11 Computer Aided Planning, Engineering and Management

Opening Vignette: Computer Aided Planning to Launch a Mobile Health Clinic Initiative

Mobile Health Clinics (MHCs), combined with the mobile computing technologies, have been highly effective in combating HIV and malaria, improving maternal health, and reducing infant mortality in Peru, South Africa, Uganda, and the Philippines. In particular, location-based text messaging applications have been highly effective to attract young people to mobile clinics that provide informational, testing, and/or clinical services. While there are many success stories about mobile clinics, numerous failures have occurred due to logistical issues (e.g., running out of supplies in the middle of nowhere), technology issues (no wireless signal in the area), procedural problems (healthcare professionals could not get visas on time), and social issues (some parents did not like their children to be invited to a clinic without parental consent).

A *Mobile Clinic Support System* is needed to address the people, process and technology issues and thus assure repeatable success of these clinics. The following figure shows a conceptual view of a support system that leverages the latest ICT developments to serve the physicians, the patients, the healthcare facilities, the suppliers of materials and the regulating authorities. Such a support system could profoundly impact the delivery of healthcare to different parts of the World because it can be offered with minimal technologies or sophisticated web and wireless support. In addition, this support system could be devoted to a single service provider or support multiple suppliers, healthcare facilities and physicians as a B2B network. How can the aforementioned Learn-Plan-Do-Check cycle be used to assure success? To gain some insights, let us go through the SPACE Planner capabilities.

Overview of a Mobile Health Clinic Support System

- **Learn:** A user (government agency or NGO) starts by first visiting the Directory and the Knowledge Repositories for case studies and information on different aspects of mobile health clinics.
- **Plan:** Go beyond case studies and actually use the Strategic Planner to generate a country and situation specific plan. The Planner guides the users through the maze of decisions in cost-benefit analysis, business process modeling, technology selection, system integration, disaster recovery, and information security that is specific to the country in which the mobile clinic is supposed to operate.
- **Do**: The generated plan serves as a solid starting point for the implementers to refine and operate mobile health clinics for different situations in different regions of the world. A wide range of simulations and business games could be used to create and exercise some what-if scenarios.
- **Check**: The operation of the mobile health clinics can be monitored through project management techniques such as "management dashboards". The lessons learned could then be used to reiterate, refine and improve the deployment of future mobile health clinics.

CONTENTS

11.1 Introduction

A computer aided planning, engineering and management environment consists of a set of tools that support the entire life cycle activities of digital enterprises as shown in Figure 11-1. Specifically, the tools in such an environment should help the key players to *Learn* what needs to be done, *Plan* how to do it right, *Do* whatever needs to be done, and *Check* to see if it is done right. Although, a great deal of knowledge about best practices, standards, and methodologies is accumulating, the actual *use* of this knowledge to make crucial decisions is sparse. The main objective of a computer aided planning, engineering and management environment is to systematically walk the users through the key decisions and automatically include the needed best practices, standards and knowledge repositories when needed.

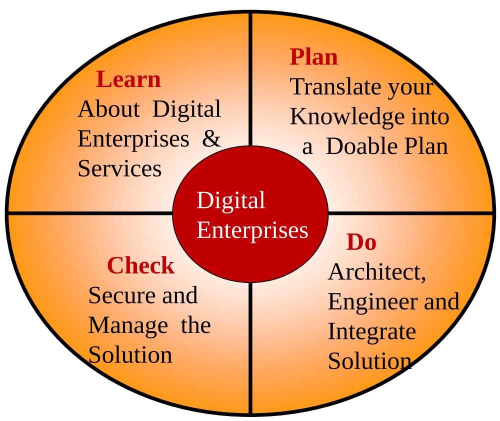


Figure 11-1: Computer Aided Planning, Engineering and Management – Conceptual Overview

This chapter introduces the main idea of a computer aided planning, engineering and management environment and attempts to answer the following questions:

- Why is such an environment needed and what are the simple examples of such an environment (see Section 11.2)
- What type of tools and techniques are used in such environments (see Section 11.3)
- How can a simple planning methodology be extended for integrated architectures (see Section 11.4)
- What is SPACE and how does it support the different planning scenarios (see Section 11.5)

11.2 Computer Aided Strategic Planning for Digital Enterprises – An Overview

11.2.1 Why is Computer Aided Planning Needed

Simply stated, planning determines what needs to be done (the objective), outlines the steps and the sequence of steps to accomplish the objective, and lists the time and effort needed to meet the objective. The result of a planning process is a document, a ***plan***, which is a repository of information about the approach, the steps, the resources needed and the time frame for an effort. Within this general framework, several levels and types of planning exist in real life situations. In enterprises, there can be several types of planning efforts (business, financial, IT infrastructure) at several levels (strategic, tactical, operational). These planning efforts cover different horizons (strategic plans are longer range than tactical or operational ones) and have different areas of focus (business plans concentrate on business issues while IS and IT infrastructure plans focus on information systems and technologies). These plans also support and feed into each other (IS plans support business plans and strategic plans feed tactical and operational plans).

IS planning is a challenging task, as shown in Figure 11-2, because it requires interactions with several others activities in an organization. At the highest level, as shown, is the business strategic planning which determines the enterprise services to be provided. At the next level, the strategic systems (financial, engineering, manufacturing and information systems) are planned. The infrastructure (facilities, equipment, human and IT infrastructure systems) needed for the services are planned at the third level. The actual development and support is conducted at the fourth level. The major planning processes and the interrelationships between the planning processes are displayed in Figure 11-2

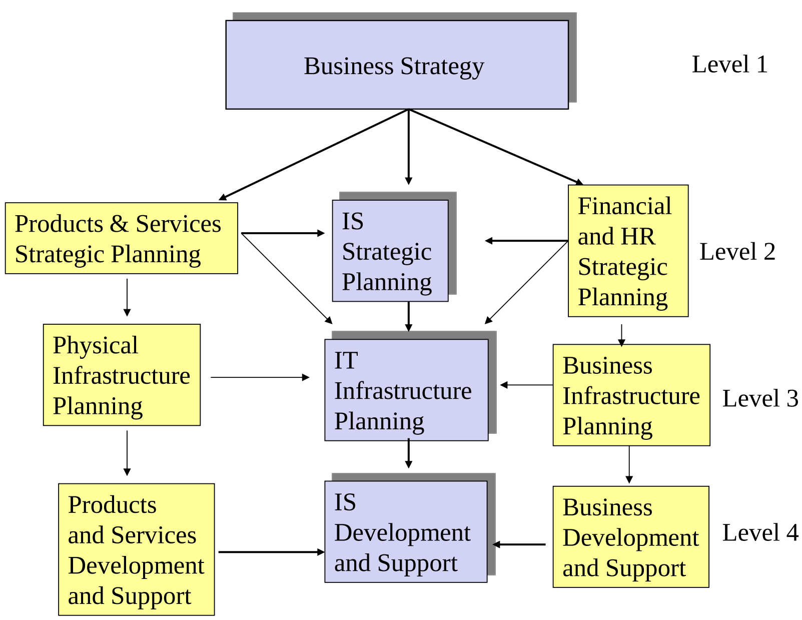


Figure 11-2: IS (Information Systems) Planning Levels -- A Simplified View

Current and next generation of public as well as private enterprises need to continuously plan a very wide range of *eservices* (ICT-based services that are delivered to the consumers). Examples of common eservices span ehealth, elearning, eprocurement, etransportation, emarketing, and the like.

These eservices provide tremendous benefits to the consumers by allowing them to easily access large amounts of information sources around the globe, purchase needed items online, join discussion groups, receive healthcare in remote areas, and learn about the latest developments in different areas of work.

However, these eservices rely on a complex ICT (information and communication technology) infrastructure that includes technologies such as Web 2.0, mobile computing platforms, wireless networks, application servers for ecommerce and B2B trade, broadband networks, cloud computing, and systems management platforms. To survive and thrive, modern enterprises need to plan their eservices and the needed ICT infrastructure quickly and correctly.

To develop solid plans for these and other eservices, enterprises need to be guided through the maze of intricate choices that involve multiple policies, procedures, technologies and suppliers. Typical challenges faced by the people involved in the planning process are:

- How to understand the business strategies and to align ICT with the business strategies
- What business processes (BPs) should be automated and re-engineered to compete and succeed
- What type of ICT infrastructure (application packages, computing platforms, and network services) are needed to support the BPs
- How to integrate new applications with the existing (including legacy) systems by using concepts such as SOA (service oriented architectures)

11.2.2 A Simple Strategic Planning Methodology

Figure 11-3 shows a simple planning framework, first introduced in Chapter 2, that has been used throughout this book. This framework shows the following horizontal layers that represent highly interdependent building blocks of a system that must be properly planned:

- **Enterprise business processes (BPs)** are a collection of *activities* that are required to achieve a business goal. At a basic level, a BP can be represented as a flowchart that specifies the orchestration of activities needed to complete the goal. For example, for a payroll *service*, several BPs have to be carried out (e.g., pay has to be computed, deductions have to be considered, overtime may need to be calculated, etc).
- **Enterprise business applications** are the computer-based information systems that provide automated support to the business services/processes. These applications are also referred to as enterprise applications, business applications or just as applications in the literature. Whatever the name, these applications are business aware. For example, a patient care application contains business logic and data that is concerned with hospitals, doctors and health insurance.
- **The information technology (IT) infrastructure** is used to build, deploy and operate the business applications. IT infrastructure, also sometimes known as computer-communication platform, consists of computing platforms (e.g., computers, operating systems, utilities) and the networks that interconnect the computing platforms. *This infrastructure enables the applications and is business unaware.* For example, the same type of networks and computers are used in airline reservation systems as well as hotel reservation systems. The best known infrastructure is the network that interconnects remote applications, databases, and users. Internet, wireless, and broadband networks are examples of vital network technologies.

These horizontal layers represent the enterprise business processes/strategies, enterprise applications, platform services and network services as the key building blocks of a plan. These horizontal layers need to be properly secured, integrated, and managed/governed (represented as vertical bars that cut across different horizontal layers of the framework). We will use this framework to establish the interrelationships between different technical and business aspects of modern enterprise and to define some basic terms. We start from the top layer (business strategy and processes) and proceed to lower layers. The discussion then moves to the vertical bars of security, management and architectures that cut across all layers.

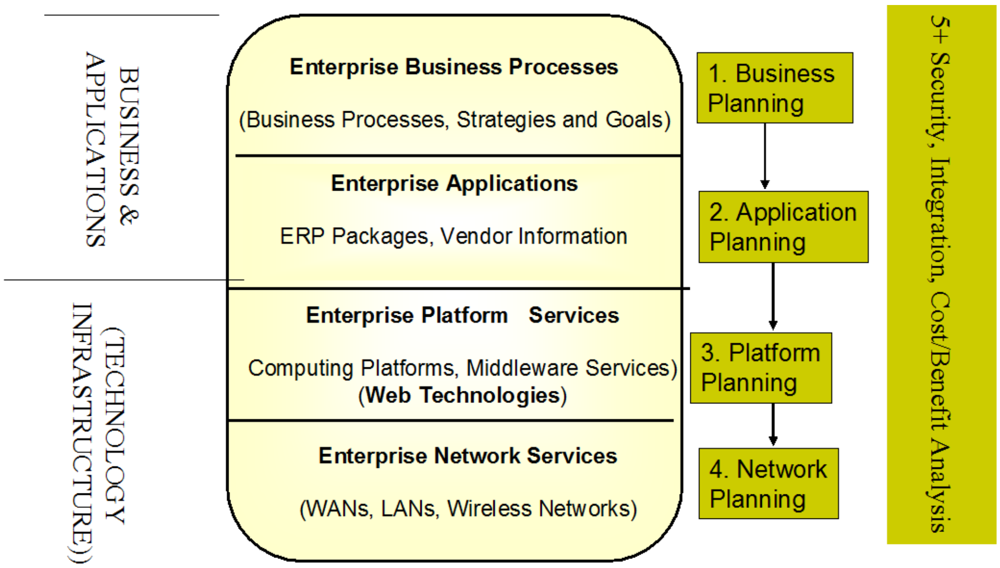


Figure 11-3: A Simple Planning Methodology

This framework helps us to develop a very simple planning methodology (darker blocks on the right) that consists of the following major stages:

Stage 1: Business planning that concentrates on business and strategic issues
Stage 2: Application planning that establishes the business applications (automated business processes) needed to support the business strategies
Stage 3: Computing platform planning that determines the needed computing hardware/software
Stage 4: Network planning that establishes how all the pieces will communicate remotely with each other
Stage 5+: Security, integration and administration planning that concentrate on how all the layers will be properly secured and administered

This simple methodology helps us to develop simple plans for small to medium businesses. We will extend and expand this methodology to handle more complex situations.

11.2.3 Example: A Computer Aided Planning, Engineering and Management Environment

A computer aided planning, engineering and management environment called SPACE (Strategic Planning, Architectures, Controls and Education) has been developed to support the methodology described here. SPACE, as we will see, can be used to support the simple methodology as well as more complex situations. SPACE consists of several components shown in Figure 11-12. Our initial focus is on the following innermost core capabilities:

- **Patterns Repository** that captures the best practices and knowledge needed by all SPACE tools. The pattern repository consists of industry patterns (e.g., health, education, public safety, public welfare, transportation), technology patterns (e.g., computing platforms, wired and wireless networks), operational patterns (e.g., security and integration patterns), and even country patterns (e.g., government patterns in different countries).

- **<u>Games and Simulations</u>** that support decisions in strategic analysis, mobile services planning, security planning, interagency integrations and health exchanges, application migration versus integration tradeoffs, risks and failure management, and quality assurance.
- **<u>Computer Aided Decision Support and Planning Tools</u>** that can be used to quickly build real life business scenarios and then guide the user through IT planning, integration, security and administration tasks by using best practices and patterns. An example is the *PISA* (Planning, Integration, Security, Administration) tool that helps small businesses plan, integrate and secure their IT systems. The main objective of these tools is to make sure that the best practices and the knowledge repositories are actually used in making effective decisions.

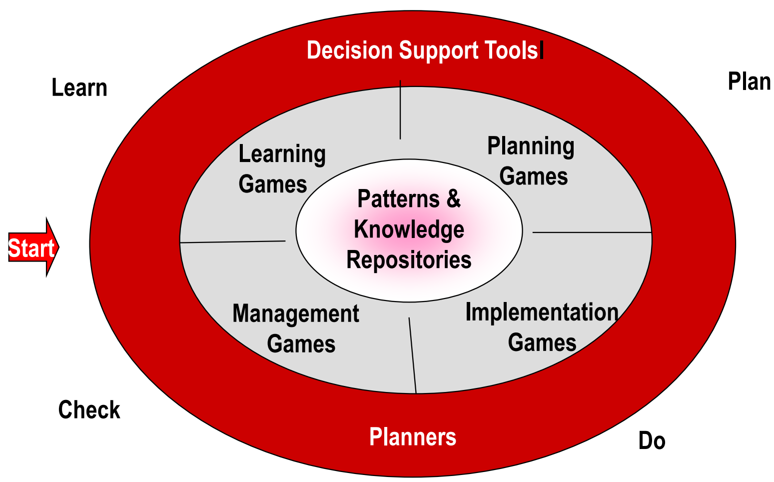


Figure 11-4: Conceptual View of SPACE (Strategic Planning, Architectures, Controls and Education)

Please note that the pattern repository captures the best practices and is at the core of the entire SPACE environment – the patterns are used in all games and detailed plus strategic planning advisors. We will briefly review the concept of patterns in the next section. The main objective of the SPACE environment is to go beyond passive repositories and to make sure that the best practices and the knowledge repositories are actually used in making effective decisions. In addition, SPACE:

- Represents a one stop shop that covers the entire Learn-Plan-Do-Check (LPDC) cycle
- Simulates a team of experts that collaborate with each other to solve problems
- Consists of games and extensive planning and decision support tools that are integrated around a common patterns repository
- Includes capabilities to support Smart systems and Big Data
- Has been and is being used extensively in developing countries

Additional information about the SPACE Environment can be found at the SPACE Site (www.space4ict.com) and research papers such as [Umar 2012, Umar 2013, Umar2014]. Computer aided planning environments such as SPACE are not readily available at the time of this writing. .

11.2.4 How to Use PISA – a Simple Planning Tool

PISA (Planning, Integration, Security, Administration), one of SPACE components, is a simple decision support environment that helps businesses plan, integrate and secure their IT (information technology) systems. At present, PISA is intended for small to medium businesses (SMBs) but can also be used for offices and divisions of larger businesses.

The PISA environment provides a family of automated consultants ("advisors") that support all stages of IT planning, integration and security projects (e.g., enterprise modeling, application planning, network planning, security planning, project planning, architecture analysis, solution evaluation). At the heart of PISA is the SPACE knowledgebase (KB) that contains an extensive patterns repository. The KB is used by the PISA advisors which are segmented into three modules:

- **PlanIT (Planner for IT)** concentrates on IT planning projects and develops a plan at the enterprise level.
- **SAM (Security and Administration Module)** provides guidance for security, project management and governance issues
- **Architecture and Integration Module (AIM)** focuses on how SOA (Service Oriented Architecture) can be used to architect and integrate the various components to form a functioning system.

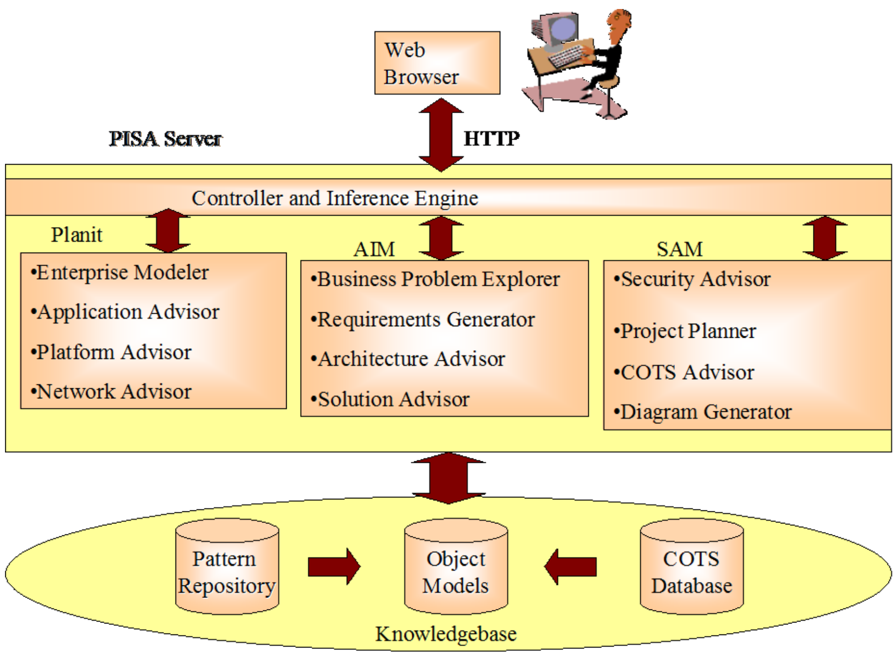


Figure 11-5: Technical Architecture of PISA

PISA uses Software as a Service (SaaS) model where the user logs on to a remote system instead of downloading and installing it. The system resides on a web server and is accessible from commonly available Web browsers. Figure 11-5 shows the technical architecture of PISA with its two major modules: PlanIT and AIM. All advisors are invoked from a controller that signs a user in, assigns a unique ID to the user, and manages user sessions. This allows each user to develop and store her own object model that is enriched as the user invokes different advisors. After being invoked, each advisor conducts its own interview, consults the object model to review the decisions made so far, makes inferences as much as possible, makes further decisions based on the user interview, extracts needed patterns from the patterns repository, and further enriches the object model. The object model is used as a basis for collaboration between the advisors. A great deal of information and documentation about PISA can be found at the website www.ngepisa.com.

11.3 Tools and Techniques Needed for Computer Aided Planning

The SPACE environment uses the following key technologies:

- Patterns are at the core of SPACE, as described in Section 11.3.1 and Section 11.3.2 (Note: these two sections are expansions of similar sections in Chapter 2). Additional details about patterns can be found in Appendix A (Section 11.7)
- Enterprise ontologies are used extensively in SPACE to develop semantically rich descriptions of enterprises in Appendix B (Section 11.8)
- Gamification, Decision Support Systems, Expert Systems and Big Data, as described in the previous chapter (Chapter 10).

11.3.1 Overview of Patterns for Computer Aided Planning

Patterns are a well-known format for capturing engineering knowledge. The idea was introduced by Christopher Alexander, a civil engineer, who wrote a series of books [2, 3] observing that well accepted buildings have common structures. Based on this, he devised a set of rules for architects to construct such buildings. According to Alexander, “Each pattern describes a problem that occurs over and over again in our environment and then describes the core of the solution to that problem in such a way that you can use this solution a million times over without ever doing it the same way twice”[3]. The "Gang of Four" extended the pattern format to software design [13]. Since then, patterns have been used extensively in software design and have been extended to e-business patterns [1, 18], requirements patterns [11, 14], architecture patterns [6], integration patterns [18], security patterns [19], and others. See the website (www.hillside.net/patterns) for extensive discussion, tutorials, and articles on patterns.

At a very basic level, a pattern *T* is a template *T(p, c, s)* where *p* is the problem to be solved, *c* is the context (under what conditions the pattern holds, i.e., why the problem needs to be solved), and *s* is the solution (what works in practice). Additional information such as examples and limitations can also be added to a pattern to help the designer. In addition, each pattern is assigned a name. Exhibit 1 shows a simplified example of a well known design pattern (Adapter) that occurs commonly in software engineering. Some patterns can be quite detailed and complex. We will discuss several patterns in this book. Due to space limitations, we cannot show every pattern completely by using the format shown in Exhibit 1. Instead, an abbreviated version consisting of few sentences will be used to highlight the essence, i.e., the solution of each pattern. For example, we will use the following sentence to describe the pattern shown in Exhibit 1:"The Adapter pattern shows how an intermediate object can be used to integrate two systems together".

Exhibit 1: Example of a Pattern

The following is a simplified view of a commonly used pattern (adapter). Patterns like this can be stored in a pattern repository and interlinked with each other.

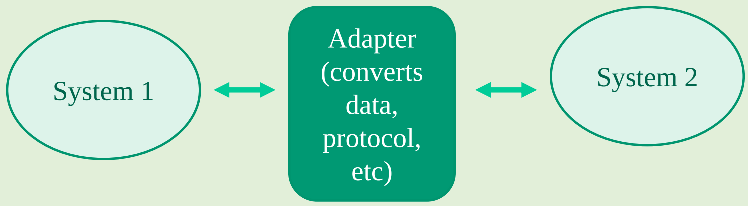
Name: Adapter

Problem: How to interconnect and integrate two different systems

Solution: Develop or buy a new component (Adapter) that does the translation between the two systems. The main idea is to imbed all the conversions in this new component and make it general so that it can be reused over and over again. The adapter may be very simple (e.g., a currency converter) or very complex (e.g., interconnects a very old system to a modern one).

Examples:
- Adapter for the US Electrical Power System and the British Electrical Power System
- Adapter for an old legacy applications that connects it to a new Web-based application.

Diagram:



Common Examples of Patterns

- Chair is a common pattern in furniture. Almost all chairs have 4 legs and a back. Of course, some chairs have arms and some don't.
- Jacket, pants and a shirt is a good pattern for men's clothes in most countries at present
- Hamburger, French fries and a soda is a good pattern for fast food
- A small house with two rooms, a bathroom, and car garage is good pattern for a starter family.

11.3.2 An Example of Using Patterns in IS Planning

Modern enterprises have to frequently develop plans for new IT services and integrate existing systems with new systems due to frequent mergers, acquisitions, business process reengineering initiatives, outsourcing of services, and fluctuations in market conditions. IS planning and integration projects typically involve heterogeneous systems from a diverse array of suppliers that have to be used by different types of customers, internal users and business partners.

11.3.2.1 Enterprise Process Pattern

There are different ways of representing business processes in an enterprise. A powerful way of representing this information is through a ***Business Process Pattern (BPP),*** shown in Figure 11-6 that captures an overall view of enterprise functional areas (e.g., sales, corporate management, back-office operations), the major business processes in each functional area (e.g., purchasing and payment within procurement) and the key interactions between these processes. A BPP can be used to identify what BPs are automated by different types of enterprise applications. For example, a purchasing application system, sometime also referred to as a purchasing application package, automates the purchasing business process.

A BPP can be used to represent an enterprise business architecture (EBA) because it can be used to represent various business processes and their interrelationships and interactions. In addition, the critical business processes can be identified to represent a business strategy. BPPs can be used to conduct quick sensitivity analysis such as the following: a) if one BP is eliminated, then what other BPs will be impacted, b) if an application package that supports a BP is replaced with another application, what other applications/BPs will be impacted, c) which application, if replaced, will have the most impact in terms of integration, d) which application, if replaced, will have the least impact in terms of integration. We will use business process patterns and other types of patterns throughout this book.

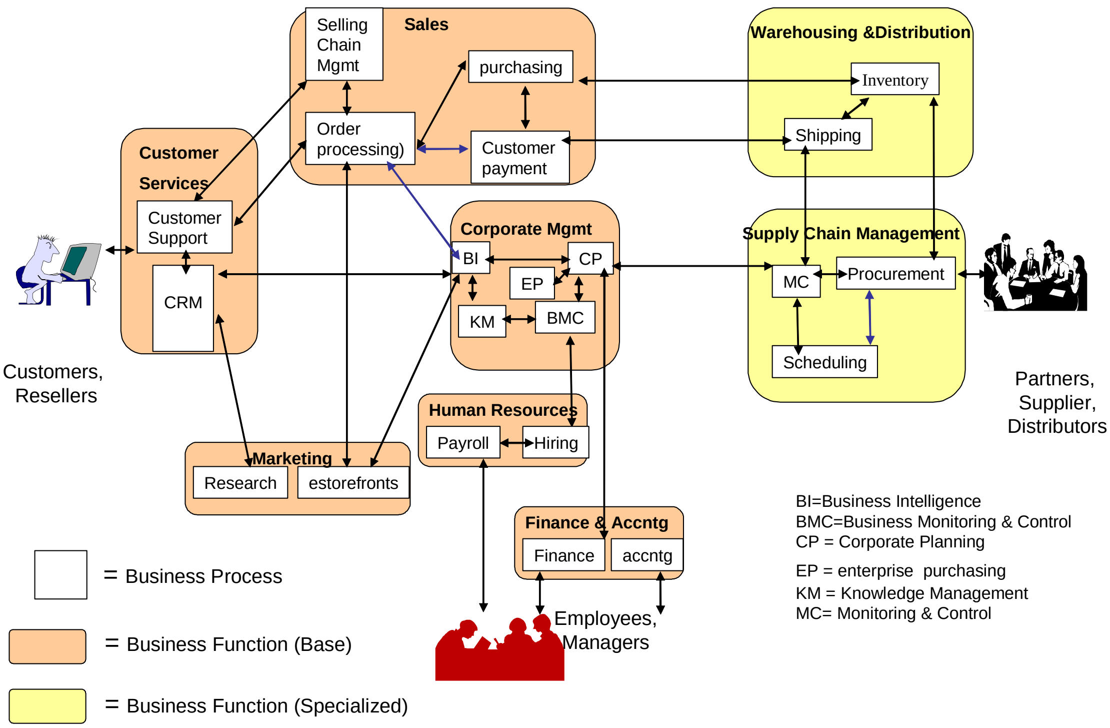


Figure 11-6: Sample Business Process Pattern (BPP) for a Retail Store

11.3.2.2 Enterprise Application Pattern

Knowledge of enterprise applications is a cornerstone of modern enterprises. Enterprise applications automate the business processes and involve many application strategies such as commercial-off-the-shelf (COTS) packages, rentals through application service providers (ASPs), outsourcing of software development, and increasing appeal of software re-use through service-oriented offerings based on Web-Services. These strategies translate a BPP to an enterprise application pattern (EAP), illustrated in Figure 11-7, that shows what BPs are automated and how (i.e., which automation strategy is used: buy, rent, outsource development, build in-house, or re-use/re-engineering of existing applications). The choice of applications and the automation strategies depends on several factors such as company size, web reliance and mobility reliance. For example, small companies in some industry segments (e.g., restaurants and local stores) typically automate only a few essential services such as back-office processes and may rent many of these services from ASPs; companies relying heavily on Web to build and deliver services will have high degree of automation and will invest in web-based purchasing and enterprise portals; companies exploiting mobile services will support mobile applications such as M-CRM. Figure 11-7 shows the resulting EAP of a small to medium store with low reliance on Web and mobile services. It is a refinement of the BPP shown earlier. The EAP suggests a sketch (a pattern) that the user can customize to build an automation vision for the business processes.

We have developed two types of application patterns to help us in choosing between the buy, rent, outsource, build, and extend/re-use (BRODE) strategies (see Figure 11-7). The APP-STRATEGY pattern helps in choosing one of the strategies and consists of five sub-patterns, one for each of the strategies. The APP-SOLUTIONS pattern suggests a detailed solution approach based on the selected APP-STRATEGY. For example, to support the buy strategy for CRM, this pattern suggests the COTS CRM packages that are used for different company sizes.

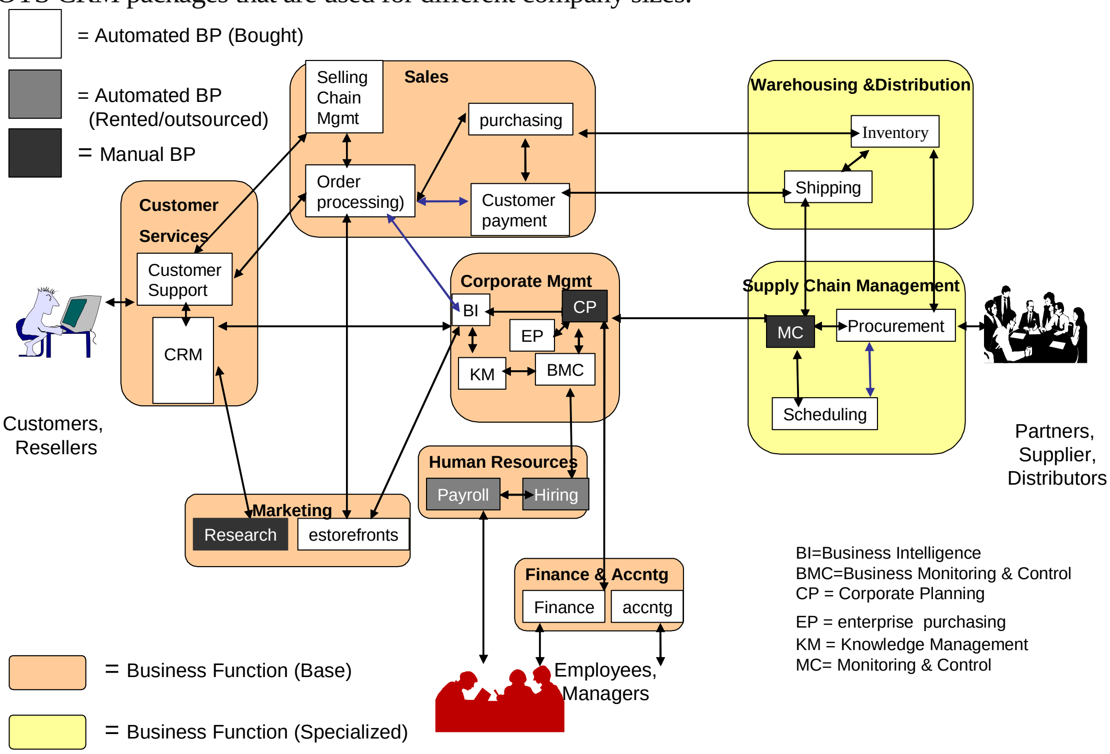


Figure 11-7: Enterprise Application Pattern (EAP) of a Retail Store

11.3.2.3 IT Infrastructure Pattern

IT infrastructure (platform) planning is concerned with determining the most appropriate technologies needed to *enable* the enterprise application plan developed previously. Examples of such enabling technologies are the Web technologies (including Web 2.0 and Web Services) used in corporate intranets, computing platforms on which the applications will reside, wireless and wired networks which connect all the computing platforms in an Intranet, and “Extranets” which connect many businesses for B2B trade. IT infrastructure planning can be subdivided into two broad stages; *computing platform planning (stage 3)* that supports the applications and *network planning (stage 4)* that interconnects these platforms with each other and the end-users. Computing platform planning consists of the following steps:

- Determine the middleware and Web services needed to interconnect the widely dispersed applications, users and databases,
- Identify the computer platforms, including servers, that will support the automation strategy and the application plan determined in the application planning stage.

- Decide which applications and databases will reside at which computing platforms (e.g., servers) at each site.
- Handle the software/hardware interdependencies (e.g., can a Windows application run on Linux platform, can an IIS server be installed on an XP machine, etc.)

Figure 11-8 shows a high level computing platform plan for a small town. The applications have been allocated to different computing platforms -- each computing platform consists of computer hardware (e.g., processor, disk drives), an operating system (e.g., Linux), some system software (e.g., MS Access), and middleware (e.g., Internet Explorer or Microsoft .NET Framework). The figure shows that the City Portal will reside on a city owned Unix server and the various services will reside on other internal machines (some of these applications may reside on the Portal Server (this decision can be made later). In addition, the city portal is connected to some other portals that reside on the public network (external portal) and on the partner network (partner portal). These computing platforms are interconnected through a network that is defined later.

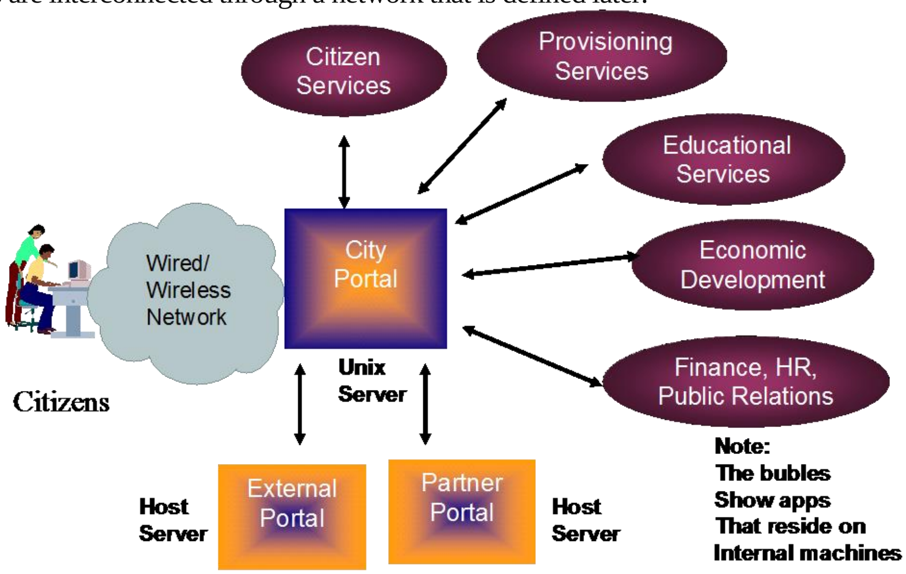


Figure 11-8: A Sample Computing Platform Plan

Network planning develops a network configuration that interconnects the computing platforms by using wireless as well as wired network elements. Figure 11-9 shows a sample network plan for this city. Network planning involves three major tasks. First, determine the workload at each site based on the work activities at each site. Second, develop a network configuration and estimate the bandwidth needed by using queuing network models. This involves, for example, network capacity planning for the internal plus external networks depending on the type of connection (wired/wireless) and the network traffic patterns. Finally, the type of connections and the commercially available network solutions need to be developed.

The network "pattern" shown in Figure 11-9, is a very good starting point for detailed network planning. This pattern represents a typical enterprise network with an Intranet for internal use, an Extranet for business partners, multiple wired/wireless LANs connected to the Intranet backbone and a Public Internet connection for the customers. This pattern can be customized, expanded, and specialized if needed.

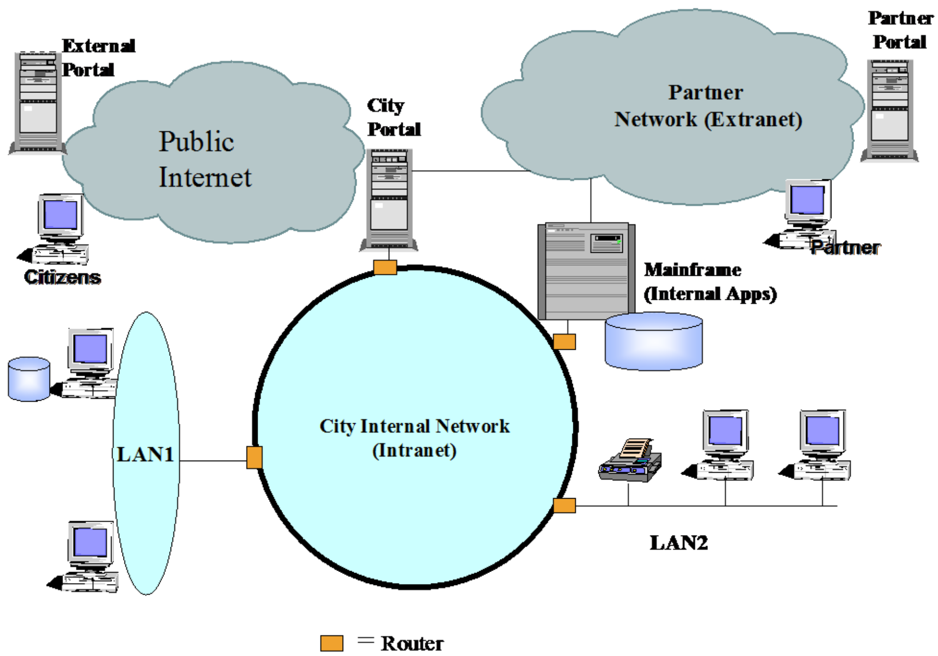


Figure 11-9: Sample IT Infrastructure and Network Pattern for a Small Company

Time to Take a Break

Suggested Review Questions Before Proceeding

- What is computer aided planning and what is its main objective
- What are the main steps of a simple strategic planning methodology
- What are the main components of a Computer Aided Planning, Engineering and Management Environment
- List the key Tools and Techniques Used in Computer Aided Planning
- How can Patterns be used for Computer Aided Planning
- What are Examples of Using Patterns in IS Planning

11.4 Extended Strategic Planning for Integrated Architectures

11.4.1 A Methodology for Integrated Architectures

The discussion so far has concentrated on simple planning scenarios that are suitable for small to medium businesses (SMBs). Figure 11-10 shows an extended methodology that is suitable for large enterprises -- it displays how individual plans can be developed and then integrated together into an enterprise wide integrated architecture (see the sidebar "Enterprise Architecture and Integrated Enterprises – A Quick Recap" for some basic definitions). This methodology consists of several stages that address several areas of focus (e.g., enterprise focus, IT infrastructure focus, integration focus and management focus). Different areas of focus are needed for different types of business scenarios. Table 11-1 shows four broad scenario types in terms of new or existing business services which need to be introduced for existing or new sites/organizational units. This table also indicates the primary areas of focus for each scenario type. The discussion is intentionally at a high level to highlight the main features of the overall methodology.

This methodology has several important features. First, it includes all stages needed to build a comprehensive enterprise architecture. Second, the methodology is asynchronous – the stages can be invoked whenever enough input is available, thus more than one stage can be executed in parallel. Third, this methodology is intelligent because many inferences are used between the stages and utilizes a knowledgebase that provides patterns and COTS (commercial off-the-shelf) information. Finally, the needed plans are developed gradually in different stages and captured in the knowledgebase (KB), thus later stages can learn from previous decisions.

Although this methodology can be, and has been, used manually, its features have driven development of the SPACE environment (introduced previously) that "gets smarter" as the user proceeds through the various stages. For example, in a manual methodology, the KB is just a collection of documents that the user can refer to. However, in an automated system, the KB is a database that contains an extensive repository of patterns, COTS information and previously generated plans. This KB can be queried, updated, and utilized by a family of "automated consultants" to infer and suggest solutions quickly (see the discussion of SPACE later).

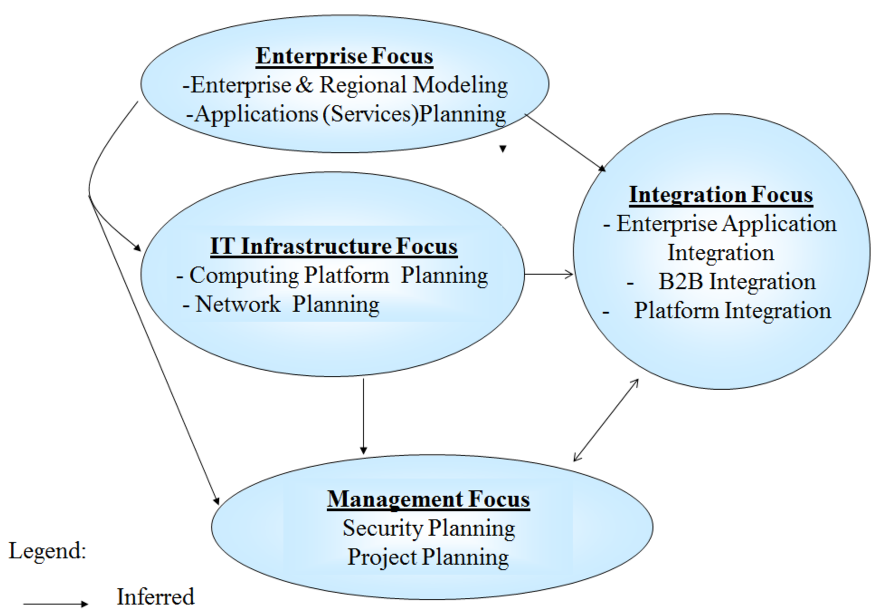


Figure 11-10: Conceptual View of Enterprise Planning, Architecture and Integration Methodology

EBP = Enterprise Business Pattern
WBS = Work Breakdown Structure

Table 11-1: Sample Architecture and Integration Scenarios

	New Site or Organizational Unit	Existing Site or Organizational Unit
New Business Services	**S1**: New services(s) for a new site **Example:** New division in Chicago focusing on a new set of business services (e.g., healthcare service) **Primary Focus:** Enterprise systems plus IT infrastructure needed at the new site.	**S2:** New services(s) for an existing site **Example:** New business services (e.g., equipment repairs) for an existing manufacturing company **Primary Focus:** Enterprise systems. It may be assumed that the needed IT infrastructure already exists.
Existing Business Services (Expanded/ Modified)	**S3**: Existing business services(s) for a new site **Example: E**xisting services (e.g., customer support) to be offered through an offshore site. **Primary Focus:** IT infrastructure. Needed enterprise systems may already exist.	**S4:** Existing business services(s) for an existing site **Example:** Existing business services (e.g., purchasing) of an existing company to be re-engineered for flexibility. **Primary Focus:** Enterprise systems plus integrated architecture focusing around purchasing.

11.4.2 Different Stages of Focus for Different Scenarios

Modern enterprises need to plan and integrate their IT services quickly and correctly to compete and survive. The methodology described in the previous section, shown in detail in Figure 11-11, can be used to systematically guide the key players through the maze of intricate choices discussed so far. However, this methodology has several limitations:

- If used manually, it is simply too complex, slow and error prone in prctice.
- It is difficult to explain the intricacies and the asynchronous processes in a paper and pencil approach. For example, it is not easy to show how different scenarios can lead to different sequence of activities
- Although the methodology is based on patterns and inferences, it is virtually impossible to fully exploit these capabilities in a manual methodology. For example, users cannot flip through hundreds of patterns and rules to locate and use the most appropriate ones.

In fact, variants of this methodology have been used in an active consulting practice and the results have highlighted the need for comprehensive automated support. A survey of available tools indicated that most tools are either drawing tools (e.g., Visio) or are too vendor specific (e.g., Cisco network designer). In addition, too much attention is paid to one aspect (e.g., network planning or business process modeling) instead of how *all* the pieces of an EA (enterprise architecture) fit together. A research project, in collaboration with a startup, was initiated with the following objectives:

- Provide automated support at EA level for all stages of the methodology discussed previously instead of one narrow area. (e.g., business process modeling or security planning).
- Simulate a consulting environment where several experts collaborate with each other to solve complex problems and provide different types of support (expert advice, help in making decisions, documentation).
- Extensively use patterns [Alexander 1977] to capture best practices in pertinet areas of work. In particular, use business patterns [Adams 2001], architecture patterns [Buschmann 2002], requirements patterns [Ferdinandi 2002], design patterns [Gamma 2006], integration patterns [Hohpe 2003] and security patterns [Kienzle 2001] to capture the available knowledge.

- Explore decision support that relies heavily on the Big Data being accumulated by extensive use of mobile phones, satellites and social media. In particular, we are interested in using the Big Dta freely available through the World Bank Open Data Institute, United Nations Department of Statistics, and the World Economic Forum (WEF).
- Take advantage of the latest thinking in design science [Hevner 2004], knowledge services [Mentza 2001], service innovation [Chesbrough 2006], and behavioral research in accuracy versus effort of information systems [Burton 1993, Todd 1992] to instantiate the system.
- Educate the users in addition to helping them make the decisions so that they can learn the basic concepts to make better decisions on their own.

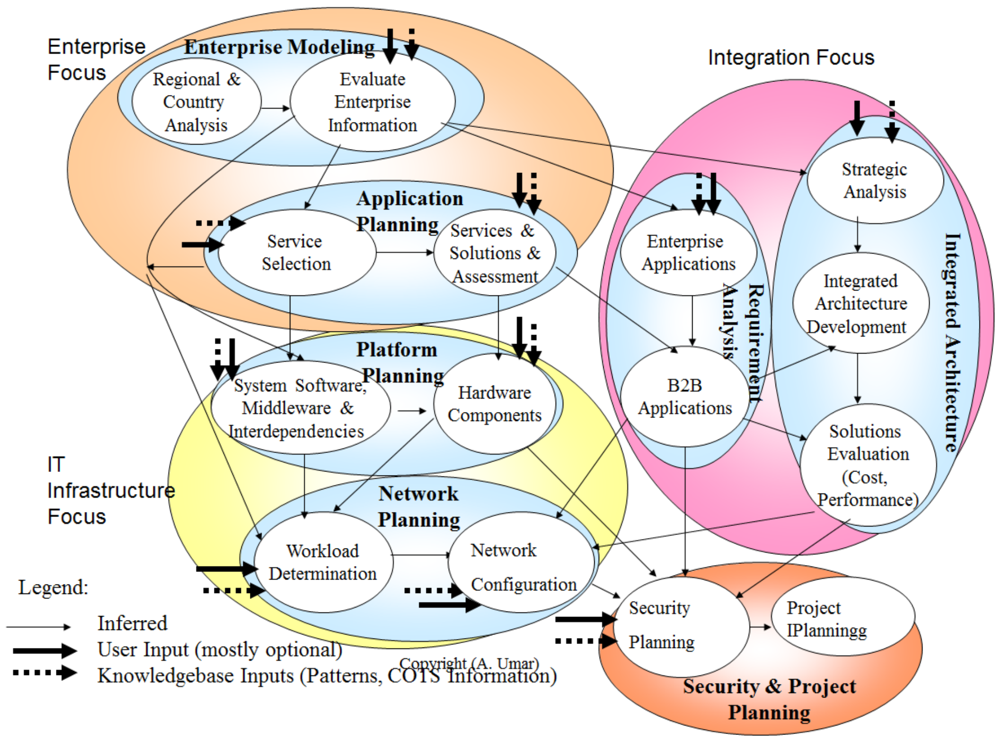


Figure 11-11: Enterprise Planning, Architecture and Integration Methodology – A Detailed View

Enterprise Architecture and Integrated Enterprises – A Quick Recap

An enterprise architecture (EA) shows components of an enterprise, what they do, and how they interface/interact with each other. EA is a consolidation of business and technology that can be of great value to the corporate management as well as CIOs. Figure A shows a conceptual view of an EA and its benefits in terms of four broad categories (planning, integration, security, and administration). This conceptual view is the foundation of the SPACE toolset described later.

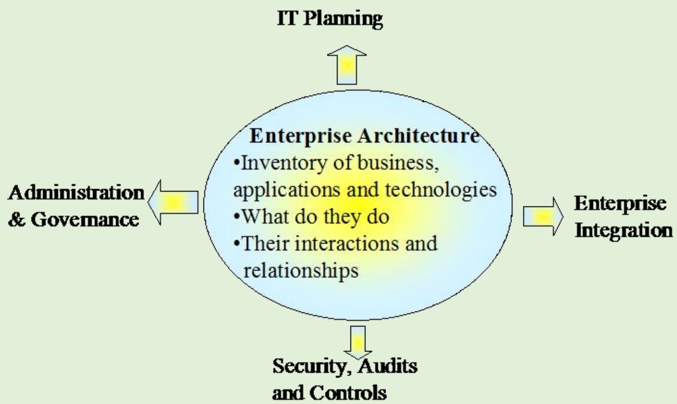


Figure A: Conceptual View of an Enterprise Architecture and its Benefits

An enterprise architecture (EA) describes the enterprise building blocks (the business layer, the application layer, and the technology infrastructure layer), what they do and how they interface/interact with each other. Let us use Figure B for a quick look at integrated architectures.. The objective of an *integrated* enterprise architecture is to show how *well* all the business plus technical components work together to serve the enterprise needs. Development of an integrated enterprise architecture, referred to as integrated architecture in this book for simplicity, generally starts with documenting the organization's strategy and goals.

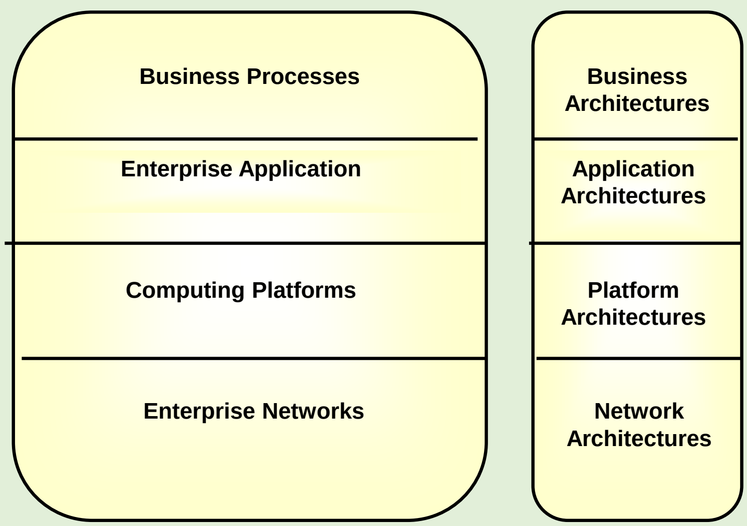


Figure B: Building Blocks of Enterprise Architecture and Integration

Based on this discussion, we can introduce the following definition of *integrated* enterprise architectures:

Definition: Integrated Enterprise Architecture = Horizontally Integrated Enterprise Architecture + Vertically Integrated Enterprise Architecture

11.4.3 SPACE – A Computer Aided Toolset for Enterprise Architecture and Integration

As a result of this research, we have developed a computer aided planning, engineering and management environment called SPACE (Strategic Planning, Architectures, Controls and Education). SPACE, currently operational as mature prototype, consists of the following components (shown in Figure 11-12):

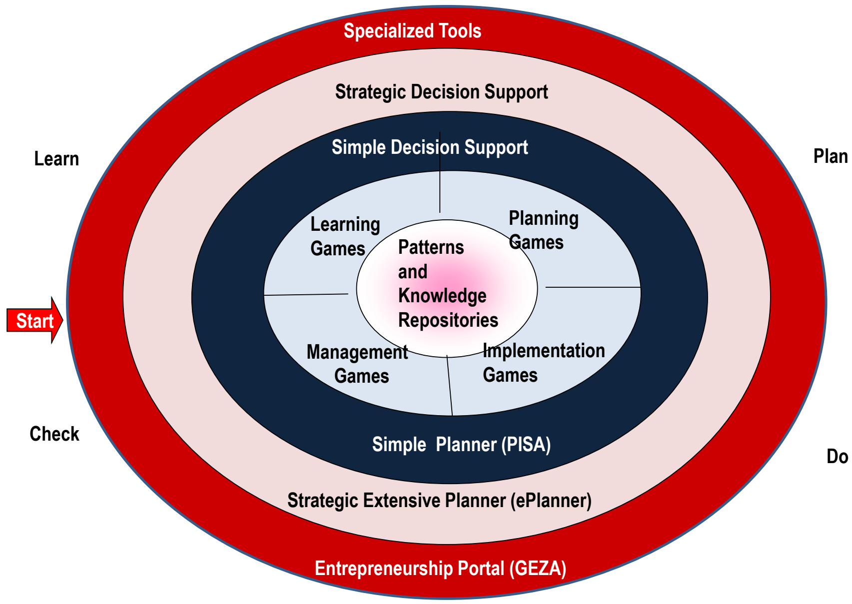


Figure 11-12: Conceptual View of SPACE (Strategic Planning, Architectures, Controls and Education).

- **Patterns Repository** that capture the core knowledge needed by all SPACE tools. The pattern repository consists of industry patterns (e.g., health, education, public safety, public welfare, transportation), technology patterns (e.g., computing platforms, wired and wireless networks), operational patterns (e.g., security and integration patterns), and even country patterns (e.g., government patterns in different countries).
- **Games and Simulations** that support decisions in strategic analysis, mobile services planning, security planning, interagency integrations and health exchanges, application migration versus integration tradeoffs, risks and failure management, and quality assurance.
- **Strategic Planner:** A strategic decision support tool for the IT officials in governments and the private sectors who need to strategically plan, architect, integrate, and manage the needed IT initiatives quickly and effectively by using the best practices.
- **Planning, Integration, Security and Administration (PISA):** A detailed decision support tool that can be used to quickly build real life business scenarios and then guide the user through IT planning, integration, security and administration tasks by using best practices and patterns.

Please note that the pattern repository is at the core of the entire SPACE environment – the patterns are used in all games and detailed plus strategic planning advisors. SPACE, spinoff of the United Nations eNabler Project, is currently being used to help developing countries and small to medium businesses to plan and engineer their systems. In addition, SPACE is being used extensively to support graduate courses and professional education in strategic planning and enterprise architectures and integration.

11.4.4 The SPACE Environment – A Quick Tour

A user of the Planner selects a service (e.g., mobile health clinic) for a given country (e.g., Nigeria) and quickly generates the following reports:

- Business plans that can be used for obtaining funding
- Detailed Planning Reports (DPRs) that show the architecture, the needed policies, and enabling technologies for the chosen service
- Standardized RFPs (Requests for Proposals) that can be used to attract the needed vendors through an open bidding process
- Project management, disaster recovery and governance guidelines for monitoring and controlling the development activities
- Education, training and public awareness campaigns needed for success

Let us briefly review how these outputs are produced by using Figure 11-13 which shows a more detailed view of the Planner. *Simply stated, the Planner is a set of intelligent apps ("advisors") that are integrated around common resources.* These advisors collaborate with each other to cover five phases (P0 to P4), shown in Figure 11-13. These advisors invoke the games, patterns, and other resources to generate the outputs shown in Figure 11-13. These outputs can be further customized by local experts and/or end users. Suppose that a user wants to develop the strategic plan for an eLearning service in Nigeria. P0 helps the user to capture Nigeria specific information and P1 helps in specification of the eLearning service. P2 generates a customized plan based on P0 and P1. P3 generates the information for RFP and requirements & integration. P4 generates outputs to support project management and governance. The outputs produced can be further customized by the users or local experts manually or by invoking specialized games and simulations. Our goal is to produce the outputs that require less than 30% of local modifications.

Using Big Data: The Planner fetches, uses and customizes extensive Big Data resources such as a set of Knowledge Repositories that provide links to a wide range of case studies and educational materials, and External Resources such as the UN Public Administration Network (UNPAN), World Economic Forum (WEF), and World Bank Institute initiative on Open Data. Rules in different phases of the Planner retrieve needed data and use it to produce outputs and/or modify decisions.

From Small and Simple Services to Large and Complex "Service Bundles". The ability to select large number of services for different countries and regions is a very powerful capability of SPACE. Specifically, the users of SPACE can do the following:

- Select a single service (e.g., a mobile health clinic) within a sector (e.g., healthcare)
- Combine different services from one or more sectors to construct "service bundles" that may represent large initiatives (e.g., Smart Cities) or interagency and B2B services (e.g., healthcare exchanges and supply chains between multiple suppliers and consumers).

Basically, a SPACE user may select an individual service or construct a service bundle for large and complex situations. Based on the choices made, the Planner automatically walks the user through the most appropriate steps and then generates very powerful outputs. Thus the Planner adjusts its behavior based on the type of service selections.

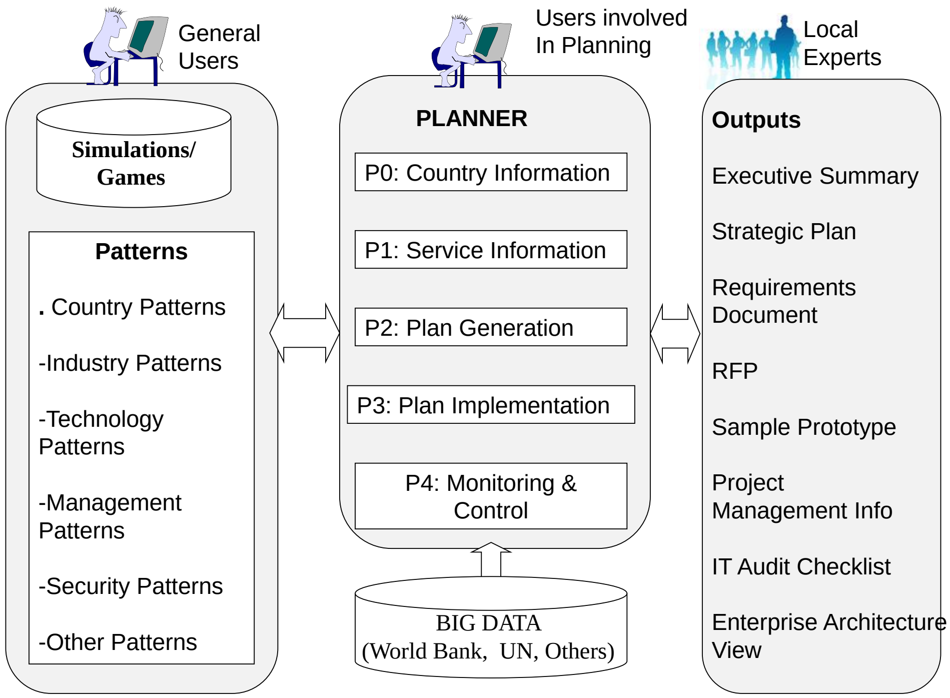


Figure 11-13: A More Detailed View of SPACE

Getting Started with SPACE

- Please go to SPACE website (www.space4ict.com)
- After reviewing the website, please go directly to the Planner section (button on the top bar). The SPACE Planner is the main capability of SPACE
- Visit the Planner Learning Corner. It has several stages that expose a visitor gradually to different features of the Planner.
- Stage 0 is just for the casual users to get them familiar with the system. No ID-PW is needed. Please use the Planner as a Guest User and run through it quickly to see the very basic features. It is very intuitive and you should be able to go through it very quickly. You can use "?" for help in any page.
- Proceed to Stage1 and review the documents in Stage1. Especially, the Conceptual Overview Document is very important. It explains the basic capabilities of the Planner and the users do not get access to the Planner Tool without reading through the Planner Overview document. Please signup for the Planner and get a Permanent ID for a better exposure to the SPACE capabilities.
- A user can go to Stage 2 and higher by reading more documents and appearing in quizzes for higher stages. Stage3 and Stage 4of SPACE are especially suited for enterprise application integration and B2B (Business to Business) integration for large and extra large scale projects.

Figure 11-14 shows the technical architecture of SPACE with its major advisors (strategic planning advisors, architecture and integration advisors, project management advisors, advisors, and gamification advisors) that utilize extensive pattern repositories to produce complete solutions. All advisors are invoked from a controller that signs a user in, assigns a unique ID to the user, and manages user sessions. This allows each user to develop and store her own object model (suggested solution) that is enriched as the user invokes different advisors. After being invoked, each advisor conducts its own interview, consults the object model to review the decisions made so far, makes inferences as much as possible, makes further decisions based on the user interview, extracts needed patterns from the patterns repository, and further enriches the object model. The object model is used as a basis for collaboration between the advisors. A great deal of information and documentation about SPACE can be found at the website www.space4ict.com.

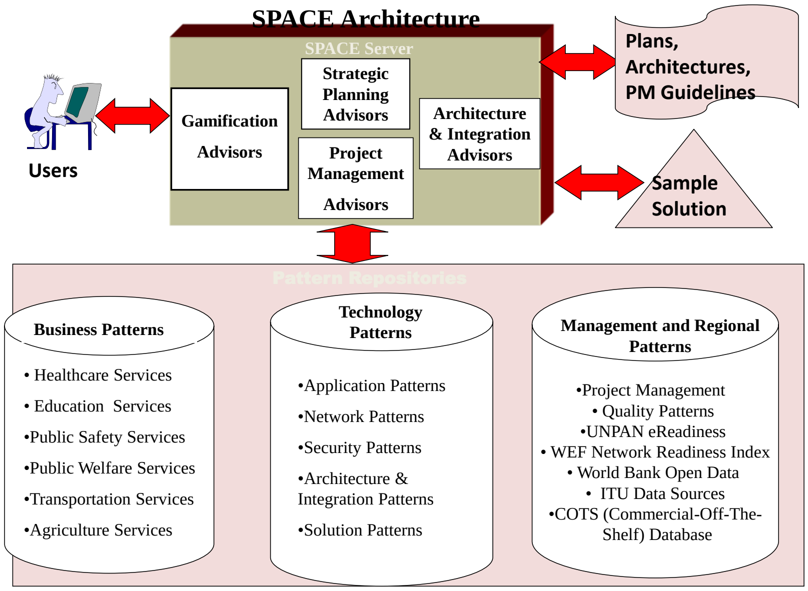


Figure 11-14: A Conceptual View of SPACE Architecture

11.5 Example of Extended Planning Using SPACE Capabilities

11.5.1 SPACE Methodology at a Glance

Figure 11-15 shows a conceptual view of the SPACE methodology that systematically guides the users through different phases of the Plan-Do-Check cycle for given eservices. This figure illustrates the flow of planning phases P0 (initialization), P1 (information gathering), P2 (strategic planning), P3 (detailed planning), and P4 (monitoring and control). The first two phases (P0 and P1) capture country and service specific information. Phase 2 generates a customized plan based on P0 and P1. P3 supports execution of the plan and phase P4 supports monitoring and control with heavy emphasis on project management and quality controls. Big Data, business patterns and intelligent rules are used in all phases of this methodology. Given a strategic project (or an initiative), this methodology identifies the main alternatives, the key business/technical issues involved in each alternative, and helps in evaluation and selection of the most viable alternatives *before* initiating the project.

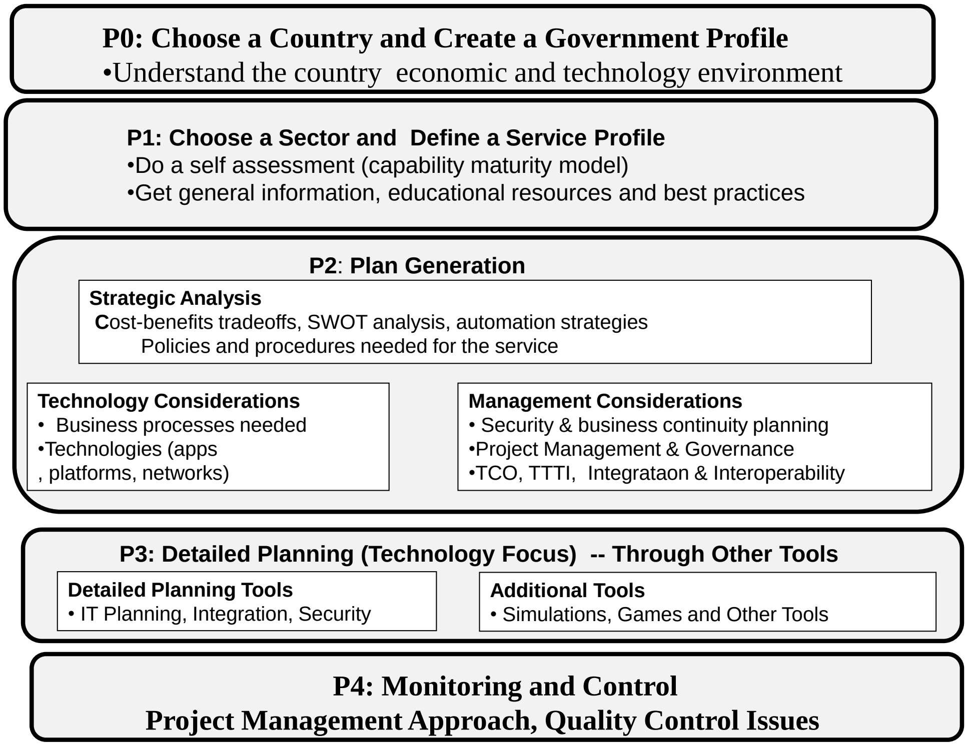


Figure 11-15: Systematic Methodology Used by the SPACE Planner

The methodology shown in Figure 11-15 can be used manually. SPACE provides computer aided support in all phases of this methodology and can produce results within an hour instead of months. How is the SPACE Planner actually used in practice? The following example illustrates the overall flow of the Planner to introduce an ICT-based *Disaster Management (DM)* service in a rural area. The purpose here is to help an agency widely provide DM services to its constituents. The following description shows the flow of the Planner, as displayed in Figure 11-15:

- In the P0 phase, the user chooses a country (e.g., Nigeria) or a region within a country. The Planner automatically fetches the most appropriate information about the country/region (e.g., population sizes, terrain, etc. This saves a tremendous amount of time and effort to the users.
- In the P1 phase, the user selects a service to be deployed (DM) and decides the types of technologies to be used for DM. SPACE consults Big Data to make sure if the needed technologies are available in the selected country/ region and gives warnings if the selected technologies are not available. SPACE then goes through self assessment (i.e., need analysis) about the DM service and automatically accesses the general information, educational resources and best practices available from the UN, the World Bank, and other Big Data sources to help the self assessment of DM service, say, in Nigeria.
- In the P2 phase, the user is led through strategic analysis (buy, rent, outsource) and cost-benefits tradeoffs associated with the DM service. The user is also guided through policies and procedures needed for the DM service.
- In the P3 Phase, the detailed planning environment can be developed through an extensive IT Planning, Integration, Security and Administration (PISA) tool, part of SPACE. Detailed IT plans can be developed easily by PISA for many sectors such as healthcare, manufacturing, education, telecommunications, retail, finance and others. The user may choose other simulations, games and decision support tools for detailed planning.

- In the P4 Phase, the progress of the project is monitored and controlled through project management techniques. In this phase, the quality of the results produced is evaluated by using the best practices in quality control.
- The final phase, not shown in Figure 3, displays the outputs produced in a well organized manner and also produces a sample prototype that can be expanded by local experts into an actual working system.

This short example highlights the main flow of the planning environment. Best practices are being used in all phases of the Planner to introduce ICT services quickly and effectively in developing countries.

11.5.2 4. From Small and Simple Services to Large and Complex "Service Bundles"

The ability to select large number of services for different countries and regions is a very powerful capability of SPACE. Specifically, the users of SPACE can do the following:

- Select a single service (e.g., a mobile health clinic) within a sector (e.g., healthcare)
- Combine different services from one or more sectors to construct "service bundles" that may represent large initiatives (e.g., Smart Cities) or interagency and B2B services (e.g., healthcare exchanges and supply chains between multiple suppliers and consumers).

Basically, a SPACE user may select an individual service or construct a service bundle for large and complex situations. Based on the choices made, the Planner automatically walks the user through the most appropriate steps and then generates very powerful outputs. Thus the Planner adjusts its behavior based on the type of service selections. Figure 11-16 shows a high level view of the services provided, bundles supported and outputs generated. These capabilities are described briefly.

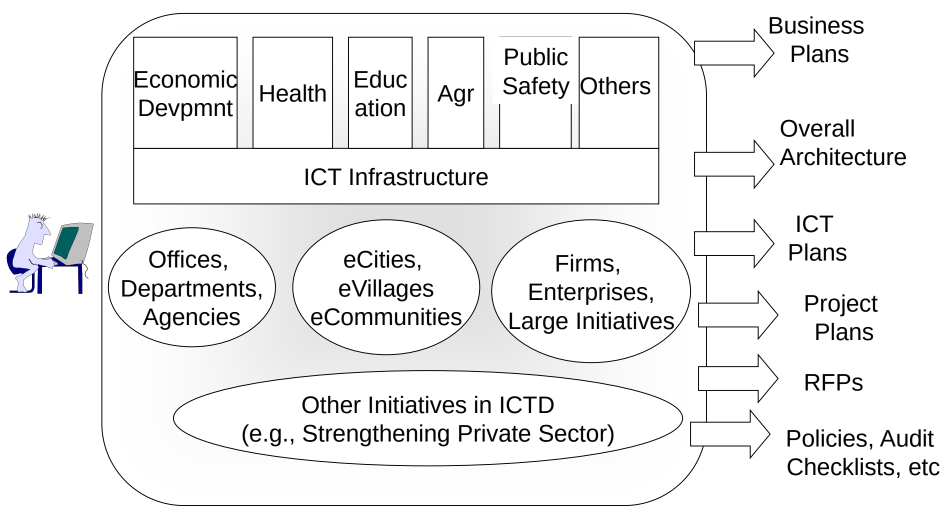


Figure 11-16: Service Types in SPACE

11.5.3 4.1 Individual Services and Sectors

The overall environment is organized into sectors and services within each sector. For example, Figure 11-16 shows sectors such as economic development, healthcare, education, and e-government. These "vertical sectors", shown as vertical bars, are supported by a horizontal sector (ICT

Infrastructure) with services such as network access and mobile computing that support all vertical sectors. Each sector provides many individual services. For example, healthcare sector provides patient care and administrative services.

11.5.4 4.2 Enterprise-Wide Service Bundles (e.g., Offices, Cities and Firms)

A user can combine different individual services into enterprise-wide service "bundles" that are managed by one organization. These service bundles, shown as circles or ellipses in Figure 4, can be used to model departments, government agencies, firms or business units, This capability of the Planner to combine several individual services from different sectors to form new service bundles is a very powerful feature that can be and has been used to represent the following real-life situations:

- Business divisions or complete enterprises in the public or private sectors such as healthcare, education, transportation, manufacturing, telecom, and others
- eCity and eVillage Initiatives that provide a wide range of ICT services that span public safety and welfare in addition to economic development and education sectors.
- Millennium Development Goals (MDGs) that span economic development, education, and other sectors.
- Mobility Initiatives that focus on introducing mobile apps and location based services in one or multiple agencies.
- Government specific initiatives at local as well as national levels in different countries (e.g., the Digital Britain Initiative).

The Planner treats each enterprise service as a single organizational unit (enterprise unit) that is managed by a central authority that can introduce and enforce common policies and procedures. This simplifies several inter-system communication problems. The interagency problems that require collaboration and coordination between multiple independent agencies are discussed next.

11.5.5 4.3. B2B Service Bundles and Enterprise Architecture Support

In addition to individual services and centrally managed initiatives in domains such as healthcare and economic development, the Planner can be used to represent large and more complex service bundles that include multiple independent agencies and organizations. The Planner provides a "Composer" that takes different services and composes them into larger and more complex service bundles such as the following (see Figure 11-17):

- A document exchange network between different government agencies
- A B2B marketplace with numerous buyers and sellers
- A supply chain system consisting of several consumers and suppliers
- A government/business network such as a health information network (HIN)

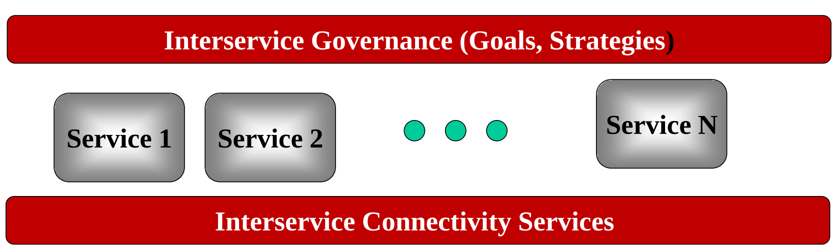


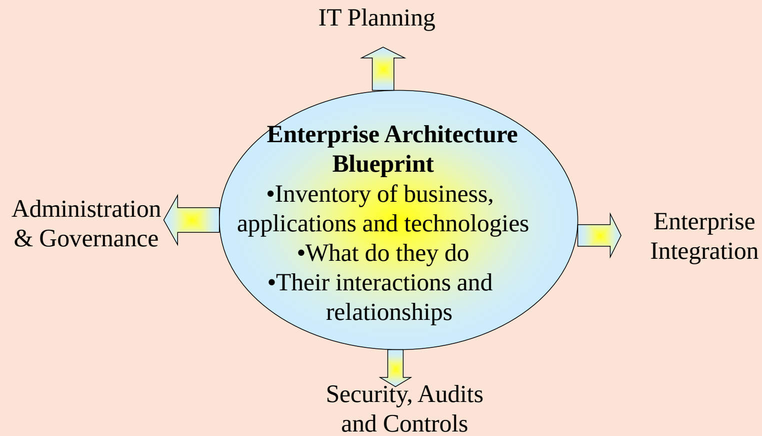
Figure 11-17: Building a Large Service from Smaller Ones

The focus here is on inter-enterprise problems that require collaboration/coordination between multiple independent agencies. The Composer treats each service developed in a session as an individual service (a reusable component) and composes large and complex servicebundles from these components by using SOA (Service Oriented Architectures). It then suggests approximate configurations with details about the governance, information exchange models (e.g., NIEM and PIP), and infrastructure components.

The Strategic Planner strongly supports enterprise architecture (EA) principles and is aligned with The Open Group Architecture Framework (TOGAF). The main phases of the planner (P0, P1, P2, P3, P4), follow the TOGAF building blocks and use a wide range of tools, techniques and standards in all phases. Additional information about EA support is provided in the sidebar "Enterprise Architecture Support in SPACE Planner".

Enterprise Architecture Support in SPACE Planner

An enterprise architecture (EA), as shown below, is basically a repository of information that can be used to plan, manage, secure and integrate an enterprise. The SPACE Planner captures and uses this information during its phases, as shown in Table 1, and generates powerful reports to support the administration, planning, integration and security activities of an enterprise.



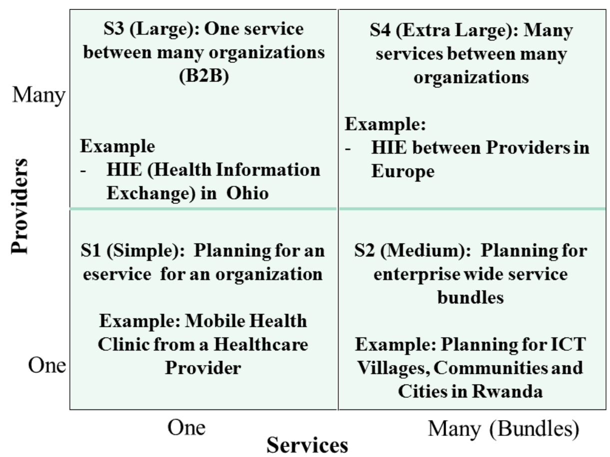
11.5.6 Examples of Using SPACE: Simple to Large & Complex

SPACE advisors in phases P0-P4 shown in Figure 4 are used to plan and architect very simple to very large and complex situations. Our initial experiments with SPACE have given us valuable insights from simple scenarios but have indicated much better cost-benefit ratios from larger scenarios. Thus we are especially interested in examining the benefits as well as the costs of large smart enterprise initiatives with many integrated smart services. The short examples in this section provide some technical insights about SPACE. It should be noted that possible real life application scenarios for a computer aided planning platform such as SPACE are potentially very large.

Figure 11-18 shows four possible categories of simple to large and complex scenarios in terms of services and service providers. Examples of scenario categories are:

- S1: This category represents single service for a single provider. The users of SPACE can select more than 100 smart services from health, education, public safety, public welfare and other vital sectors. For example, a user can select Mobile as examples of these simple scenarios.

- S2: This category represents a service bundle by a single provider. SPACE users can combine many individual services to form service bundles that represent doctor's offices, health clinics, telemedicine centers, integrated practice units (IPUs), assisted living centers, health community centers, and healthcare agencies. A Telemedicine IPU is an example of these medium scenarios.
- S3: This category represents a service shared by multiple providers. This scenario category can be used to model a large number of B2B services such as Health Information Exchanges (HIEs) between different healthcare providers and interagency services in governments.
- S4: This category represents service bundles between multiple providers. This scenario can be used to model large and complex projects such as interagency projects and large health exchanges within a state or country. We will discuss an HIE (Health Information Exchange) between different providers as an example of large and complex scenarios for smart health and human services.



Service Scenarios – From Small to Large and Complex

Figure 11-18: Handling Small to Large Scenarios in SPACE

Suggested Review Questions Before Proceeding

- What is SPACE and what are its main components
- What are the main steps of the SPACE methodology
- How SPACE uses "Service Bundles" to handle large and complex enterprises
- How SPACE supports Enterprise Architectures
- List the names of patterns for used in SPACE
- What are enterprise ontologies
- Conduct a literature survey and identify 2-3 tools that are similar to SPACE

11.6 Concluding Comments and Next Steps

In its mature prototype (Beta) mode, the SPACE Environment is available at www.space4ictd.com and can also be accessed from the UN-Gaid eNabler site (www.enabler4mdg.org). Potential users can choose more than 100 individual services spanning health, education, agriculture, public welfare and economic development and generate detailed planning reports that contain business plans, policies, requirements, technologies and project management recommendations. In addition, SPACE fully supports composition of these individual services into enterprise-wide and inter-enterprise services. The eBusiness capabilities are provided through a similar environment called PISA (Planning, Integration, Security and Administration) available at www.ngepisa.com.

We have learned several invaluable lessons in this project. The key positive finding is the significant reduction of time (from 4-5 months to 2-3 days) and increased chance of success due to consistency of processes and quick availability of common practices. This reduces cost and reduces expensive retries and thus could possibly lead to equality at a global level. The major challenge is training of the practitioners in the underserved sectors. To address this challenge, we have been improving the training and educational capabilities of the SPACE environment and have reorganized the SPACE website so that different user types are exposed to different sections of SPACE.

Our long range goal is to make the SPACE environment a very powerful tool that can play a crucial role in advancing eGovernment and eBusiness initiatives in underserved segments around the globe. Some of the future directions are:

- Expand the "Learn and Replicate" capabilities by extensively using a social network between the users of the system. This will help the users to exchange ideas, views, experiences and lessons learned.
- Significantly expand the games and simulation capabilities. Most of the SPACE advisors at present are implemented as Web Services so that they can be invoked from another advisor or from a game.
- Support more complex services that span multiple agencies (e.g., multiple government agencies from multiple countries). This is currently operational but we want to expand it more.
- Expand the intelligence capabilities of the inference engine by improving the reasoning and learning features through use of recent developments in machine learning, fuzzy logic and case-based reasoning.

11.7 Appendix A: Patterns-based Decision Support for IS Planning – A Closer Look

11.7.1 introduction

Patterns are a well-known format for capturing engineering knowledge. The idea was introduced by Christopher Alexander, a civil engineer, who wrote a series of books [Alexander 1977, Alexander 1979] observing that well accepted buildings have common structures. Based on this, he devised a set of rules for architects to construct such buildings. According to Alexander, "Each pattern describes a problem that occurs over and over again in our environment and then describes the core of the solution to that problem in such a way that you can use this solution a million times over without ever doing it the same way twice"[Alexander1979]. The "Gang of Four" extended the pattern format to software design [Gamma1994]. Since then, patterns have been used extensively in software design and have been extended to e-buisness patterns [Adams 2001], requirements patterns [Ferdinandi 2002], architecture patterns [Buschman 1996], security patterns [Kienzle2001], and others. These patterns make it easier for the developers to reuse the experience of others. The SPACE environment introduced in this chapter is heavily based on patterns. See the website (www.hillside.net/patterns) for extensive discussion, tutorials, and articles on patterns.

At a very basic level, a pattern *T* is a tuple *T(p, c, s)* where *p* is the problem to be solved, *c* is the context (under what conditions the pattern holds, i.e., why the problem needs to be solved), and *s* is the solution (what works in practice*)*. Additional information such as examples and limitations can also be added to a pattern to help the designer. In addition, each pattern is assigned a name. Exhibit 1 shows a simplified example of a well known design pattern (Adapter) that occurs commonly in software engineering. Some patterns can be quite detailed and complex. We will discuss several patterns in this paper. Due to space limitations, we cannot show every pattern completely by using the format shown in Exhibit 1. Instead, an abbreviated version consisting of few sentences will be used to highlight the essence, i.e., the solution, of each pattern. For example, we will use the following sentence to describe the pattern shown in the sidebar "A Pattern Example (Simplified)". The Adapter pattern shows how an intermediate object can be used to integrate two systems together.

A Pattern Example (Simplified)

Name: Adapter

Problem: How to interconnect and integrate two different systems

Context: Whenever different systems need to communicate with each other

Solution: Develop or buy a new component that does the translation between the two systems. Imbed all the methods that do the translation in this new component and make it general so that it can be reused over and over again. .

Example: Adapter for SAP applications that connects SAP applications to connect with other

vendor applications.

The main value of a pattern is the solution *s* it specifies. The solution represents the best practice and what works in real life situations. The solution s is provided to a designer as a generic solution -- a sketch -- that can be refined and specialized based on the situation, additional inputs, or inferences from other patterns For example, the Adapter pattern is a generic solution that can be refined based on how different the two systems are. For example, if the two systems only use different currencies, then the adapter can do the currency transformation. However, if the two systems use completely different middleware technologies, then the adapter will be more complex.

The solutions in a pattern may be derived from other patterns. For example, taking the civil engineering examples of Alexander, the design of a living room depends on the design of the house (larger homes can have larger living rooms with more windows). To support decisions, different patterns need to be interconnected so that solutions in one stage may influence solutions in a later stage. The chained patterns can be represented as rules. For example, consider two patterns: *T1(p1, c1, s1)* and *T2(p2, c2, s2).* Let us assume that -> indicates an "influence" operation, then the following rule will interconnect the two patterns:

```
If (s1 -> c2) then use s2
```

The main value of patterns in decision support is that they suggest proven solutions to given problems. First, the problems become relevant due to some situations. In addition, the solutions are not standalone, they are dependent on previous choices that are represented by other patterns. Thus a hierarchy of patterns can be developed where solution in one pattern is used to influence and invoke solutions in other patterns Thus mechanisms are needed that interconnect different patterns together where a solution in one pattern leads to another set of problems that become relevant and need new solutions. We will expand on interconnected patterns in the next section.

Patterns can be of different types. For example, they may provide high level guidelines, called ***meta patterns***, or provide implementation details such as program patterns. Within the context of IS planning, we are primarily interested in meta patterns because they are more useful to IT managers. These patterns may be obvious to software designers and thus taken for granted, but they are not obvious to IT managers. Unlike software patterns that concentrate on issues such as layout of program objects in main memory, meta patterns concentrate on higher level issues such as how to do outsourcing, what should be automated in a small business, what type of COTS to use, what type of computing platforms to use, what type of network to use, and the like. Another class of patterns of value to us is "***anti-patterns***" that represent the solutions that *should not* be used. Anti-patterns are useful in IS planning to warn the IT managers about some solution approaches that simply do not work.

We should also note that our objective is to build a decision support tool, thus we are quite comfortable with not knowing the exact solutions at present. As more and better solutions become available, we will update the tool database. In fact, one of our design criteria is to use an external ‘patterns’ database’ that will drive the process. This database can be updated quickly and easily so that new solution approaches can be added as they become available. This is in sharp contrast to other tools such as the eBusiness Framework developed by IBM that contains most of the pattern information in static HTML files.

Patterns: Definitions, Examples and References

An early definition of patterns is offered by Alexander:

- “Each pattern describes a <u>problem that occurs over and over again</u> in our environment and then describes the core of the solution to that problem in such a way that <u>you can use this solution a million times</u> over without ever doing it the same way twice” Alexander, “A Pattern Language”, 1977.

A DARPA Working Group developed the following definition in 1997:

- “<u>*Patterns* are reusable abstractions of best practices</u> that can be documented and logged in a pattern repository. They range from detailed algorithms to enterprise architectures and to other fields like conventional architecture”.

Examples of Patterns

Why worry about patterns? Patterns are useful because they represent a common library that experienced professionals are likely to see over and over again. Patterns can also be used to cluster similar things and make inferences based on the similarities.

Initial focus of patterns in computing has been on technical areas such as design patterns, architecture patterns, and pattern languages. However, considerable attention is now being paid to business patterns, security patterns, integration patterns and the like (see references on patterns below). These patterns are being used for the following purposes:

- Business patterns can be used to capture the common business processes in various industry segments. For example, a healthcare business pattern can represent common healthcare business processes.
- Security patterns can be used to represent solutions to commonly known security problems. For example, an 802.11 security pattern shows the best practices in security solutions for 802.11 networks.
- Integration patterns can be used to represent common approaches to integration problems. For example, an SOA pattern can be used for enterprise wide application integration.

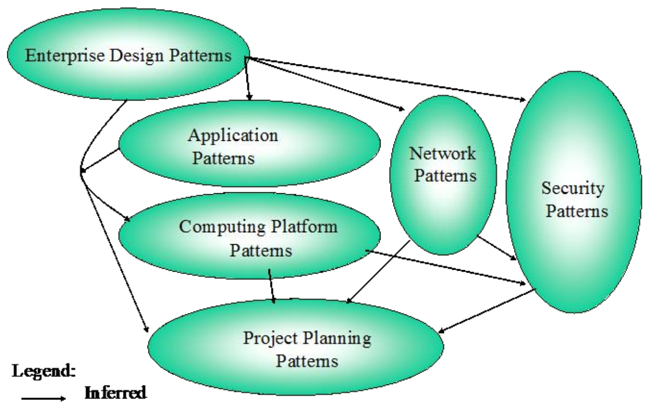
Some patterns are available over the Internet. An example is the IBM's Patterns for e-business -- an extensive web site with a great deal of information on this topic (http://www-106.ibm.com/developerworks/patterns/).

Selected References on Patterns:

- Adams, J., et al, *Patterns for e-Business: A Strategy for Reuse*, IBM Press, October 2001.
- Alexander, C., *The Timeless Way of Building*, Oxford University Press, 1979
- Alexander, C. et al , *A Pattern Language*, Oxford University Press, 1977
- Buschmann, E., et al, "Pattern-Oriented Software Architecture, Vol. 1: A System of Patterns", John Wiley, 1996.
- Ericksson, H. and Penker, M., *Business Modeling with UML – Business Patterns at Work*, John Wiley, 2000.
- Ferdinandi, P., *A Requirements Pattern: Succeeding in the Internet Economy*, Addison-Wesley, January 2002.
- Gamma, E., et al, *Design Patterns*, Addison Wesley, 1994.
- Hagge, L. and Lappe, K., "Sharing Requirements Engineering Experience Using Patterns", *IEEE Software*, January-February 2005, pp. 24-31.
- Hohpe, G. and Woolf, B., *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*, Addison-Wesley, 2003.
- *IBM e-Business Framework* website http://www-106.ibm.com/developerworks/patterns/
- Kalakotta and Robinson, *E-Business 2.0*, Wiley, 2002 (Chapter on Business Patterns)
- Kienzle, D., and Elder, M., "Security Patterns for Web Development", *DARPA Contract No: F30602-01-C-0164*, June 2001. Weblink: http://www.scrypt.net/~celer/securitypatterns/final%20report.pdf.
- Patterns Web Site, http://hillside.net/patterns

11.7.1 Patterns Based IS Planning

Figure 11-19 shows a pattern-based IS planning model that illustrates the interactions between major patterns such as enterprise design patterns, application patterns, computing platform patterns, network patterns, security patterns, and project planning patterns (a more detailed view is presented in Figure 2 and will be discussed later). These patterns support the various activities of IS planning and are the foundation of patterns-based IS planning. The main idea is to start with enterprise design patterns (EDPs) that represent commonly occurring business situations. Computer aided tools can use these patterns to infer various business as well as technical issues that are associated with an EDP. For example, online-purchasing is an EDP that represents a business decision of a company to sell its products over the Web. From this pattern, one can infer the technologies needed to support this pattern (e.g., a Web browser, a Web Server, an online catalog, a payment system, a shopping cart, an order processing system, etc.). In addition, several business issues related to this pattern can be identified (e.g., what are the costs and risks involved, should part or all of the system be outsourced, what are the security constraints, what type of systems need to be integrated with this system, are there any regulatory issues involved, etc.). These EDPs can be combined to build models of companies and inferences about these patterns can be refined successively through interactions with a user.



Enterprise Architecture Support in SPACE Planner

Figure 11-19: A Pattern-Based Research Methodology

11.7.2 Patterns for Enterprise Design

Enterprise design patterns (EDPs) capture the essence of proven solutions to recurring problems in enterprise design. Numerous patterns can fall into this category. Our goal in this paper is to identify the core EDPs that significantly influence the decisions made in IS planning. The following EDPs represent a sample:

- **EDP-BPI (business process identification):** This pattern suggests that similar business processes (BPs) exist in similar types of businesses. For example, small manufacturing companies have many common BPs, so do small banks and travel agencies. Thus a business ontology repository could be very useful to provide a list of BPs for different industry segments. An example of such a repository is the "Open Source Business Process Ontology" (www.jenzundpartner.de/Resources/RE_OSSOnt/re_ossont.htm). Another example is the SAP's Business Maps (www.sap.com). We have used a combination of Open Source BPO and SAP's Business Maps and modified them for SMBs as a starting point for EDP-BPI.
- **EDP-CSF (critical success factor);** This pattern suggests that the CSF methodology can be used successfully to identify the critical business processes that must be supported properly. The idea of CSF, popularized by [Rockart 1979], is very simple -- in any organization certain factors are critical to the success of that organization and must be crucial players in the planning process. The CSF approach has been used successfully in many cases and has been extended to facilitate IS planning [Peffers 2003].

- **EDP-BPO (business process outsourcing):** This pattern highlights the important issue of outsourcing business processes. Although this area is evolving, some approaches and guidelines for BPO can be used to develop answers to typical outsourcing questions.
- **EDP-BPA (business process allocation):** This pattern addresses the issue of what business processes would take place where (i.e., should manufacturing be done at a different site, should design be co-located with manufacturing or research). Business process allocations also lead to workgroup (WG) formations. For example, workgroups are typically formed around BPs at the same sites. The classical principles of maximizing cohesion and minimizing coupling can be used to form effective WGs. For example, human resource departments are typically co-located with corporate management due to the frequent interactions between the two. As we will see later, EDP-BPA and the WGs formed will impact the choices of applications, computing platforms, network design, and security.

11.7.3 Patterns for Application Planning

Application planning in the 21st century is quite different as compared to the previous century where application planning concentrated primarily on in-house software development. This is largely due to the availability of highly configurable commercial-off-the-shelf (COTS) packages, rentals through application service providers (ASPs), popularity of software development outsourcing, and increasing appeal of software re-use through service-oriented offerings based on Web-Services, .Net and J2EE. Due to this, we focus on capturing those requirements that help in choosing between the buy, rent, outsource, build, and re-use strategies. In addition, we need to capture the common choices made in application planning. Application patterns represent the requirements as well as the solutions:

- **Application Requirements Patterns** represent common requirements. Specifically, REQ-CORE, REQ-COTS, REQ-INTERFACE, and REQ-DOC patterns show the recent approaches used to capture different aspects of requirements. For example, REQ-CORE represents the core (functional) requirements of an application. For many existing application, a core requirement pattern exists that can be specialized for certain situations. For example, all inventory management systems have many common functional requirements. This is also true for the customer relationship management and supply chain management systems. The REQ-COTS pattern captures requirements needed for COTS (commercial-off-the shelf) selection and can be based on the PORE (procurement-oriented requirement engineering) approach suggested by Maiden and Neube [Maiden 1998]. The REQ-DOC pattern suggests good approaches for documenting the requirements. These include the one suggested by [Haage and Lappe 2005] and the requirements patterns suggested by [Ferdinandi 2002].
- **Application Solution Patterns** suggest solutions to the requirements patterns and show the common application decisions such as automation strategy and COTS selections. The APP-STRATEGY pattern helps in choosing between rent/buy/develop/re-use strategies and suggests a few factors (time constraints, budgetary constraints, cost considerations, privacy concerns , uniqueness of the system, flexibility requirements, in-house expertise, existing systems, etc). This pattern also suggests automation strategies, such as the following: "`IF (time.LT.month) AND (privacy.EQ.Low) THEN STRATEGY = rent`". In these rules, the conditions are recorded in the requirement patterns. The APP-SOLUTIONS pattern suggests a solution approach for applications. For purchasing, this pattern suggests the successful practice of buying “integrated application suites” instead of individual packages. For development, this pattern suggests concurrent software approach – a more common practice at present. These patterns depend on the EDPs because manufacturing companies choose different types of applications than financial companies. Similarly a small manufacturing company would choose different type of applications.

11.7.4 Patterns for Computing Platform Planning

Computing platforms represent the computing hardware/software needed to support the application plans. Computing platform patterns (CPPs) capture the best practices in computing platforms (laptops, PDAs, Desktops), the system software (OS. utilities, compilers) and middleware services (e.g., Web browsers/servers, distributed database and transaction managers) in contemporary environments. These patterns are influenced by the application and enterprise design patterns.

- CPP-WORKSTATIONS. This pattern shows the characteristics of user workstation (hardware, system software, middleware) for different types of users as determined by the EDP-BPA pattern. For example, a human resource department clerk will need a different type of workstation than a CAD engineer.
- CPP-SERVER: This pattern shows how the server platforms can be selected. Due to the decreasing cost of servers, each work group cam be assigned one server. In addition, common server configurations from suppliers such as Dell are well known and described in some detail by the suppliers.
- CPP-INTERDEP: This pattern captures the interdependencies between the various application software, system software, and computing hardware devices on various platforms. For example, different devices need different device drivers, IIS Web server requires Windows Professional Edition to operate, etc.

11.7.5 Patterns for Network Planning

Network plans support the enterprise, application, and computing platform plans. Network patterns capture the successful network (wired as well as wireless) solutions. These patterns are influenced by the application, computing platforms and enterprise design patterns. Examples of these patterns are:

- NET-WM: This pattern suggests that a workload model can be inferred from workgroups and the applications that run in the workgroup. For example, in a human resources WG, the workload can be estimated based on the common activities performed in human resources (e.g., payroll, benefits, etc). Similarly, network workload generated by software developers includes large downloads and frequent website searches – a pattern not found in clerical staff.
- NET-LAN, NET INTERNAL, and NET-EXTERNAL patterns represent common practices in modern network design. For example, Ethernet is used commonly in wired LANs, Wi-Fi is common in wireless business LANs, and Bluetooth and Zigbees are used most frequently in consumer electronics. In addition, the Public Internet is used commonly with VPN (Virtual Private Network) support for added security where needed. For highly secure network transport in Intranets and Extranets, privately owned T1 and T3 lines are used commonly in wide area networks. For wireless local loops, LMDS (Local Multipoint Distribution Service) is being used commonly and FSO (free space optics) has gained popularity for highly secure transmissions over shorter distances (1 to 2 Kilometers). In other words, even in the highly complex network world, successful patterns have emerged.

11.7.6 Patterns for Security Planning

Security plans are needed to support the enterprise, application, computing platform, and network plans. Security patterns suggest security solutions to common security problems. Some work on security patterns has been reported by Kienzle and Elder [Kienzle 2001]. However, this work is limited to cataloging security patterns for Web development and is not suitable for security planning. For the purpose of this research, we have identified the following security patterns:

- SEC-WIFI, SEC-INTERNET, and SEC-WEB patterns capture the best practices in developing security for Wi-Fi, the Public Internet, and the Web. SEC-WI-FI pattern is a set of successful recommendations such as the following: a) Place the Wireless LANs outside instead of inside the corporate firewalls so that all such LANs internal or external will have to go through the firewall to access any corporate resources, b) Encrypt the wireless LAN traffic by using solutions such as wireless VPN, c) Heavily protect the resources accessed through wireless LANs. By using higher-level security offered by SSL, PGP, and SET to protect Web resources, email, and financial transactions, and d) Explore the new standards such as 802.11i for improvements in 802.11 LANs. SEC-INTERBET and SEC-WEB patterns suggest use of VPN and SSL, respectively, with some technical details.

- SEC-AUDIT and SEC-POLICIES patterns suggest the commonly used approaches for security audits and policies/procedures. A security audit and control checklist has been presented in [Umar 2004] that has been used as a SEC-AUDIT pattern and has been customized for several industrial settings in consulting assignments. Development, deployment, and enforcement of security policies is a non-trivial task in the modern digital enterprises and is an important aspect of security planning. SEC-POLICIES pattern suggests proven approaches to developing security policies quickly. For example, the best approach is to obtain pre-written policies, and then customize them to satisfy your needs. The British standard, known as BS7799 (also ISO 17799), provides a very good pattern that has been customized successfully by many organizations. An example of a good policies review pattern is the the NIST Program Review for Information Security Management Assistance (PRISMA).

11.7.7 Patterns for Project Planning

Project planning is an essential aspect of IS planning. In particular, project plans represent the work breakdown structure (WBS) and staffing needed to support the decisions made in application, computing platform, network, and security planning. Project plan patterns (PPPs) suggest typical WBS for different types of activities. For example, if an SMB embarks on a new application development project, then a project plan would reflect the new applications to be developed and the changes to network and computing hardware. The PP-DEV pattern represents the WBS needed for software development. This pattern has pre-fabricated WBS because we know what each development process requires and how are these activities. In addition, certain defaults can be assumed based on common practices in SMBS. WBS-DEV, WBS-BUY, WBS-RENT, and WBS-OUTSOURCE patterns represent the most commonly used WBSs in application development, buying, renting, and outsourcing, respectively. These WBSs represent the activities, their precedence relationships, and the typical length of activities.

11.8 Appendix B: From Enterprise Patterns to Enterprise Ontologies [1]

Simply stated, ontology is a common vocabulary and a “formal, explicit specification of a shared conceptualization” [Gruber 1993]. Enterprise ontologies are at the core of enterprise integration due to their ability to provide common vocabulary between diverse systems. Figure 11-20 shows a conceptual view of the major domains of our ontology (enterprise processes, enterprise applications, integration technologies, and computer-communication platforms), their components and interrelationships.

- The enterprise processes of an enterprise consist of business functions (BFs) that represent the functional areas of a business (e.g., marketing, sales, human resources, finance, manufacturing, etc) and the business processes (BPs) within each functional area.

[1] This section is extracted from a paper [Umar 2009].

- The enterprise applications that automate the enterprise processes are represented in terms of an enterprise plan that automates the enterprise and is further subdivided into application suite (e.g., a supply chain management suite from i2), and application packages (e.g., order processing package from Amazon.com).
- The integration technologies provide the software that interconnects and integrates applications at B2B level (B2B integration platforms), at enterprise level (Enterprise Application Integration -- EAI -- platforms), and at individual application level (adapters). The computer-communication platforms provide the necessary enabling technologies for the enterprise applications and consist of system software (e.g., operating systems, database managers, and utilities), networks (routers, gateways, wireless access points) and computer-communication hardware. Due to space limitations and to stay focussed on application integration, we will henceforth not discuss this domain in this document.

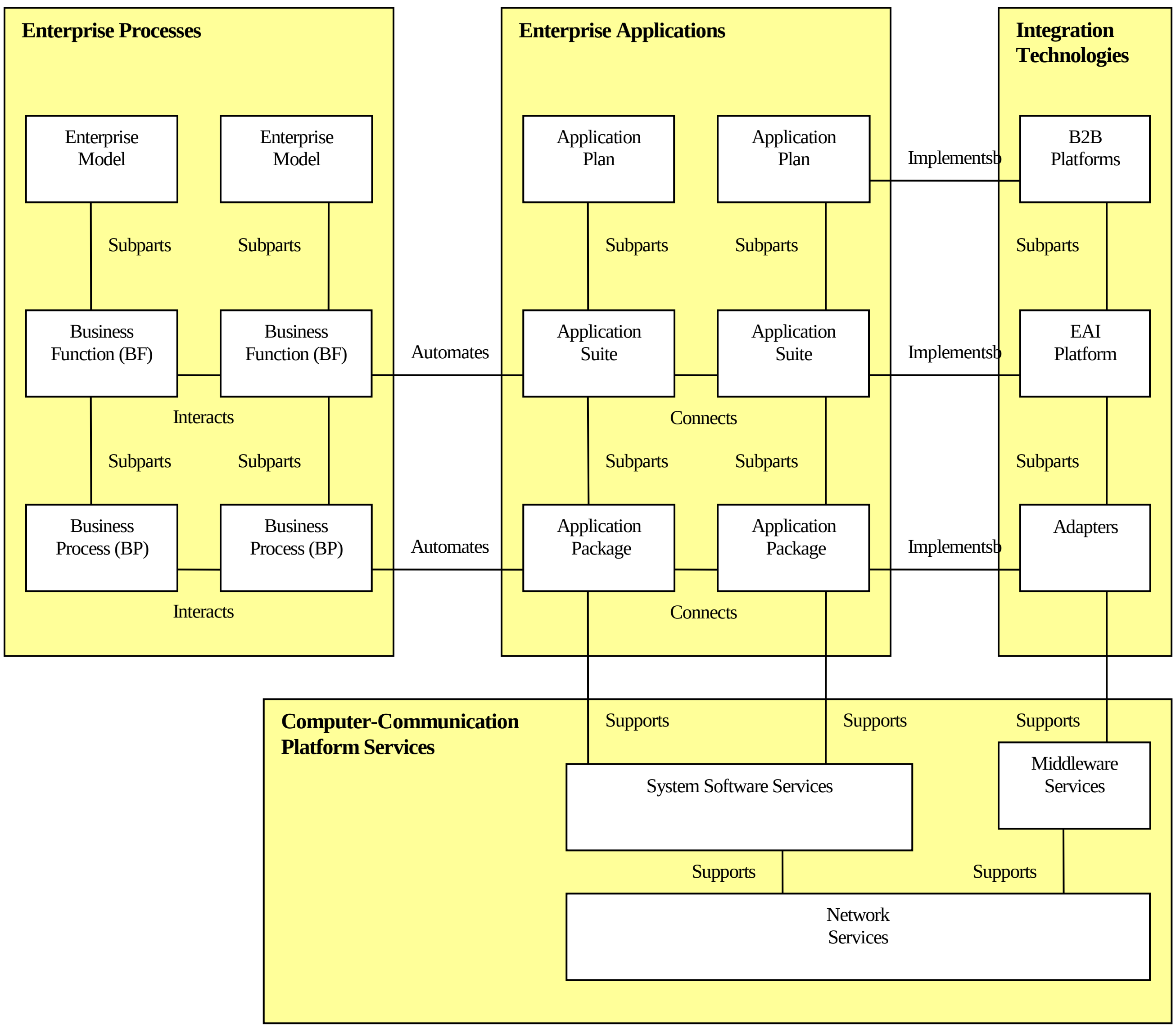


Figure 11-20: High Level View of Proposed Enterprise Ontology

Figure 11-20 also shows the main relationships (subparts, interacts, automates, connects, and implements) between the key concepts of the proposed ontology. These relationships carry semantics.

For example, subparts represents a consists-of relationship and implies that lower level objects inherit common properties. The relationships are within a domain (e.g., “interacts” between two business processes) or between domains (e.g., an application "automates" a business process). The core elements of the ontology are represented as domain class diagrams (only names, attributes, relationships). These elements are only at the level of granularity needed for IS planning and integration. These domains comply with the OMG MDA (Model Driven Architecture) taxonomy because business processes represent a CIM (computation independent model), enterprise applications represent the PIM (platform independent model), and integration technologies and computer-communication platforms represent the PSM (platform specific model). This ontology also allows for EAI and B2B integration.

Table 2 shows the key elements of the proposed ontology along with attributes that capture the properties of the different objects and further define the object type in different domains. For example, the enterprise objects represent industry type (e.g., manufacturing), industry size (small, large), and position in organization (C2B, B2B, B2E). The position attribute helps infer many different aspects of a plan. For example, C2B applications typically use the Public Internet and thus rely on Web technologies very heavily. The AKA (also known as) attribute is used to represent synonyms and other terms used to represent similar objects. AKA is very useful for semantic translation (same object named differently).

Table 2: Elements of Enterprise Ontology

Domain Granularity Level	Enterprise Processes	Enterprise Applications	Interconnectivity Technologies
Coarse	Enterprise represents a business entity . Attributes: industry Type (manufacturing, etc), size (large, medium, etc), AKA (company, firm)	Enterprise Application Plan represents a collection of applications. Attributes: Plan Type (long range, short range, etc). AKA (company, firm)	B2B Integration (B2BI) Platform represents a collection of B2B interconnectivity software Attributes: EAI Platform Type (Web-services based, proprietary). AKA (B2B server)
Medium	BF, represents business functional areas. Attributes: Specificity (common/industry specific). Position (c2b, b2b, b2E), AKA (functional area)	Application Suite: represents an integrated set of applications Attributes: Technology-used (Web-services, etc). AKA (ERP system, Enterprise System, Enterprise Component)	Enterprise Application Integration (EAI) Platform represents a collection of interconnectivity software Attributes: EAI Platform Type (Web-services based, proprietary). Focus (front-end, back-end), AKA (message brokers)
Refined	BP (Business Process) Attributes: Specificity (common/industry specific). Position (C2B, B2B, B2E), AKA (business services)	Application Package Attributes: Specificity (common/industry specific). Position (C2B, B2B, B2E), AKA (Business component)	Adapter Attributes: Adapter Type (data, function, user interface level). AKA (mediator)

References

- Adams, J., et al, “Patterns for e-Business: A Strategy for Reuse”, IBM Press, October 2001.
- Alexander, C., “The Timeless Way of Building*”,* Oxford University Press, 1979
- Alexander, C. et al , “A Pattern Language”, Oxford University Press, 1977
- Blokdijk, A. and Blokdijk, P., "Planning and Design of Information Systems", Academic Press, 1987.
- Boar, B., "The Art of Strategic Planning for Information Technology", Second Edition, John Wiley, 2001.
- Boar, B., “Practical Steps for Aligning Information Technology With Business Strategies”, John Wiley, 1994.
- Bode, S., & Riebisch, M. *Tracing the Implementation of Non-Functional Requirements.* Retrieved Feb 2, 2014, from igi-global.com: http://www.igi-global.com/viewtitle.aspx?titleid=52227&sender=aa801e50-1c1f-45d0-96e9-348334b0606f, 2011
- Burton, F. et al, “An Application of Expectancy Theory for Assessing User Motivation to Utilize an Expert System”, *Journal of Management Information Systems*, Vol. 9, Issue 3, pp. 183-199., 1993.
- Buschmann, E., et al, “Pattern-Oriented Software Architecture, Vol. 1: A System of Patterns”, John Wiley, 1996.
- Cardwell, G., “The influence of Enterprise Architecture and process hierarchies on company success”, Total Quality Management & Business Excellence, 19(1/2), 47., 2006.
- Chesbrough, H. and Spohrer, J., “A Research Manifesto for Service Science”, *Comm of ACM*, July 2006, pp. 35-40.
- Dada, D., *The Failure of e-Government in Developing Countries*, EJIDSC, Vol 26, no. 7, 2006, link: http://www.ejisdc.org/ojs2/index.php/ejisdc/article/viewFile/277/176.
- Gamma, E., et al, *Design Patterns*, Addison Wesley, 1994.
- Gruber, T. “Towards principles for the design of ontologies used for knowledge sharing,” presented at the Int. Workshop on Formal Ontologies., Padova, Italy, 1993.
- Ferdinandi, P., "A Requirements Pattern: Succeeding in the Internet Economy", Addison-Wesley, January 2002.
- Hagge, L. &Lappe, K., "Sharing Requirements Engineering Experience Using Patterns", IEEE Software, January-February 2005, pp. 24-31.
- Hevner, A., et al, "Design Science in Information Systems Research", MISQ, Volume 28, Number 1, 2004, pp. 75-105.
- Hohpe, G. and Woolf, B., "Enterprise Integration Patterns : Designing, Building, and Deploying Messaging Solutions", Addison-Wesley, 2003.
- Kienzle, D., and Elder, M., “Security Patterns for Web Development”, DARPA Contract No: F30602-01-C-0164, June 2001. Weblink: http://www.scrypt.net/~celer/securitypatterns/final%20report.pdf.
- Maiden, N.A., and Neube, C., “Acquiring COTS Software Selection Requirements”, IEEE Software, vo.l. 15, No. 2, 1998, pp. 46-56.
- Mentza, G. et al, “Knowledge Services on the Semantic Web”, CACM, , June 2007
- Peffers, K., Gengler, C., and Tuunanen, T. “Extending Critical Success Factors Methodology to Facilitate Broadly Participative Information Systems Planning”, Journal of Management Information Systems, Vol. 20 No. 1, Summer 2003, pp. 51 – 85.
- Porter, M. E., and Lee, T., *The Strategy That Will Fix Health Care*, Harvard Business Review, October, 2013.

- Rockart, J.F. “Chief Executives Define their own Data Needs”. *Harvard Business Review,* **57** (2), 1979, 238-241, 1979
- Rosenfeld, E., *After all this…. Why Do Projects Fail?*, http://www.adaptivepartners.com/projfailb.htm.
- Todd, P. and Benbasat, I., “The Use of Information in Decision Making: An Experimental Investigation of the Impact of Computer-Based Decision Aids”, MIS Quarterly, Vol. 16, No. 3 (Sep., 1992) , pp. 373-393
- Umar, A., and Zordan, A., “Enterprise Ontologies for Planning and Integration of eBusiness*”, IEEE Transactions on Engineering Management*, May 2009, Vol. 56, No. 2, pp. 352-371.
- Umar, A., “Computer Aided Planning, Engineering and Management of IT Services”, IEEE International Technology Management Conference, Dallas, June 2012.
- Umar, A., “Computer Aided Consulting for Developing Countries”, IEEE International Technology Management Conference, Hague, June 2013.
- Umar, A., “Decision Support for Smart Healthcare”, IEEE International Technology Management Conference, Chicago, June 2014.
- Umar, A., and Zordan, A., “Enterprise Ontologies for Planning and Integration of eBusiness*”, IEEE Transactions on Engineering Management*, May 2009, Vol. 56, No. 2, pp. 352-371.
- Umar, A., Intelligent “Decision Support for Architectures and Integration of Next Generation Enterprises ”, *Informatica*, V. 31, No. 14, pp. 141-150, 2007
- Umar, A., "Information Security and Auditing in the Digital Age”, NGE Solutions, May 2003, Revised August 2004.
- Ward, J., and Griffiths, P., "Strategic Planning for Information Systems", Wiley Information Systems Series, 1996.
- Weblink1: Standish Group Chaos Report, published annually by the Standish Group (www.standishgroup.com), last accessed: April 22, 2014
- Zachman, J.A., "Business Systems Planning and Business Information Control Study",IBM Systems Journal, Vol. 21, No. 1, 1982, pp.31-53.