measures the application's robustness and maintainability. It is a measure of whether the application is well written with easy-to-follow, structured code and sufficient program comments to facilitate enhancements or maintenance. A high technical quality application will have data definitions (or other frequently changed items) included in the code as tables or copy members rather than hard coded within the programs. Similar logic will be coded once and referenced in other sections of the program or application rather than physically replicating. In general, a high level of technical quality would be an application that already follows the principles of common interfaces and consistent definitions that forward-thinking organizations usually adopt during the architecture framework phase of the SBA.

Strategic value This assessment criterion measures the application's importance in achieving strategic objectives. This should be assessed by users in the context of the strategic drivers as defined in the business context phase of the project.

Upon receipt of the above assessments and after the target architecture is developed, a fourth criterion for assessment should also be applied to the existing application portfolio–that of technical evolution.

Technical evolution This assessment criterion measures the application's positioning to evolve effectively into the target architecture and to take advantage of envisioned advances in information technology. For example, an application that is written for a hardware environment and language that will become part of the target technology architecture would normally have a higher technology evolution rating than one that is written for an environment that will not be carried forward into the target environment. Likewise, an application that is written in a "portable" language has a higher evolution rating.

Based on the analysis of this data, a summarized assessment can then be developed. These criteria should now be mapped in the following pairs on four-quadrant matrices to allow a high-level determination of the recommended disposition of each application:

- User satisfaction versus Strategic Value
- Technical Quality versus Strategic Value
- Technical Quality versus Technical Evolution.

The combination of this information can be used to generate a summary assessment similar to the example presented below.

For this summary matrix, each application has been classified by placing it in the quadrant that most appropriately represents the combined classifications, also taking into account the likely recommended disposition of existing hardware and system software platforms that currently support the applications (i.e., the likely "technical evolutionary" capability of the hardware/systems software platforms themselves).

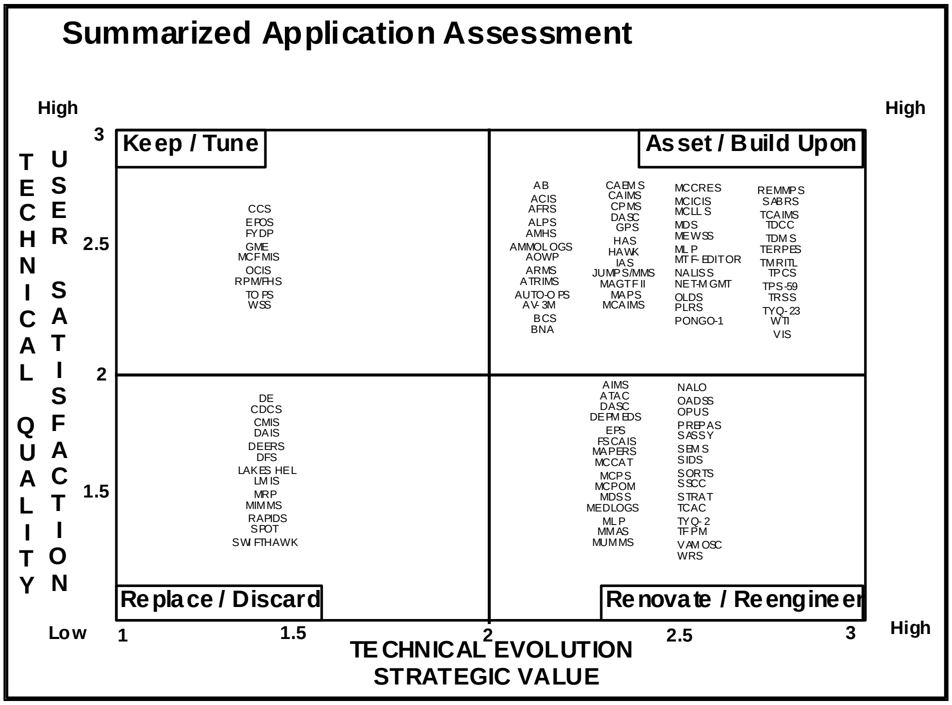


Figure B-3
Sample Summarized Application Assessment

The individual source matrices are typically completed by both users and IS staff within multiple functional areas with input from the client core AWG.

For the summary matrix, each application is classified in a quadrant that most appropriately represents the combined classifications from the source matrices. The following provides a bit more detail on each of these assessment quadrants.

Replace or discard

The application has low user satisfaction, technical quality, technical evolution, and strategic value. If the application is absolutely necessary to the business, it should be completely replaced with a newly developed application or purchased package.

Renovate/reengineer

The application has low user satisfaction and technical quality but high strategic value and operates in a technical environment that can evolve into the target architecture relatively effectively. The application might be given a revamped user interface for improved usability or maybe the programs can be restructured for better reliability and maintainability, perhaps utilizing some reverse engineering tools.

Keep/tune

The application has high user satisfaction and high technical quality but low strategic value and is written in an environment that will be difficult to evolve into the target architecture. Because the application is technically sound and the users seem satisfied currently, keep the application as is for now doing minimal tuning and maintenance to keep it running.

As it reaches the end of its normal life cycle, or as other applications in the new environment have an increasing need to integrate with this application, it may have to be replaced. However, because it is stable and has low strategic value, it should be one of the last applications to be redeveloped or replaced.

Asset/build upon

The application has high user satisfaction, technical quality, and strategic value, and it operates in an environment that can evolve into the target architecture relatively effectively. It should be retained as one of the core applications upon which to build. Applications that fall into the other three categories above should begin to migrate into this category as they are redeveloped, replaced, or converted over the agreed-upon architecture implementation interval.

Rules of thumb

For this summary matrix, each application has been classified by placing it in the quadrant that most appropriately represents the combined classifications, also taking into account the likely recommended disposition of existing hardware and system software platforms that currently support the applications (i.e., the likely "technical evolutionary" capability of the hardware/systems software platforms themselves).

Because these ratings on the source matrices occur independently, there is the potential for a given application to fall in different quadrants on each of the three source matrices (User Satisfaction versus Strategic Value, Technical Quality versus Strategic Value, Technical Quality versus Technical Evolution).

The following rules of thumb should be used in arriving at the summary assessment when independent sources place a given application in more than one quadrant or when the definitions of the quadrants themselves are insufficient to make the determination:

- The strategic value rating on the User Satisfaction versus Strategic Value matrix should be used because this source matrix is completed by the user community (rather than IS support staff). The assumption is that the end user has the best feel for the value of the application to the operational area it supports.

- If an application has low technical quality and low strategic value and low technical evolution combined with high user satisfaction, the application is placed in the “keep/tune” quadrant (i.e., the high user satisfaction and low strategic value combination move the application to the keep/tune rather than replace/discard). The user is happy and it is not a strategic application so, for the moment, keep it going with minimal investment. It will be one of the later applications to be replaced in the new environment.

- If an application has high user satisfaction and high technical quality but low technical evolution and low strategic value, it should be placed in the replace/discard quadrant.

- If an application has low user satisfaction and low technical evolution but high strategic value and high technical quality, it should also be placed in the replace/discard quadrant.

- If an application has high user satisfaction and high strategic value but low technical evolution and low technical quality, it should be placed in the keep/tune quadrant. Over the long term it will need to be replaced because of the low evolution and quality ratings; however, it should not be replaced right away because the users like what they have and it is important to the operation.

- If an application has low user satisfaction but rates high on technical quality, technical evolution, and strategic value, it should be placed in the renovate/reengineer quadrant. The basic application is probably reasonable as a building block, but perhaps it lacks critical functionality or the user interface may be cumbersome. A facelift may be all that is needed to increase user satisfaction.

- If an application rates high on user satisfaction, technical evolution, and strategic value but is rated low on technical quality, it should also be placed in the renovate/reengineer quadrant. The low technical quality is probably of the sort that is not visible to the user, such as unexpected crashes or incorrect data returned. If it were, the user satisfaction would probably not be high.   The reason for low technical quality ratings in this case are probably due to difficulty in maintaining, debugging, and enhancing these systems due to poorly structured programs. Given the high user satisfaction, this is an application that should be reengineered internally for more efficient maintainability and execution while maintaining the look and feel it has today.

- If an application is rated low on user satisfaction and technical evolution but high on technical quality and strategic value, it should be placed in the replace/discard quadrant. Regardless of how high the technical quality is in the current environment, if the environment will not be carried forward into the target, the ultimate fate of this application is to be replaced or discarded, depending on the strategic value. In this case, where the application is strategic, the choice will be to replace it with an application that functions in the target technical environment.

- If an application has low user satisfaction, technical quality, and technical evolution but has high strategic value to the enterprise, it should be placed in the replace/discard quadrant. As in the case above, the lack of evolution capability alone is enough to place it in this category. This, coupled with low technical quality in the current environment, provides two compelling reasons to build a new application to support this strategically important functionality.

- Finally, some judgment calls will need to be made where applications end up near the borderline of two or more quadrants. In these cases, low technical evolution ratings should generally pull applications having high strategic value into the replace/discard quadrant.

Appendix C: Detailing the Target Architecture

Most efforts at detailing a target architecture tend to settle on a three-tier model of computing. Each of these tiers is detailed below.

Enterprise tier

It is envisioned that some systems will support virtually all functional areas. In fact, these systems will have enterprise-wide impact through the data they capture and make accessible to users. They will reside at a minimum number of locations (usually only one, but certainly only one within each major area of operation).

These enterprise-wide systems support operations that are common to all work groups. Also, the kind of activities supported do not typically require split-second response times and real-time currency of information, although this may be desirable. The key aspect of systems falling into this classification is that they typically process large volumes of information, and this information needs to be accessed in a consistent way by many users who are usually geographically and organizationally dispersed.

The technology architecture will provide for computer processing to support the enterprise-level systems in a central location(s) with appropriate disaster recovery and security capabilities. All users will be able to access these facilities via network connections.

These systems will probably be positioned to run on high-capacity processors (depending on data volume, response time requirements, etc.). The final decision will be made when the detailed design of the specific application system is undertaken.

Work group

A work group is composed of individuals who share common requirements and needs for information access to perform their function. There are typically multiple work groups within the organization. They typically have a need for quick access to current, function-specific information.

The technology architecture will provide for computer processing in close proximity to the work group to support these quick access requirements. Work group level systems will be deployed to physical locations that support a critical mass of individuals within a work group. Therefore, work group systems may be replicated over the network computing environment.

Within the work group classification will be multiple specialized, but interconnected servers, specifically application servers, communications servers, data servers, etc. These processors will support each individual workstation's need for access to work group data and connectivity to other servers beyond the immediate work group, as well as the enterprise processor(s) that house the enterprise applications.

Individual

The individual level of architecture is the individual worker equipped with a computing platform that is networked to allow access to work group and enterprise facilities. Application systems deployed at the individual level fall into two categories: supporting "tools," such as E-mail, word processing, spreadsheets, and calendaring/scheduling tools; and the "client" portion of work group or enterprise systems, which allow access to data and services that reside on enterprise and work group computing levels. These types of systems may be made available on an individual's workstation to allow maximum customization and autonomy while allowing continued connectivity to other work group and enterprise systems via the network.

Within the work group and individual levels there are further classifications:

Transportable

This is the case of the work group level of computing where one or more work groups are physically transported to a temporary base operation but, once there, they remain fixed for a period of time. An example would be a deployed medical treatment facility in temporary quarters.

Mobile

This is a special case of the work group and individual levels of computing where one or more work groups and individuals are "on the move" and require access to individual, work group, and enterprise computing resources while mobile.

Each level of computing has some unique characteristics in terms of the topology and the components needed to make up the total computing environment at that level. The following is a graphic depiction of each level and how it will interoperate.

To support the location profiles discussed above, the technology architecture has taken the form of a three-tiered network computing environment. This environment provides maximum flexibility to support both common and unique local applications while providing the connectivity required for information sharing. This architecture also provides a measure of local control over systems operation for the various work group locations by allowing critical applications to be co-located with the local work group staff. The three tiers of the target network computing environment are:

Local area networks

A local area networks (LAN) provides terminal and/or workstation access to individual, work group, or enterprise computing resources as well as file sharing and peripheral sharing. The LAN also provides communication with members of the local work group via electronic mail, local office automation tools, and simple localized application systems, which run either on the workstations themselves or on LAN-based processors (referred to as "LAN servers"). The LAN will always provide the link to other network components that, in turn, will link to computer processors. There are exceptions only in cases of deployed mobile computing at the individual level where LAN connectivity is not feasible.

Campus area networks

A campus area networks (CAN) interconnects LANs within a physical work location (or "campus"). Each major fixed physical location will have a single CAN as a "backbone." These fixed physical locations would typically be in CONUS or host facilities that have been provided for long-term usage. CANs will support higher speeds than LANs for rapid message and file transfer between loosely coupled applications that run on multiple work group processors (work group servers) at a physical location or that run on LAN servers as described above.

Workstations and/or terminals will never directly connect to the CAN. These devices will gain access to the CAN only via their LAN connection.

Wide area network

The third tier of the network provides connectivity to a wide area network (WAN) that interconnects all physical locations. The WAN may be a combination of privately owned network facilities including, but not limited to, radio, satellite, and cable; leased lines; and public network services such as Electronic Data Interchange (EDI), packet switching, frame relay, etc. from Value-Added Network (VAN) suppliers. The WAN provides the high-speed, long-haul communications links to interconnect the dispersed physical locations. The WAN provides the capability for applications running on LANs and work group processors on CANs to communicate with remote site applications via message and file transfer or, if necessary, in a more tightly coupled, interactive fashion. The WAN connectivity also allows access to applications that run centrally on an enterprise processor.

Connectivity options

At enterprise and work group locations, LANs will not connect directly to a WAN; instead, they will gain access to a WAN through their connection to a CAN. Workstations and terminals likewise will not connect directly to a CAN; instead, they will gain access to the CAN via their connections to a LAN. Work group processors and enterprise processors will connect into a CAN as well. This allows all workstations and terminals to gain access to all processors via a standard set of network connections.

For the cases of deployed mobile locations, connectivity into the network of computer processors will come through the use of wireless data transmission via a range of wireless technology including, but not limited to, microwave and satellite capability. Depending on the situation, a "traditional" cable-based LAN may be deployed that will be interconnected to the larger community, or an individual computing platform may use wireless LAN technology or individual wireless capability to achieve connectivity directly without a LAN. Anytime such wireless capabilities are used, care must be taken to deal with the issue of security and the possible requirement of not revealing the location of the installation to hostile parties.

Why this computing approach?

This network computing approach with distributed applications and data minimizes the impact of network or processor failure on the enterprise (i.e., the failure of one part of the network), or a local computer will not bring all work group systems down. Also, backup and recovery of the work group that has had the failure can be accomplished by switching it, with minimal disruption, to one or more of the other distributed platforms, if the network connectivity remains intact.

The network computing approach also provides the infrastructure and connectivity required to easily support common services such as E-mail and EDI. These common services have been defined as a required part of many application systems of the future and will be a key enabler to effective information capture and sharing.

Finally, the network computing approach will enable the organization to take advantage of emerging “groupware” packages that allow common work activities to be more effectively automated. Common office automation tools, such as word processing, calendaring, and business graphics, are all more effectively implemented and managed in an environment where connectivity is assured. These work group and individual productivity tools fit naturally on LAN-based platforms. A measure of standardization on these tools and platforms will be necessary for the organization to reap the productivity and effectiveness gains it seeks in the coming years. The network environment will facilitate this process.

Questions to consider when detailing the architecture

There are a number of questions the AWG should ask itself when working through standards at each layer of the DoD reference model. While this is not intended to be an all-inclusive list, here are some questions that a work team can begin to ask when developing the target architecture:

- What opportunities exist for application and technology environment portability within our existing baseline architecture?
- Which of our existing standards meet these functionality requirements?

- What is the impact of DoD standard systems on my functional area architecture?

- What gaps do we have in our standards? Which ones are needed but do not exist? Which ones exist but haven't been implemented in our organization? Why?

- What advantages could be derived through making our current architecture more "portable," more "scalable," or capable of a higher degree of interoperability?

- What kind of benefits are these—cost savings or "value-added" (such as rapid response to wartime situations)?

- What are the "diversity costs" for operating multiple environments across the Logical Operating Unit (LOU)?

- What payoff does standards implementation afford our organization? When and where? What is the business case?

- What is the impact of Federally mandated standards on my functional area's architecture?

- Should we build standards within specific vertical applications, or should we integrate them within specific technology platforms across the organization (e.g., implement standards within a customer service application versus a specific platform area, such as user interface, across the LOU)?

- How much of the existing embedded "legacy" system(s) do we keep? What needs to be replaced? What is the IT and business case for either solution?

- Can we implement these standards? Is the plan realistic? When will we achieve results? What time frame considerations merit review?

Key questions

In addition to the general standards questions, there are specific standards issues to be addressed at each level of the standards-based model. The following questions are presented solely as guidelines. These question sets should be extended by the AWG in every area relevant to the enterprise's architecture.

User interface

- What are the user requirements for user interface (UI) across the functional area(s) with which I am working?

- How global a UI do I want to put in place? Do I want one or several?

- What UI standards do I want to adopt—the X/Windows model, a proprietary-based implementation, or both?

- Will the UI(s) run on proprietary and "open system" workstations? Will they run on both POSIX-based and non-POSIX–based workstations?

- What UI standards does my existing environment support? Can I migrate my current UI environment into a common standards-based set of UIs? How global a UI standard do I want?

- What is the "diversity cost" of the set of UIs in place? Is there an opportunity to eliminate and simplify?

- Which de facto or de jure standards in this area can I make use of now? How standards-compliant are my options?

- What role will my UI play in an overall client/server strategy or cooperative processing architecture?

- If I am not conducting true multiuser/multitasking interactive applications, what is the value of implementing a common UI?

- Is there a suite of "seamless" UI "shrink-wrapped" (commercially available) software available for these "standalone" workstation applications? Neither solution offers the advantages of a true proprietary (VAX, OS/2) or open system-like (POSIX) multitasking environment.

- Does this UI support true multitasking/multiuser work group requirements, or does it really only provide "task switching"?

- Do the UI products I am evaluating support the target platforms I am designing? How do they handle application binary interfaces?

- What will the total cost for my UI strategy be, including costs to upgrade workstations, LAN wiring, implementation, and retraining?

Database

- What are the user requirements for database across the functional area(s) with which I am working?

- What is the "diversity cost" of this set of various databases? Is there an opportunity to eliminate and simplify?

- Which of the de facto or de jure standards in this area can I make use of now? How standards-compliant are my options?

- What is my overall database strategy and architecture? What is the outlook for relational database proliferation? How will functional area(s) handle database related activities in the future?

- To what degree is my current database architecture SQL compliant? To what degree should my target architecture be SQL compliant?

Applications

- What is the scope, depth, and number of the application portfolio across the functional area(s) with which I am working?

- What is the *"diversity cost"* of this set of various applications? Is there an opportunity to eliminate and simplify?

- What will I do with systems that are currently under development but may not be "playing by the standards" I am proposing?

- Which of the existing applications described in the baseline effort should be considered candidates for:

  - Porting to a new open systems environment
  - Recoding into a new environment
  - Redesigning into a new environment
  - Starting from scratch
  - Killing and eliminating?

- How will the existing applications support our target GAE and GTE requirements? How will they coexist with new applications if they are not going to be replaced? What is their life cycle?

- Which applications are redundant? Which of the target applications could be modularly “reused”? Does the code permit reusability?
- What standard programming languages does the application support? Do these languages map to my language standards strategy?
- Are new target applications available in “shrink-wrap” form? What platforms do they currently support? How does my existing and target architecture support these products today *and* tomorrow?
- Are there de facto or de jure standards in this area I can use now? How standards compliant will my target option be?
- What target tools will I use in the conversion process if conversions are deemed necessary? What CASE-tool standards should I use? Do they support evolving CASE standards?
- Does my existing vendor offer porting services for the target application suite in the GAEs and GTEs I am designing? How will I handle conversion if they do not?
- How will new target applications support interfaces to databases, user interfaces, languages, operating systems, communications, management services, and other services? Are there application portable interfaces for these applications in place? What are they? Are they compliant? Are they X/Open XPG compliant?
- What will be the total cost for my application strategy, including costs to migrate and retrain? What are the costs and risks associated with migration?

Languages

- What is the scope, depth, and number of languages in the language portfolio across the functional area(s) with which I am working?
- Have I complied with the DoD policy regarding the use of ADA?
- What is the “diversity cost” of this set of various languages? Is there an opportunity to eliminate and simplify?

- Are there de facto or de jure standards in this area I can make use of now? How standards compliant will my target option be?

- Do I have the professional set of employee resources to sustain and support the existing language requirements? Do I have the right resources to support a new target set of languages?

- How portable is the language(s) and what binary conversion capabilities does it possess? What "degree of freedom" do I have with my existing language portfolio suite?

- What applications and other system components will my existing languages support (e.g., applications, databases, operating systems, user interfaces, communications, management, and other services)? Which one should be:

  - Used in the future?
  - Slowly phased out?
  - Used only for system maintenance?
  - Totally eliminated as quickly as possible?
  - Acquired because we do not have them now but will need in either the short or long term?

Operating system

- What is the "diversity cost" of this set of various operating systems?

- What is the smallest number of languages that I can standardize on today? How many of the languages currently in place do I want to retain in the future?

- Of the languages currently in place, how many of the ANSI-compliant languages have proprietary extensions to them which effectively render them "proprietary" in nature?

- What applications must operating systems support in the target architecture?

- What system calls and operating system standard interfaces do my current operating systems support? What about my target operating systems?

- To what degree will my target architecture operating system environment support standards for network computing? Cooperative processing? Client/server applications?

- What standards framework should I adopt for remote procedure call (RPC)?

- How should my target future operating system handle security?

- How should I integrate new operating systems to be inserted into my existing technology base with embedded systems in place?

- Does the target architecture for operating systems have a migration road map associated with it?

- If I do select a new target set of operating systems that is different than those in place today, will the target architecture support a realistic conversion plan?

Communications

- What is the *"diversity cost"* of this set of various communication systems, platforms, and protocols?

- What target applications should my new communications architecture support in the future?

- To what degree should the new architecture support LAN-based standard environments? To what degree should my new architecture support standards associated with network computing (LU 6.2, DCE, RPCs, etc.)?

- What standards model do I want to adopt in my future architecture? Is there (or will there be) enough product in the marketplace to implement my target architecture?

- To what degree can we implement the OSI model within our new target architecture 1) with existing embedded base products and services, and 2) with new products and services emerging in the marketplace?

- Which of the developing standards (such as X.500) represent significant breakthrough standards that may be of use in more than 36 months (i.e., they are not available for several years but are concepts upon which we would like to establish our target architecture)?

- What new standards, not currently in use, are there that could result in a significantly new way of conducting our functional area's mission, such as EDI (X.12), ISDN or FDDI, or SONET (fiber optic transmission for image and other high bandwidth requirements)?

- What set of target protocols and target services do I want to support in the target architecture?

- If the client/server model is to be implemented in the target architecture, what roles will respective applications have (client or server) to one another?

- What role will the various platforms have with regard to the applications that they run?

- What network management standards do I want my new architecture to support?

Management services

- What is the *"diversity cost"* of this set of various management services located and maintained in different non-compatible environments?

- What is the set of management services that I want in my target architecture? Where should they be located in the target architecture—on one platform or many?

Other services

- What is the *"diversity cost"* of this set of various "other services"? Is the functional requirement cost or opportunity loss of not having certain management services such as access control, authorization, authentication, time, directory, cryptographic, file, data, print, EDI, presentation and monitor/sensor, or actuator? Which of these services should we add, and where should they be located in the architecture?

Application placement within the infrastructure and recommended style of computing

The SBA project participants spent a significant amount of time discussing the descriptions and characteristics of applications and related information subjects in order to provide input on the decision regarding which processor types on which tier of the network would be used to support the applications. The physical location of applications and information can be determined using the technology architecture platform profile described earlier and the cross-reference matrices from other views of the architecture.

Application to Generic Application Environment Matrix

The Application to Generic Application Environment Matrix characterizes each application in terms of the GAEs that will be required to support the functionality of the application. Each target application opportunity cross-referenced to one or more GAEs. This matrix was used as input to the recommendations on application placement across the technology environment. An excerpt from the matrix is shown in Figure C-1 below.

Client/server model

The DoD TAFIM document specifies the client/server model as the preferred standard for distributed network computing. Within the client/server model, five "styles of computing" can be used. Each of these styles of computing has strengths and weaknesses that must respectively be exploited and minimized. The reader is referred to TAFIM Volume 2 for a detailed description of each of these styles of computing. The styles are described below in graphic form in Figure C-2.

Major Business Areas	Applications	Batch Processing	Broadcast	Computer Conferencing	Decision Support	Document Processing	Document Storage & Retrieval	Electronic Mail	Electronic Publishing	Enhanced Telephony	Expert Applications	Hyper Media Processing	Inquiry Processing	Real Time Control	Shared Screen Teleconferencing	Text Processing	Transaction Processing	Video Processing	Video Teleconferencing	Voice Processing	Voice Mail
Establish Direction	THS Agreement Management Application	X				X	X	X	X		X	X			X	X				X	
	THS Guidance Management Application	X				X	X	X	X		X	X				X				X	
	Medical Total Quality Management Application							X			X	X	X				X				
	THS Medical Situation Management Application							X			X	X	X				X				
	THS Medical Options Development & Evaluation Application					X	X	X			X	X	X				X				
Acquire Assets	Defense Medical Service & Materiel Management Application					X	X	X				X	X				X				
	THS Procurement Management Application					X		X				X	X				X				
	THS Assets Positioning Management Application							X				X	X				X				
	Health Statistics Tracking Application	X			X	X	X	X				X	X			X					
	THS Disbursements & Receivables Application					X		X				X	X				X				
	THS Facilities Management Application				X							X	X				X				
Provide Capabilities	THS Medical Transportation Assignment Mgmt Application				X						X	X					X				X
	Theater Medical Site Management Application					X	X	X	X			X	X				X				X
							X	X	X			X	X				X				X
											X	X									
																X	X			X	X

Figure C-1. Application to Generic Application Environment Matrix

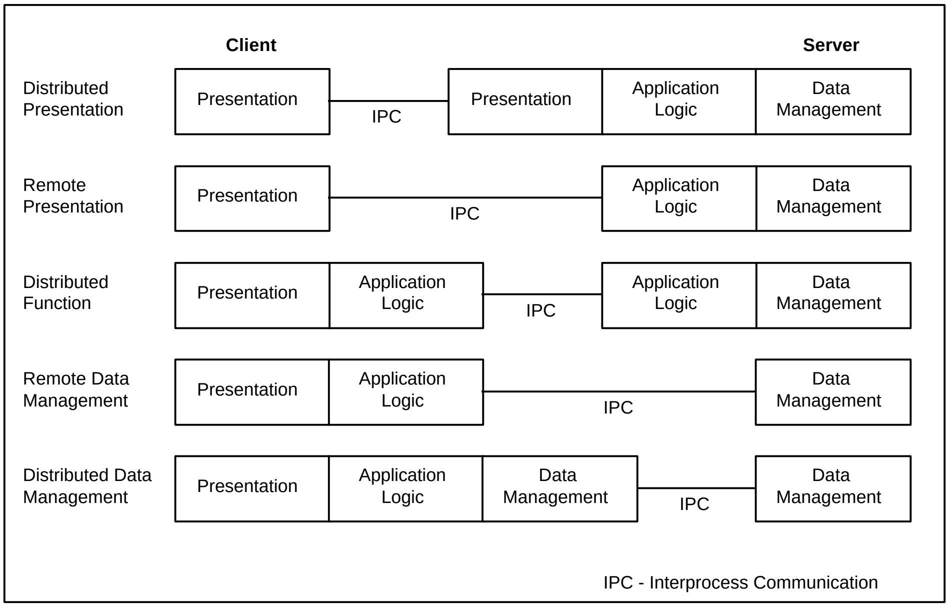


Figure C-2. Client/Server Style of Computing Model

As the above descriptions show, the location(s) of applications and related information is highly dependent on the style of computing chosen for the application and the degree to which a given data grouping (or set of data groupings) is accessed by other applications that may or may not be operating in the same style of computing. Therefore, the logical progression in making these determinations is to analyze each application and its associated information characteristics and linkages depicted in the architecture models and to recommend a preferred style of computing for each application based on these combined characteristics.

The following components of other views of architecture contributed directly to the decisions on the style of computing for each application.

Work view

Logical operating unit to logical work location

- Logical operating unit to data grouping
- Logical operating unit to application.

Information view

- Characteristics of information

Information model.

Applications view

- Application description
- Characteristics of applications
- Application to data grouping

  Application to GAE.

Recommended style of computing and application placement

The following is an excerpt of the recommended style of client/server computing for each target application opportunity.

With these styles of computing in mind, a general mapping of applications and information to the location types was done. Figure C-4 shows an example of the high-level placement of applications and information at one of the three levels within the technology environment:

	Application \ Client/Server Style	Distributed Presentation	Remote Presentation	Distributed Function	Remote Data Management	Distributed Data Management
Establish Direction	THS Agreement Management System	X				
	THS Guidance Management System	X(1)		X(2)		
	Medical Total Quality Management System	X				
	THS Medical Situation Management System	X(1)	X(2)			
	THS Medical Options Development & Evaluation System	X(1)	X(2)			
Acquire Assets	Defense Medical Service & Materiel Management System		X			
	THS Procurement Management System		X			
	THS Assets Positioning Management System	X				
	Health Statistics Tracking System	X				
	THS Disbursements & Receivables System	X				
	THS Facilities Management System			X		
Provide Capabilities	THS Medical Transportation Assignment Mgmt System		X			
	Theater Medical Site Management System	X				
	THS Personnel Management System	X				
Employ Health Service Assets	Public Health System			X		
	[illegible]			X(1)	X(2)	X(3)

Figure C-3. Recommended Style of Client/Server Computing Matrix

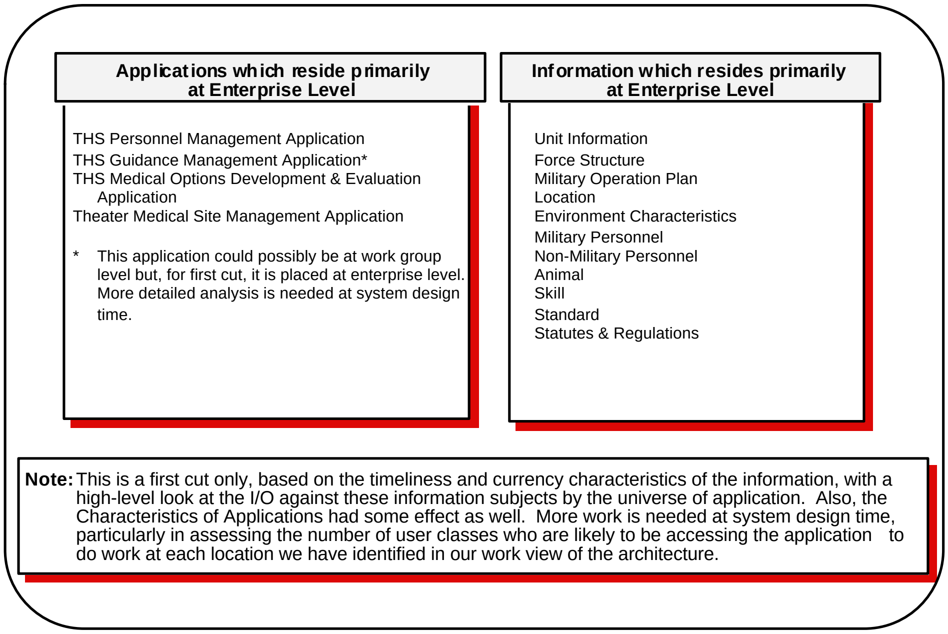


Figure C-4. Enterprise Level Applications/Information Model

When a distributed network computing environment is envisioned for an organization, the “location” of the application and information is not definable in concrete terms at the architecture level. By definition, this kind of technology environment will support both distributed data and application processing. Specific instances of a given data grouping and an accessing application within our architecture model may appear at a number of dispersed locations, either through the techniques of replication, fragmentation, or a combination of both.

An example of this might occur in the health care equipment data grouping and the applications that access it, such as the defense medical service and materiel management application. Records containing the data elements that describe a particular piece of equipment may appear on a computer system in the work group where the equipment is in use. However, some specific data elements about this same type of equipment may appear on another computer system in another department, which may happen to have some of this equipment in use there.

Because information can appear in many locations and computer system platforms in a distributed computing environment, our applications and information architecture implementation must support methods of data synchronization and control that are independent of the applications accessing and updating the data groupings.

Other applications and supporting technology platforms

The next area to consider is inter-enterprise connectivity. Connectivity with other systems in the three services, DoD units, and other Federal agencies is increasingly important. The touted benefits of open systems technology (i.e., portability, interoperability, and scalability) can most effectively be used in this arena. With the adoption of open systems (as described in TAFIM), in conjunction with the mission-specific architecture developed in this SBA, the needed building blocks are available to "link" to entities "beyond the boundary," as needed, in an effective way.

The need to evolve to a minimum set with common components

In transportable and/or mobile locations, a key issue is to "economize" the various communications so that they can be routed through an efficient set of voice/data switching and transmission systems. The goal should be to evolve these systems to a minimum set that meets the currently envisioned needs but which, like the networked computing environment with which they must interface, are built using "standard" components or building blocks.

This will not only move the organization ahead in terms of interoperability but also should reduce the number of unique repair parts and end units. This will provide cost and operational efficiency benefits that complement the increased productivity that seamless communications can bring.

Cross-service compatibility is a key issue

The issues of compatibility and interoperability within the body of existing and planned communications are significant. In the joint environment, there are still major issues with mismatches on communications protocols, as well as with system and applications software, which cause severe hardships when joint operations are attempted.

We are reminded that various armed services networks utilize commercial facilities for both switching and transmission to augment private networks. AUTOVON is an example of a service that rides on leased commercial facilities. There are pros and cons to each approach.

When commercial facilities are used, the organization gains something in that the operation of the network is not a burden, and the underlying technologies and services are constantly being upgraded by the common carriers and VAN vendors. However, the use of commercial facilities introduces the need for minimum service levels and a monitoring process to ensure compliance. Also, these commercial networks have many more opportunities for security breaches than do fully controlled private networks. Based on industry experience, however, when traffic does not require specific security considerations and when they are readily available, commercial facilities are an advantage because of operational and feature-related factors cited above.

The major drawback to dependency on commercial network services is that they may not be available in the diverse geographic and political environments within which the DoD may have to operate. If they are available, their reliability may not be guaranteed. These considerations may lead the DoD to rely almost exclusively on facilities and services that are totally under its control.

The need to further investigate capacity and improved mobility

For the DoD to realize the full potential of the networked computing environment defined in this document, the area of mobile communications needs further investment in two areas:

- Additional capacity for gear that is currently effective
- Gear that will provide new capabilities to transmit and receive a significantly increased amount of digital data in wireless mode.

This is an area that needs to be explored in more detail as each application opportunity moves into the design stage.

From the USMC work done earlier in the year, the SBA team has found that for mobile telecommunications two distinct approaches are in use today:

- Deterministic routing (used by the Navy, USMC, and Air Force)
- Flood search routing (used by the Army).

The flood search routing technique is used in the Army's Mobile Switch Routing Telecommunications (MSRT) system. The MSRT system most closely replicates the features and capabilities of the commercial cellular phone network. It allows "full service" while on the move. In this regard, it is superior to the deterministic approach and should be explored as an evolutionary path. Cellular technology is well tested in the commercial arena and is undergoing continual refinement. This should allow the DoD to take advantage of reduced costs and increased reliability and bandwidth in the long run wherever this technology is feasible. It must be recognized, however, that this technology is not as easily established in a deployed environment as the deterministic method.

From the USMC project, it is understood that the Army is the executive agent for all DoD tactical switching. This includes defining, scoping, planning, scheduling, and determining the operational impact of changes to the tactical switching environment across the DoD services and agencies. Other components of the DoD would be well served by assigning resources to work closely with the Army in this area exploring mobile and transportable switching and transmission facilities options for interconnecting its IT computing platforms.

Security considerations

Security should be implemented at a minimum according to DoD directives. TAFIM Volume 2 refers to a number of standards for security implementation. They are:

- Open systems security
- Multi-domain information security
- Multi-channel processing security
- Distributed processing security
- Security management.

Within the specific components of the technology architecture, there will be opportunities to implement various degrees of security. Security can be implemented at the application level, the operating system level, the database management level, and at the external environment (platform/facility) level.

Multilevel security for secured clients and servers in the technology environment, as well as the possibility of network encryption units (NEUs) for secured network nodes, are just a few examples of the areas covered in the referenced standards and guidelines. The following is a high-level view of the components of a secured architecture.

1. Two constructs provide security extensions to the Target Technology Architecture: (1) Multi-level Secured (MLS) operating systems; and (2) Network Encryption Units (NEUs).
2. The operating system on each processor provides for multiple levels of security.
3. Each processor has two connections to the network, one for an unclassified network address, the other for multiple addresses (one for each level of security).
4. NEUs provide protection of information passing from processor to network and from network to processor.

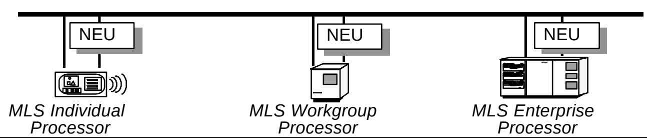


Figure C-5. High-level View of the Components of a Secured Architecture

The reader is referred to TAFIM Volumes 2 and 3 for a treatment of this subject. At a minimum, the DoD should adhere to the standards put forth in these documents paying particular attention to any interfaces between supporting establishment and tactical systems. Of course, the unique nature of delivering health services may actually make the need for security less of an issue than for military operations (i.e., there may be value in identifying a given location as a medical facility).

A look ahead

The next phase in the SBA is the opportunity identification phase. In reality, a significant portion of this phase has been completed during the development of the application architecture view of the SBA.

Migration options follow the opportunity identification phase. This plan will identify and prioritize project initiatives for the next 5 years. The approach will include bundling the projects identified into implementation phases.

Once the project initiatives are grouped into implementation phases, the implementation planning phase begins. These plans will provide more detailed descriptions of the near-term (those started in the first 2 years) projects identified in the migration plan.

When both the migration and implementation plans have been developed and reviewed by the ASC, implementation of the projects can begin. However, the SBA process is not complete until an SBA administration process is defined that will keep the SBA planning process alive and current with changes within the DoD.

This page intentionally left blank.

Appendix D: GAE/GTE/GTP Definitions

Introduction

This appendix contains the definitions of the Generic Application Environments (GAEs) and Generic Technology Environments (GTEs) introduced in Sections 3 and 4. This is an initial set and is not intended to be "all inclusive." Simply put, these should help to get a work team started on its quest to define the necessary GAEs and GTEs for its functional area(s).

GAE sample definitions

Batch processing

Batch processing environments are characterized by their ability to queue work (jobs) and manage the sequencing of processing based on job control commands and lists of input data. The results of this processing include updated information files or databases and often printed reports or special forms that are themselves queued as output jobs.

As such, work is performed asynchronously from the users requesting the job or waiting for its printed output. In most cases, the direct users of the environment are specially trained computer system operators.

These environments have been the mainstay of data processing operations since their inception and will continue to perform critical recordkeeping and background processing functions in conjunction with their related interactive GAEs.

This is evidenced by the major transition that has occurred since the punched card and paper listing days of the sixties. This transition has seen the migration from key punch, through remote job entry (RJE) and optical character recognition readers, to the use of on-line, interactive data entry systems (a transaction processing environment) and inquiry processing systems that share a common database.

Use of file transfers between environments will continue as an effective means of interfacing with batch processing environments, only in a network server context rather than the conventional host computer relationship.

Batch application attributes include number, source, and nature of data capture transactions; timing and sequencing requirements; and volume and type of printing requirements.

Transaction processing

Transaction processing environments support on-line capture and processing of information in an interactive exchange with the user. These typically involve predetermined sequences of data entry, validation, display, and update or inquiry against a file or database.

Environments using character keyboard entry/displays typically base screen designs around the use of menus and electronic forms. Those using GUIs are moving toward the use of icons and images to support command activation and information display.

On-line transaction processing applications have grown out of document processing applications where timeliness and currency of processing a functional area transaction and capturing its associated information is important.

Typical transaction processing application attributes include number, size, source, location, and complexity of transactions; response time; and peak usage requirements. The nature, size, and complexity of associated subject databases (or files) also need to be determined along with the degree of sharing with other applications—as derived from the information model.

Inquiry processing

Inquiry processing environments support functional area activities requiring interactive selection, extraction, and formatting of stored information from files and databases. They are used in conjunction with batch and transaction processing environments to provide information retrieval using either structured (routine) or ad hoc (definable) queries. They are intended to replace the need for extensive reporting systems by providing only needed information on demand.

These environments typically provide user-oriented languages and tools (often referred to as fourth generation languages) to simplify the definition of searching criteria and aid in creating effective presentation of the retrieved information (including use of graphics).

Attributes include frequency of inquiries (prestructured or ad hoc), types and complexity of searching, associated files or databases, and types of presentation required.

Decision support

Decision support environments provide interactive modeling and simulation tools that allow the user to analyze the effects of alternative decisions. These modeling and simulation tools typically work in conjunction with files and databases that were created from batch or transaction processing environments.

As with inquiry processing environments, GUIs are used to simplify the interactions for both building and using the decision support models.

Attributes include the type and complexity of models and simulation algorithms required, the frequency of use, the associated files and databases, the complexity of presentation required, and response time.

Expert systems

Expert systems environments use a type of artificial intelligence built with inference engines and knowledge or rule bases that take or recommend actions based on presented situations and past "experience." They are used to augment human decision-making processes where the "expertise" or thought processes of the decision maker can be defined as rules.

Expert systems are now finding their way into many functional area applications, especially those involving assessment or estimating processes, such as credit risk assessment. These environments are quite specialized today and are based on tight relationships between the "shells," within which relationships are defined, and the corresponding knowledge bases. As experience with applying these environments grows, they will likely become more integratable with other environments.

Attributes involve size and speed of processing, the type of knowledge base used, the type of inferencing processing involved, and whether it is used in batch or interactive mode.

Real-time control

Real-time control environments support event-driven processes supporting monitoring and actuation of physical processes. For this reason, they are often referred to as sensor-based systems. They are designed to handle and process interrupts from a variety of sources (typically involving some kind of sensor device or timer), process associated information through some type of capture or control algorithm, and respond, if necessary, with an appropriate signal to a control or actuation device.

Unlike in the process, manufacturing, and raw materials industries, real-time control environments have a minor presence in financial organizations. They have a role in building security and facility management in such applications as access control systems, fire detection and alarms, energy management, and elevator controls. There are some applications, such as access controls, where integration with other security management environments may be appropriate.

Text processing

Text processing environments support the creation of text documents. They have evolved from the early word processing systems of the seventies to be popularized as part of the explosive application growth of desktop personal computers. They offer greatly improved editing and revision capabilities over the typewriters that they were designed to replace.

Because of their character and word orientation, they offered only limited abilities to improve the presentation and appearance of the final printed document. As a result, they are now losing ground to the graphics-oriented, document processing environments.

Text processing environment attributes include editing and formatting features, mail/merge capabilities, and document filing requirements.

Document processing

Document processing environments extend the basic capabilities of text processing to take advantage of the graphics capabilities of today's workstations and laser printers. Consequently, they provide powerful document and presentation tools for the end user.

These environments use an object-oriented approach to composing documents, allowing the incorporation of graphics, images, and even voice annotation, along with stylized text. They provide advanced formatting and editing features such as style guides, spell checking, use of multiple columns, table of contents generation, headers and footers, and outlining tools.

They require a GUI and often include support for scanning images into bit-mapped representations. This SBA Guide, for example, was prepared using such an environment.

Attributes include types of objects supported, editing, style and formatting features, resolution of display and printing, graphics generation capabilities, color or gray-scale usage, search and retrieval facilities, and document filing requirements.

Electronic publishing

Electronic publishing environments extend document creation and production tools to provide formal publishing capabilities. This includes incorporation of photographic quality images and color graphics, very advanced formatting and style features, such as wrapping text around graphic objects or pictures, and kerning (overlapping characters to optimize spacing).

These environments range from desktop versions to sophisticated corporate publishing systems and are often used through external publishing services. They generally require specially trained "operators" who possess document design and layout skills. They also interface with, or incorporate, sophisticated printing and production equipment.

Attributes include resolution and color; editing, formatting and style features; type, size, and binding of printed output; and printing production rates.

Hypermedia processing

Hypermedia processing is a new environment that extends the object-oriented approach to organizing and displaying information by utilizing various relationships between the stored or created objects. As such, it overcomes the limitation of the printed page and allows the user to "navigate" through the compiled information based on mixed form objects in a manner that is consistent with the needs and capabilities of the user, not some fixed presentation format.

Through the use of the GUI and its extensions to include voice/sound as well as video capabilities, hypermedia presents the ultimate in user communications. In effect, a dynamic document is created by integrating the full range of information display capabilities interacting with associated files and databases under user control.

Attributes include the type and quality of mixed objects supported, the types of relationships allowed, and navigation tools.

Video processing

Video processing environments support the creation of video “productions,” either as sequential presentations or as interactive presentations, under user control. They involve both video and sound capture and editing, as well as incorporating still graphics and title generation capabilities.

They are becoming increasingly popular in corporate education as an adjunct or replacement for classroom training. They are also useful for marketing and product promotion or in packaging general information and inquiry services.

Attributes include nature (i.e., analog or digital) and quality of capture and reproduction, editing facilities, ability to integrate user commands, and sequential or direct file access.

Document storage and retrieval

Document storage and retrieval environments are used to retain large volumes of stored information in document formats. Originally these systems were based on microform media using film or fiche with special readers to magnify the information. Computer output microfilm (COM) systems are used to store computer-generated listings or reports.

More recent introduction of optical storage technologies is allowing for storage of scanned or computer produced documents using digital storage techniques. These are now available for use on PC networks as well as for large corporate applications such as archiving. “Juke boxes” are now available to load compact disks under computer control to achieve incredibly high storage and on-line access volumes. Compact disks show considerable promise as a means of distributing reference material with frequent updates possible at low cost.

Attributes include type of media, speed and resolution of scanners, compression techniques, ability to modify or update stored material, access frequency and response, media storage life, and retention volumes.

Electronic mail

Electronic mail environments support the storage and forwarding of directed messages, mail, and other documents or files between sender and one or more recipients. They provide the sender with facilities to create or define the message(s) or file(s) to be sent, use directories and distribution lists for routing information, assign priorities, use preformatted electronic forms, and trace the status of messages sent.

The recipient is typically provided a mailbox with a summarized listing of incoming mail, a log of mail received and read, the ability to file or print mail or documents, and the ability to reply to or forward messages.

These environments are now capable of interfacing amongst themselves to extend their reach from work group to public level (international) distribution. Some are capable of “reading” text messages back via phone access through the use of voice synthesis.

Attributes include sending and receiving features, number of direct users, extent of directory and distribution list management, interconnection capability, and security facilities.

Voice mail

Voice mail environments offer the storage and forwarding of voice messages for a designated set of recipients. They are usually used as an extension of the phone system to provide an alternate to message centers. They typically allow the recipient to retrieve recorded messages remotely from any touch-tone telephone.

Attributes include quality of voice recording, user features, size of directories, and message management facilities.

Enhanced telephony

Enhanced telephony environments provide improved means of using the phone system for interactive audio exchanges between users. Features include call forwarding, call waiting, programmed directories, teleconferencing capability, automatic call distribution (useful for busy customer service areas), and call detail recording.

These can be provided at the local (facility level) or across corporate or public networks.

Attributes include the features supported and the ease of use or help facilities provided through voice response and/or intelligent handsets or integrated voice/data workstations.

Shared screen teleconferencing

Shared screen teleconferencing environments are another newly emerging type of system aimed at supporting more effective remote communications in an interactive mode between two or more users. They combine an audio teleconferencing capability with shared common workstation “windows” that are refreshed on every conferee’s workstation whenever someone displays new material or changes an existing display.

In this way, conferees present and discuss displayable material interactively as in a meeting. They can graphically annotate or modify the shared conference window. The attractiveness of this environment is that it can cost-effectively support many of the communication requirements of remote meetings using normal telephone linkages with properly equipped workstations.

Attributes include display quality, refresh and transmission rates, and conference control features.

Video teleconferencing

Video teleconferencing extends the remote meeting environment to include full motion display of events and participants in a bidirectional manner. Thus, the facial expressions and body language of presenters and questioners is displayable to all participants in a conference.

There are a variety of schemes for directing the cameras ranging from fixed position to sender directed to receiver directed to automated sound pickup. This technology has seen limited application because it required studio facilities and was very expensive in its introductory phases. Breakthroughs in charge-coupled cameras, display technology, and high bandwidth communications should see a resurgence in interest and application of video teleconferencing.

Attributes include picture and sound quality, refresh and transmission rates, and camera and conference controls.

Broadcast

Broadcasting environments provide one-way audio or audio/video communications between a sending location and multiple receiving locations. They include the use of private TV facilities that can be purchased or leased for corporate purposes. Many organizations are taking advantage of these facilities and offsetting travel costs for use in corporate announcements and product introductions.

Some information providers are now producing special-purpose TV shows for corporate subscribers as a substitute or adjunct for attending conferences (e.g., *The Computer Channel)*. These are often combined with audio return links for question and answer sessions.

Attributes include the quality of production facilities and the scope/range of the receiving network.

Computer conferencing

Computer conferencing environments combine the merits of document creation, E-mail, and conferencing by allowing groups and subgroups to participate in "conferences" via computer workstation. These conferences, however, do not occur in real time. The conferees discuss proposed topics through interacting over time. Conferees, or invited guests, can drop in or out of conferences or subconferences at will. The ability to trace the exchanges is provided.

These environments have become popular among academics and within university circles, beginning with basic text capabilities. Combining the richness of hypermedia with computer conferencing would create an environment in which the most capable and experienced individuals could be brought together remotely to focus on a critical topic using the most powerful electronic means of communicating ideas.

Early forms of these environments are now available to users of graphical workstations. Attributes include types of documents exchanged, conference management and recording facilities, and search and retrieval capabilities.

GTE sample definitions

Each GAE is supported by one or more GTEs. The combination of the GAEs and GTEs provides the infrastructure components for delivering systems and services to the organization.

User interface services

User interface services provide the basic means for users to interact with the computing environment, managing the user interface for any class of user interface device from a simple character terminal to an advanced graphic workstation. User interface services also provide support for the user in navigating through to the appropriate system or server, authenticating the user and managing the user's desktop.

User interface services must support various input and output devices defined in the GAEs for each user class. There will need to be a variety of presentation servers used by user interface services to support the various classes and types of interface devices. For example, there may be an X/Windows-based high-end GUI server and a lower level character-based server for different users.

User interface services interact with all other GTEs providing them with the ability to receive and present information to and from the user. Client applications and users can be reasonably isolated from differences in the underlying technology through the various presentation servers incorporated in user interface services. For example, the user interface should operate in a similar way on a Mac, a PC, or a POSIX workstation.

Optional servers can provide encryption, data, and file management for user interface services. These may or may not be configured into the environment.

System management services

System management services support all activities dealing with the management of the computing environment, interacting with all other GTEs to provide the management capability to monitor and control the total environment.

The objectives of system management services include providing adequate availability and performance across the environment, accurate and complete billing, change control, and failure recovery. This environment provides the basis for implementing specific applications and tools to provide these capabilities.

GAEs and all other GTEs make use of system management services.

Communications management services

Communications management services is another GTE that is used by all of the other GTEs that want to communicate. This environment implements the communications infrastructure consisting of various communication servers, name and directory services for resolving addresses, and authentication and access control for ensuring the appropriate level of security. Thus, all the technology associated with communications and connectivity is bundled into this environment.

Specialized application servers for bandwidth management and other communications functions would also be provided.

Database management services

Database management services consist of the servers required for managing files and data within the technology environment. It consists of data servers that implement databases and file servers that provide local and remote access to various types of files.

Specific application servers may be implemented to isolate the other environments from the physical structure and location of data. Implementation of a distributed data management environment would require a set of specific application servers to support access, manage the physical datasets, and provide the appropriate level of integrity.

Hypermedia

An emerging area for information management is hypermedia. Hypermedia provides a highly flexible way of linking objects. Over time, documents, images, and other objects could be linked in hypermedia databases resulting in the elimination of document management services as a separate entity.

Standards for information management will be required to deal with traditional data management as well as newer technologies for storing other forms of information. Distributed data management capabilities are appearing in vendor's products and need to be addressed through appropriate standards for their usage.

Transaction management services

Transaction management services implement the environment required for managing transaction processing. This environment includes the basic functionality and servers required to implement a transaction processing application. In today's world, CICS would fit under transaction management services. In the future, it is anticipated that a client/server environment will become the norm.

Transaction management services receives requests (transactions) from user interface services and actually performs the transaction processing. It may interact with information management, document management, or distribution management services to update a database or pass the message on to another environment for processing. For example, a transaction generated by a user interface services environment (i.e., a user using a workstation) could link with several environments before the transaction is completed.

The type and nature of the link will depend on the application requirements. For example, the link may be a real-time interactive link requiring completion by the server before the client can do something else or may be a message transfer link where the message or transaction is passed to the other environment for later processing.

This environment consists of authentication and access control servers to control access to transaction processing and at least a data server with which to update or interact.

Document management services

Document management services are analogous to information management services, providing other environments with the means to access and manipulate documents—either text only or some combination of data, text, voice, graphics, and image (a compound document). The key difference between these two technology environments today is the level at which we can manipulate basic elements of information. In information management services, we can access and manipulate each field within the file or database. In document management services, we generally access the entire file or document using application specific formats for manipulating portions of the document. For example, the format for a Microsoft Word document is different from WordPerfect; likewise, the way graphics is stored in each differs.

However, the distinction between the two is one that is based on currently available technologies. Once we have compound document architecture standards and databases that can handle document objects well, it's likely that the two environments will merge and become one.

Conferencing management services

Conferencing Management Services supports the real-time exchange of information from one or more user clients. It permits a user to address a communication to any member of a group without needing to know exactly who is in the group receive communications from all or selected members of the group without needing to know who is currently in the group, and to reply to them in a like manner.

Conferencing services include various types of real-time services including voice conferencing (audio only), video conferencing (audio and video) and computer conferencing (shared screen).

The Conferencing Management Services GTE utilizes Name and Directory services to establish the parties for the conference and is closely linked with the Communication Services GTE to establish the physical linkage. It may also closely link with hypermedia (in information management services) to provide a dynamic subject- and task-oriented asynchronous conferencing environment.

Distribution management services

Distribution management services support the distribution of messages, transactions, files, and any other information between technology environments and physical locations. This environment consists of servers that implement electronic mail, voice mail, and EDI. It also is tightly linked to the communications management services GTE to provide the actual communications between components.

Development services

Development services provide support for all aspects of systems delivery including all phases of the development life cycle, prototyping, and end user development. This environment interacts with the other GTEs to access information on the current infrastructure and to implement changes and enhancements.

Development services is built upon several servers to provide authentication, location of objects (name and directory servers), and to implement specialized applications. CASE tools and compilers are considered to be application servers in this environment.

Repository services

Repository services is an emerging GTE that will provide the repository environment for managing the technology environment and the applications and data stored in the environment. The repository can store information about any “object” in the technology environment including, but not limited to, the physical processors, application modules, data, and processing functions. All of the GAEs, GTEs, components, and servers defined in this document would be entities in a repository.

Repositories for system construction

A passive repository, such as those being introduced by IBM, DEC, and others, can provide the dictionaries and system encyclopedias needed for defining and constructing application systems and data. This type of repository is the essential underpinning of a CASE environment, as it provides the basis for storing information at each phase in the development cycle and transferring that information from one phase to another.

Repositories for systems management

Another type of repository, called the active repository, can be used to store system information and to dynamically manage the IT environment. For example, with the capabilities of an active repository, system management services could manage the execution of applications to optimize performance and reliability.

Conceptually, repository services will interact with other GTEs to provide a “single system image.” This is an environment where the computing and network infrastructure appears to the application and user as one “computer.” In this environment, repository services would define the single-system image and manage where and how processes are actually executed.

Server definitions

Figure D-1 lists several server types. It illustrates a sample set of logical components of an organization’s technology infrastructure. Entries may be added or modified.

Name	Translates network-wide logical names to network address
Directory	Identifies logical names based on attributes
Authentication	Establishes the needed identity of a network user
Access control	Establishes access to desired applications or data
Cryptographic	Provides encryption and key management services
Communications	Establishes linkages for a client (switching, router, gateway)
Time	Ensures common network time
File	Provides transparent access to network files
Data	Provides remote data services (database access)
Print	Remote printing and print management
Mail	Provides electronic mail services
EDI	Provides electronic data interchange
Applications	Provides application-specific services
Presentation	Manages the user interface for a client user (a person)
Sensor monitor/actuator	Manages interfaces to physical sensors, actuators, and timers

Figure D-1. Server Classes

Name server

The name server provides a means of finding an attribute of an entity given the unique name for any entity within the technology environment. Entities can be physical components (computers, workstations, network nodes), logical components (application modules, data storage locations), or users.

The name server will be accessed frequently by clients to find addresses for servers and other objects. Consequently, it needs to be implemented so it can provide high-performance response to queries. The search will be by unique name (unlike the directory server) so quick response can be provided.

Name servers will also likely be highly distributed, so clients cannot assume the attribute provided by a name server is the latest version. While in 99 percent of the cases it will be correct, clients will have to implement a recovery mechanism to deal with the exceptions.

There are few vendor implementations of name servers in the market today. Likewise, the standards bodies are still drafting industry standards for name servers and application programming interfaces to name servers. DEC has an early implementation and architecture for a distributed name service and is worth investigating.

Directory server

The directory server provides a means of finding a set of entity attributes based on qualifiers, such as a telephone number or other descriptive characteristic. Unlike a name server, the searches are often ambiguous and based on a combination of attributes.

Clients may use a directory server in the future for queries such as, “find me a vector processor with 40 MIPs performance” or “find me a storage device with 40 MB free space.”

Directory servers will not be accessed as frequently as name servers. Performance will not be as critical as the name server’s because of the lower rate of access and the fact that the access by directory server clients is done on an ad hoc-query basis, often under the direction of a user (e.g., find John’s telephone number).

Like the name server, clients cannot assume that the attribute provided by the directory server is the latest version. While in 99 percent of the cases it will be, clients will have to implement a recovery mechanism to deal with the exceptions.

The directory server may become a client to a name server to resolve physical and logical addresses.

Authentication server

Validation of users, nodes, programs, and other required objects is performed through the authentication server. Secure channels using encryption and/or some form of trusted communications provide the linkage between client and server.

Access control server The access control server maintains the access control lists for each object within the technical environment. The access control server determines whether access to the requested system object is authorized.

Cryptographic server Encryption services for any process are provided by the cryptographic server. The cryptographic server also manages keys and handles distribution of valid keys among the cryptographic servers. A centralized key management server may be required.

Communications server The communications server forms the basis of managing connections between objects in the environment. It provides connections between objects independent of the physical implementation of the network and ensures accurate delivery of messages between objects.

The communications server, from the point of view of the GTEs using it, provides OSI Level 7 services to the environment. Gateways, routers, bridges, and protocol converters are included in the communications server but are invisible to the clients of the communications server. Bandwidth and capacity management support are also incorporated in the communications server to provide the basis for optimizing the capacity and reliability of the network.

Utilization of this server will provide applications with transparent access to communications services in the environment. The communication server has the ability to support a transparent computing environment where applications and users do not have to be concerned with the logical and physical implementation of the technology.

Time server A critical need in distributed environments is to make sure that time is synchronized throughout the environment. This is especially important in distributed transaction processing applications and database environments where logs need to be kept synchronized to support transaction backout and recovery.

The time server provides time synchronization services to all objects within the environment. Individual objects will call on the time server to get an accurate, consistent time for their use.

There has been limited vendor and standards activity in this area. DEC has proposed their time server to OSF as part of the distributed computing environment request for technology.

File server

The file server provides transparent access to files from workstations and other clients. Unlike a data server, the file server provides access and linkages to the file directories and is not aware of the contents of the file. Processing of contents of a file needs to be performed by the client. The file server does no client-visible manipulation of the data within a file. Essentially, the file server provides the client with the use of a virtual disk drive and little else. For example, in a workstation environment, the workstation would perform all the processing on the file.

This can create synchronization and reliability problems when the file server is used as the place for storing databases and other files that are accessed by several users. The file server is best used when accessing an entire file such as a word processing document or a spreadsheet.

Over time, the file server may be replaced by a data server because of its improved controls and better management capabilities.

Data server

The data server provides data services to clients. A client will send a request to a data server (sometimes called a data*base* server) and the server will respond with the results of the request. The accessing and updating of the data maintained on the data server is performed by the data server, not by the clients.

The data server can provide additional services. For example, recovery and rollback capabilities can be provided. It supports the implementation of better controls by managing access to the data resident within the server.

The data server can also be optimized to the type of data it is being asked to manage. For example, a data server could support archiving and be based on optical storage technology rather than magnetic. In the future, data servers will likely provide access to multiform data that includes voice, text, and image objects as well as data.

Print server The print server provides common printing services to clients within the environment. The print server provides transparency between the client and the physical printer. For example, differences between different vendors' laser printers should be transparent to the client.

In addition to device-independent printing, the print server also provides queuing, priority management, and other print management services so that the physical printers can be effectively managed.

Printers can be any local or remote output device capable of printed output, including traditional character and line printers, laser printers, fax machines (the printing portion), and even microfilm printers.

Mail server The mail server provides mail transfer capabilities for a community of users. The basic function is to support the store and forward of interpersonal messages between users. The mail server moves messages based on the contents of the *message envelope* not the message's contents.

The mail server also manages the users' mailboxes. It can automatically acknowledge delivery to a user's mailbox.

The server will support multiform mail transfer (voice-mail, graphics). In the near future, compound mail documents could be transferred using this server.

EDI translation server The EDI translation server interprets the content of EDI messages and routes them to the appropriate EDI partners. The EDI server works hand in hand with the mail server but needs to interpret the EDI message to translate it or route it to the correct recipient.

The EDI server also provides queue management facilities and assured delivery of messages.

Application server An application server provides a set of standard application services to clients. It is a form of packaging an application as a commonly used and reusable component of the infrastructure.

The application server must:

- Have a defined application programming interface and message structure
- Be independent of the client
- Provide a set of generic services that can be utilized by a variety of clients (versus a set of services directly linked to a specific application system)
- Hide its underlying process and data from the application—be essentially a black box.

Breaking specific application systems into a client/server model of design is desirable, but the result is not necessarily an application server. The key is to have independence from the client so the server can be utilized by a variety of clients throughout the organization.

Presentation server

The presentation server provides presentation services for a client application and/or a person. It creates a generic presentation environment that is independent of the underlying technology and provides a means for users to interact with the technology environment.

The presentation server is the most user-visible portion of the technology environment. It is the place where the "look and feel" of the organization's infrastructure will be implemented.

Various models and standards for the user interface are available. It should be noted that standards and available products for the user interface are at a very early point in their evolution.

The presentation server will need to accommodate character-based terminals for the foreseeable future, but we expect a migration to graphic-based terminals to occur over time.

Sensor monitor/ actuator server

The sensor monitor/actuator server provides client applications and users with an interface to physical devices such as cash dispensers, building monitoring systems, or any other device that interacts with physical control systems.

This server is used extensively in manufacturing applications. It can provide the interface to manufacturing equipment, robots, and the like.

Generic technology platforms

There are six technology constructs, or GTPs, are used to provide the fundamental building blocks in a standards-based architecture. Each GTP can function as a fully independent "architecture" in that they each have an interface along with processing, storage, and communications capabilities. As such, each GTP may offer alternative choices in delivery of the same GAE. For example, all six constructs are capable of supporting some form of electronic mail, with different associated strengths and weaknesses.

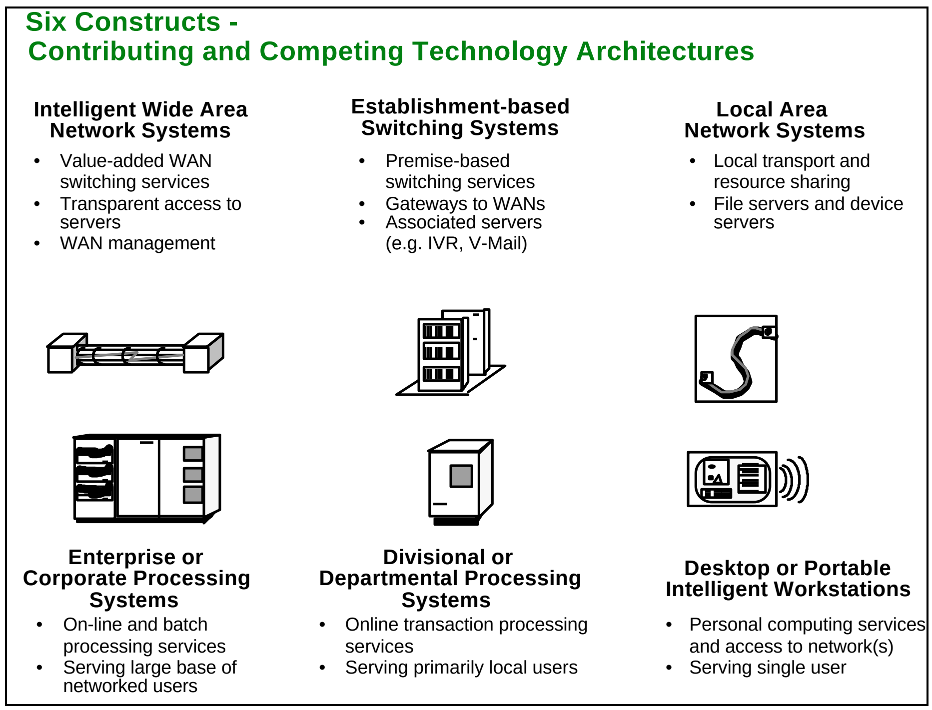


Figure D-2. Six Generic Technology Platforms

It is also important to note that the GTPs do not connote a particular size/capacity. The names for the GTPs connote the usage of the processor, not size. In fact, departmental processors may be larger or smaller than enterprise processors. Some processors acting as LAN servers could well be larger than departmental or enterprise processors depending on the way a given company wishes to organize its work.

Used in combination, these GTPs can be used to describe any architecture environment that current information technology can deliver. Most large organizations are already using multiple combinations of these GTPs. Having determined the appropriate combination of GTPs to support the organization's application requirements, the key to integration is in defining standards that will ensure the highest level of compatibility and portability across the GTPs at both the application and technology platform levels.

Figure D-3 shows a first level of decomposition of each GTP, illustrating the principle components for which standards need to be defined.

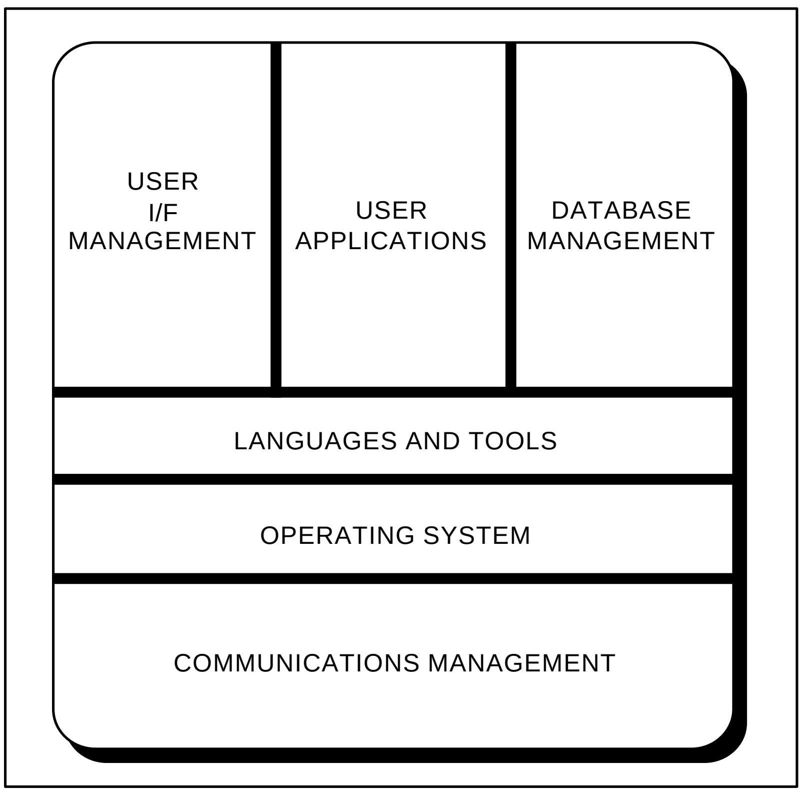


Figure D-3. Components of Generic Technology Platforms

At the component level, we see that all six of the GTPs share a similar structure. Thus, the key to effective integration and sharing in the technology environment is to adopt standards for each component of GTPs, which minimizes the number of different interfaces among components. In today's technology marketplace, vendors are increasingly agreeing on standards at the interface, from GTP to GTP, and within the components of the GTPs themselves. Organizations should adopt technology standards which take advantage of this trend.

This page intentionally left blank.

Appendix E: Migration and Coexistence

The framework for migration

The selection of migration in support of change is always a difficult task and fraught with difficulties and risks. Because the task of migrating information systems and technology is risky, the constraints of migration have to be taken into account in selecting direction and strategy. Many worthwhile projects have floundered because migration was not adequately scoped prior to adoption. In the future, the adoption of open systems and standards-based architectures will reduce the complexity of many migrations to the point where migration will become just one of the scheduled phases, without exposure and without impact on the viability of the strategy. In the meantime, great care is needed to embark on the journey with safety.

As the information infrastructure extends throughout an organization, users draw more and more on the services of a variety of systems. An essential part of migration planning is to accommodate change in one area while accommodating continuity of service in other areas. Coexistence requirements are often as difficult to meet as migration requirements.

What are the migration objectives?

Any migration planning exercise needs to have a clearly defined statement of objective and specification of requirements. In the planning process described, the objectives and primary requirements will emerge from Phase 1, architecture framework, with some refinement of these emerging from Phase 3, target architecture.

For some organizations, the selection of objectives and the movement towards openness will proceed in close cooperation with the development of new functional area systems. For organizations that have a significant investment in infrastructure, or have a multivendor environment, the migration objectives may be very much more technology oriented. Some of the typical migration objectives in the latter category are:

- To move away from dependence on a proprietary infrastructure that has an uncertain future
- To introduce increased interoperability between platforms in the current environment
- To introduce increased openness and integration across platforms in the current environment
- To introduce increased standardization in the current environment so that economies are realized
- To standardize a multivendor environment
- To introduce standards of compliance providing a level playing field for equipment acquisition
- To achieve portability and scalability
- To increase the extent of reuse of technology, applications, and people
- To create an environment that better accommodates new non-proprietary technology
- To introduce new technology
- To facilitate interconnection and interpretability with other organizations
- To work towards the network computing vision within the organization or with other organizations.

Development of these objectives so that they provide clear improvement rather than just a rationalization of costs will flow by examination of key technology issues as they affect the functional area within DoD. Typical questions may define requirements for openness and standardization:

- What interconnection with suppliers is required to improve service/support or reduce costs?
- What interconnection with internal customers is required to improve the service, provide superior products, or reduce costs?
- To what degree can information technology improve or create services?

- Are there particular forms of technology that will change the nature of information processing within the DoD?
- What forms of electronic product distribution (within the DoD) would benefit our functional area?
- What industry-based technology initiatives do we need to come to terms with or accommodate?
- What are the interpersonal communication flows on which our organization depend? What will the benefit of interorganizational electronic mail be?
- What functional area/transaction documents flow with other organizations (outside the DoD)? What benefits would accrue by passing these electronically?

Dilemmas

The evolution of standards is proceeding on many fronts but not at the same pace. The dynamics of standards evolution relate to the complexity of the subject area and the extent of vested interest supporting standardization versus the extent of vested interest resisting standardization. The scene is complicated by the variety of standards bodies and the spectrum of standardization covering de facto standards through to de jure activity.

Of the technology components identified as major building blocks, the most significant level of standards activity is proceeding in the areas of database interface, operating system interface, graphical user interface components, and communications network protocols. In addition, languages have traditionally been an area of standards activity.

The drive for change comes with the attendant problems. While they have been dealt with in some detail in the architecture sections, they remain to be addressed by migration strategies. The dilemmas are repeated here and described in slightly more detail because they have a direct impact on migration.

GUI vs. character vs. block mode terminals

The significant attention given to GUIs flows directly from the level of functionality and ease of use that they can provide. To fully utilize this technology, applications must be modified to support the selected GUI interface.

The conversion of existing character mode or block mode programs to support GUIs requires significant change in program structure and presentation programming. The support of the enhanced functionality requires work to establish and support pull-down menus, pointing devices, and context sensitive tools. The introduction of a GUI approach will, in most instances, require distribution of some part of the application functionality or presentation. Support of distributed function requires an infrastructure that provides services such as program distribution, software inventories, remote diagnosis, and file transfer. These increase the size of the migration activity.

Another area of difficulty is that the selection of a GUI comes with its own set of infrastructure assumptions. Any standards-based initiative reflects its heritage. For example, X/Windows emerged from the character-based segment of the industry. Selection of an X/Windows-based implementation creates a demand for network facilities that accommodate character mode terminals. For reasonable response times, X/Windows needs a local host; thus, the infrastructure requirements may even be in conflict with the needs of character terminals that are currently connected to a remote host.

Selection of a GUI creates a need to examine impacts and migration strategies for both applications and networks.

Peer-to-peer vs. master-slave

A common thrust and assumption in many standards-based activities is that information technology will be deployed in a peer-to-peer manner thus accommodating distribution in any of its many forms. Again, this assumption requires quite a different infrastructure than that used to support the conventional character mode or block mode terminals, both of which reflect a master-slave orientation.

Peer-to-peer connections require a communications network that embodies capabilities such as those inherent in LANs and wide area packet networks. By and large, the WANs established to support block mode terminals are packet based and are thus well suited to support peer-to-peer interoperability. Character mode WANs are unsuitable for support of peer-to-peer communications nor are packet networks able to adequately support character mode applications across the network. Therefore, in this case, the movement to standardization is more easily accommodated within a block mode world than it is within a character mode world.

Database

Another area of significant standards activity is that of databases. The adoption of SQL and the relational model establishes the cornerstone of standards in this area. While standards have established significant standardization in terms of the data interface language, other areas of significance for programming, such as interoperability and distribution, have not received the same attention and do not have communality across the marketplace.

This area of standardization also illustrates the conflicts between standardization and innovation. The emergence of object-oriented databases disturbs the status quo and calls into question the breadth of applicability of the incumbent standards.

Again, converting programs to make use of the relational model is no simple matter. While it is possible to develop migration tools that allow programs with old forms of data navigation to access SQL databases, this does not exploit the capabilities of SQL. To gain the full benefit of the SQL model requires that information be remodeled and that applications be redesigned.

Finding an answer

It is impractical to simply toss a coin when selecting a standard. It is essential that any drive towards standardization be initiated in the context of a well-thought-through architecture for the organization. The trends toward distributed processing and GUIs are immutable. The deployment of these styles of computing needs to be approached carefully by operating within the constraints of available technology and being consistent with the structure of technology placement that matches the long-term direction and shape of the organization.

In resolving these dilemmas, the migration plan will have to adopt a strategy that reflects an assessment of:

- What do we wish to protect and what are we prepared to discard?
- To what degree do we wish to standardize?
- Do we want standards to be vendor neutral, or are we satisfied with proprietary standards?

Taking control and responsibility

Answering questions such as these is, for some organizations, an entirely new activity. For many organizations, the issues of longer term technology architecture and direction are simply left in the hands of the selected supplier. At this stage in the development of standards-based systems, a standards-based policy requires the organization to accept responsibility for its own direction. The organization must clearly understand that it is choosing to pursue its own path through the morass of technology choices rather than simply following the lead of a particular vendor.

Making this decision entails some risk and requires that the organization retains staff with the time and ability to guide the organization. Against these costs will be balanced the benefits that flow from openness. Pursuing this path requires determination and commitment from the entire organization.

Determinants of migration size and complexity

As the scenarios show, the extent of migration activity varies significantly according to the:

- Current architecture
- Target architecture
- Organization size
- Value of technology to the functional area
- Organization complexity
- Extent of change
- Impact on culture
- Cost.

For some organizations, the migration activity may be minor and may not need to be supported by extensive structure and analysis. For these organizations, the extent of planning implied in this appendix may be totally inappropriate. It may be that they can simply "just do it."

For others, the issues of migration and maturity of the standards-based products will be such that, after analysis, the migration costs and issues will loom sufficiently large that the organization will determine that its best interests are served by the retention of a proprietary strategy (at least for the interim—until the costs become less prohibitive).

Baseline characterization

The inventory activities of this phase will provide key information for migration planning on the valuation of existing assets and the identification of risk. From a migration point of view, the necessary inputs may include:

- Valuation of existing investments in hardware, software, applications, development staff, operations staff, users, management, and management process.
- Critical evaluation of existing suppliers, their prospects for survival, and continuity of their product lines. Is the vendor a special-purpose vendor and thus likely to survive in its niche, regardless of standards support?
- An estimate of risk, cost, and opportunity cost relating to the current inventory. For vendors or product lines that may not survive, what is the cost to the organization of loss of impetus as a vendor winds down investment and turns attention to alternative product lines? What is the cost arising from reduced market support? What is the opportunity cost from use of obsolete equipment?

Target architecture — examine alternatives

The selection of a target architecture requires some understanding of migration impacts in order to move towards a practical target. Selection of a target will need to take into account the issues that emerge from the baseline phase while addressing the objectives and targets. Some of the questions that may help the issues emerge are:

- Do we have requirements that can only be addressed with a proprietary-based architecture?
- What is the impact of past investment? What base must be protected?
- What are the general levels of costs associated with different architectures? What is the impact on the total level of expenditure across the organization across time?

Alternative architecture targets may emerge by looking at the organization from various views. Looking at the organization in terms of its functional areas will highlight standardization within a related application set and may subsequently identify pilot opportunities that are not closely coupled with other application systems. Viewing the organization in terms of work organization and the need for application access of each grouping of staff and department will provide input to the needs of the organization in terms of GUIs and integration on the desk.

An inventory-oriented view focusing on the proliferation of platforms will focus on the need for rationalization of platforms and uniformity in infrastructure. Such a view needs to include the network platforms.

A management view of the organization will focus on the integration of information and the needs of the organization.

Opportunity identification

For some organizations, the opportunities for migration will be in the form of specific functional area initiatives with supporting applications. The difference will be that implementation will be based on the adoption of a standards-based architecture. From the functional area point of view, these projects may not represent a significant change from the normal approach of justifying and proceeding with information system implementation. Where such opportunities are limited in scope and proliferation, they make ideal pilot candidates.

For some organizations, open systems adoption will require a gradual modification and migration of the infrastructure. In these situations, there is a significant need for the commitment of the organization to sustain migration over a long period.

Migration options

The evaluation of migration requires that the alternative migration strategies be examined to determine the effort, cost, and adequacy of the approach. This requires research and validation of the elements of each possible migration solution. Typical questions that need to be asked are:

- Is it viable?
- What products does it need? On what standards are they built?
- When will the products be available?
- What can we do to position for future decisions?
- What education and learning must be undertaken?

- How do we introduce the consequent cultural change? What is the cultural change for development staff, operational staff, users, and management?
- What are the relative costs of each option?
- What benefits are delivered by the option?

A caution to the reader

The migration scenarios selected are hypothetical and have been developed for the purpose of illustration only. They do not attempt to portray real life situations. Care must be taken in using the scenarios in that, while the DoD must individually assess its own requirements, the scenarios presume requirements. While the DoD will evaluate migration options based on the latest market knowledge, the scenarios presume the market at a point in time.

The comments and conclusions made about the scenarios are general only, they are not complete. They should not be cast in the light of recommendations. It should also be realized that the solutions presented in each scenario are not necessarily the only ways of solving the hypothetical problems. The investment decision process and relative sensitivity to costs are different for every organization. These scenarios do not provide guidance or commentary on the relative costs of alternative migration options.

Scenario 1: proprietary vendor with a commitment to POSIX

This is a general scenario that covers a medium-sized vendor offering POSIX interfaces to a proprietary operating system as part of a general commitment to vendor-neutral standards. It is assumed that the vendor also commits to XPG and OSI.

In this case, the vendor is committing to comply with the open APIs so that applications written to the standards are portable onto or from their platform. Vendors providing this level of standards support aim to accommodate portability of applications across platforms but have a view that the platform, as supplied by the vendor, is complete.

The alternative view, that standards should be used to allow interchangeable components within generic platforms, has not been considered in any scenario. This concept of openness is not supported by hardware vendors but does receive some support from software vendors and third-party peripheral suppliers.

While every vendor offering POSIX-compliant platforms has a proprietary offering below the interface, the class of vendors represented by this scenario differs from the provision of a POSIX-compliant UNIX in that:

- The capabilities of the proprietary offering are maintained intact within the platform; thus, a single platform can operate in either of the two modes.
- The platform benefits in that the proprietary environment is probably more mature than the UNIX environment. This presumption may not always be correct and will change as the UNIX-based offerings develop.
- The platform will be developed by the vendor in response to two client sets (proprietary and open). It is possible that the proprietary mode will always receive functional enhancement first.
- The development of a new function is limited by the resources of the vendor. The vendor will not normally be able to roll in a function developed by the industry for the UNIX vendors or by the two groupings of UNIX-based platforms (OSF and UI).
- The vendor's solution will not be able to benefit from the ideas of component interchange should the marketplace force vendors along this path.

Current architecture

The current architecture is shown in Figure E-1. The primary characteristics of it are:

- A proprietary CPU running proprietary operating systems with proprietary file systems but with POSIX compliance.
- A platform that is able to include an SQL DBMS.
- Language support that includes COBOL, proprietary languages, report writers, and query languages.
- The platform includes a number of mission-critical applications that operate using on-line update to the databases.
- Normal terminal support that uses block mode terminals, and all applications written to support block mode terminals.

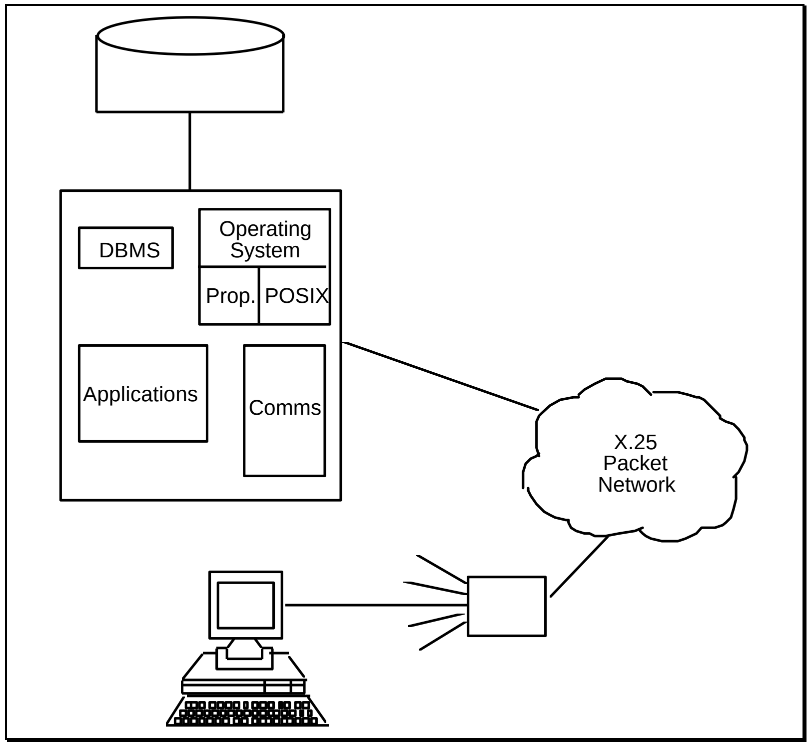


Figure E-1. Current Architecture

- The vendor has committed to support OSI, has an X.400 product in place, and an FTAM product due to be released—it is expected that the vendor will fully support the level 7 OSI protocols a little behind market adoption.
- The vendor has support for X.25, and terminals may access applications via X.25 operating in block mode.

Migration objectives

There is a significant investment in application software; thus, there is a desire to protect this investment. There is no desire to change the user interface for existing applications.

There is a significant investment in block mode terminals. It is required that these be retained for their life rather than be discarded.

There is a desire to use a GUI for some new applications, which creates a requirement for both the GUI and block mode operation to be accommodated.

There is a desire that both old and new applications be able to operate on one platform and share the networks and infrastructure.

There is a requirement that data belonging to old or new platforms be available across both types of applications.

There is a requirement that the software for existing operations, network management, capacity management, storage management, etc. continue in use.

Target architecture

The target architecture requires that:

- The POSIX interfaces be enabled
- The DBMS be SQL based
- The programming language be a standard language
- A GUI be introduced.

Migration options

The scenario assumptions have resolved much of the discussion regarding alternative strategies. The scenario assumes coexistence is available.

Option 1

Leave all old applications intact and write all new applications using the POSIX-defined interfaces. Ignore the need for a GUI and continue to use block mode terminals with the existing networks.

On analysis, this is practical for only a small percentage of applications. Few applications can live within the bounds of the implemented POSIX standards. A number of batch, OLTP, and process control applications cannot operate within the bounds of the available POSIX specifications and/or support. New applications requiring this functionality must use the proprietary facilities.

There are also some conflicts between the POSIX-defined interfaces and block mode operation.

Option 2

Same as Option 1 but also make use of a non-open GUI.

This requires some distribution of the presentation layer. The selected GUI is Microsoft Windows. By using PCs on a LAN with block mode emulation to the host, it is possible to accommodate both the block mode terminal applications and the GUI-based applications, but the GUI is not open.

Option 3

Same as Option 1 but using X/Windows as the GUI from the central host.

This option proved unviable. X terminals were not able to support block mode emulation. Workstations able to support the block mode operation and X terminal emulation could not viably attach through the network to the host-based X applications.

Option 4

Same as Option 1 but using X/Windows and distributed presentation.

Apart from the issues raised in Option 1, the X/Windows-based GUIs are somewhat incompatible with the LAN facilities required to support the block mode terminals. The solution requires that each terminal be replaced by a workstation, with a presentation layer being distributed to the individual workstation. The presentation layer then requests service from the applications in the central host.

By using the existing block mode as the interface, it is possible to use X/Windows over existing applications.

Option 5

Move to OSI network while retaining block mode terminals and supporting X/Windows.

Again, this scenario is only viable where functions can be distributed to the workstation. The use of X/Windows precludes the use of OSI all the way to the terminals. The use of X/Windows also displaces the currently mature network facilities and network management capability.

Preparing for migration

The scenario revealed a number of exposures. The following activities are warranted:

- An assessment of the viability of the supplier. Should the supplier be unable to continue to maintain development of the two product lines, this scenario will revert to be similar to Scenarios 1 and 2.
- An assessment of the vendor's development funding is necessary to determine what confidence there is that new open functionality will be delivered to match the marketplace. It is assumed that the vendor will be prepared to reveal internal information to indicate the viability of the strategy.
- A brief on standards activity is needed to fully understand the complexity of standards compliance in an environment that must also continue to support the proprietary standards.

Preferred migration

The preferred migration option is Option 2 based on the existing network and a non-open GUI.

The improved compatibility with the installed base over the X/Windows options is significant. Given the inability to fully comply with open standards, it is not clear what the benefits are of partial compliance.

The selected approach includes:

- Use of POSIX standards only where the whole application is able to operate within the standard
- The ability to distribute a presentation layer but no obligation to do so for all applications
- The ability to make use of the GUI for new applications but no obligation to do so
- The introduction of standards-based LAN platforms and workstation platforms to replace the existing terminals and cabling system
- The ability to purchase application packages that work to the POSIX interface standards.

This option provides the confidence of staying with the old while being able to watch the emerging marketplace activity in the open arena.

Conclusions

- A migration is not possible without a total commitment to open standards.
- The use of X/Windows does not fit well with the block mode orientation of the vendor.
- The use of OSI requires some distribution of function.
- The movement away from proprietary networks and block mode operation raises some issues of transaction integrity and recovery. Even though a LAN can support these requirements, the devices attached to the LAN may not unless they emulate the block mode operation.

For example, a remote check printing application that requires confirmation from the printing device that printout has completed without a paper jam as a condition of transaction commitment will not be able to obtain the required status advice under several of the migration options.

- The use of a GUI requires some distribution of function.
- The accommodation of distributed and centralized applications is difficult. The mix of proprietary for centralized and open for distributed is difficult.
- The adequacy of the strategy assumes that the vendor will survive.

Scenario 2: complex multivendor installation

This scenario covers a large conglomerate organization having a variety of vendors represented in different parts of the organization. It is assumed that there is a mix of vendors including IBM, Digital, Unisys, UNIX platforms, and PCs.

Current architecture

The current architecture is shown in Figure E-2. The primary characteristics of it are:

- There are no corporate systems. Each vendor's equipment has a reason for being there, but none is seen as the corporate system.
- Each platform has its own network and terminal set. All of these operate in the mode native to that supplier.
- The IBM platforms utilize 3,270 applications with an SNA network.
- The Digital platforms make use of character mode terminals.
- The Unisys 1100 platforms cover a variety of UNIX suppliers. All make use of ASCII character mode terminals and applications. None has an extensive network.
- PCs proliferate throughout the organization and operate standalones as well as in terminal emulation mode for any of the major platforms.

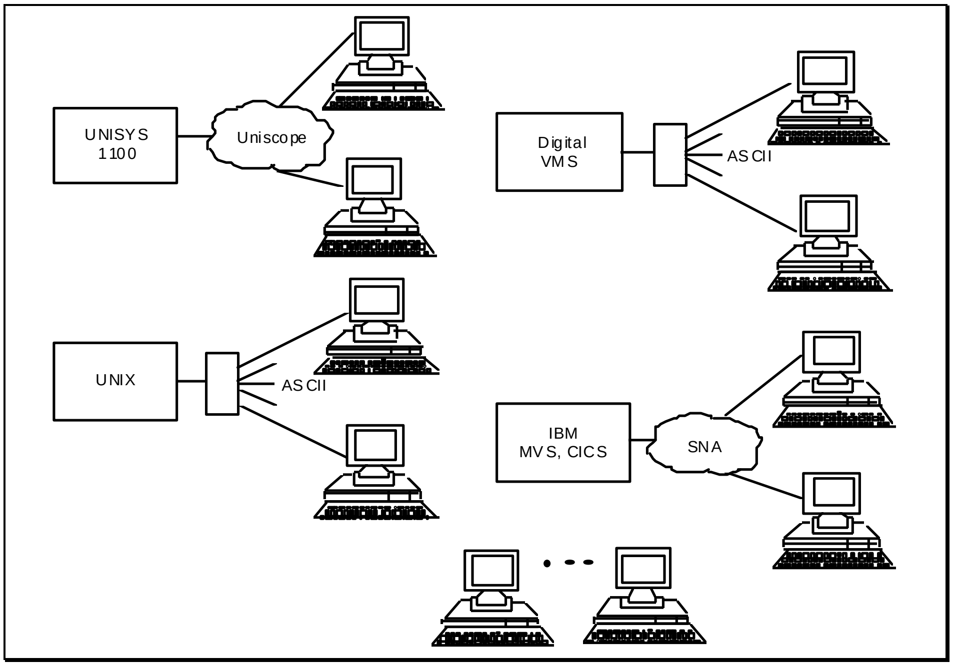


Figure E-2. Current Architecture

- There are no LANs in place.
- There are no shared networks other than at the physical level where TDMs are in place to comb the leased line requirements where these are required.
- Applications cover the range of GAEs including OLTP, interactive computing decision support, office automation, real time, and special purpose. There is no integration of office automation functionality.

Migration objectives

The migration objectives are multiple. None are obligatory, but in order of importance they are:

- Move to a single user interface (preferably a GUI) across the whole organization.
- Move to an environment where any user can access any application.
- Move to a single programming environment so that any development staff can be deployed on all projects.
- Move to a single operational management environment, so that IS operations management can manage the total investment in one coordinated way.

- Move to an integrated information environment where all data can be shared. There is a requirement for both centralized corporate data on the mainframe platforms and distributed work group data on LANs.

The organization has indicated it is willing to redevelop any applications in order to address the migration objectives.

Target architecture

The target architecture is shown in Figure E-3. It is characterized by:

- Multiple mainframe platforms
- A shared network
- A single workstation type able to access applications on all mainframe platforms
- Platforms that provide terminal access from any terminal to any application
- Platforms that provide access to data on any platform from any application or workstation.

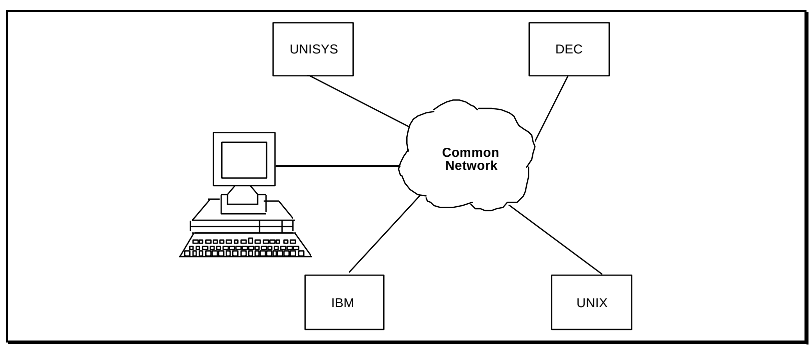


Figure E-3. Target Architecture

Migration options

The alternative migration options are:

Option 1

Implement a shared network that attaches to all hosts and is able to support the variety of terminal types such that any terminal can access any application.

This option proves to be unworkable. The two main problems are the conflict of character mode versus block mode and the need to convert the proprietary protocols and formats.

A network of LANs with bridges and routers can pass character mode traffic in a responsive way but does not accommodate the various protocol converter requirements. Additionally, the cost of the network is significant because of the bandwidth required to sustain responsive network transit. The solution is of doubtful adequacy in addressing the future support of X/Windows unless distribution accompanies the introduction of the GUI. There is no capability of using X/Windows on a broad scale.

While protocol conversion facilities are superior, it is still impractical to provide an "any-to-any" capability. Products are available to support almost all of the combinations, but the technique for addressing the need of each is quite different. In some cases, it requires a back-end solution, while in others it requires a front-end or protocol conversion. Combining them all requires installing some navigational intelligence at the front end and requires significant definitional coordination. Some custom software is needed for ASCII to UTS but can be modeled on available software.

The net conclusion is that this is not a viable approach.

Option 2

Same as Option 1 but convert all character mode applications to operate in line mode with local pad devices.

This approach is assessed as not strategic. It does not move forward; it reduces functionality for some applications and does not facilitate the introduction of a GUI.

Option 3

Review all applications in terms of GAE requirements and work toward a rationalization of platforms by redeveloping applications on fewer platforms

This does not increase openness or integration, it simply reduces the diversity at the cost of significant redevelopment.

Option 4

Redevelop applications on the platform that combine the most mature environment with the potential for future openness. In the redevelopment, use techniques that will ensure future portability, regardless of the standards, through the use of insulation layers and local high-level language facilities. In practice, the selection of a single platform would need to give weight to the extent of the existing investment and the availability of alternative off-the-shelf applications. This option ignores these practical issues for the sake of illustration.

In terms of the GAE functionality covered by the baseline definition, the IBM environment provides the greatest match and maturity. The IBM environment is also supported by all other platforms to some degree or other but mostly acting in 3270 terminal emulation mode. This eases migration phases. The IBM environment is not amenable to open systems development within CICS or IMS.

The Digital environment is assessed as providing significant maturity, particularly in terms of the connectivity options that it supports, while also providing significant opportunity for open development. It combines support for the proprietary solution with OSI and POSIX compliance from within the one platform. It would be the selected platform under this option.

Option 5

Move as many applications as viable onto UNIX platforms and assess the remainder for rationalization onto a single platform. Migrate to a WAN capable of supporting the selected platform's protocols.

Option 6

Move everything to UNIX regardless of suitability and put up with the inadequacies. Implement a standard network and an X/Windows-based window manager. Implement data server functionality across all platforms.

This approach suffers in that it forces distribution onto the workstation in order to get X/Windows functioning. It also requires LANs with TCP/IP for the network with an eventual migration to OSI. These present difficult migration phases for some of the proprietary platforms.

Option 7

Distribute as many applications as possible and, for the remainder, distribute presentation with all the existing platforms being retained as application servers.

This option would permit the implementation of X/Windows with the front-end host then using a variety of techniques for accessing the application servers, including RPC for hosts that support it and terminal emulation for the remainder. This would require that character mode applications be converted.

Option 8

Leave existing platforms, applications, and networks intact but define a new environment with a shared network for use in developing new applications. Over time, the applications will migrate as they reach normal end of life.

The most open new environment is the use of X as a GUI with a presumed distribution of the presentation layer or the whole application. Where access to new applications is needed, a LAN is implemented with access to both the block mode hosts and the new network. Alternative products such as xterm 3270 can be used to provide access from within a window.

The shared WAN does not have to carry either block mode or character mode traffic because these remain on the existing networks. Thus, it can be based on TCP/IP without needing to review enhanced capabilities such as 3270 over the network. It would be possible to run an X.25 network, but this would require TCP/IP to run over X.25 to support the NFS/RPC protocols, which is not preferred.

Existing centralized character mode applications require separate network facilities with access to these from the LAN. Solving the character mode requirements creates a complex solution that is difficult to support.

Preparing for migration

This scenario revealed a number of exposures. The following activities are warranted:

- A critical view of work flow to determine what the real need for integrated access to applications is, compared with the presumed desirability of full integration
- A critical view of management processes to determine what information consolidation is required now and in the future
- A critical view of application-to-application flows, including a forward looking view that postulates future requirements
- A critical view of platform characteristics, GAE requirements, and an assessment of these against the adequacy of products available in the marketplace
- An activity to review all applications to determine the suitability of distributing them to operate within a local

work group or to distribute the presentation layer with application service calls being used to request service from the centralized platforms

- A plan to rationalize the number of platforms over time.

Preferred migration

The preferred migration option represents a combination of elements from the other options and an attempt to get the best of everything. A schematic of the option is shown in Figure E-4. The characteristics of the option are:

- All existing applications remain on the existing hosts. Office automation is to be introduced as a local capability with a corporate electronic mail and document storage/retrieval capability.
- A rationalization project is initiated to reduce the variety of platforms over time. In the meantime, each will be supported with only some modification. It is assessed that resources are better directed to tasks other than redeveloping applications.

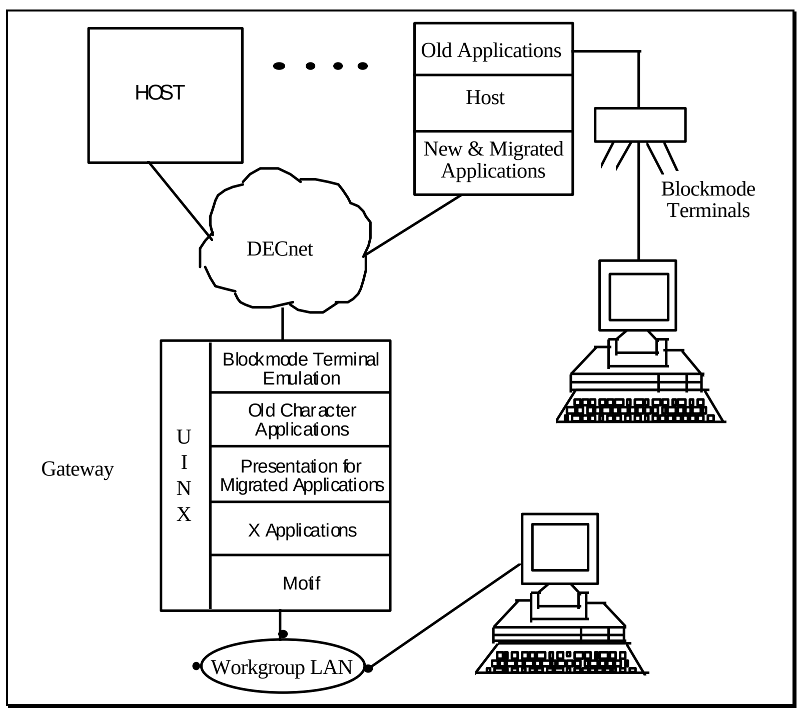


Figure E-4. Preferred Migration Option

- All character mode applications will be reworked to become either:
  - Line mode and centralized
  - Character mode and distributed
  - X/Windows and distributed
  - Distributed presentation (X/Windows) with RPC connection to the centralized application server.
- Preferably, new applications will be implemented based on POSIX, a GUI, and standards but, where close coupling exists with existing applications, there may be a need to continue implementation on other platforms.

Thus, the applications will use RPC to access a POSIX host or some form of client server using block mode or other protocols to access application servers operating on IBM, Unisys, or Digital. Access to Digital hosts can be accommodated in either of the above styles.

- Motif has been selected as the GUI of choice given the presence of Digital platforms. It is based on X/Windows. Motif may not be supported by some vendor environments.
- Existing block mode terminals will be retained where possible.
- The standard LAN platform will provide access from workstations to a UNIX-based gateway local host that will provide access and conversion facilities as required.
- A single network is to be established that will carry all traffic. It will be based on DECnet.

Network considerations

The analysis of network options encompasses a review of open networks as well as the use of proprietary networks. It is assumed that, apart from the existing block mode terminals, the network will need to support TCP for the RPC connections to UNIX and DECnet for similar access to the Digital hosts.

The analysis of the use of a neutral TCP/IP network is difficult due to the scarcity of information. TCP/IP networks are often established on a systems-integration-basis with components sourced from a variety of vendors. Carrying SNA and DECnet traffic over IP is understood to be possible, although the product quality is not known. Carrying the character mode traffic is impractical and carrying the UTS traffic requires protocol converting UNIX minis and replacement of terminals with PCs.

The same limitation for character mode also applies to the use of X.25. While some classes of SNA traffic can be carried on X.25 (3270 and PUT4), and DECnet and UTS can be carried on X.25, the strategy is not favored.

If SNA is used to provide the network, a number of shortcomings exist. There is no way of carrying TCP over the network, thus there is no simple means of carrying RPC. It would be possible to implement an RPC transport mechanism based on APPC, but it is suspected that the approach would also need the IBM CSFI product set. Handling DECnet over SNA is also a problem area unless it is transported over X.25 over SNA. The approach is very complicated.

The engineering solution based on shared bandwidth and separation of the logical networks is also not preferred. It involves a significant outlay for additional equipment and suffers from a lack of flexibility.

The selected approach is to use DECnet as the transport mechanism. It provides good support for RPC and potentially supports the TCP/IP protocols as well as accommodating SNA over DECnet in a variety of forms. It cannot accommodate PUT4 but, in this configuration, this is not an issue. Provision of UTS traffic is by using Unisys 3270 support to replace the UTS terminals with 3270s. This has minimal impact on the Unisys applications.

Other considerations

There is a need to control the development of new applications so that over time the organization moves to a more cohesive architecture. The organization is determined that compliance with standards and uniformity across the organization will not be at the expense of functionality and, thus, has a willingness to continue with some proprietary systems where there is a demonstrated need.

A process of architecture review is to be introduced as part of a tighter approval process ensuring that there is a movement towards rationalization.

Conclusions

- The needs of character mode applications and block mode applications fight each other all the way down the line.
- Introducing a GUI requires distribution that will require redevelopment of the application regardless of whether it is character mode or block mode.
- Introducing distribution requires a uniform transport mechanism. Accommodating coexistence creates a complexity of requirements that may be impossible to meet.
- The standards-based approaches represent a particular style of solution. There may be more appropriate solutions, but they may not be open.
- Distribution requires careful planning and analysis. Again, the various open and proprietary products assume different architecture for distribution.
- While a solution on paper has been identified, it is not completely open; and it requires a significant level of validation to demonstrate its viability.
- The questions of operational viability and the adequacy of the selected products in real life still remain to be verified by test laboratories and pilot projects.
- The process requires significant planning skill as well as access to technical planners who are familiar with the products and the environment. Pursuing the selected path will require major commitment from executive management.

Appendix F: Cost/Benefit Analysis

Introduction to a business case analysis approach introduction

This appendix describes the process of performing a cost/benefit analysis (CBA) of information systems alternatives that support a Business Process Redesign (BPR) and systems technology. It is part of an overall economic analysis framework for evaluating the economic effects of one or more subsystems within an object business system.

The object business system can be thought of as an organization, such as the DoD, the Office of the Secretary of Defense (OSD), or a department within OSD and/or a particular work group within the department that transforms inputs into products and services. Furthermore, an object business system can be thought of as a particular work system or set of business processes that are carried out within a particular organizational context, supported by a particular information systems architecture and technology resources.

The DoD has previously implemented an important information management improvement plan known as the Technical Reference Model for Corporate Information Management. This initiative calls for the financial assessment of BPR and information system investments, denoted as Financial Economic Analysis (FEA). DoD guidance on FEA is found in the draft Memorandum for IRM Points of Contact, Budget Bulletin Number 92-04.

The overall approach for performing a CBA applied to BPR and standards-based architecture planning is discussed with the help of an example.

CBA is a systematic financial procedure for evaluating the costs and benefits of an investment opportunity. The investment opportunity may include changing an organization’s work system or business process, information systems technology, and/or work group resource assignments. It provides the financial information necessary for management to make decisions about the benefits of adopting new business processes, information technology, and work group arrangements in order to improve productivity, accuracy, timeliness, and reduced life-cycle costs.

Performing a CBA for a new system architecture is a complex task, especially when combined with corresponding required changes in the business process and work groups. It is a task that involves defining the baseline costs for the current object business system, and assessing the potential effects of possibly applying different technologies, different standards, different applications, different human resource assignments, different business processes and different levels of technological experience to successfully satisfy the mission of the organization. In this appendix we have:

- Defined the business baseline
- Defined the technology baseline
- Defined the financial and standards criteria
- Ranked the alternative system architectures
- Presented the key elements in performing a CBA
- Presented the key financial measures and risks.

Determining the business baseline and benefits

CBA focuses on the evaluation of alternative investment strategies and management practices aimed at improving user and management productivity and reducing life-cycle costs. A different analysis compares current baseline operational and management costs with the expected costs for one or more investment alternatives.

The framework for analysis in determining the business baseline and benefits is found in Figure 1-1. The process begins by first defining the object business system and scope of the analysis. The object business system in this section focuses on a particular business function.

Second, a functional analysis of current work activities is performed, and the time and costs for performing the work is collected and analyzed. In addition, output volume, work flow times, technology used, and resources allocated to perform the functions are analyzed. From this information, a Cost Breakdown Structure (CBS) is derived that classifies costs according to a life-cycle orientation. The life-cycle costs may be transformed to fixed and variable cost elements to support the FEA requirements.

This process can be very time consuming, especially when the costs for the activities are not recorded in terms of fixed and variable costs. Therefore, the data may need to be converted by an approximation method with reduced accuracy. Costs are then summarized into their life-cycle phases for activities to provide a cost profile for the work processes.

Third, alternative work processes are identified in order to improve overall productivity and reduce costs. This may include new work flows, activities and tasks, and possibly work group rearrangements to support the updated business processes. The fixed and variable cost structure for the alternatives are estimated with corresponding risk. At this time, the business requirements for standards-based systems, applications, and networks may be identified at a high level.

Fourth, a pro forma estimate of benefits and costs for each alternative is prepared. The costs are estimated for each alternative over the useful life of the systems (e.g., 5 years).

Fifth, the CBA for each alternative is computed with associated risk factors for each alternative. The CBA provides a financial profile of effectiveness measures in terms of their cash flow equivalencies. Costs and benefits are equivalent if they have the same effect. Cash flow equivalence compares the costs and benefits of alternatives in the same terms consisting of: (1) the amounts of the sums, (2) the time of their occurrence, and (3) the interest rate. Interest formulas provide the time value of money viewpoint as a standard for comparing alternative investment proposals. The future amount of a sum can be calculated using the compound interest formula (1):

$$FV = PV\,(1 + i)^n \qquad (1)$$

where the Present Value (PV) represents the current or present sum of money, and FV represents the Future Value, given a rate of interest, *i*, for a period of *n* years. The present value (PV) of a sum for *n* years for a given rate of interest can be easily determined by solving equation (1) for PV. This relationship is applied to assess the PV of benefits for the business process alternatives and system alternatives illustrated in the following examples.

Example 1: logistics support services

A BPR study team is assigned to logistics support services to improve productivity. This function is performed across multiple departments. The function is responsible for supplying and maintaining electronic spare parts at selected sites to support the mission of the department.

The cost summary in Figure F-1 represents the baseline costs for the current business process at three sites. The total combined baseline cost breakdown, human resources, and output for the function at three locations is summarized as follows.

COST ITEM	SITE 1	SITE 2	SITE 3	TOTAL
Personnel	$24,000	$50,000	$100,000	$174,000
Transport	$5,000	$10,000	$20,000	$35,000
Facilities	$6,000	$60,000	$60,000	$136,000
IS Services	$5,000	$30,000	$40,000	$75,000
Total Cost	**$40,000**	**$150,000**	**$230,000**	**$420,000**
Number of Staff	160	1,800	2,000	3,960
Number of Parts Shipped Per Year	220,000	900,000	1,000,000	2,120,000

Figure F-1. Summary Baseline Cost, Personnel, and Output
($ in thousands)

Figure F-2 shows the summary baseline costs per part serviced and maintained. Each person services and maintains, on the average, 535 parts. The service and maintenance cost breakdown per part includes:

- Average personnel costs per part $ 82.08
- Average transport cost per part $ 16.51
- Average facility cost per part $ 75.47
- Average IS services per part $ 35.38

Total unit cost (rounded) $209.43

COST ITEM	SITE 1	SITE 2	SITE 3	TOTAL
Personnel	$109.09	$56.56	$10.00	$82.08
Transport	$22.73	$11.11	$20.00	$16.51
Facilities	$27.27	$66.67	$70.00	$75.47
IS Services	$22.73	$33.34	$40,000	$35.48
Total Cost	**$181.82**	**$167.68**	**$140.00**	**$209.43**
Parts Serviced Per Person	1,375	500	500	535

Figure F-2. Summary of Baseline Costs Per Part Serviced and Maintained ($ in thousands)

Figure F-3 summarizes the cost alternatives of two business process alternatives compared to the baseline business process. The PV cost (rounded) for each alternative is:

- Current baseline (PV) $1,677 million
- Alternative A (PV) $1,599 million
- Alternative B (PV) $1,841 million

	COST ITEM	B.P Current Baseline	B.P Alternative A	B.P Alternative B
1	Annual Recurring Cost		$150,000	$336,000
2	BPR Investment & Migration Cost	-0-	$1,000,000	$500,000
3	5-Year Present Value (PV)	$1,676,934	$598,905	$1,341,547
4	PV (2+3)	$1,676,934	$1,598,905	$1,841,547
5	NPV Benefit	n/a	$78,029	($164,613)

Figure F-3. Business Process Redesign (BPR) Alternative Benefits Compared to the Baseline ($ in thousands)

The PV represents the current or discounted value of a set of recurring cash flows for a predetermined interest rate (8 percent) over a period (e.g., 5 years) plus an initial investment cost for migration. This concept is based on the idea that a sum of money in the future is worth less than that same amount in the present.

The annual recurring cost for the baseline case is $420 million. Discounted at 8 percent, plus the migration costs of $0, gives the PV cost for the baseline case $1,677 million (rounded) over 5 years. Likewise, the annual recurring costs for Alternative B is $150 million. Discounted at 8 percent, plus the migration cost of $1,000 million in the first year, gives a present value of $1,599 million (rounded) over 5 years. Thus, Alternative A costs $78 million less to implement than the baseline over a 5-year period. Similarly, Alternative B has a higher cost than the baseline and therefore is the least attractive financially.

Expected 5-Year Equivalent Costs				
			Risk	**Risk**
			(-10%) A & B Low	**(+10%) A & B High**
Work Process		Percent		
Baseline	$1,676.934	100	$1,676,934	$1,676,934
Alternative A	$1,598,905	95	$1,439,015	$1,758,796
Alternative B	$1,841,547	110	$1,657,392	$2,025,702

Figure F-4. Risk Adjusted Cash Flow Equivalent

Defining technology baseline

The process of assessing the benefits of alternative investments in systems and architectures begins with defining the scope and business objectives for technology change. The need for technology change can involve many factors. There may be a need to improve user productivity for accessing data or applications. The need may involve improving development efficiency or promoting portability and interoperability among several systems. Finally, the need may involve improving the operational efficiency and effectiveness of networks, or subsystem components, or reducing the life-cycle costs of systems and/or applications.

Once the scope and objectives are defined, the next step is to determine the target object system and baseline operational costs associated with using and maintaining information technology. This process involves identifying the operational costs for maintaining the hardware, software, and applications. Also, it may include the cost of database access and conversion, the cost of maintaining networks and paying for communication line charges, and the cost for vendor support services including training.

Once the baseline costs for the object system are collected, the next step is to define alternative architectures and systems that meet the business, technical, and organizational requirements and objectives. The system acquisition, operational cost, and utilization cost for each alternative must be collected and analyzed. The initial investments (acquisition costs) for each alternative need to be determined. This involves collecting all non-recurring costs for acquiring, installing, and making the systems ready for productive use. Some of the costs may be fixed charges such as hardware and software maintenance and reuse. Others may vary with the level of system use (variable costs) such as conversion costs, communication access and usage charges, and database storage costs.

Define financial criteria and review open system standards criteria

Prior to performing the cost/benefit analysis and determining cost saving alternatives for the alternative architectures and systems, the financial and standards-based architecture criteria need to be defined. The financial and standards-based criteria need to be incorporated into the business case analysis to support the overall decision-making process. In addition, a method for assessing the degree to which alternative systems support the agreed-to criteria needs to be established. The financial criteria may include cost, productivity, quality, and degree of conformance to standards-based system criteria.

The financial criteria for classifying costs need to be defined. This can have bearing on the overall result. Costs can be classified into their fixed or variable components. Costs can also be classified as direct and indirect, as recurring and non-recurring, and as sunk or past. The fixed and variable costs are based on a level of activity. Those costs that do not vary with the level of activity are called fixed costs; those that do vary with activity are called variable costs. Examples of system costs are fixed disk storage drives, terminals, and workstations. Examples of fixed costs are maintenance costs, depreciation, insurance, and interest on capital equipment. Variable costs are ordinarily defined as those costs that vary in some relationship to the level of operating activity, for example, the network line usage charges, package software and license fees, network support service charges, and computer supplies. Direct costs consist of three components: direct materials, direct labor, and direct expense. Indirect costs consist of indirect materials, indirect labor, and indirect expenses. The prefix direct refers to the fact that the materials or labor used under this classification can be directly associated with the output produced or service delivered, whereas indirect costs cannot. The labor costs for performing the functions or work processes are considered as direct costs. Fringe benefits costs for management services are indirect costs. Both cost classifications are useful; however, when indirect costs are large, the fixed and variable cost structure is preferred. Recurring costs refer to those costs that occur again and again or at specified intervals; for example, the cost of network support services, systems performance analysis, and/or management services activities that all occur throughout the system life cycle. Non-recurring costs refer to "one time" costs that are not repetitive, such as system installation costs, application design and development costs, and application conversion costs.

In addition to cost, productivity and quality standards need to be specified. Productivity is a measure of how well resources are combined and utilized to accomplish specific, desirable objectives or results. It can be thought of as the ratio of results achieved to the resources consumed. The total results achieved are called effectiveness. The total resources consumed are referred to as efficiency. Quality is defined in terms of what is wanted and when it is needed. The "what" is the means for providing the end user with outputs or service that accurately match requirements and expectations. The "when" implies providing users and customers with what is needed on a timely basis; therefore, standards for quality are measured in terms of accuracy and timeliness.

Likewise, the criteria for open system standards need to be established for a given standards-based systems project. The degrees to which interoperability, scalability, and portability are specified in the system requirements need to be determined. Interoperability focuses on the communication methods between machines that provide accurate and reliable transmission of data without affecting the applications that are running. This is the requirement for access or interconnection. It also includes the requirement for distributing or sharing the applications and data across that network. This need to connect, distribute, and share software is the requirement for interoperability. The portability standard addresses the need for application software to be able to run on a variety of computer systems without any work on the part of the user and without any changes to the software. All versions of the software are identical, and the output is readily usable on other machines. Scalability refers to the ability of the same application software package to run with accepted performance on systems of varying size, from microcomputers to minicomputers to mainframes. The degree to which open system standards are represented in alternative systems needs to be established and assessed. The standards for evaluation are found in the Technical Reference Model for Corporate Information Management. The criteria for evaluating standards in this model included level of consensus, product availability, completeness, maturity, stability, de facto usage, and problems and limitations. The standards that are being considered or required to support alternative architectures under consideration need to be ranked for each system alternative. A method for performing this qualitative assessment is shown in Figures F-5 and F-6.

	Standard	Weight		Architecture	
		(1-5)	Baseline	A	B
1	OS/POSIX	5 points	1	8	3
2	Network GOSIP	4 points	8	8	8
3	SQL DB	4 points	8	8	8
4	Languages ADA	3 points	8	8	6
5	User Interface X/Windows	3 points	1	2	6
	* Eight point evaluation scale: 1=lowest, 8=highest.				

Figure F-5. Relative Ranking of Standards-Based Architectures

	Selected		Architecture	
	Open System Standard	**Baseline**	**A**	**B**
1	OS-POSIX	5	40	15
2	Network-GOSIP	32	32	32
3	SQL /DB	32	32	32
4	Language-ADA	24	24	18
5	User Interface X/Windows	3	6	18
	Total Points	**96**	**134**	**115**

Figure F-6. Rank Score of Standards-Based Architectures

Rank and prioritize alternative standards-based technologies

Alternative systems and standards are assessed using a relative ranking method to arrive at a figure of merit. The alternative systems under consideration are matched against the selected standards. A weight is applied for the specified standards. The baseline and alternative systems are assessed on a scale of one to eight (see Figure F-5). The weighted scores are compared to the baseline score (see Figure F-6). This process is illustrated in Example 2.

Example 2: baseline architecture

This system supports the current work process in Figure F-1. It is a large mainframe proprietary computer by one of the leading computer manufacturers. It supports applications. The system supports an SQL database. The WAN and LAN use GOSIP with over 200 active terminals. (Note: Federal agencies are no longer required to use GOSIP; the protocol is specified here as an example only.) The current user interface is propriety and not compliant with X/Windows. The vendor has no plans to meet this standard.

Alternative Architecture A

Alternative A is a multiple minicomputer-based system that supports over 2,000 terminals and personal computers. The operating system is propriety but POSIX compliant. The WAN and LANs support GOSIP. The data based on both systems support SQL, although some vendor options have been implemented. The programming languages are ADA, FORTRAN, and COBOL. The propriety graphic user interface (GUI) is partially compliant with the X/Windows user interface.

Alternative Architecture B

Alternative B represents multiple client/server systems that each support 640 personal computers and over 1,360 workstations. The system has a propriety UNIX Operating system. The WAN and LANs support GOSIP. The database system supports SQL. Languages supported are COBOL, BASIC, C++, and FORTRAN. There is a GUI, but it is not fully compliant with X/Windows.

In summary, the relative ranking in Example 2 of the alternative architectures indicates that the multiple minicomputer architecture (Alternative A) ranks the highest in terms of standards compliance with an index number of 140. Second is the client /server architecture (Alternative B) with an index number of 120. Alternative A is 20 points higher than alternative B as compared to the baseline case of 96 points (index 100).

Perform economic assessment

To perform the economic assessment, we need to include all the costs in each phase of the system life cycle. An overreaching goal of the life-cycle cost (LCC) process is to develop high-quality standards-based architectures and systems based on response to established need. In the DoD, this means deploying standards-based architectures and systems that are competitive in performance, quality, and LCC. The generic LCC model should be applied to assessing the costs of systems from the acquisition phase through the utilization phase. The system's life cycle begins with the identification of need and extends through system planning, systems analysis, systems design and construction, installation, evaluation, acceptance and functional use, maintenance and support, system reuse and, ultimately, phase out. The process represents the life-cycle activities of many systems projects. Although these activities may vary somewhat from one open systems architecture program to another, it reflects a common process for all.

The LCC for each alternative needs to be organized into a Cost Breakdown Structure (CBS). The CBS is a top-down structure that links objectives and activities for each phase of the systems project. It forms a logical subdivision of costs by functional activity areas and major phases. All life-cycle cost elements are considered and identified in the CBS. The costs are coded and entered into a cost/benefit model or database and serve as input to the cost/benefit analysis.

Once the costs for the system alternatives are determined by CBS, the costs and benefits for the alternatives are analyzed and the financial measurements can then be computed. This involves determining the acquisition and utilization costs over the useful life of the system, coinciding with the planning horizon of the organization. Once the costs have been determined over the useful life of the system, the costs and benefits for each alternative can be calculated. In addition, a level of uncertainty can be assigned to the cost elements in the cost/benefit analysis model. A risk assessment can be performed to provide management with a range of benefits that are most likely and least likely to occur. The result of this cost/benefit analysis is then documented and reported to management for decision making.

Summary of financial measures

Assessing the costs and benefits of alternative systems can be represented by one or more measurements using the cost/benefit model. The most commonly used measures are payback, internal rate of return (IRR), and net present value (NPV). A sensitivity analysis can also be performed to determine the range of risk and benefits given a set of risk factors. The payback measure indicates the average of the number of months or years a systems project can take to recover its initial investment. The initial investment usually represents the total cost of acquisition or the cost for planning, designing and implementing, and making the system ready for use.

The IRR is the rate of interest the systems project earns over its useful life. It is the interest rate that makes the equivalent discounted costs and benefits equal; the higher the IRR, the greater the benefits delivered by the systems project.

The NPV calculation represents the net cost equivalent or discounted cash flow value for a systems project. It is one of the most reliable outcome measures of the cost/benefit analysis and is illustrated in the examples that follow. The initial investment costs are subtracted from the sum of the discount cash flows to provide the NPV or net benefit. The NPV takes into consideration the time value of money over the useful life for each systems alternative under consideration. It transforms the costs and benefits for each year into a present equivalent form for comparison. Selected risk factors can then be applied to each of the costs in the CBS. The NPV is then recalculated to produce the risk-adjusted NPV.

The use of sensitivity analysis provides an expected range of benefits, such as optimistic, most likely to occur, and pessimistic. The analysis is performed by assigning probabilities to the CBS for each system alternative. The risk-adjusted NPV provides a level of confidence for decision making.

The initial investment costs, or acquisition costs, are the costs for getting the systems project started, such as acquiring hardware and software. Additional examples include the contract price, shipping, installation costs, license fees, and conversion and/or migration costs. The initial investment costs are the one-time, non-recurring costs for acquiring and implementing system solutions.

The criteria for performing the financial analysis include:

- Agreeing on the cost classification to be used to collect the cost data
- Determining the economic life of the alternative systems and architectures
- Determining the discount rate or time value of money
- Agreeing on the financial measures to be used for comparison, such as NPV.

This page intentionally left blank.

Appendix G: Architecture Security Planning Considerations

Introduction

The purpose of this appendix is to describe the overall architecture security planning considerations that are an integral part of the standards-based planning process. It is essential to realize that IT security is not an add-on that can be fitted or not, like an optional extra for a car. IT security is both a mind-set and a management tool. It is not merely a concern for the confidentiality of data but also for its integrity and, most importantly, its availability.

IT security is not a negative, restrictive management tool but a facilitating one. Its purpose is to find a safe path through the hazards of business and technology problems. Two elements taken together form the purpose of IT security: the first is to ensure the availability of the resources of an organization to the potential user, when required, to the level required, and in safety; the second is to deny resource availability to unauthorized users. In essence, *IT security* equates with *resource maximization*.

The open systems/SBA concept represents a significant pattern or paradigm shift in the way in which 1) information technology is applied to data and information handling, and 2) the organization must be structured to make use of both.

Paradigm shifts have occurred in the past. The first occurred when organizations had to insert “data processing” into a completely manual organization. This produced the “fortress MIS” phenomenon. Security was relatively simple and, in most cases, merely required a wall be built around the mainframe computer.

The second paradigm shift was distributed systems when microcomputers spread like an infection to the extent that, in some organizations according to a recent report, there are more microcomputers than staff. This second shift presented a security problem in that it was no longer possible to put a wall around all the places where computing equipment appeared. Even IT planning became disseminated.

With the introduction of cooperative/networked processing on top of the unassimilated microcomputer spread, the pressure for change became so great that the degree of shift, or change in the paradigm, ushered in a new era in information handling. All that had gone before was referred to as Era I in DMR's *Strategies for Open Systems*, and all that followed the paradigm shift is Era II. This second era is one where the whole organization will be involved in information handling technology. If we were reliant on the computer before, we will be doubly so in the future. The organization will be planned around information flows and be fully dependent on IT technology. We will have come so far that it will be impossible for us to go back.

Under these circumstances, the applications that will be developed must be as reliable as possible while being flexible and responsive to change. This means that information protection requirements must be considered from the very beginning of IT planning, through to the stage where all the applications that are spawned are operational, and beyond.

The basis of the Era II environment is that a standards-based, networked infrastructure will become the norm, and that hardware, software, and applications can be "plugged in" easily. This has significant impact in terms of providing sufficient levels of security.

Security must be built into the infrastructure and into each feature using the infrastructure. The only effective way to do this is to insert security into the total IT process from architecture planning through to implementation.

If further justification is still needed for the use of IT security at all planning phases, consider that the thrust of the new SBA approach is to design for continuous change. Change means possible danger; if it is not monitored and controlled, a false step may lead to organizational damage and loss. The proposed control is through principles, generic models and the adoption of standards, and continual iteration. The result is a process that creates a systems environment that evolves and changes continuously rather that being cast in concrete. Under these circumstances, the widespread use of IT security is essential.

IT security architecture must produce the following for every application system or group of systems:

- A clear understanding of the security requirements and architecture for each application system and the IT security results of any interaction
- A detailed depiction of:
  - The IT security services and resulting mechanisms required
  - The boundaries of the IT security service
  - An overview, where possible, from beginning to end of the application or group of applications of the IT security service required.
- The capacity to apply different methodologies to the various application systems depending on and focusing on implementation requirements.

Planning the new architecture

Many organizations are now beginning to realize that they are all competing together in time. Their CEOs are demanding IT results now. The old static linear model, because it took too narrow a view of the business world, is now obsolete. Non-performance, or some form of extended response time, is no longer acceptable with the shorter planning cycle predicated by the new paradigm.

In such a speeded-up environment, it is easy to overlook the importance of IT security. In the push to get results, IT security and quality assurance are usually among the first things to be dropped or to which only lip service is paid. Business professionals who know what end results they want will often push for faster delivery times and deliberately overlook certain technical requirements for data and information protection. Their aim may be oversimplified as *"getting a working application as quickly as possible and with the minimum expenditure."* The technical specialists, on the other hand, are looking for the most efficient and effective technical solution. IT security can often be overlooked by both groups to the detriment of both their aims. Because of this, it is important to include on the AWG at least one IT security specialist who can identify the requirements rather than wait for a non-specialist to become familiar enough with the technology to be able to perform this service. It is easy for the unpracticed eye to overlook a situation that is a security situation in the making. Consideration of the five models and the architecture principles that lie at the heart of the standards-based architecture approach will show how intricately intertwined IT security is in the use of that approach.

Business model

This model identifies the business functions performed by the organization in fulfillment of its mandate. It also shows the informational flows required by each function and their interlinkages. This level is also the starting point for analyses of the impact on the organization of loss of each of the business functions. A business impact analysis of this type helps identify the levels of security required by each function. Coupled with an analysis of the recovery options, this will result in the development of contingency plans for the operation of each of those functions and for the organization as a whole. It can also be the starting point, depending on the criticality and size of the development effort, of a development contingency plan (see “Architecture Framework” below) designed to protect the development investment.

All through the planning process, the planning team must continually ask such questions as:

- Is this legal?
- Is this safe?
- What could go wrong?
- What are the risks attached to this decision and have they been evaluated?
- What is the level of risk involved in each case?
- What are the data protection, security, and safety aspects of the alternatives/proposed action?
- Which alternative is better from an IT security point of view?

Architecture principles

These act as the guides for the subsequent IT architecture views that will be developed. They should include the principles that begin to define the type of IT security or data protection architecture that the organization needs to support. To what level, for example, will subunits of the organization be allowed to handle their own IT security and how much, if any, central coordination will be provided? It is important to begin thinking of these things at the earliest possible stage. Protection and safety requirements can then be built in relatively easily, usually more cheaply, and certainly more effectively, than if they are retrofitted.

Work organization

This model provides an indication of the impact of the proposed changes on the organizational structure. Here the primary IT security concern is accountability. This is mainly a factor of responsibilities and their separation; for example, audit responsibilities should report to the highest level in the organization and should be independent of the line organization that must be audited. This avoids the situation where any individual or group is required to be judge and jury in its own case. The reporting responsibilities for security in general, and IT security in particular, are also important. Those positions responsible for granting access to the database, the issue and currency of passwords, and key management, for example, must be identified. There will, however, be other less obvious occurrences that must be identified and dealt with appropriately.

An important decision at this stage, if it has not already been mandated, is who is responsible for security. A number of legal decisions have been handed down in the United States where CEOs, whether they were aware of their responsibility or not, were fined and jailed for not adequately protecting their organization’s data/information when “disasters” occurred. Consequently, if the decision is made that the user manages IT security with IT playing an advisory role, it is important to identify where the responsibility lies to ensure the user takes good advice, and who enforces it. If this is omitted, the lack of clear-cut responsibilities will usually result in time-wasting wrangling or a standoff in which nothing useful in the way of protection is achieved.

Information model

This model identifies the information requirements for the organization. For each data group identified, it must include the requirements for security as well as the data and information required by, for example, audit trails. Consideration must also be given to the advisability of mixing data and information of varying levels of sensitivity. Data aggregation can result in levels of sensitivity that the component data items do not attain.

Application model

This model analyzes and describes the functions and sub-functions that will be supported or automated through information technology and groups them into potential system applications. As part of this process, all logical dependencies and relationships among the application opportunity areas are identified. Defined at this stage are the scope and interfaces of applications that then provide the basis for detailed design. Identified at this time are IT security criteria that include:

- The sensitivity levels of the data handled by the various applications and the resulting sensitivity level of the applications
- The impact of linking applications of disparate sensitivities on potential users and on hardware and software choices
- The known security/protective strengths and weaknesses of the proposed hardware choices.

Technology model

The three components of the technology model define the hardware, software, and communications environment required to support the organization's business. Each element of these components requires an IT security profile showing not only its strengths and weaknesses but a general picture of what it can and cannot do and the way in which it does it. Thus, the hardware profile must include a definition of the security required to protect each element in conformity with the requirements identified for the business as a whole. This security profile, if not already identified, must be identified for each element considered by the planning team. The information derived from these profiles, if properly used, can improve the efficiency and effectiveness of the standards-based architecture being developed and will play a part in subsequent development decisions.

Implementation

There are seven phases in the planning process to implement a standards-based architecture in an organization:

1. Architecture framework
2. Baseline characterization
3. Target architecture

4. Opportunity identification
5. Migration options
6. Implementation planning
7. SBA administration.

The models discussed above fit into the target architecture phase and form part of the deliverable for that phase, the *Target Architecture Document*. However, all the statements made about the need to include IT security and information protection architecture considerations at the earliest possible point in the planning process still hold true. Consequently, elements of IT security will be found in each of the other six phases. Each phase is discussed in more detail below.

Architecture framework

This is a general definition of the current environment and the architecture direction to be taken for the target architecture. Any lapses in the current environment, as perceived by IT security, must be identified so that corrective action can be included in the new standards-based architecture. This means that a security review of the environment must be carried out for the organization or at least that area of it covered by the architecture being developed.

In developing the deliverable, the business and IT issues must be identified and the areas of interaction described in some detail. Where there is concern, for whatever reason, the causes must be outlined. Some problem areas will be apparent only as the result of identification by the security review, and some areas of general concern may have an IT security mandated solution.

The general description of the current IT organization, environment, and technology must include IT security, its responsibilities, and who is responsible for the delivery and enforcement processes included within it. There are a number of areas where IT security should operate, and its presence or absence should be noted; for example:

- Security administration roles and responsibilities
- Software development
- Change control

- Physical access controls
- Logical access controls
- Reliability and availability analyses
- Startup and shutdown procedures
- Security violation detection
- Protection from possible capture and/or overrun
- Key management
- Damage limitation
- Network management
- Recovery procedures and contingency planning.

These areas and others should be included the review of existing standards and any absences noted.

The review of existing opportunities should consider the impact of their implementation, from an IT security viewpoint as well as from others. At this point, all data elements to be handled by the standards-based architecture, which usually means the organization's total data holding, should have been reviewed and a sensitivity (confidentiality) level assigned. This, along with integrity and availability, determines the level of IT security required for the data covered by the architecture and the systems that handle that data. Without such determinants it is very difficult, for example, to be sure that the correct level of countermeasures has been applied. The cost of implementing the necessary data protection capabilities may vary significantly between the available opportunities. A wrong decision could result in significant additional, and unnecessary, costs in some instances. Security can be expensive, and money spent on protecting information assets that do not have a high value for one or more of the determinants may well be wasted. Also, a wrong decision at this point concerning opportunities could well alter a preference list based solely on other, non-IT security criteria.

Since IT security is really "good clean living with the computer," the architecture principles must include those that will protect the data and information in terms of the appropriate levels of confidentiality, integrity, and availability.

The other determinants of the level of IT security required in a system are:

- **Accountability:** This concerns the ability to identify and authenticate the source of an action and is essential to the audit process.
- **Access control:** This concerns the control of access to facilities and to components of systems. The controls may be mandatory (MAC) and rule based (RBAC), or discretionary (DAC) and identity based (IBAC). The controls may include labeling requirements and the restriction of downgrades and upgrades.
- **Non-repudiation:** In the transmission of data and information, it is important to know precisely who originated it and who received it. Therefore, proof of origin and proof of receipt are vital.
- **Assurances:** There must also be ways of assuring the users that the system architecture and the application planning and development process (systems development life cycle) can be relied upon to produce applications safe to use.

A potentially important consideration at this stage is the production of a development contingency plan. Depending on the size of the development effort and the criticality of the work being developed, a contingency plan should be put in place to ensure that the development work may be continued with the minimum of disruption and extra expense in the event of an emergency during the development period. As the development process continues, the cost increases. The loss of most or all of this development effort could be a severe setback to any development program because the replacement of the lost work may be impossible if the additional development funds are unavailable.

Baseline characterization

This activity defines the existing applications and technology platforms that form the foundation or baseline from which the standards-based architecture must develop. This definition phase includes a description of the baseline IT security measures in place for the protection of these existing applications and technology platforms.

Consequently, any imperfections in IT security terms and in the protective requirements of the baseline and the minimum level of IT security required across all the applications and platforms, must be identified. This may already have been done as the result of a security review or audit of some type. If it has not been done, then it must be done as part of this activity. Failure to do so runs the risk of building a new edifice (architecture) on faulty foundations. Deficiencies in the IT security baseline may then be made good before the new development begins or be planned as part of the new development work. Either way, the omissions will be remedied.

Target architecture

Using many of the directional elements developed in the architecture framework phase, this phase defines in greater detail the architecture aimed at or targeted. It should represent the *idealized vision of the architecture to be implemented* with the proviso that this idealized vision must make allowance for IT security requirements.

In developing a standards-based infrastructure architecture, the AWG takes all business, work organization, application, and information models as input, all of which have been considered from an IT security viewpoint. The target architecture phase uses those models to develop the architecture for the generic application and technology environments. In addition, the target standards and technology platforms on which those environments will reside are fully described. In this way, the IT security requirements are carried through what has been described as *"the essence of SBA planning."* Figure G-1 illustrates the familiar standards-based model, and every element indicated has an IT security aspect.

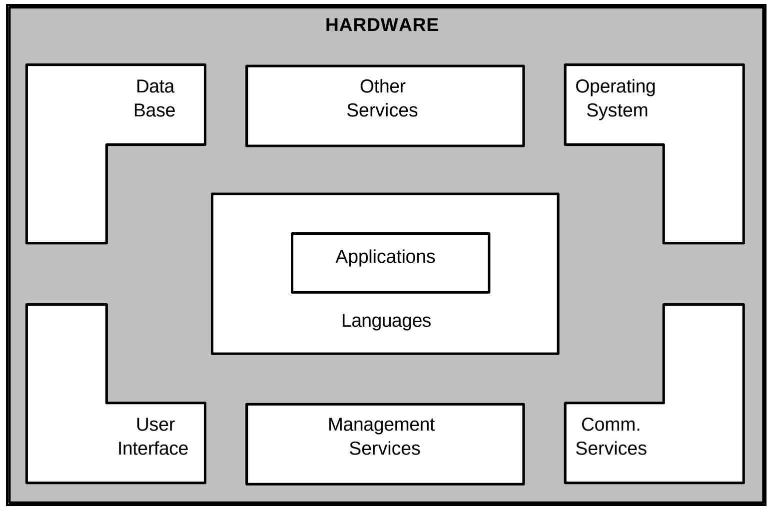


Figure G-1. Standards-Based Model

Opportunity identification

This phase takes a closer look at the opportunities identified in the previous phase, the target architecture. The opportunities identified may require researching and testing. This classifies them according to a number of criteria, including IT security criteria. In the case of software, the evaluation criteria, rationale, and guidelines for use are derived from DoD 5200.28.STD *Department of Defense Trusted Computer System Evaluation Criteria.* The IT security criteria for databases are provided in NCSC-TG-021 *Trusted Data Base Management System Interpretation Criteria.* These sets of criteria, depending on the circumstances, can have a significant effect on architecture flexibility and interoperability and, of course, on the costs.

Migration options

This phase involves sizing migration steps and identifying the “trigger points” on the implementation path where specific actions must take place for the successful implementation of the standards-based architecture. The migration path must allow for organizational change and must also be flexible enough to accommodate changes in the architecture itself that occur as the migration plan is being implemented. There are four areas where migration activity may be focused.

Work flow and organization

This includes the organization of work procedures and business operations at the user level and how users conduct business activities with regard to the active use of information technology. It is important to ensure that the organization of the work flow does not contravene any of the IT security principles, policies, and guidelines already identified. It is easy, when moving from generalities to the next level of detail down, to miss the observance of some security criterion agreed upon at an earlier stage.

Data and information

IT security must be sure that the data and information resources of the organization are not put to any uses that run contrary to IT security requirements and guidelines and do not contravene good management practice.

Applications

These are the tasks performed by IT or to which IT is *applied* in support of the business functions of the organizations. IT security must monitor a number of aspects of application development to ensure that reliable systems are produced to the correct level of security. Therefore, allowance must be made for IT security to perform such tasks as:

- The development process to ensure, for example, that no Trojan horses have been inserted in the code or security features disabled
- The quality assurance process
- The organization and level of separateness of the development, testing, operations, and maintenance units
- Applications handling data of disparate sensitivity levels are not linked.

Technology platforms

The underlying hardware, communications, and system software components used by the delivered applications have security strengths and weakness. IT security must ensure that the secure limits are not exceeded or liable to be exceeded. Thus, every effort must be made to avoid a security failure or incident.

Implementation planning

This phase, harvesting the benefits from the new architecture, endeavors to identify the short-term gains achieved. Once these have been identified, the focus becomes broadening the awareness of the successes throughout the organization to induce "*culture change*." An IT security awareness program should be considered as part of this. Part of the reason for the success of that particular project is improved availability and integrity measures built in as part of the application development process associated with standards-based open systems.

SBA administration

"Reality testing" the elements of the standards-based architecture once they have been implemented is done by conducting a comprehensive review of the *Architecture Framework Document* produced in Phase 1, as well as the *Baseline Characterization Document* produced in Phase 2 of the overall implementation process. The output is a self-critical document used to modify the overall *Architecture Framework Document*. This phase closes the loop in what is a cyclical process. Modifying the *Architecture Framework Document* starts the process afresh. IT security must, therefore, be represented in this phase, as in all others, to ensure that IT security requirements are not overlooked. They may be given due attention during the first iteration of the cycle but can subsequently be erased if they are not given additional attention.

New legislation or changes to old legislation may also require changes to the IT security infrastructure. Technological developments may necessitate changes or modifications to the IT security approach taken. Again, is flexibility is emphasized. Although IT security requirements must be considered and included at an early stage, they cannot be considered “set in concrete” or otherwise immutable.

The provision of IT security capabilities should not be seen as a hindrance to a project or as an unnecessary budget item. They identify an integral component of the information itself, and information handling in general—ease and safety of use. Our increasing reliance on computerized applications demands that this component be present so that effort can be concentrated on using information technology to the fullest, rather than worrying that the organization will be left high and dry by IT failure.

This page intentionally left blank.

Appendix H: How To Do SBA Administration

SBA administration

Most organizations recognize the "need for SBA governance" on or about the time that the initial SBA planning project comes to a close. It is strongly recommended that the DoD adopt a mechanism for keeping the SBA up to date.

It is not uncommon for an organization to establish an SBA administration function that coordinates the review of SBA-related projects and resets project priorities based on architecture evolution.

Typically, this coordination is managed through semi-annual SBA review meetings held with SBA representatives from each of the major functional areas participating in the SBA effort (representatives are selected by the ASC). SBA representative are responsible for keeping the SBA administration function abreast of changes in project status and direction. In turn, the SBA administration uses the representatives to execute changes in the general SBA strategy (consult the *Implementation Plan Document* for more details). The final pages of this SBA Guide describe a recommended process that can be used to support the goals of the SBA.

Process overview

An SBA Management Team (SBAMT) will be established to maintain the SBA. This team will work directly with project managers responsible for developing SBA projects as well as with the functional managers and their staff responsible for overseeing project implementation.

It is paramount that the SBAMT build into the overall administration process a review system to ensure compliance with the objectives set forth in the *Architecture Framework Document, Target Architecture Document, Opportunity Identification Document, Migration Options Document,* and *Implementation Plan Document.*

Monthly project coordination meetings will be held between the SBAMT and all project managers developing SBA-related efforts. The purpose of these reviews will be two-fold:

- Provide an opportunity for project managers to report any issues that will impact the delivery of their projects to the SBAMT, who will approve changes to project plans
- Create an environment whereby SBA project managers can meet to discuss cross-project issues and actively identify opportunities to reuse code and build integrated systems.

On a quarterly basis, the SBAMT will sponsor a status review with the executive sponsor. This quarterly review will provide top decision makers within the organization an opportunity to review the progress of key IT initiatives while lending guidance to the SBAMT.

When the SBAMT is not meeting with project managers or the executive sponsor, they are updating the SBA project plans and communicating all changes to these plans through a myriad of communication vehicles intended to provide needed information to all members of the organization's stakeholder community. (See the "communication vehicles" part of this appendix for more details.)

Key elements of the SBA management process

Following are several important elements in the SBAMT process:

- Establishment of the SBAMT
- Addition of SBA to the duties of the executive sponsor
- Implementation of the project coordination meetings
- Institutionalization of the quarterly SBA reviews.

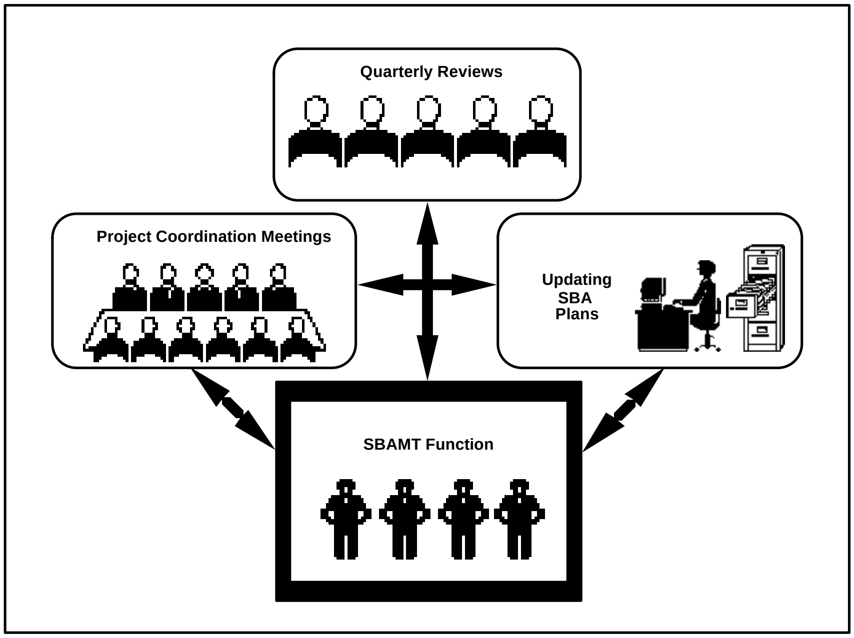


Figure H-1. The SBA Management Team (SBAMT)

The SBAMT

The first step in the SBA administration process is to establish an SBAMT. The SBAMT is charged with keeping the SBA up to date. This is done by managing the coordination of the projects defined in the SBA *Implementation Plan Document*. The people assigned to this function will employ such devices as monthly meetings with SBA project managers as well as quarterly reviews with the executive sponsor in order to ensure that the SBA projects are evolving as planned.

The team should be staffed with experienced planners and technologists who have a deep-rooted understanding of IT implementation projects (i.e., data processing, communications, and systems analysis). Typically, the team is situated in the IT systems development area enabling it to oversee the development activities. If not, standards and policies defined by the SBAMT could be ridiculed because the process "was not invented here." If reorganization occurs, it is important that the SBAMT be placed with the highest ranking IT officer to ensure continued execution of the SBA plans.

Many organizations are beginning to place SBA administration functions under the command of the senior-most executive (i.e., the CEO) in order to ensure that the most crucial IT applications are being developed in unison with the organization's strategic plan. This is highly recommended and represents the best case scenario.

Once established, the team must conduct a general assessment of the SBA projects to see if, in fact, the projects are being implemented in compliance with the overall architecture. This is done by mapping project progress against the implementation plans as well by as the team asking itself (and the responsible project managers) some hard questions like:

- Is the architecture framework still valid? Should any of the architecture principles be modified? Which ones and why? What has changed?
- What are the benefits to be had from changes to the implementation plans? Are there any cost savings, value-added benefits, or softer, long-term intangible benefits?
- Have IT standards been materially implemented in the organization? How far along the standards road have we traveled thus far? How far, given this "process check," do we have yet to go? Have we gleaned 80 percent of the benefit already, or is there still payoff down the road?
- Has the enterprise recognized any benefit from the work achieved?
- Given the current state of implementation, have any other payoffs been obtained that may not have been originally predicted?
- In general, do the plans and their delivery schedules appear to be changing?
- Have any standards, targeted as important, not yet matured as much as originally anticipated?
- What is the status of the technology that was selected for implementation? Has it "shown up on time" in the marketplace? Have we secured its acquisition?

After these questions have been answered, adjustments to the original plans should be made (i.e., if a given project is not maturing as originally scheduled, specific steps must be developed to produce “workarounds”).

Primary responsibilities

- Conduct monthly project coordination meetings
- Conduct quarterly executive sponsor meetings
- Update SBA plans
- Communicate SBA changes to the stakeholder community
- Review SBA project status
- Facilitate cross-project sharing of information/code
- Identify opportunities to consolidate systems development efforts
- Assist project managers in adjusting SBA project plans
- Coordinate complimentary voice and data development efforts.

Executive sponsor

In industry, perhaps the largest constraint in SBA implementation work is senior management’s unwillingness to participate in the review and nurturing of the IT architecture. To keep SBA in the forefront of activities in the systems development arena, this attitude must change.

An IT steering committee must be formed, charged with overseeing the prioritization of SBA projects, as well as final approval for all changes and adjustments to the SBA project scope and delivery schedules. This duty would be appropriate for the executive sponsor. This team of senior officers should be prepared to commit the necessary resources required to make SBA a success.

Typically, the steering committee (executive sponsor) members participate in quarterly reviews of the SBA project status and actively seek to incorporate input from the quarterly SBA reviews into their budget/planning (i.e., POM) process. These decision makers assist the SBAMT in implementing the necessary changes to the SBA by communicating shifts in priorities to their subordinates.

In this new kind of “top-down,” “function-driven” environment, assessment and review become less personally and politically charged. The result is that the

SBAMT process becomes easier to conduct successfully. Ultimately, this form of organizational behavior leads to the establishment of a successful and repeatable implementation process.

Primary responsibilities

- Participate in quarterly SBA reviews
- Make decisions regarding SBA project priorities and adjustments
- Oversee SBA project implementation within the functional areas of the enterprise.

Quarterly SBA reviews

Quarterly SBA reviews are a vehicle to help executive sponsor members keep abreast of SBA progress and be aware of all the changes that occur during the SBA project evolution. Information conveyed in these reviews should be incorporated into the budgeting process within the enterprise. In this way, the enterprise will reduce the dollars being squandered on insignificant IT projects.

Also, these reviews are an important means by which the SBAMT can gain an understanding of the desires of senior officers (i.e., balance current priorities with new requirements). This insight will be needed to better manage changes to the SBA project plans and to define new SBA projects.

Primary objectives

- Executive management review of the SBA progress
- Approval and prioritization of new SBA projects
- Approval and prioritization of changes to existing SBA plans
- Providing a means for functional areas to articulate new IT requirements.

Monthly project coordination meetings

Project coordination meetings are held between the SBAMT and all the SBA project managers responsible for building SBA projects. These meetings are a way for the administrators to understand the issues affecting SBA efforts, enabling them to make changes to the SBA.

Furthermore, these meetings are used to encourage project managers to discuss interproject issues, like software reuse and data integration. When this communication vehicle hits its stride, it can be used to deliver information regarding new IT standards and policies to all project managers represented in the coordination meetings.

Primary objectives

- SBAMT review of SBA projects (plans and budgets)
- Announcement of adjustments in SBA plans
- Cross-project discussions on coordination issues (i.e., data sharing, etc.)
- Delivery news on IT related issues (i.e., standards adoption, etc.).

Communication vehicles

As mentioned earlier, it is extremely important to staff the SBAMT with seasoned IT professionals. To do otherwise can be disastrous. Team members must come to the planning table with experience in technology planning and the sensibilities to understand the inherent cultural and political climate.

The next most important factor in conducting successful architecture administration is the establishment of a set of effective communication mechanisms that can help the administration team distribute important information, such as project planning documents, and receive critical feedback without having to become immersed in the typical "red tape" that such work usually entails.

Figure H-2 highlights this issue and suggests several ways the Marine Corps can keep the communication lines open while effectively distributing valuable information about the status of its SBA projects.

Quality review meetings

Sometime during the first year of SBA administration, the SBAMT should develop a quality review process that will be applied to each SBA project as it matures through the phases of the project development life cycle. This "process check" should conform to existing Total Quality Management (TQM) initiatives and, as such, provide a "quality assurance" dimension to the overall architecture administration process.

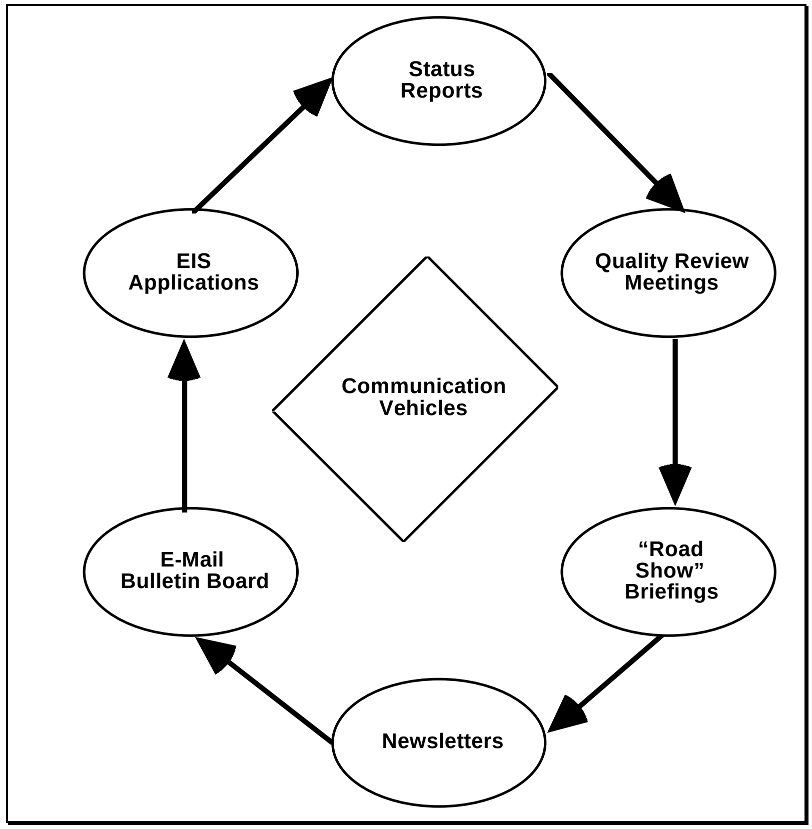


Figure H-2. Some Important Communication Vehicles

A review process based on the Continuous Process Improvement Cycle (see Figure 8-1) is recommended. The notion is that a project is planned, work begins, the result is checked against the plan, and opportunities for improvement are defined and acted upon through modifications to the next plan (or project phase, whatever the case may be). The use of this technique will help the enterprise learn from its SBA experiences.

Each review meeting can be used as a way for the SBAMT to communicate suggested changes in the project development process to SBA project managers (internal as well as external personnel), contributing to the creation of the "learning organization," which is fundamental to TQM objectives.

Status reports

Status reports are another way to improve communication within the SBA development environment. By documenting such things as causes of project delays or scope changes, the SBAMT can begin to define ways to proactively address them. These “lessons learned,” together with the modified plans, should be included in a quarterly SBA status report and delivered to all designated personnel.

Often overlooked, documenting the “lessons learned” (see Figure 8-4) becomes very valuable to future project development teams, particularly when defining modifications to SBA project plans helping future project managers to “never make the same mistake twice.”

“Road shows”

Another important way to inform enterprise personnel about the significance of SBA is to establish an SBA awareness program (or “road show”). The road show will involve the creation of an SBA briefing that describes the SBA process and explains the impact it has on the enterprise. (See Figure H-3.)

The SBAMT will schedule briefings at all major sites. All personnel would be expected to attend one of these briefings. Once all personnel have been exposed to the SBA project, the next phase of the awareness program would take the form of annual status meetings delivered at the same sites.

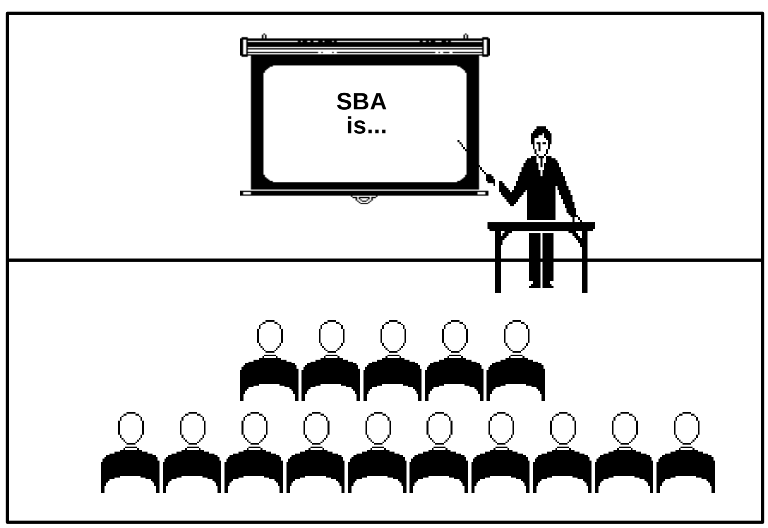


Figure H-3. The SBA “Road Show” Will Take the Message to the Troops

Newsletters

An SBA newsletter could also be created as a means of keeping all personnel informed of the SBA progress. The newsletter could be published quarterly, and its production should coincide with the IT executive sponsor meetings. This way, news concerning executive management decisions about SBA events can be delivered to the entire community.

Electronic bulletin boards

An electronic bulletin board dealing with SBA subjects can be established within the E-mail environment. (See Figure H-4.) It can become a very useful broadcast mechanism, since many personnel use it on a daily basis. In fact, many organizations in the commercial world use such devices as a way to solicit improvement ideas from personnel, transmit newsletters, distribute results from quarterly reviews, and deliver project progress reports to SBAMT-like groups.

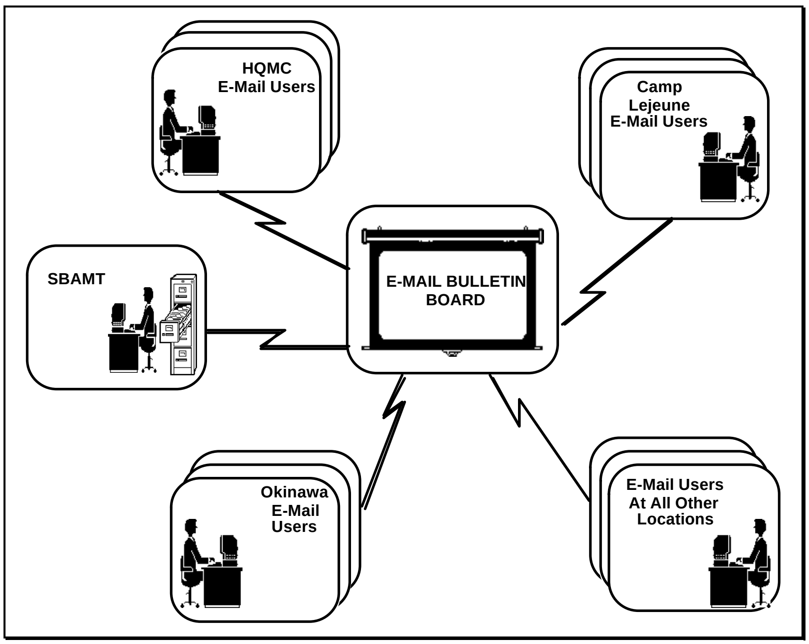


Figure H-4. The E-mail Bulletin Board Posts All SBA News for All Personnel to Access

EIS applications

The development of an SBA Executive Information System (EIS) is another effective communication tool. The primary focus of such a system is to provide an electronic means of keeping senior management aware of changes in SBA projects.

The typical EIS system is easy to use, has user-defined triggers and a myriad of other features that make such a system a very useful tool. (For example, each executive can define areas of particular interest so that when one of his SBA projects is affected in any way, an electronic message is sent to his computer; similarly, other changes that are not of interest never show up on his screen).

Architecture remodeling

When should you remodel? When any of the principles developed in the architecture framework phase have changed. Another reason could be a major change in technology significant enough not to have been anticipated in the target architecture phase; however, such changes will become increasingly rare. One of the major benefits of standards planning is that standards, unlike the underlying technology itself, change far more slowly.

In theory, one should never have to change the architecture if the architecture principles do not change; however, they do change from time to time. When this happens, the SBAMT should discuss and confirm the perceived changes with the SBA executive sponsor and all IT project managers before taking any action.

This page intentionally left blank.

Appendix I: Sample Deliverable Table of Contents

This section provides general outlines for each of the deliverables in the SBA planning process. These may be amended and customized by the AWG for presentation to the ASC. The individual circumstances surrounding the organizational culture and IT environment will also influence the deliverable.

The standards-based architecture

The standards-based architecture is composed of seven deliverables, which are released on a phased basis. Figure I-1 outlines the individual components of the model.

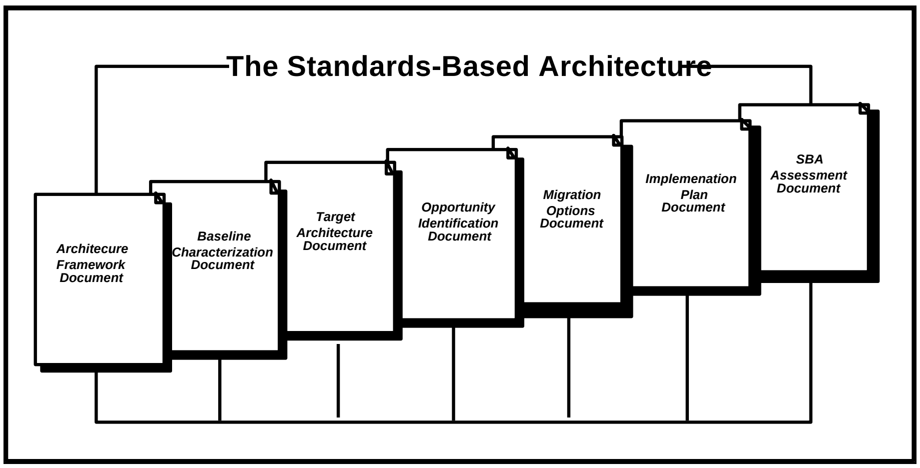


Figure I-1. The Standards-Based Deliverable Set

Staged deliverables throughout the process

A key aspect of the standards-based planning process is the manner in which the architecture is developed. It is recommended that at each phase of the planning process an interim deliverable be produced by the team. Figure I-2 illustrates the phases and their associated deliverables.

Deliverable style All of the deliverables should be “executive style” in scope, easy to read, and highly visual in nature. The key attribute of these deliverables is that they are distributed across the organization and are used to communicate the chief attributes of the architecture to the various constituencies within the enterprise.

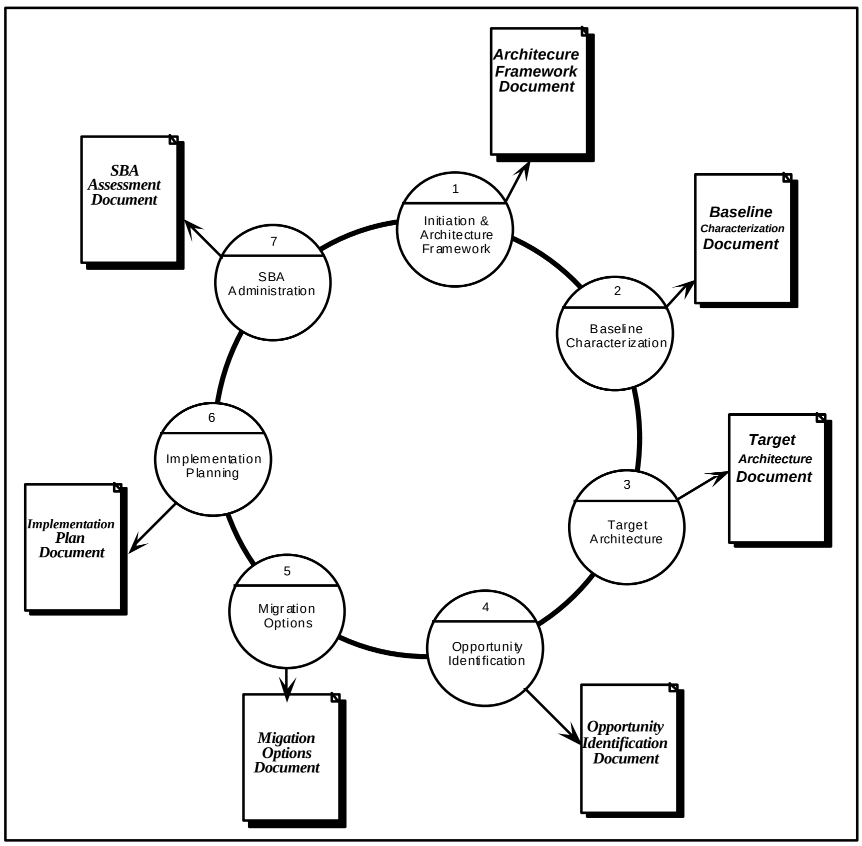


Figure I-2. The Standards-Based Deliverable Set

The length of each document should be between 25 and 45 pages. This will assure that the documents actually get read by individuals in the organizations.

Architecture Framework Document

SAMPLE TABLE OF CONTENTS

Baseline Characterization Document

SAMPLE TABLE OF CONTENTS

I. Executive summary

- Project status
- Key issues

II. Key architecture baseline characterization issues

III. Scope and approach

IV. Classification and description

- Platform classification
- Generic application model
- Generic technology model
- Work flow model
- Generic information model
- Standards support description
- Security evaluation
- Connectivity support model
- Cost/performance data

V. Summary assessment of design issues and constraints of current environment

VI. Implications for target architecture design

VII. Next steps

Target Architecture Document

SAMPLE TABLE OF CONTENTS

SAMPLE TABLE OF CONTENTS

Migration Options Document

SAMPLE TABLE OF CONTENTS

I. Executive summary

- Project status
- Key issues

II. General cost/benefit definition

III. Migration project scope definition

IV. Technology standard implementation strategy

V. Time lines and trigger points

VI. Project cost and time frame considerations

VII. Specific business case and cost/benefit analysis for identified opportunities

VIII. Project deliverables definition

IX. Organizational change process requirements

X. Next steps

Implementation Plan Document(s)

This is not a formal presentation document, rather it is the aggregate set of project plan documents produced by the individual functional unit.

Presented below is a suggested set of topic areas to include in each plan. These may vary widely depending upon the implementation project but should comply with all DoD project management standards.

I. Project description

II. Objectives

III. Scope

IV. Deliverables

V. Critical success factors

VI. Constraints

VII. Task list

VIII. Effectiveness measures

IX. Technology requirements

X. Staffing skills

XI. Completion criteria

XII. Other issues

SBA Assessment Document

SAMPLE TABLE OF CONTENTS

I. Executive summary
- Project status
- Key issues

II. Scope of architecture review

III. Key review findings

IV. Implementation adherence to IT principles and target architecture
- Processes
- Information
- Platforms
- Standards
- Migration issues
- Architecture organization and personnel issues

V. User views of benefits and functionality delivered

VI. Review of cost/benefit implementation delivered

VII. Continuous process improvement recommendations

VIII. Next steps

This page intentionally left blank.

Appendix J: Glossary

American National Standards Institute (ANSI): The principal standards coordination body in the United States. ANSI is a member of the International Organization for Standardization (ISO).

Application: The use of capabilities (services and facilities) provided by an information system specific to the satisfaction of a set of user requirements. [P1003.0/D15]

Application Entity: The part of an application process that interacts with another application process.

Application Layer: Layer seven of the OSI Reference Model. It serves as a window through which applications access communication services.

Application Model: A term used to describe those functions of an organization that can be supported or automated through IT. It is used for grouping or clustering functions into applications. It provides the application developers' views of the IT architecture.

Application Process: The part of an application that resides in a single end system.

Architecture: Architecture has various meanings depending upon its contextual usage.
(1) The structure of components, their interrelationships, and the principles and guidelines governing their design and evolution over time. [IEEE STD 610.12]
(2) Organizational structure of a system or component. [IEEE STD 610.12]
(3)The disciplined definition of the IT infrastructure required by a business to attain its objectives and achieve a business vision. It is the structure given to information, applications, and organizational and technological means—the groupings of components, their interrelationships, the principles and guidelines governing their design, and their evolution over time.

Bridge: The hardware and software used to connect circuits and equipment in two networks with the same protocol.

Common Applications Environment (CAE): The X/Open term for a computer environment in which applications can be ported across X/Open vendor systems. It includes standards for the operating system, languages, networking protocols, and data management.

Computer-Aided Acquisition and Logistics Support (CALS): Standards for electronic file format interchange and data management adopted by the U.S. Department of Defense to acquire, process, and disseminate technical information in digital form. CALS will facilitate the transfer of logistic and technical information between industry and Government by leveraging existing international standards. Among the industry standards used in CALS are IGES (CAD, vector graphics), SGML (automated publishing), GRP 4 Raster or TRIF (raster scanned images), and CGM (illustrations).

Computer-Aided Software Engineering (CASE): A set of software tools that automate and contribute to the improvement of the software development process.

Conformance: Meeting standards. By running standard test scripts, conformance testing ensures that a product meets standards.

Connection: In data communications terminology, a logical link established between application processes that enables them to exchange information. In the OSI Reference Model, an association established by one layer with two or more entities of the next higher layer for the transfer of data. In TCP/IP, it is a logical TCP communication path identified by a pair of sockets, one for each side of the path.

Data Link: An assembly of two or more terminal installations and an interconnecting line.

Data Link Layer: Layer two of the OSI Reference Model. It controls the transfer of information between nodes over the physical layer.

Directory Services: A service of the External Environment entity of the Technical Reference Model that provides locator services that are restricted to finding the location of a service, location of data, or translation of a common name into a network specific address. It is analogous to telephone books and supports distributed directory implementations. [TA]

Distributed System: A system consisting of a group of connected, cooperating computers.

Distribution List: A list containing the names of mail users and/or other distribution lists. It is used to send the same message to multiple mail users. It can be private or public.

Electronic Mail: The electronic generation, transmission, and display of correspondence and documents. Electronic mail is a GAE.

Entity: An active element within an open system layer (e.g., session entity, transport entity). It can represent one layer, one part of a layer, or several layers of the OSI Reference Model. One layer can include several entities.

Exterior Gateway Protocol (EGP): The service by which gateways exchange information about what systems they can reach.

Gateway: A device for converting one network's message protocol to the format used by another network's protocol. It can be implemented in hardware or software.

Generic Application Environment (GAE): A term used to describe the set of architecture components that describe the different possible types of IT applications.

Generic Technology Environment (GTE): A term used to describe the set of architecture components that describe the different types of services required to support a GAE.

Generic Technology Platform (GTP): A term used to describe the different types of delivery components that can be used to support IT applications.

Government Open Systems Interconnection Profile (GOSIP): A government (e.g., U.S. or U.K.) profile of functional applications that outlines a national policy and strategy for converting to a communications system based on OSI. Use of GOSIP is no longer mandatory.

Host: A computer, particularly a source or destination of messages, on a communications network.

Information Model: A term used to describe the information resources of the organization and their interrela-tionships. It is used to support data modeling and resulting database and document storage design requirements. It provides the information resource managers' views of the architecture.

Institute of Electrical and Electronics Engineers (IEEE): An accredited standards body that has produced standards such as the network-oriented 802 protocols and POSIX. Members represent an international cross section of users, vendors, and engineering professionals.

Integrated Services Digital Network (ISDN): The recommendation published by CCITT for private or public digital telephone networks where binary data, such as graphics and digitized voice, travel over the same lines. ISDN will unite voice and data transmission, including imaging, over the same kind of digital network that links most telephone transmissions in use today.

Interface: A connecting link between two systems. In the OSI Reference Model, it is the boundary between adjacent layers.

International Standard (IS): Agreed international standard as voted by ISO.

International Organization for Standardization (ISO): An organization that establishes international standards for computer network architecture. Its OSI Reference Model divides network functions into seven layers. (Membership is by country, with more than 90 countries currently participating.)

Interoperability: (1) The ability of two or more systems or components to exchange and use information. [IEEE STD 610.12]. (2) The ability of the systems, units, or forces to provide and receive services from other systems, units, or forces, and to use the services so interchanged to enable them to operate effectively together. The conditions achieved among communications-electronics systems or items of communications-electronics equipment when information or services can be exchanged directly and satisfactorily between them and/or their users. [Joint Pub 1-02, DoD/NATO] [JOPES ROC]

(2)The ability of applications and computers from different vendors and architectures to work together on a network.

Interoperability Testing: Procedures for ensuring that a computer product or system can communicate in a multivendor network.

Layer: A level of the OSI Reference Model. The model divides functions for transferring information between systems into seven layers, grouping the related functions or tasks and making them easier to understand. Each layer performs certain tasks to move the information from sender to receiver. Protocols within the layers define the tasks for networks but not how the software accomplishes the tasks. Interfaces pass information between the layers they connect.

Local Area Network (LAN): A data network, located on a user's premises, within a limited geographic region. Communication within a local area network is not subject to external regulation; however, communication across the network boundary may be subject to some form of regulation. [FIPS PUB 11-3]

Message: A block of information sent from a source to one or more destinations.

MS-DOS: The personal computer operating system developed by Microsoft Corporation.

Multivendor Network: A computer network with hardware and software from more than one vendor.

National Institute for Standards and Technology (NIST): The division of the U.S. Department of Commerce that ensures standardization within Government agencies. NIST is responsible for the Applications Portability Profile—a set of standards and guidelines for U.S. Government procurement. NIST was formerly known as the National Bureau of Standards (NBS).

Network: A system of connected computers.

Network Layer: The third layer of the OSI Reference Model. This layer controls underlying telecommunication functions such as routing, relaying, and data link connections.

Node: A point in a network, either at the end of a communication line (end node) or where two lines meet (intermediate node).

Open Network: A network that can communicate with any system component (peripherals, computers, or other networks) implemented to the international standard (without special protocol conversions, such as gateways).

Open Software Foundation (OSF): An organization created by major IT vendors to define specifications, develop software, and make available an open, portable environment.

Open Systems: (1) A system that implements sufficient open specifications for interfaces, services, and supporting formats to enable properly engineered applications software: (a) to be ported with minimal changes across a wide range of systems, (b) to interoperate with other applications on local and remote systems, and (c) to interact with users in a style that facilitates user portability. [P1003.0/D15] (2) Software environments consisting of products and technologies that are designed and implemented in accordance with "standards" (established and de facto) that are vendor independent and commonly available.

Open Systems Interconnection (OSI): A set of standards that, when implemented, let different computer systems communicate with each other.

Operating System: A group of programs operating under the control of a data processing monitor program. It manages such functions as memory, processing tasks, and interprocess communication in a computer system.

OSI Reference Model: The seven-layer model, defined by the ISO, that provides the framework for building an open network. The seven layers, ranging from highest to lowest, are application, presentation, session, transport, network, data link, and physical.

Password: A string of characters required to gain access to directories, files, or applications.

Peer Protocol: The protocol governing communications between program entities that have the same function in the same layer in each of two OSI networks.

Physical Layer: The first layer of the OSI Reference Model. It governs hardware connectors and byte-stream encoding for transmission. It is the only layer that involves a physical transfer of information between network nodes.

Portable Operating System Interface for Computer Environments (POSIX): An IEEE standard operating-system interface defining the external characteristics and facilities required to achieve the portability of applications at the source-code level.

Portability: (1) The ease with which a system or component can be transferred from one hardware or software environment to another. [IEEE STD 610.12] (2) A quality metric that can be used to measure the relative effort to transport the software for use in another environment or to convert software for use in another operating environment, hardware configuration, or software system environment. [IEEE TUTOR] (3) The ease with which a system, component, data, or user can be transferred from one hardware or software environment to another. [TA]

Porting: The process by which a software application is made operational on a computer architecture different from the one on which it was originally created.

Presentation Layer: The sixth layer of the OSI Reference Model. It allows an application to properly interpret the data being transferred.

Process: A general term for any computer operation on data.

Profile: A set of one or more base standards, and, where applicable, the identification of those classes, subsets, options, and parameters of those base standards, necessary for accomplishing a particular function. [P1003.0/D15]

Protocol: A set of rules governing network functionality. The OSI Reference Model uses sets of communication protocols to facilitate communication between computer networks and their components.

Quality of Service (QOS): A set of characteristics of a connection as observed between the connection end points. In the OSI session and transport layers, acceptable QOS values are negotiated between the service users when the connection is established.

Scalability: The ability to use the same application software on many different classes of hardware/software platforms from personal computers to super computers (extends the portability concept). [USAICII] The capability to grow to accommodate increased work loads.

Server Type: A class of servers in a client/server architecture.

Service Provider: The resource that provides the facilities of the relevant OSI Reference Model layer. The OSI session and transport layers are the service providers for the session and transport services, and the X.25 network gateway or X.25 message control system is the service provider for the network service.

Service User: The software application using the facilities of one of the layers of the OSI Reference Model. For example, a program that calls the programmatic interface to the session layer is a session service user.

Session Layer: The sixth layer of the OSI Reference Model. It provides the means for two session service users to organize and synchronize their dialogues and manage the exchange of data.

Store-and-Forward Message System: The communication process that allows messages to be stored at intermediate nodes before being forwarded to their destination. X.400 defines a message handling system that uses this process.

System:–People, machines, and methods organized to accomplish a set of specific functions. [FIPS PUB 11-3]

TCP/IP Gateway: A device, or pair of devices, that interconnects two or more networks or subnetworks, enabling the passage of data from one (sub)network to another. In this architecture, a gateway contains an IP module and, for each connected subnetwork, a subnetwork protocol (SNP) module. The routing protocol is used to coordinate with other gateways. A gateway is often called an IP router.

*Technology Mode*l: A term used to define and describe the components of the technology infrastructure that support the other architecture models. It is in this area that the enabling effect of standards-based architectures is felt the most. The technology model provides the technology managers' views of the architecture.

UniForum: A trade association dedicated to promoting UNIX and open systems. UniForum sponsors UNIX events, publishes magazines, directories and technical overviews, and proposes specifications.

UNIX: An operating system that has become a de facto industry standard, supported on a wide range of hardware systems from a variety of vendors.

UNIX International: The consortium that defines and promotes the UNIX operating system and related software products.

Wide-Area Network (WAN): A public or private computer network serving a wide geographic area.

Work Organization Model: A term used to describe the impact on business operations at the work group and user

levels. It is used by organizational change designers to manage the impact of introducing new IT systems. It provides the users' views of the architecture.

X.25: Recommendations developed by CCITT that define a protocol for communication between packet-switched public data networks and user devices in the packet-switched mode.

X.400: The international standard for a store-and-forward message handling system in a multivendor environment.

X/Open Company Ltd.: *A* nonprofit corporation made up of vendors and large corporate users who are investing in the specification of the X/Open Portability Guide (XPG), an open environment based on standards. X/Open also brands products.

This page intentionally left blank.

Appendix K: Proposing Changes to TAFIM Volumes

Introduction

Changes to the TAFIM will occur through changes to the TAFIM documents (i.e., the TAFIM numbered volumes, the CMP, and the PMP). This appendix provides guidance for submission of proposed TAFIM changes. These proposals should be described as specific wording for line-in/line-out changes to a specific part of a TAFIM document.

Use of a standard format for submitting a change proposal will expedite the processing of changes. The format for submitting change proposals is shown below. Guidance on the use of the format is subsequently provided.

A Configuration Management contractor is managing the receipt and processing of TAFIM change proposals. The preferred method of proposal receipt is via e-mail in ASCII format, sent via the Internet. If not e-mailed, the proposed change, also in the format shown below, and on both paper and floppy disk, should be mailed. As a final option, change proposals may be sent via fax; however, delivery methods that enable electronic capture of change proposals are preferred. Address information for the Configuration Management contractor is shown below.

Internet: **tafim@bah.com**

Mail: **TAFIM**
Booz•Allen & Hamilton Inc.
5201 Leesburg Pike, 4th Floor
Falls Church, VA 22041

Fax: **703/671-7937**; indicate “TAFIM” on cover sheet.

TAFIM Change Proposal Submission Format

a. Point of Contact Identification

(1) Name:

(2) Organization and Office Symbol:

(3) Street:

(4) City:

(5) State:

(6) Zip Code:

(7) Area Code and Telephone #:

(8) Area Code and Fax #:

(9) E-mail Address:

b. Document Identification

(1) Volume Number :

(2) Document Title:

(3) Version Number:

(4) Version Date:

c. Proposed Change # 1

(1) Section Number:

(2) Page Number:

(3) Title of Proposed Change:

(4) Wording of Proposed Change:

(5) Rationale for Proposed Change:

(6) Other Comments:

d. Proposed Change # 2

(1) Section Number:

(2) Page Number:

(3) Title of Proposed Change:

(4) Wording of Proposed Change:

(5) Rationale for Proposed Change:

(6) Other Comments:

n. Proposed Change # n

(1) Section Number:

(2) Page Number:

(3) Title of Proposed Change:

(4) Wording of Proposed Change:

(5) Rationale for Proposed Change:

(6) Other Comments:

Format Guidance

The format should be followed exactly as shown. For example, Page Number should not be entered on the same line as the Section Number. The format can accommodate, for a specific TAFIM document, multiple change proposals for which the same individual is the Point of Contact (POC). This POC would be the individual the TAFIM project staff could contact on any question regarding the proposed change. The information in the **Point of Contact Identification** part (**a**) of the format would identify that individual. The information in the **Document Identification** part of the format (**b**) is self-evident, except that volume number would not apply to the CMP or PMP. The proposed changes would be described in the **Proposed Change #** parts (**c, d,** or **n**) of the format.

In the **Proposed Change #** parts of the format, the Section number refers to the specific subsection of the document in which the change is to take place (e.g., Section 2.2.3.1). The page number (or numbers, if more than one page is involved) will further identify where in the document the proposed change is to be made. The Title of Proposed Change field is for the submitter to insert a brief title that gives a general indication of the nature of the proposed change. In the Wording of Proposed Change field the submitter will identify the specific words (or sentences) to be deleted and the exact words (or sentences) to be inserted. In this field providing identification of the referenced paragraph, as well as the affected sentence(s) in that paragraph, would be helpful. An example of input for this field would be: "Delete the last sentence of the second paragraph of the section and replace it with the following sentence: 'The working baseline will only be available to the TAFIM project staff.'" The goal is for the commentor to provide proposed wording that is appropriate for insertion into a TAFIM document without editing (i.e., a line-out/line-in change). The c (5), d (5), or n (5) entry in this part of the format is a discussion of the rationale for the change. The rationale may include reference material. Statements such as "industry practice" would carry less weight than specific examples. In addition, to the extent possible, citations from professional publications should be provided. A statement of the impact of the proposed change may also be included with the rationale. Finally, any other information related to improvement of the specific TAFIM document may be provided in c (6), d (6), or n (6) (i.e., the Other Comments field). However, without some degree of specificity these comments may not result in change to the document.