

2 Case Studies and Examples

- 2.1 INTRODUCTION
- 2.2 SHORT EXAMPLES OF E-BUSINESS STRATEGIES
 - 2.2.1 *Staples Gets into e-business*
 - 2.2.2 *Nestle Corporation*
 - 2.2.3 *Visteon*
 - 2.2.4 *Wired Wellington -- Wiring up New Zealand's Capital*
 - 2.2.5 *Stock Research Group (SRG)*
 - 2.2.6 *Auto-by-Tel*
 - 2.2.7 *Charles Schwab*
 - 2.2.8 *Dell Computers*
 - 2.2.9 *Miller Industries*
 - 2.2.10 *Northern Telecom*
 - 2.2.11 *Synchronized Supply Chains*
 - 2.2.12 *Review Questions*
- 2.3 E-COMMERCE/E-BUSINESS EXAMPLES
 - 2.3.1 *Xerox Document Processing Equipment*
 - 2.3.2 *Amazon.Com Site*
 - 2.3.3 *Amp Inc. Electronic Connectors*
 - 2.3.4 *W.W. Grainger - Facilities Management Supplies*
 - 2.3.5 *Mitre Corporation - The DISCUS Data Interchange System*
 - 2.3.6 *Cisco - Data Communications Equipment*
 - 2.3.7 *Web Security at Exostar*
- 2.4 E-COMMERCE -- ONLINE PURCHASING EXAMPLES
 - 2.4.1 *Getting Sam Started in e-commerce*
 - 2.4.2 *A Simple C2B Purchasing Example -- An Electronic Store Front*
 - 2.4.3 *A Simple B2B Purchase Example*
- 2.5 A FINANCIAL MARKETPLACE
 - 2.5.1 *Overview*
 - 2.5.2 *EMarket Transactions and Transaction Flows*
 - 2.5.3 *An Architectural View*
 - 2.5.4 *Review and Discussion Questions*
- 2.6 CITY OF SEATTLE PUBLIC UTILITIES GO WIRELESS
 - 2.6.1 *Introduction*
 - 2.6.2 *Choosing the Computing Platform and Wireless network*
 - 2.6.3 *Application Architecture*
 - 2.6.4 *Software Approaches*
 - 2.6.5 *Review Questions*
- 2.7 INSURING OUTSOURCED ENTERPRISES IN E-COMMERCE -- A METHODOLOGY
 - 2.7.1 *Overview*
 - 2.7.2 *An Approach to Estimate Risks*
 - 2.7.3 *Review Questions*
- 2.8 AN INTEGRATED MANUFACTURING SYSTEM
 - 2.8.1 *Overview*
 - 2.8.2 *Review and Discussion Questions*
- 2.9 A CUSTOMER RELATIONSHIP MANAGEMENT PORTAL -- A TECHNICAL VIEW
 - 2.9.1 *Overview*
 - 2.9.2 *Usage Scenarios*

- 2.9.3 *Detailed Design -- Solution Architecture*
- 2.9.4 *Review Questions*
- 2.10 MIGRATION OF A PROJECT MANAGEMENT SYSTEM
 - 2.10.1 *Overview*
 - 2.10.2 *Step 1: Business Drivers and Requirement Definition*
 - 2.10.3 *Step 2: Development of a Solution Architecture*
 - 2.10.4 *Step 3: Detailed Design and Implementation*
 - 2.10.5 *Review Questions*
- 2.11 CASE STUDIES AND EXAMPLES OF NETWORKS
 - 2.11.1 *ANX- Example of an Extranet*
 - 2.11.2 *An Enterprise Network Example*
 - 2.11.3 *Example of Public Internet*
- 2.12 OLDER CASE STUDIES -- A FRESH LOOK
 - 2.12.1 *Hewlett Packard's Migration to Client/Server Architecture*
 - 2.12.2 *Off-Shore Reengineering*
 - 2.12.3 *University of Florida Reengineers via the World Wide Web*
 - 2.12.4 *3Com's Data Warehouse*
 - 2.12.5 *British Telecom's Data Warehouse*
- 2.13 REFERENCES

2.1 Introduction

This chapter contains several case studies and examples that illustrate different aspects of distributed systems and e-business. Review and discussion questions are included to help better understand the concepts, and to extend and apply them to other situations. Hints to help with the review questions are provided and sources where needed information can be found are listed (for obvious reasons, most information sources point to different modules of this book!)

This handbook has almost 100 case studies and examples. In addition to the few case studies in this chapter, each module has several pertinent case studies. In particular, each module concludes with a chapter that discusses numerous case studies to illustrate state of the practice in the subject area.

The objective of the few case studies in this chapter is to illustrate the vast material in this book from different perspectives. The topics and examples are chosen so that they complement and, in some cases, reinforce (with some duplication) the XYXCorp case study that is used throughout the book. Some case studies are very short -- a paragraph -- while others are quite detailed with discussion of choices and tradeoffs. An attempt is made to capture the essential aspects of each case study as briefly as possible -- unnecessary and irrelevant information that goes on for 20+ pages is actively avoided (we all have seen enough of those case studies!). The chapter is organized so that the case studies progress from short to detailed. In some cases, the questions raised in the first few case studies are indirectly answered in the later case studies. So a general hint to the reader is to first scan through the entire chapter and then look at review questions.

Most case studies involve multiple issues that span strategies, applications, architectures, integration, networks, middleware and platforms. Some of these case studies appear in specific chapters of the book. They have been extracted and placed in this chapter because analysis of these case studies applies to the whole subject matter and not just to the topic being discussed in the said chapter. Many interesting case studies were published in the mid 1990s when the enterprises needed to grapple with transition from mainframe to then new client/server platforms. A few of those case studies ("Oldies but Goodies") are included here to raise the question: how would you do it today?

The case studies and examples should be viewed within the context of enterprise-wide planning tasks that span business strategies to IT infrastructure issues. These tasks, shown in Figure 2-1, are introduced to

guide the discussion of the XYZCorp case study that is used throughout the book at the end of each chapter (see Chapter 1 of the "Overview Module" for an Introduction to the XYZCorp case study and a discussion of the planning tasks). In other words, you should assume that the case studies are results of one or more of these planning tasks. You should then ask the question: how are these results related to other possible tasks. The review and discussion questions at the end of each chapter are intended to raise these questions.

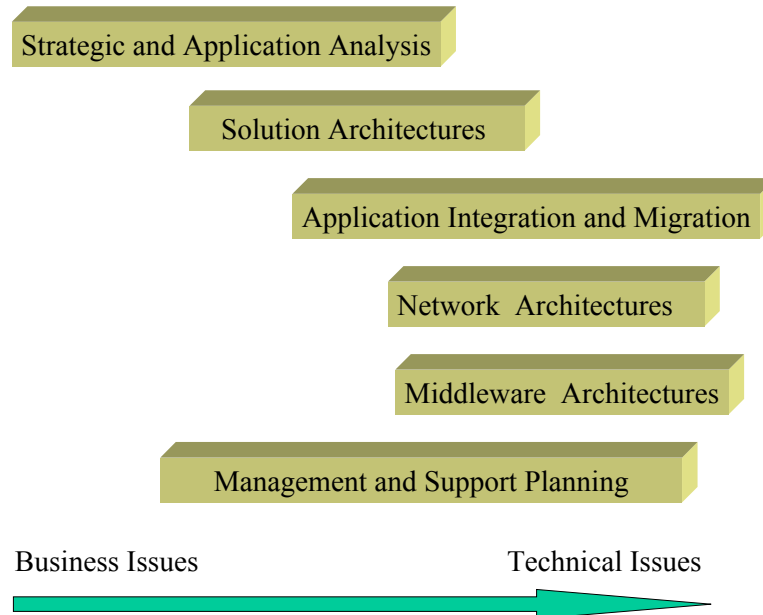


Figure 2-1: Typical Enterprise Planning Tasks

This case study will help us to apply the techniques which we will learn in this book to a realistic enterprise.

A wide range of sources are used to extract these case studies. In addition, many case studies are based on abstractions of selected consulting and work assignments that I have worked on over the past several years. Examples of the sources are (these are also good sources for additional case studies):

Books:

- Severance, D. and Passino, J., "Making I/T Work", Jossey-Bass, 2002¹.
- Kalakota, R. and Robinson, M., "e-Business 2.0", Addison Wesley, 2001.
- Turban, E., et al, "e-Commerce -- A Managerial Perspective", 2nd edition, Prentice Hall, 2002.
- Huff, S., et al, "Cases in Electronic Commerce", Irwin-McGraw Hill, 2000.
- McNurlin, B. and Sprague, H., "Information Systems Management in Practice", Prentice Hall, Fifth Edition, 2002.
- Harmon, P. and Morrissey, W., "The Object Technology Casebook -- Lessons from Award-Winning Business Applications", John Wiley, 1996.
- "E-commerce Case Studies", Telcordia Technologies Handbook, 1999.
- Laudon, K. and Laudon, J., "Management Information Systems", 7th edition, Prentice Hall, 2002.
- Lithicum, L., "Enterprise Application Integration", Addison Wesley, 2000.

Selected Magazines and Web sites:

- CIO magazine (www.cio.com), and CIO-insight magazine (www.cioinsight.com) publish interesting IS management and strategic case studies regularly.

¹ This book is based on the experience of the authors while working on several ERP projects. The book develops a hypothetical case study (General Manufacturing Inc.) based on the actual companies and also has several well known case studies.

- Information Week magazine (www.informationweek.com) publishes general trends and approaches frequently.
- Computerworld magazine (www.computerworld.com) also publishes several real life examples and case studies.
- Forbes Magazine (www.forbes.com) publishes many business oriented case studies.
- <http://www.idea-group.com/cases/> -- many detailed case studies are available through this Web site for a price -- some are oriented towards networks.
- Web site "www.mobileinfo.com" -- many case studies and examples about m-commerce and mobile applications.
- Harvard Business Review -- a classic source for case studies and examples.
- <http://www.idea-group.com/cases/> -- many detailed case studies are available through this web site for a price.
- Aberdeen Group (www.aberdeen.com) has many relevant case studies -- some are free.
- www.microsoft.com/net/use/casestudies.asp -- Microsoft site shows several case studies of XML. Web services. Some are interesting business and application examples.
- IBM's web site (www.ibm.com) also publishes several relevant case studies.
- www.redherring.com/discussions/ -- a good site for Red Herring magazine. Publishes case studies.
- www.knowledgestorm.com -- a good site for numerous examples.
- Magazine "eAI Journal" (www.eaijournal.com) is a very good source of information on integration examples and discussion of integration technologies.

2.2 Short Examples of e-Business Strategies

Here are a few highly abbreviated examples of how some organizations have established strategies and used IT to survive and thrive.

2.2.1 Staples Gets into e-business

Staples is getting out of the business of paperwork. According to an August 26, 2002, Wall Street Journal advertisement, 3.3 million corporate end-users benefit from realtime order information on over 80,000 items. The items are available at staples.com. Staples uses IBM's WebSphere to display on-hand inventory instantly and handle customer orders.

2.2.2 Nestle Corporation

Nestle, an international food and pharmaceuticals company operating in more than 70 countries, decided to standardize its business processes to compete in e-business. The factories in different countries did business according to local rules and culture. But this did not allow the company to use its worldwide buying power for commonly used raw materials. The company introduced a single Enterprise Resource Planning (ERP) system to streamline its material requirement and planning systems and significantly reduced operational costs.

2.2.3 Visteon

Visteon, a spin-off of the Ford Motor Company, manufactures auto parts. To compete in the fiercely competitive auto parts market, Visteon embarked on a streamlining initiative. The goal of the initiative is to cut product development time from 30 months to 10 months, and to reduce manufacturing time from 5 days to 1 day. At the core of this initiative is an integrated supply chain management system that links suppliers, designers, and production planners in a uniform manner.

2.2.4 Wired Wellington – Wiring up New Zealand's Capital

The city of Wellington, New Zealand -- the country's capital -- developed Vision 2020, a vision for the city to be a "wired city" by the year 2020. The key idea behind this project is to string fiber-optic cables throughout the city's core business section. This will inter-link all the downtown buildings and allow the businesses to develop the cables for whatever purpose they wish. The Vision does not specify who will make use of the network and how. The city council only encourages the businesses to increase use of Internet for the advancement of its status and business activity. This is an interesting example of e-government.

2.2.5 Stock Research Group (SRG)

SRG is a financial services information broker that serves as a collection point for information needed by investors. SRG, in effect, pulls potential investors to its sites and then channels them to the appropriate companies for investment. In other words, SRG is an "infomediary" in the investment business. To accomplish this, SRG has developed an IT infrastructure that serves the customers and then also connects the customers to the target companies. Most target companies are smaller mining companies. SRG also develops Web sites for the mining companies -- this gives them an opportunity to build uniform user interfaces so that the customers do not see fundamentally different Web sites as they go from one company to the next.

2.2.6 Auto-by-Tel

Web-based auto sales companies such as Auto-by-Tel started with the strategy that the auto prices for the same car vary widely between dealers and that potential customers cannot visit all auto dealers to find the best deal. Auto dealers bought into this idea because it increased their sales channels. The large auto manufacturers are also responding to the Web-based auto sales as a "virtual dealerships" that can sell cars. This is a good example of a solution that addressed an existing problem.

2.2.7 Charles Schwab

Charles Schwab made customer focus the core of its business strategy to stay competitive. This required that the Schwab brokers have complete and recent information about customers as well as investment opportunities (i.e., minimize "I will call you back with more information"). To achieve this, Schwab linked all of its sales and customer service organizations with one another and with all the customer-interfacing parts of the company. A new information infrastructure was needed to capture and integrate information about customers and their investment patterns.

2.2.8 Dell Computers

Dell computers is a well advertised case study. The main question is: how did Mike Dell who was repairing and selling used PCs get to the point where he sells more computers than IBM. The answer is a combination of strategic moves. The first is devotion to sending a pre-configured system to the customers -- this appeals to the users because they can just turn on the machine without having to go through a lengthy installation and configuration process. The second is to use Internet very effectively as a sales and distribution channel -- Dell sells directly to the customers thus bypassing the intermediaries. In addition, Dell uses a very effective outsourcing and virtual integration strategy that allows Dell to have nearly just-in-time assembly of computers -- different components are produced by different suppliers and assembled when a customer buys a computer. Dell has also deployed its Premier Page Program to thousands of corporate customers. The program allows Dell to enhance its successful direct-sales practices by leveraging the Web to offer a hybrid solution combining enhanced order management capabilities (product configuration, personalized pricing, order status, and shipment tracking) with customer relationship functionality (contact information, document repositories, realtime access to customer/technical assistance). The returns are quite impressive. Dell reports that nearly 40% of its daily online revenues come from the Premier Page Program. Further, customers report higher satisfaction levels and intentions to sole-source Dell PCs (see the Dell Case Study in Chapter 1 of this module).

2.2.9 Miller Industries

Getting ready for the year 2000 was a prime driver behind a Baan ERP implementation by Miller Industries Inc., a manufacturer of towing and recovery equipment in Chattanooga, Tennessee. The company met its January 1999 deadline for the project, but reported a \$3.5 million operating loss for the fourth quarter of 1999. It attributes the loss in part to costs and inefficiencies resulting from the new system.

CIO Lance St. Clair says the unexpected costs resulted from hiring additional staff to support a change in the production process imposed by the Baan software, and outsourcing part of its business in order to focus on Baan. "It takes a while to get everyone used to doing things a new way," he says. The company has gone back to its own systems for ERP in the towing and recovery equipment business.

2.2.10 Northern Telecom

Northern Telecom in Brampton, Ontario, developed a comprehensive Invoice Management (CIM) system that won the Computerworld award in the "Best Use of Object Technology To Integrate Legacy Systems" category. CIM uses the Northern Telecom Customer Order Database (CODB) which consolidates all separate and distinct legacy systems data into one common database. From this database, CIM extracts data for the processing of orders and updates CODB where needed. The application re-uses objects for its internal operations and hides many of the legacy systems interfaces by using object wrappers.

2.2.11 Synchronized Supply Chains

Many interesting examples of legacy system access and integration are appearing in manufacturing organizations. These organizations rely on a chain of consumer-supplier for their items. For example, an auto manufacturer has to rely on several suppliers who in turn rely on their suppliers. This "supply chain" needs to be synchronized to shorten the cost and the time involved in the supply chain. Technically, this involves interfaces and integration of numerous legacy systems that exist in the supply chain. Many of these systems are being integrated by using Web technologies.

2.2.12 Review Questions

For each short case study discussed above, do the following (make assumptions where needed) :

- List the type of applications that will be needed to support the strategies. Show how these applications need to be interlinked (i.e., build a logical application architecture).
- Draw a physical architecture that shows the IT infrastructure needed to support the applications.

Hints: These questions assume that you know commonly used e-business applications in organizations. For a discussion of applications, see the "Applications" Module. Many examples of physical architectures can be found in the "Architecture" Module (naturally!). You may want to consult the original sources (e.g., [Kalakota 2000]) for additional information.

2.3 e-Commerce/e-Business Examples²

The following examples, largely based on [Telcordia 1999, Huff 2000, Harmon 1996], are somewhat more detailed and show a high level architectural solution.

2.3.1 Xerox Document Processing Equipment

Xerox is the world's largest retailer of cut sheet paper, toner, document processing systems, and services. It also leases large copy machines. Xerox's products are sold directly through its own sales organization or through resellers.

² Some of these examples were prepared by my colleague and friend Paolo Missier.

The main issue is that pricing of products is usually uniform, but some accounts have special prices through pre-negotiated contracts. Therefore, some authorized customers are shown special prices, different from published prices.

Figure 2-2 shows the overall architecture of the Xerox System. To avoid conflict with resellers, customers browsing products sold by resellers are automatically guided to a choice of resellers. Some customers prefer to browse products online, but complete transactions via phone, therefore a call center with a toll-free number is provided. Available selection is huge with a large variety of choices. For the sake of simplicity, customers are shown a limited selection of the most popular choices.

Product search is supported in the following ways:

- When the customer knows the name of the product, he can directly get the detailed description and pricing.
- The customer can browse through product categories that show product listings.
- The customer is asked questions about requirements, then products are recommended.
- The site supports mostly business customers, therefore, it cuts down on fancy, unnecessary graphics.

Xerox site architecture is based on IBM's Commerce.Net, which can create dynamic catalogs, extract customer and product information dynamically, and support different prices for products. Commerce.Net also provides security mechanism.

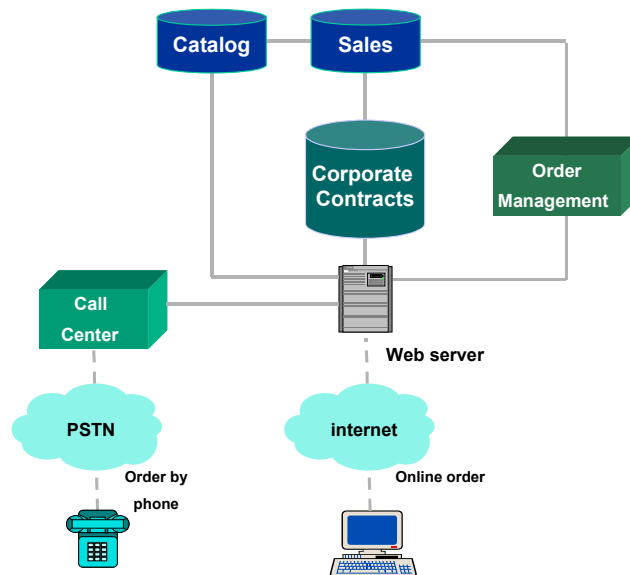


Figure 2-2: Xerox Online Order Processing System

Review and Discussion Questions

- Showing different prices to different customers happens in many business situations. Can the approach developed by Xerox be used to show a dozen or more prices for different customers? Be specific.
- How can this architecture be extended to include dealing with Xerox partners?
- Show the logical flow of the order processing system through a diagram (e.g., a flowchart).
- Survey the COTS products that can be used to support this architecture. Will you replace CommerceNet with another platform? Why or why not?
- Develop a network configuration to support this architecture.

Hint: Review the discussion of online purchasing examples discussed in Section 2.4 before answering these questions. Also, the chapters in the "Platforms" Module and the "Network" Module should be consulted. For a few sample network configurations, see Section 2.11.

2.3.2 Amazon.Com Site

Amazon.com is a big success story in consumer to business (C2B) E-commerce. The overall model of Amazon.com is shown in Figure 2-3. The basic idea is that Amazon.com does not maintain large physical inventories. It basically exercises virtual enterprise by using the following: :

- Several bookstore and publisher partners
- When a book is ordered, the site closest to the customer is searched for available items
- Available items are shipped from the closest site to the customer
- Complex B-To-B agreements for service levels

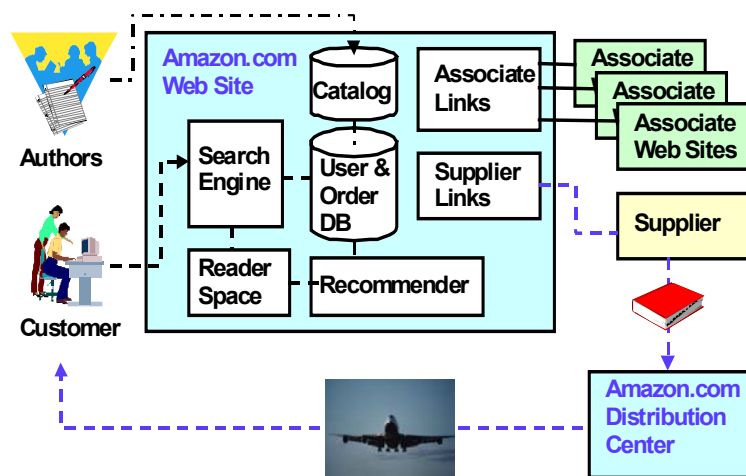


Figure 2-3: Amazon.com Model

Review Questions

- Expand this case study by using the great deal of information that has been published about Amazon.com (one possible source is [Kalakota 2000]).
- Show a detailed scenario that describes complete flow of an order from customer to shipping and receiving.
- Using the high level model shown in Figure 2-3, cast it into a detailed implementation model that shows the COTS components and detailed IT infrastructure.
- Describe the format of messages in XML that are exchanged between Amazon.com and the business partners.

Hint: Review the discussion of online purchasing examples in Section 2.4.

2.3.3 Amp Inc. Electronic Connectors

Amp Inc. is one of the world's largest manufacturers and suppliers of electrical and electronic connectors. Most of its products are sold directly, but it also uses distributors.

The company has a huge product base, around 145,000 SKUs. These change continuously (around 200 changes per day). Paper catalogs constituted the company's primary means of contact with customers. The company's Web site, shown in Figure 2-4, was originally developed to replace the paper catalog.

The online catalog provides product information, specifications, as well as high-quality graphics of products. COTS search engines were not suitable for searching Amp's product catalog, therefore the company decided to develop a custom search engine called "Step Search" to enable searching by function rather than keyword.

Since Amp has a large customer base outside the US, the catalog was originally designed to support multiple languages without having to duplicate the catalog. The catalog is currently available in seven languages. The catalog dynamically downloads product information in the language selected by the user. Amp used Lucent Technologies' software for translation. This ground-up approach to globalization saved the company a lot of trouble that other companies face with globalization efforts. The catalog is implemented using the Oracle 7 DBMS.

Amp has also added online ordering capabilities to its site. This functionality is available for registered users. Large customers can use EDI for their orders, while small customers use a shopping cart. Personalized user profiles have been recently added for guests.

In addition to viewing product specifications, non-registered users can query the system for price and availability.

Amp's Website also provides 3D models for some 6000 products. These models can be downloaded by design engineers to incorporate into their designs. Each downloaded model is estimated to save engineers 2 days of design work.

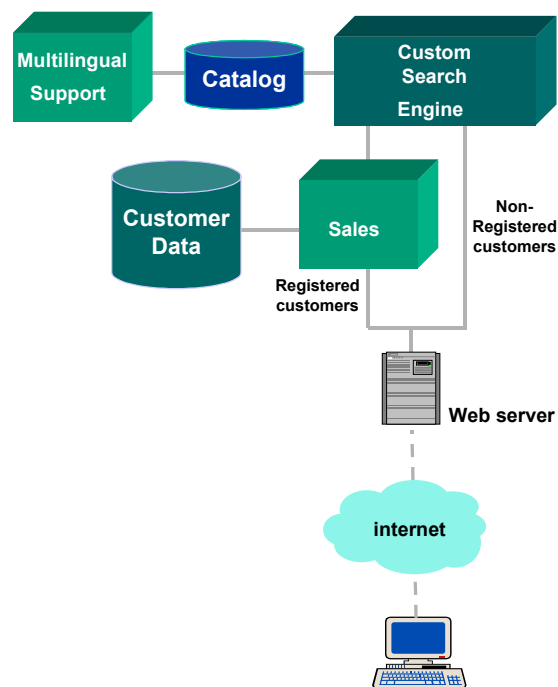


Figure 2-4: Amp Inc Architecture

Review Questions:

- How can you include security in this system? What will be the different levels of security? Discuss the flow of security in this system.
- How can XML be used in this system? Give details.
- Survey, evaluate and pick a COTS platform that can support this architecture.
- Translate Figure 2-4 into a physical architecture ("solution architecture") that shows the middleware and network. You should also map existing COTS technologies to this architecture.
- How will you support mobile users for this system?

- Develop a detailed network design for this system. Show the routers, gateways, firewalls, wide and local area networks, Internet and Intranet, etc. (Optional Question)

Hints: Most of these questions can be answered by reviewing the chapters in the "Platforms" Module. The "Architecture" Module and the "Network" Module provide a great deal of information on the detailed physical architecture.

2.3.4 W.W. Grainger – Facilities Management Supplies

W.W. Grainger Inc. is a leading distributor of maintenance, repair and operations (MRO) supplies in North America. With 220,000 repair and replacement parts from 550 suppliers, Grainger's business depends on service reputation and supplier partnership.

Grainger's business is characterized by high-volume and low-value orders. The cost of paper-based orders processing is, therefore, high.

Business is repeated with the same customers. 95% of purchasing transactions are completed using open accounts. The Grainger system, shown in Figure 2-5, chose to implement a separate server for financial transactions using the Secure Electronic Transaction (SET) standard (www.setco.org). The site uses Open Market's AM Transact for transaction processing.

Grainger supplies 78,000 brand name products and 220,000 repair and replacement parts. Grainger's site is backed by comprehensive electronic databases which enable product search by description, brand name, manufacturer's model number and Grainger stock number. The catalog's pages include description, specifications, and drawings of products.

Registered users can search for products in different ways and complete orders electronically. Customers can also view product availability and special pricing.

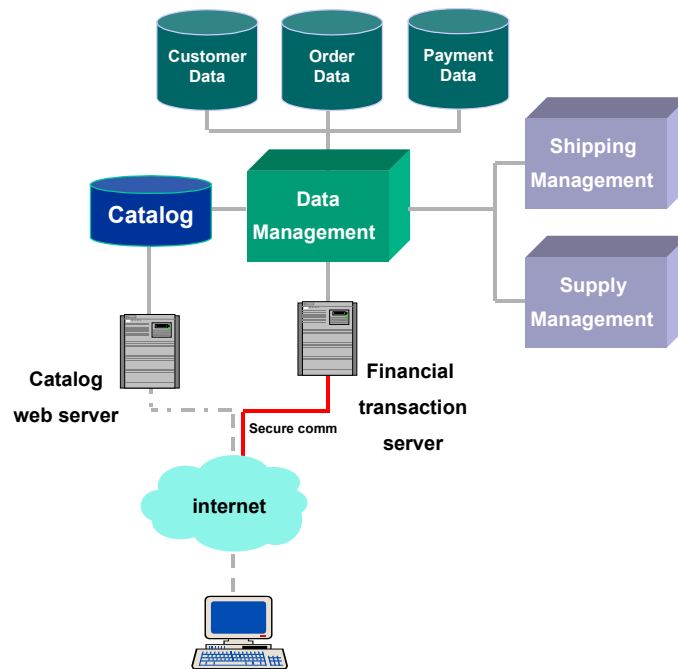


Figure 2-5: W.W. Grainger Architecture

Review Questions:

- How will the catalog be populated? Survey and choose a COTS catalog product that can satisfy the Grainger requirements.

- Can a supply chain management system be used in this system? What are the business advantages of using an off-the-shelf SCM system? What type of technologies and standards will you use in this case?
- Will you use EDI or XML in this project? Where and how? What XML standards are relevant in this area (Biztalk, Rosettanet)?
- How can you include security beyond SET in this system?
- Survey, evaluate and pick a platform that can support this system.
- Translate Figure 2-5 into a physical architecture ("solution architecture") that shows the middleware and network. You should also map existing COTS technologies to this architecture.
- How will you support mobile users for this system?

Hints: Most of these questions can be answered by reviewing the chapters in the "Platforms" Module. The "Architecture" Module provides a great deal of information on the detailed physical architecture.

2.3.5 Mitre Corporation – The DISCUS Data Interchange System

Mitre Corporation developed a Data Interchange and Synergistic Collateral Usage System (DISCUS) that integrates multiple legacy application systems. The conceptual architecture of DISCUS, shown in Figure 2-6, is based on an object request broker that receives requests from multiple applications and accesses various databases. The information from one database is converted to another by using translators that are built as part of the project. The end users of this system are primarily information scientists who require access to multiple forms of information: text, images, maps, and relational data. This information needs to be accessed by a variety of legacy applications that include Map front-ends, image processing, and other tools/applications. The system uses CORBA (Common Object Request Broker Architecture) for accessing remote data.

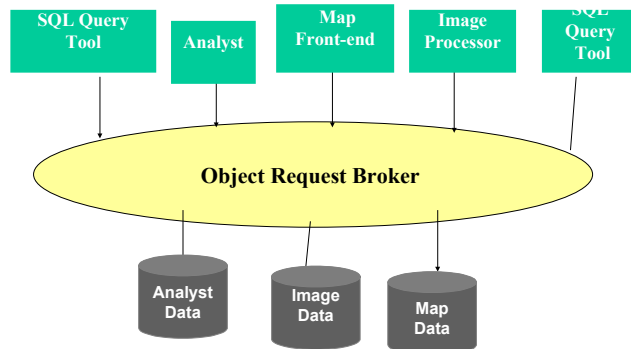


Figure 2-6: DISCUS Data Interchange System

Review Questions.

This system was developed by using CORBA. How would you use the EAI (Enterprise Application Integration) platforms to achieve the same goals? What are the tradeoffs?

Hint: The "Integration" Module extensively discusses various integration choices and describes the EAI platforms.

2.3.6 Cisco – Data Communications Equipment

Cisco is the supplier of 80% of the internet's backbone equipment -- LAN and WAN switches, routers, network management tools, etc. Cisco estimates that its market share is number 1 or number 2 in every market segment that it participates in.

Cisco calls its E-commerce site "The Cisco Connection Online (CCO)". CCO allows users to browse the product catalog and to access standard pricing. Customers can also place orders online, track shipments and access maintenance and warranty information.

An important part of Cisco's business is the maintenance and network design services. In order to help customers design their own network solutions and to reduce the time customers had to wait for answers to technical problems, Cisco's CCO site includes two unique features:

- A Network Configuration Tool: This tool is based on a constraint-based configuration engine from Calico. The network configuration tool allows users to configure their own networks, to validate them and price them. The customers' benefit from this tool is that they are in charge of their purchasing decisions.
- A customer user group: This user group is a Q&A forum where customers can place questions about their technical problems. Answers are provided either by other customers or by the Cisco technical support engineers. Before placing their question, customers can query the Q&A database to see if a similar problem has been answered before. Customers can receive notifications when answers to their problems are posted. Customer surveys show that customers are satisfied with the hybrid self-help/technical support system since it leverages the expertise of both other customers and Cisco's technical support. The system also reduces the time a customer needs to wait in order to get an answer to his technical problem.

Another important feature of Cisco's E-commerce site is the flexible integration with large customers, partners, as well as suppliers through its Global Networked Business (GBN) model (see <http://www.cisco.com/warp/public/756/gnb/>).

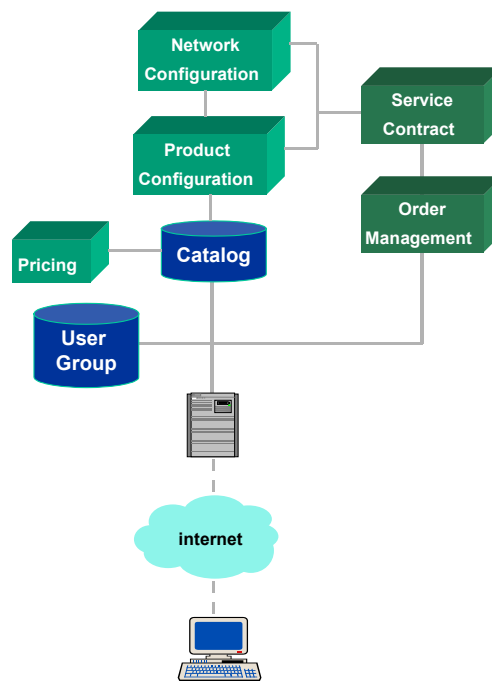


Figure 2-7: Cisco Architecture

Review Questions:

- Can emarkets be used in this system? What are the business advantages of using emarkets in this system? What type of technologies and standards will you use in this case?
- Will you use EDI or XML in this project? Where and how? What XML standards are relevant in this area (Biztalk, Rosettanet)?

- How can you include security in this system? What will be the different levels of security? Discuss the flow of security in this system.
- Survey, evaluate and pick a platform that can support this system.
- Translate Figure 2-7 into a physical architecture ("solution architecture") that shows the middleware and network. You should also map existing COTS technologies to this architecture.
- Develop a detailed network design for this system. Show the routers, gateways, firewalls, wide and local area networks, Internet and Intranet, etc.
- How will you support mobile users for this system?

Hints: Most of these questions can be answered by reviewing the chapters in the "Platforms' Module. The "Architecture" Module and the "Network" Module provide a great deal of information on the detailed physical architecture.

2.3.7 Web Security at Exostar

Members of the defense business are obviously concerned about security breaches. In August 2002, a 23-year old hacker nicknamed RaFa broke into a highly secure NASA database, and stole 43 megabytes of sensitive design data about planned space vehicles. The documents were created during a collaborative effort by Boeing, Aerojet and Pratt & Whitney.

So when companies such as Boeing, Rolls Royce, Lockheed Martin, BAE Systems and Raytheon invested \$100 million to create online aerospace marketplace Exostar in 2000, they chose to implement 87 security requirements. This included data encryption not only while data is in transit but also while it is stored in the database. Exostar also maintains detailed background information on each user and a 12-month record of every file being accessed, what changes were made, by whom and allows no one person or company (even at Exostar) to have complete access to all the data. Many vendors collaboratively built the Exostar security system.

Rolls Royce put Exostar to the test using Exostar's electronic collaboration services. By using these services, its engineers could securely share CAD patterns and project management systems with other design engineers at Fiat-Avio, Goodrich Corporation, Honeywell, Volvo, and others. At the start of the project, Rolls Royce appointed a project manager to initiate the process. The manager logged onto the system to start a session. Then Verisign verified the project manager's identity and authority to work on the project. Verisign gave the project manager a password and a digital certificate. The certificate resided on the manager's computer so it was only possible to access the system from that computer. The manager then invited others to join the project and specified the level of access to which each user was entitled. After initial processing, the partners from different sites were ready to share information and worked on the same document using their personal digital certificates for verification, the file itself was encrypted with a 128-bit key. After the session ended, the file was then sent to Exostar's data center, for high levels of physical and network security. Whenever another project member wanted to access the file for revisions, it was encrypted again before traveling over the Internet to his computer, where it remained encrypted until the engineer with the authorized key opened it.

This security solution is not an overkill if you are in the defense business. Exostar has over 11,000 members who think that the extra security is worth it. It is estimated that by collaborating over the Web, Rolls Royce saved as much as 60% on its travel budget, reduced project management errors by up to 50% and cut the product development cycle time by up to 40%. In addition, so far, no break-ins have been reported.

Source: Niall McKay, "Case Studies: Digital Do-Overs -- Web Security: Xtreme Exostar", Forbes, 7 October, 2002

Review Questions:

- In your view, is this security an overkill? Explain your answer.
- What is the main piece of sensitive information that needs to be protected? Why should it be protected?
- Are there some security issues that are not being addressed in this case study?

Hints: Most of these questions can be answered by reviewing chapters 3 and 4 in the "Management" Module.

2.4 e-Commerce – Online Purchasing Examples

These examples are based on some real life online purchasing systems. They have been extracted from the "Platforms" Module for broader discussion.

2.4.1 Getting Sam Started in e-commerce

Sam owns a small shoe shop in a suburb of Chicago and wants to sell his shoes over the Internet. What should he do? Here are the key steps.

Step 1: Set up a Web site

Sam can build his own Web site or he can outsource to Web hosters. In many cases, an ISP can provide you with a free Web site with adequate disk space to get started. A very extensive list of ISPs and Web hosters can be found at the Website (www.Webhostdir.com).

Step 2: Get a domain name

Sam will have to get a domain name to be in business. A Web hoster can get Sam a domain name or Sam can do it himself. If Sam chooses to do it himself, the Web site (www.allwhois.com) is very useful -- it allows you to search for a desired name and gives you contact information if the name you want is taken. After finding a suitable domain name, Sam will need to purchase a domain name. The Web site (www.networksolutions.com) can be used to purchase a domain name -- it roughly costs \$35 per year.

Step 3. Get shopping cart software

Almost all e-commerce purchasing systems use shopping cart software. Sam can build his own shopping cart or outsource his shopping cart processing. The Website (e-commerce.about.com) has very useful information in this area.

Step 4. Set up a payment system (how does Sam get paid?)

If a customer buys something from Sam over the Internet, Sam may want to be paid (a very safe assumption). Payment systems in many e-commerce sites take credit cards. Once again, Sam can develop his own credit card processing system (why?), buy and install a credit card processing system, or connect to an ASP that specializes in credit card processing. Cybercash (www.cybercash.com) is an example of a company that provides a variety of credit card processing software as well as hosting services for credit card processing.

But how does Sam get the money? Most credit card processors need a merchant account to transfer money into. Merchant accounts can be set up by banks or Sam can use companies such as Cybercash or authorize.net for setting up merchant accounts.

In addition to credit cards, payment systems do use other means such as electronic fund transfer, letters of credit, loans, and purchase orders, etc. We will discuss these payment systems later.

Step 5: Set up a Delivery System

Sam will also have to make arrangements to have the shoes delivered to the customers. He can outsource this also to carriers such as Federal Express or UPS.

Review Questions

- Should Sam outsource the entire process, part of it or none? What factors will influence his decisions? Develop a decision chart. Also, specifically suggest a few ASPs for Sam.
- If Sam does choose to do it himself, should he custom build the on-line purchasing system or consider buying a commercial-off-the-shelf e-commerce platform? If so which one?

Hints:

- The discussion about outsourcing in Chapter 2 of the "Applications" Module can be converted into a decision model. As suggested previously, the Web site (www.Webhostdir.com) has a very large collection of possible service providers for e-commerce.
- Many e-commerce platforms (e.g., IBM Websphere, Oracle E-commerce Platform, Microsoft EC Site Server) are available commercially and should be considered because these platforms package many facilities on the same computing platform. These platforms are discussed in the chapter "EC Platforms" of the "Platforms" Module.

2.4.2 A Simple C2B Purchasing Example – An Electronic Store Front

Let us consider a simple example of a company that wants to establish an electronic store front, i.e., allow customers to buy the products from the company over the Internet. The following discussion shows the usage scenarios and elaborates the principal activities of Ipurchase, a system developed to support the storefront. Figure 2-8 shows a simplified view of Ipurchase with various Web technologies such as HTTP, HTML, etc. The usage scenarios are presented from three different perspectives: customer usage, business administrator usage (e.g., purchasing department), and product administrator usage (e.g., IT Group).

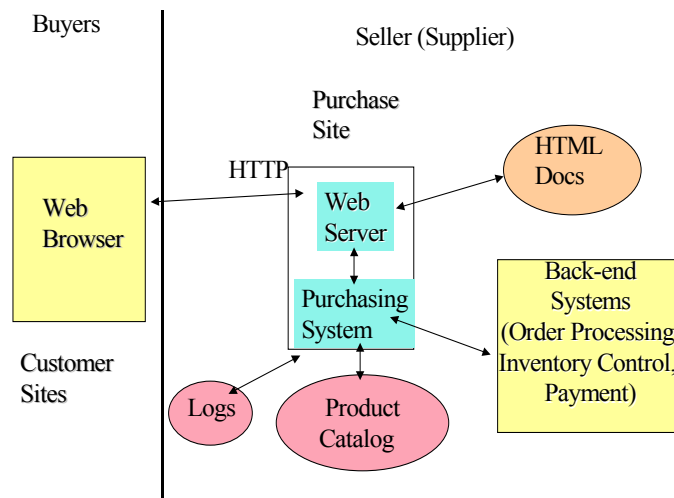


Figure 2-8: A Simple Internet-based Purchasing System

2.4.2.1 Customer Processing

i) Initial Processing

- The customer gets on the corporate Home Page.
- IPurchase Greeting screen shows and walks the customer through various informational and marketing pages.

ii) Search and Browse Catalog and Select Items

- A product review and selection screen is shown to the user as a default. Other views are also possible (e.g., vendor view, manufacturer view).
- The customer browses and/or searches through the catalog based on product attributes (e.g., price, name, manufacturer, etc.), synonyms, and full text searches (e.g., “find me a laptop”).
- If an item is not available (as indicated in the catalog), the customer can choose to terminate the session or browse for other items.
- The customer selects the product(s) to purchase.
- A “shopping cart” is populated with the items selected by the customer.
- The customer verifies the items to be purchased and clicks on “purchase”.
- The system asks for customer ID or payment information such as credit card number.
- The validation process is triggered to verify the payment information.
- The customer is notified whether to proceed (Catalog Review and Selection) or not (appropriate errors/guidance messages on how to correct the errors).
- The purchase information is also validated for exceeding the purchase limit. The employee is given help in making corrections (shuffling the shopping cart), and resubmitting.

iii) Order Generation

- An order entry is created with a control number.
- The order log has information that can also be used for General Ledger (i.e., customer ID, name, items ordered, total quantity, etc.).
- The order is logged in the log database.
- The order is sent to the order processing system.
- The user is given an end of order screen and is given the control number.
- An email may also be sent to the user documenting the order.

iv) Order Processing

- The order processing system receives the order and initiates its order processing.
- The supplier sends the needed items to the customer through its shipping department.
- If for some reason, the order cannot be full-filled, then the system notifies the employee through email or phone call.

2.4.2.2 Business Administrator Usage Scenarios

The business administrator (e.g., purchasing department) will be involved in the following steps to reconcile the orders and to print various reports. These operations will require password protection.

v) Reconciliation and customer support

- The order log database is reconciled with the shipment data.
- Error reports are generated for mismatches.
- Purchasing and shipment resolve the errors.
- Customer issues are resolved by the customer support department.

vi) Payment processing

- The payment is handled mainly through credit cards.
- High priced items are handled through purchase orders, invoices and accounts payable (i.e., accounts payable processes the invoices, and invokes other back-end processing such as general ledger).

viii) Management reporting

- Generate and send reports to managers periodically (who ordered what in their department).
- Allow report generation on an ad hoc basis by the managers
- Generate reports on system activities (how many people logged on, how many actually purchased something, etc.).

2.4.2.3 Product Administrator (“Web Master”) Usage Scenarios

The administrators perform the following functions (after the system has been developed and installed):

- Purchase system installation and configuration procedures
- Design and population of the catalog
- Replenishment of the catalog periodically (e.g., daily)
- System performance monitoring, tuning, backup/recovery, etc.

Review Questions:

- Give more details about how the payment system will work through a credit card (i.e., show how a merchant bank and a credit card processing agency will participate in this system).
- Discuss how the PO processing will take place. Show the key players, their roles and the flow between the players.
- How can you include security in this system? What will be the different levels of security? Discuss the flow of security in this system.
- How can XML be used in this system? Give details.
- Survey, evaluate and pick an e-commerce platform that can support this storefront.
- What will you need to do to convert this storefront into a virtual storefront (i.e., the customer can choose items from multiple suppliers)?
- Translate Figure 2-8 into a physical architecture ("solution architecture") that shows the middleware and network. You should also map existing COTS technologies to this architecture.
- How will you support mobile users for this system?
- Develop a detailed network design for this system. Show the routers, gateways, firewalls, wide and local area networks, Internet and Intranet, etc.

Hints: Most of these questions can be answered by reviewing the "E-commerce Platforms for C2B" Chapter in the "Platforms' Module. The "Architecture" Module and the "Network" Module provide a great deal of information on the detailed physical architecture.

2.4.3 A Simple B2B Purchase Example

For sake of discussion, let us now introduce a simple B2B purchasing system, Bpurchase, that differs from the C2B Ipurchase system in two respects: a) It involves a B2B relationship between a buyer corporation and multiple suppliers, and b) the buyers are employees of the buying corporation with proper authorization for purchasing.

Bpurchase was developed for ordering low-cost, fast delivery (within 24 hours) items with a maximum order of \$1,000 from multiple suppliers. We assume that less than a dozen suppliers (vendors) participate in the system handling around 40,000 orders per year. The information from the vendors is maintained in a Master Product Catalog that is loaded and updated periodically to reflect the latest vendor products. Bpurchase replaces a current corporate purchase system for employees that is primarily phone-based. In the phone-based system, the employee calls the vendor, the vendor asks the employee a few questions (e.g., items needed, employee ID, project number, etc.), supplies a control number to the employee, ships the requested items to the employee, and then sends an invoice to the company.

The purpose of Bpurchase is to minimize operational costs and improve customer satisfaction by automating the phone-based purchase system. The basic idea is to allow the employees to search and select the items to be purchased and create an electronic "purchase cart". From the purchase cart, an automatic order is generated that is sent to the vendors for purchasing.

The main B2B consideration is that an “open purchase order (PO)” agreement exists between the buying organization and the sellers, i.e., the sellers (vendors) send a monthly invoice to the buyer that shows all the

purchases made that month. Open PO is a convenient way of buying low cost items such as office supplies (a PO does not have to be issued for each pencil).

The following discussion shows the same usage scenarios as discussed for the Ipurchase system. The usage scenarios are presented, as before, from three different perspectives: employee usage, business administrator usage (e.g., purchasing department), and product administrator usage (e.g., IT Group).

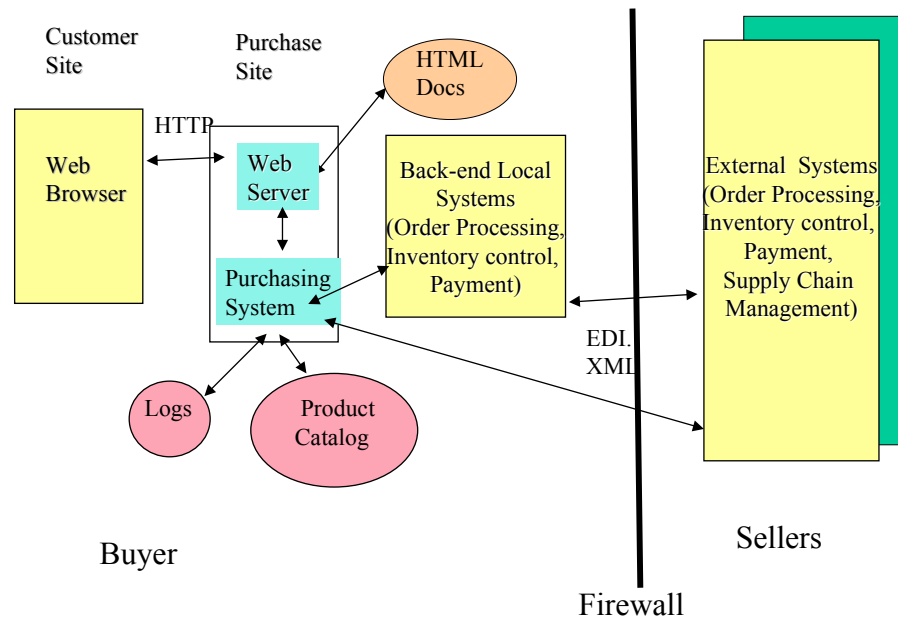


Figure 2-9: Conceptual View of B2B Purchasing

2.4.3.1 Employee Usage Scenarios

The employees will use the following steps to order supplies:

i) Initial Processing

- The employee gets on the corporateHome Page.
- The BPurchase Greeting screen is shown to the user and asks for employee ID, project ID, etc. Passwords are not required to use the system.
- The validation process is triggered to verify employee and project information.
- The employee is notified whether to proceed (Catalog Review and Selection) or not (appropriate errors/guidance messages on how to correct the login errors).

ii) Search and Browse Master Product Catalog and Select Items

- The user initiates a search and browse of the Master Product Catalog by clicking on an icon.
- A product review and selection screen is shown to the user as a default. Other views are also possible (e.g., vendor view, manufacturer view).
- The employee browses and/or searches through the catalog based on product attributes (e.g., price, name, manufacturer, etc.), synonyms, and full text searches (e.g., “find me adhesive tapes”).
- If an item is not available (as indicated in the catalog), then the employee can optionally generate an email to the vendor asking about item availability. There is no back-order processing in this system. The user can search for the needed item from another vendor catalog or choose to terminate the session (i.e., call vendor off-line).
- The employee selects the product(s) to purchase.
- A “shopping cart” is populated with the items selected by the employee.
- The employee verifies the items to be purchased and clicks on “purchase”.

- The purchase information is validated for exceeding the IPurchase limit, project information, budget, etc. The employee is given help in making corrections (shuffling the shopping cart), and resubmitting.

iii) Order Generation

- An order entry is created with a control number.
- The order log has information that can also be used for general ledger (i.e., employee ID, name, project, task, items ordered, total quantity, vendor ordered from, etc.).
- The order is logged in the log database.
- The order is sent to the vendor through Email (other options for interfacing IPurchase with vendor systems are being investigated).
- The user is given an end of order screen and is given the control number.
- An email is sent to the user documenting the order.

iv) Vendor Processing

- The vendor receives the Email and initiates its order processing.
- The vendor sends the needed items to the employee.
- If for some reason, the vendor cannot fill the order, then the vendor notifies the employee through email or phone call.
- The invoices with additional details (e.g., what was purchased, who purchased it, etc.) are sent by the vendor to purchasing once a month. At present this information is sent on a floppy diskette.

2.4.3.2 Business Administrator Usage Scenarios

The business administrator (e.g., purchasing department) will be involved in the following steps to reconcile the orders and to print various reports. These operations will require password protection.

v) Reconciliation

- The order log database is reconciled with the vendor provided invoices.
- Error reports are generated for mismatches.
- Purchasing and vendors resolve the errors.

vi) Invoice processing

- The approved invoices are sent to accounts payable.
- Accounts payable processes the invoices.
- Other back-end processing (e.g., general ledger).

vii) Payments

- Accounts payable sends the checks to the vendors (once a month).

viii) Management reporting

- Generate and send reports to managers periodically (who ordered what in their department).
- Send an email to employees indicating what they have ordered and how much the company was billed for it (this will be used to verify that the employee has actually received the items).
- Allow report generation on an ad hoc basis by the managers.
- Generate reports on system activities (how many people used it, how many through Web, etc.).

The business administrators also initialize and maintain the IPurchase system (e.g., define vendor information such as email and contact, define and modify validation rules, add/delete vendors, restrict vendors to certain products, etc.).

2.4.3.3 Product Administrator Usage Scenarios

The product administrators perform the following:

- System installation and configuration procedures

- Population of the Master Product Catalog
- Replenishment of the consolidated catalog periodically (e.g., daily)
- System performance monitoring, tuning, backup/recovery, etc.

Finally, notice that, because users are authenticated against existing databases, we can eliminate the need to build and maintain a new system just to authenticate remote access users. This existing database can be Unix, NetWare, or Windows NT based.

Review Questions:

- How will the Master Product Catalog be populated? Discuss the logistics involved in getting the information from the vendors and loading it into the catalog? How frequently should this catalog be updated? How can you make this process completely automated (i.e., extract information from vendor catalogs and load it into the product catalog)? Discuss the detailed design of a catalog loader with exact technologies that you will use (e.g., middleware services such as FTP, email, XML, HTTP, etc.).
- Will you consider a virtual catalog (i.e., the information is maintained in the vendor catalogs and not stored in a single product catalog) for this project? How exactly will this work? What is the major challenge in building virtual catalogs? Show a detailed design of a system that will support a virtual catalog.
- The company is thinking of making this system available to external customers. What changes will have to be made to the system? Identify at least five changes that should be considered? For each change, discuss the technical as well as the management aspects.
- The system currently uses a PO system. How can payment from a credit card be introduced in this system?
- Can emarkets be used in this system? What are the business advantages of using emarkets in this system? What type of technologies and standards will you use in this case?
- Will you use EDI or XML in this project? Where and how? What XML standards are relevant in this area (Biztalk, Rosettanet)?
- How can you include security in this system? What will be the different levels of security? Discuss the flow of security in this system.
- Survey, evaluate and pick a B2B platform that can support this B2B system.
- Translate Figure 2-9 into a physical architecture ("solution architecture") that shows the middleware and network. You should also map existing COTS technologies to this architecture.
- Develop a detailed network design for this system. Show the routers, gateways, firewalls, wide and local area networks, Internet and Intranet, etc.
- How will you support mobile users for this system? Can you support vendors that are mobile? Can you find application of mobile agents in this system? Where and how?

Hints: Many of the product catalog issues and the tradeoffs between centralized versus distributed and virtual catalogs are similar to the issues encountered in data warehouse design. Consult the Data Warehouse chapter in the "Integration" Module. The changes to introduce for C2B can be found in the "E-commerce Platforms for C2B" chapter of the "Platform" Module. The discussion of emarkets can be found in the "EB Applications and Models" chapter of the "Applications" Module. Discussion of the EDI versus XML and the use of XML standards is discussed in the "B2B Trade" chapter of the "Platforms" Module.

2.5 A Financial Marketplace

2.5.1 Overview

A large international bank, let us call it **Xbank**, and an emarket provider, let us call it **Zmarket**, have formed a partnership to deliver financial services to B2B exchanges. Xbank will provide a wide suite of financing to B2B members through a link with Zmarket. Zmarket will route financial transactions to the Xbank link.

For those transactions that Xbank processes, Xbank will route the incoming messages from Zmarket to its respective systems and operations areas. For those transactions not processed by Xbank, Xbank will route the transactions to its correspondent banks, which will make up the **Partner Bank Network (PBN)**.

Figure 2-10 shows a logical view of the B2B Service. This Service involves a B2B exchange, Zmarket, Xbank, and the PBN.

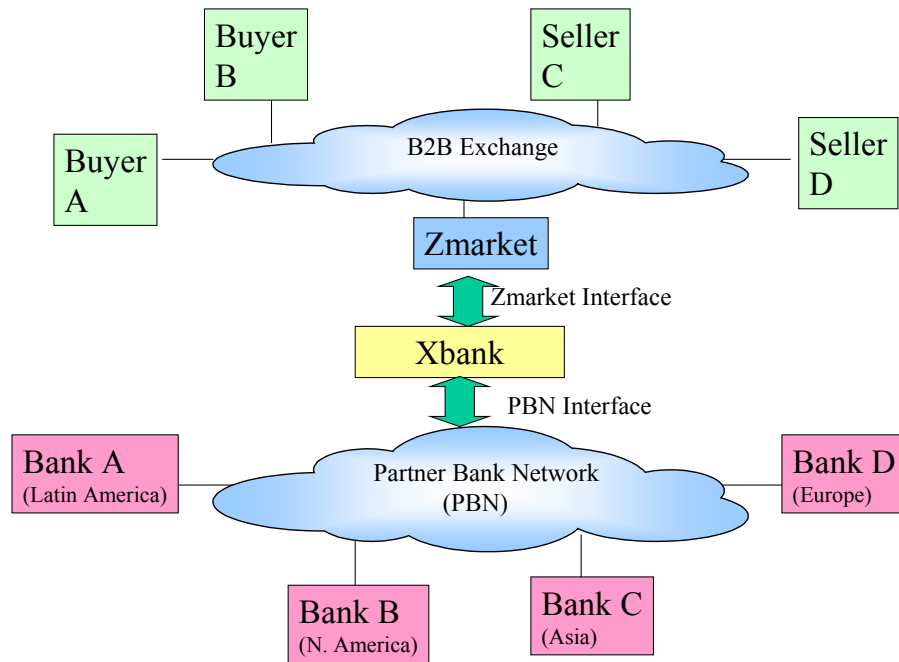


Figure 2-10: Conceptual View of a Financial Marketplace

B2B Exchange. This exchange provides members the ability to meet trading partners and arrange a deal between buyers and sellers. However, for the buyers and sellers to close a deal, credit and payment settlement needs to be made. Zmarket passes the financial transactions to Xbank for processing. For example, a seller will send the financial information about a buyer to Zmarket to verify. Zmarket will process the information and then send it to Xbank for financing and payment.

Zmarket. Zmarket will provide the authentication and certification that will underlie the transactions conducted on the exchange, as well as the transactions sent and received between the exchange and Xbank. The certificates are expected to enable trading partners to close their deals online as well as request transaction fulfillment via digital signatures.

Xbank. The bank will provide financial services to the B2B exchanges through Zmarket. Xbank may process the transactions itself, in which case the transaction request would be routed to the appropriate processing system and/or operations area. If Xbank *does not* process the transaction, then it will route the transaction to one of its partner banks for processing (depending on the transaction request and the underlying client).

Partner Bank Network. The B2B exchange can benefit from a network of financial institutions that are business partners of Xbank. Instead of financial institutions maintaining direct relationships with the members of the exchange, they can use Xbank's interface to the B2B exchange. Basically, Xbank receives

the financial transaction (e.g., request for a letter of credit) and attempts to process it by itself. If not, it broadcasts the transaction to the partner network for bids. It thus serves as a clearinghouse for the partner banks.

2.5.2 EMarket Transactions and Transaction Flows

Let us concentrate on two transactions, Payment and Finance.

Payment System. Figure 2-11 shows the general payment flows that involve direct payment (without intermediary banks) as well as payment through intermediaries. The following describes a possible payment flow process in the B2B exchange environment.

- Buyer sends a payment request on the B2B exchange.
- Zmarket receives the transaction request, certifies the sender and the instruction and passes the authorized transaction to Xbank through the Xbank-Zmarket interface in the agreed format.
- Xbank receives the instruction and validates that the authentication conditions have been met. If the Buyer has an account with Xbank and there are available funds, then Xbank will debit the Buyer account and make the payment to the Seller bank through electronic fund transfer or other means. If Xbank does not have an account for the Buyer, then Xbank will determine if there is an account for the Buyer's Bank on the Partner Bank Network.

Typically, both the Seller and the Buyer require some form of advice that the payment has been made/received. In addition, each instructing party in the chain usually requires some confirmation/advice that their instruction has been executed. Each bank has specific advising arrangements with each of their clients with set prices for the services. Advices may be sent via phone, fax, mail, or other means.

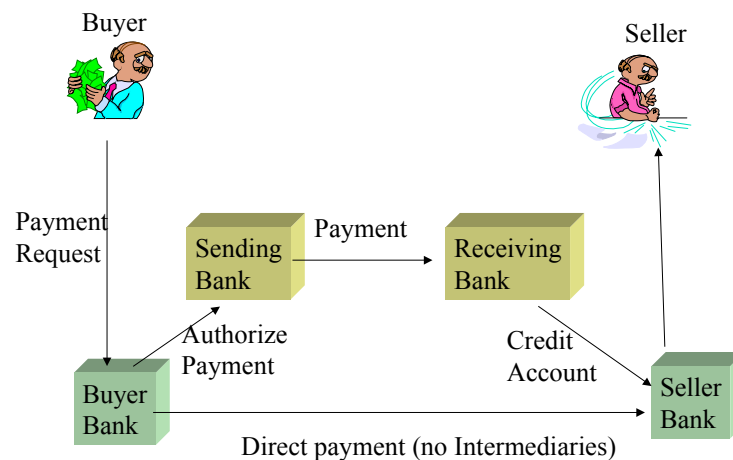


Figure 2-11: General Payment Flows

Let us work through a few scenarios of payment, depending on whether the Remitter has an account with Xbank or with a PBN bank.

The flows are represented as sequence diagrams that are used heavily in system modeling techniques such as UML (Unified Modeling Language). System models are typically built in UML by using tools such as Rational Rose. These tools not only perform consistency/completeness tests but also generate, if needed, skeleton code in Java and C++ to speed up the system building process.

Figure 2-12 shows a sequence diagram for Scenario 1, i.e., when Xbank has an account for the Buyer. In this case, Xbank after authentication and verification of available funds, will debit the Buyer's account and make the payment. After the payment has been made the payment advice is sent back to the Buyer.

Figure 2-13 shows the sequence diagram for scenario 2, i.e., Xbank does not have an account for the Buyer's Bank, but the Buyer's Bank is a partner bank of Xbank. In this scenario, Xbank will forward the instruction to the Buyer's Bank requesting them to make the payment.

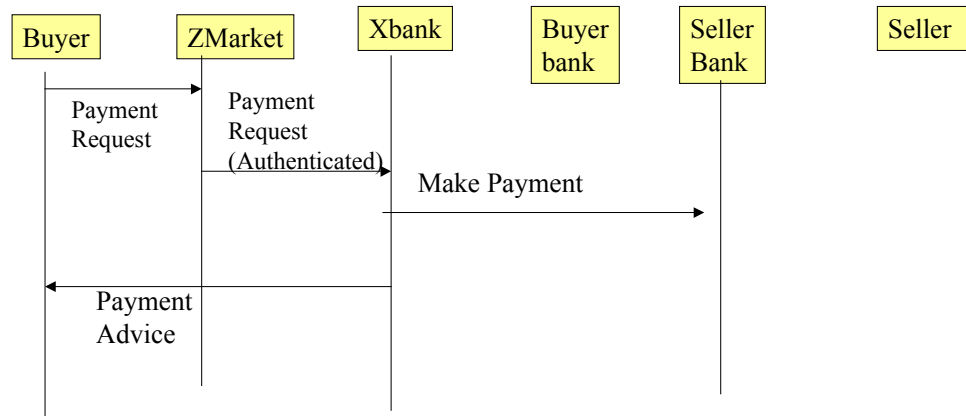


Figure 2-12: Payment Scenario 1 - Xbank Account

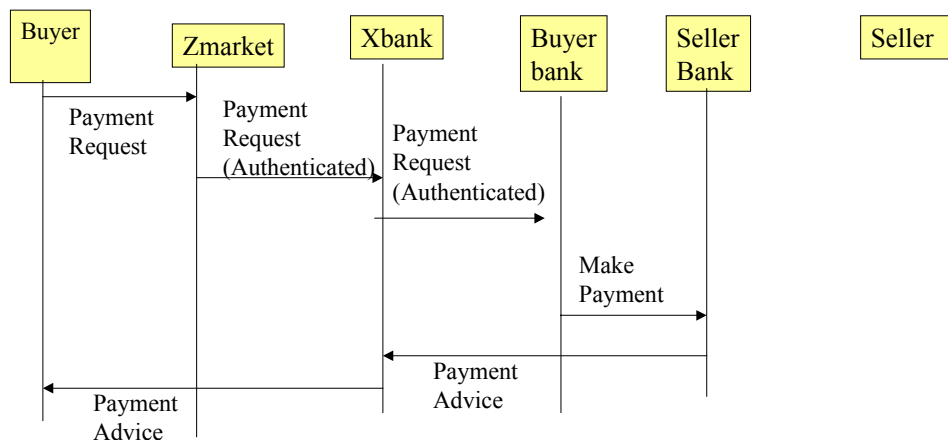


Figure 2-13: Payment Scenario 2 - PBN Xbank Account

Financing Flows. Figure 2-14 shows the general financing flows. There are several variations on the types of financing and the relationships between the various parties. The buyer, seller or both may require financing. Additionally, the banks may request financing and credit enhancements from other financial institutions in support of the payment assurance they provided to their clients. Various financing arrangements and variations are currently utilized in cross-border commerce. It is beyond the scope of this case study to discuss different financing arrangements. For the purpose of discussion, we will assume Buyer/Seller Financing. In buyer financing, also known as import financing, the buyer lender agrees to finance the buyer (importer), thereby taking the credit risk of the buyer. In seller financing, also known as exporter financing, the seller lender agrees to finance the seller (exporter), thereby taking the credit risk of the seller.

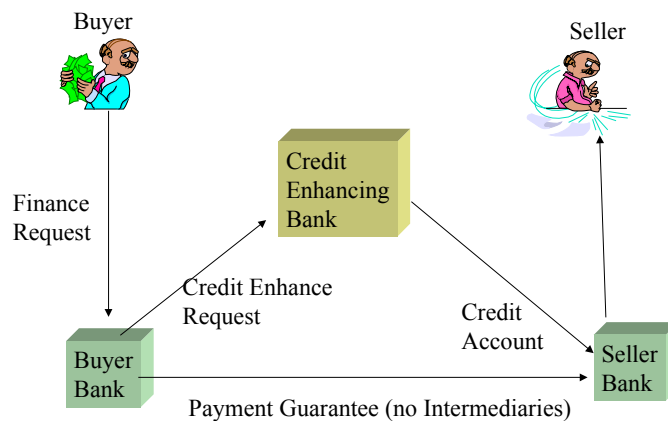


Figure 2-14: General Financing Flows

The following describes possible financing process flows in the B2B exchange environment for Financing

- The financing may occur prior to, as a prerequisite of or after the deal closing between the seller and the buyer.
- Buyer/seller requests financing for a deal on the B2B exchange. The request goes through the Zmarket certification process and is forwarded to Xbank. The financing request includes the name, address, account and financial institution of the requesting party, amount and tenor of financing, payment trigger, along with the details of the underlying transaction (i.e. goods involved, etc.).
- Xbank receives the financing requests and determines if it will provide the financing or if it will pass the transaction on to the Partner Bank Network. If Xbank provides the financing, it will confirm back to the exchange (via Zmarket), the terms of the financing (i.e. rate, maturity, terms, etc.). If the transaction is passed on to the Partner Bank Network, then the PBN will be given notice of the financing request including the details mentioned above. The Partner Banks will bid on the request and then pass the financing information back to Zmarket through Xbank.

2.5.3 An Architectural View

Figure 2-15 shows a high level architecture diagram of the pilot. This architecture is a more detailed view of the logical view presented in Figure 2-10. This high level architectural diagram shows that:

- B2B Exchange is based on the public Internet (i.e., it uses HTTP messages).
- The traders on the exchange that use Zmarket services use the Zmarket client to communicate with a Zmarket server that resides at the Zmarket site.
- Zmarket is connected to Xbank systems through a line that may be a dedicated T1 line. Xbank clients transfer the information between Zmarket and Xbank.
- The message transported between Zmarket and Xbank will be in an XML variant such as FinXML.
- The Xbank systems receive the messages from Zmarket and send them over the Partner Banking Network (PBN).

Let us go through a few more details about this architecture.

B2B Exchange. B2B exchanges provide members the ability to meet trading partners and arrange a deal. The Exchange is based on the public Internet (i.e., the members use Web browsers, Web servers, and HTTP for communications over the IP network).

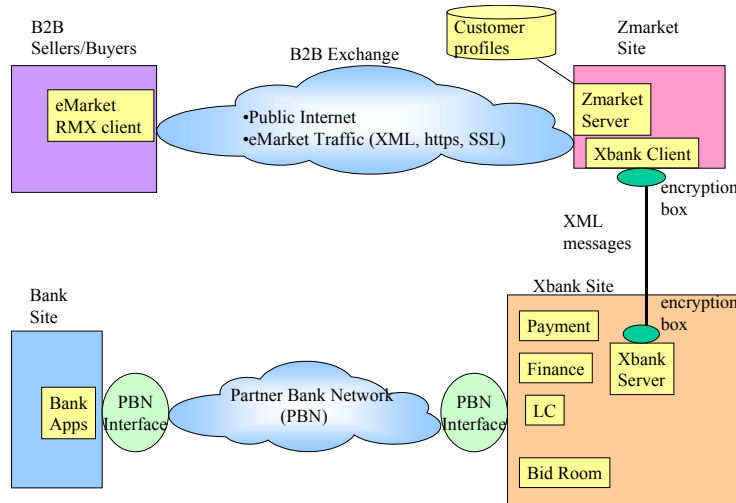


Figure 2-15: High Level Architecture of a Financial eMarket

Zmarket. Zmarket provides the authentication and certification through a server that will underlie the transactions conducted on the exchange, as well as the transactions sent and received between the exchange and Xbank. The certificates are expected to enable trading partners to close their deals online as well as request transaction fulfillment via digital signatures.

Xbank to Zmarket Link. Due to high security concerns, there will be a direct CPU-to-CPU link (a T1 line) between Zmarket and Xbank. Hardware encryptors will be installed on both ends of the T1 line for security. The Xbank software will transfer information between Xbank and Zmarket. The interface between Zmarket and Xbank is expected to change over time.

XML Messages Exchanged Between Xbank and Zmarket. The messages exchanged between Xbank and Zmarket will be based on XML. The following XML standards are worth consideration: FinXML, FpXML, and SWIFTML (a markup language used in the Swift network - www.swift.com).

Deutsche Bank Applications. Xbank software will receive the information from Zmarket and will invoke one of the following applications: Payment, Finance, and LC (Letter of Credit). These applications, as the names imply, will handle the payment, financing and letter of credit scenarios (some of these scenarios have been discussed previously). Let us discuss a new application (bid/auction room) that will need to be developed specifically for PBN.

Xbank wants to create an online mechanism, similar to an auction room for the financial institutions to participate in financial transactions originated through an exchange and routed through Xbank. This “auction room” will get the PBN banks involved in cases where Xbank does not execute the transaction, or in cases where the buyers/sellers may not have direct accounts with the PBN banks. The main idea is that a possible financial transaction can be broadcasted to the PBN for bidding. The auction room will obtain the bids, summarize them, and send them to the customer (buyer/seller). Based on customer selection, then a financial transaction is executed.

2.5.4 Review and Discussion Questions

- What is the business motivation for this financial emarket? Specifically, what are the strategic benefits that the traders, the emarket providers/owners (e.g., Zmarket), the principal banks (e.g., Xbank), and participating banks (e.g., the PBN banks) have in joining this marketplace?

- Figure 2-10 shows one possible way of implementing the financial marketplace. What other variations, with tradeoffs, can exist in this marketplace? Specifically, why cannot all the banks directly join the marketplace instead of going through Xbank and Zmarket?
- How will you define the security of such a system? What will be the different levels of security and how will the authentication be passed from one player to another (e.g., traders to Zmarket, Zmarket to Xbank, and from Xbank to PBN)?
- Define the customer profile information that Zmarket will store. This information will be vital to identifying exchange members and their relevant financial data as well as other unique information that could be used for authorizations, reporting, and other purposes. Describe the minimum, (core) and optional information that Xbank will require for transaction processing purposes.
- Discuss how the Xbank communicates with each of the financial institutions on the Partner Bank Network (PBN).
- Choose a variant of XML that will be most suitable for this emarket.
- Xbank wants to create an online mechanism, similar to an auction room, for the financial institutions on the PBN to participate in financial transactions originated through an exchange and routed through Xbank. Discuss the design of such an auction room. In particular, identify the proper message protocols and communication mechanisms for the auction room.
- What type of legal and business issues can arise in this model? In particular, how will the PBN banks accept the responsibilities and who will be liable in case of frauds?
- Define additional flows for payment and financing and define message formats between the flows.
- Develop a complete technology implementation plan for this emarket. The plan should include how Xbank systems will be affected, the technologies and components to be purchased/developed for pilot and final implementation, documentation and procedure development, organizational Implementation Issues, and include legal considerations.

Hints:

- In determining different configurations and strategic benefits, keep the tradeoffs between the following factors in mind: the issues of making money for the participants, getting buy-ins from participants, management and support issues, handshakes in security, and fraud detection/avoidance concerns.
- The security issues in this case go from physical network to the financial transaction level. A comprehensive security architecture is needed.
- You can assume that the customer name, address, and financial institution name and account number are always needed.
- The Xbank can communicate with each of the financial institutions on the Partner Bank Network in a variety of ways, depending on the nature of the transaction. For instance, SWIFT network is widely used as the authenticated communication vehicle for payment and trade transactions in the banking industry (www.swift.com). Some documentation is still exchanged in paper form and signatures are maintained by respective institutions. By extension, there are significantly more vehicles for the financial institutions to communicate with their clients including telephone, fax, Internet, dial-up systems, and in many cases, clients still walk in to a local branch to initiate their financial transactions.
- For deciding the XML variant for this emarket, FpXML, FinXML, swiftXML and other similar players in the financial industry should be reviewed. It is best to produce a table such as the following for analysis (a few initial entries in the table are given to get started).

| FpML | FinXML | SwiftML |
|-----------------------------|-----------------------------------|---|
| May be for derivatives only | Wide range of instruments covered | SWIFT may probably accommodate XML standards/needs of clientele |

- The “auction room” can be thought of as a vehicle to obtain new business in cases where Xbank does not execute the transaction, or in cases where financial institutions may participate in the credit process associated with the transaction. You should identify the proper message protocols and determine how much can be leveraged from the existing SWIFT messaging infrastructure. You should attempt to minimize the amount of infrastructure cost that each financial institution would incur in order to participate in this model.
- A variety of legal and business issues can arise in this model. For example, will the Partner Banks accept a Zmarket certificate and/or instruction/request from Xbank in lieu of their traditional requirements from the Remitter. A major issue centers around liabilities for fraud, etc. Another key issue is that SLAs (service level agreements) with Partner Banks on the PBN need to be clearly specified. A key problem could be delegation of authentication across different players (e.g., trader to Zmarket, Zmarket to Xbank, Xbank to participating banks).
- Definition of additional flows for payment and financing requires knowledge of banks and financial institutions. People knowledgeable in these areas should undertake this exercise. To develop the exact message formats between various players, you could assume these messages to be SWIFT messages. These formats will help in choosing the right flavor of XML (e.g., FinXML versus SwiftML) for messages between Xbank and Zmarket.
- The following details could be considered, for example, in a Technology Implementation Plan for this emarket:
 - Emarket client interfaces -- the B2B exchanges could provide their own GUIs for the participating traders.
 - Xbank internal and back-end (PBN) system interfaces -- they should stay the same as much as possible.
 - Data mapping: The required data for each transaction will need to be converted into the format Xbank requires in order for Xbank back-end systems to process. This mapping will take place at Xbank site. The task of mapping (e.g., translate FinXML to SWIFT) could be automated.
 - It may be a good idea to develop the front-end at Xbank that will allow Zmarket to interact with the Xbank internal systems.
 - You need to define the exact XML transactions with standards such as ebXML, FinXML.
 - You also should design the internal Xbank applications (e.g., letter of credit) and determine the app to app message flow within Xbank and with PBN apps. A publish-subscribe model may be useful for this architecture.
 - Exact selection of middleware (e.g., to support publish-subscribe), physical network, and computing platforms

2.6 City of Seattle Public Utilities Go Wireless

The following case study is a summarization of a case study described in an article by Peter Rysavy, [Rysavy Research](#), for "PCS Data Today" online journal, April 30, 1999. The purpose of this case study is to see how you would approach the problem now.

2.6.1 Introduction

City of Seattle Public Utilities has spearheaded a wireless project. The overall goal is to provide field workers and crew chiefs with online access to inventory databases, work orders, maps, e-mails, schedules, and other essential information from anywhere through wireless access. The applications the utility is extending to its mobile workers include both work management and office applications. Developing wireless networking solutions requires special considerations.

The utility's work and inventory management application is MAXIMO, a system that uses an Oracle database, developed by PSDI (Bedford, MA, <http://www.psd.com/>). MAXIMO is a vertical market-type application. The office-based applications that the utility provides to the field include Novell's GroupWise

and Web-hosted applications on the utility's intranet. These are horizontal market applications -- their use is not restricted to any particular job function or type of industry. The goal of these applications is to provide reliable remote operation with preferably the same user interface as a direct LAN connection.

The utility has two types of remote workers who will use the wireless system: leads that will use MAXIMO primarily and crew chiefs who will use the office applications in addition to MAXIMO. The key goal is to provide a flexible wireless architecture that allows new applications to be added easily. In addition, wireless connections should be no less secure than existing remote access methods based on dial-up connections. Finally, the utility wants an approach that allows it to migrate easily to other wireless technologies in the future.

2.6.2 Choosing the Computing Platform and Wireless Network

The utility considered Windows 95 and Windows CE platforms and chose Windows 95 due to scalability issues. They considered many wireless networks such as the analog cellular network, PCS technologies, and four wireless packet networks with service in the Seattle area.

The utility decided to base their applications on TCP/IP communications and a packet-based approach to better support the frequent communications needed for workers in the field. This requirement eliminated circuit-switched cellular connections. Since packet-based services were not available for digital PCS networks, the remaining choices were CDPD and the Metricom Ricochet network. CDPD and Metricom Ricochet are both IP-based packet networks. Data services for GSM and CDMA digital PCS were candidates (these services were not widely available in 1999).

2.6.3 Application Architecture

The utility wanted to implement an architecture that insulates its applications from the actual network used to the maximum extent possible. The IP-based approach, where applications make no assumptions about the nature of the physical connection, achieves this goal.

Migrating between network types is possible, though some adjustments may be necessary for each network. For example, CDPD uses fixed IP addresses and Metricom Ricochet uses dynamically assigned addresses. This difference could affect how firewalls are configured. The effective throughput rates of Ricochet and CDPD also differ, with Ricochet operating at 20 to 30 Kbps and CDPD at about 10 Kbps.

2.6.4 Software Approaches

Application designers need to consider wireless network features (i.e., they operate at lower speeds with higher latency, and connections can be lost at any moment, especially when mobile). There are several approaches:

The first approach is to use TCP/IP protocols between client software installed on the mobile computer and the various servers. Because some workers will be working with the same applications both in an office environment and in the field, the advantage of this approach is that the user interface stays the same in both environments. A disadvantage is that this approach does not address some of the connectivity issues associated with wireless, such as throughput and latency. Another disadvantage is the requirement for software installation on field computers, which can add to maintenance and support.

Another approach is to use Citrix MetaFrame (combined with Microsoft Terminal Server), where applications run on an application server at a central location, and mobile nodes operate as terminals (thin clients). The utility has already deployed Citrix MetaFrame to support dial-up users. The advantage of this approach is that installing and maintaining mobile computers is simplified because they only need the Citrix client software to access multiple applications. The disadvantage is that Citrix MetaFrame has some significant limitations when operating over wide area wireless connections. We learn about these limitations in the next section when we look at test results.

The third approach is to use wireless middleware (specialized software installed on a mobile computer and on a centralized server that acts as an intermediary between client applications and server processes) to optimize communications. The utility has looked at wireless middleware designed specifically for MAXIMO, as well as general purpose middleware that optimizes IP communications over wireless links. The advantage of wireless middleware is it allows applications to run with much better response times and much greater reliability, however, it increases complexity and adds cost.

Because wireless coverage is not always available everywhere the workers spend time, an approach also considered was Oracle Lite where workers can download a subset of the database they need, operate on it locally during the day, and then synchronize at the end of the day. This approach reduces the demand for wireless connectivity, but it is not as flexible as the other approaches where field workers remain in constant communications during the day and can respond quickly to changing circumstances.

2.6.5 Review Questions

Read this case study and describe how you would approach the problem now. Specifically:

- What will be your choice of wireless network?
- What middleware service will you chose and why?
- Will you choose a mobile application server, which one and why?
- What additional mobile applications can you think of in this case?

Hints: Most of the answers can be found in the "Wireless and Broadband Networks" Chapter (Networks Module) and "Mobile Application Servers" Chapter (Platforms Module).

2.7 Insuring Outsourced Enterprises in e-Commerce – A Methodology

2.7.1 Overview

Insurance companies are having trouble insuring the current corporations that have many of their e-commerce services outsourced. Consider, for example, Pete's PC Shop (a fictitious PC store), that wants to be insured for a \$1 million per year potential business loss due to fire. The insurance companies will visit Pete's Shop to verify that fire-hoses and other anti-fire equipments are working properly and that the company indeed does do a \$1 million per year business before writing an insurance policy. Consider, now, the situation where Pete wants to be insured against possible IT infrastructure failure (Pete supports Web advertising and online-purchasing -- failures of these services could cost PC quite a bit). The complication is that Pete has outsourced Web advertising as well as online-purchasing to ISPs (Internet Service Providers) and/or ASPs (Application service Providers). How will the insurance companies write an insurance policy in this situation?

Insurers write policies insuring companies against various types of failures. Insurers need to be able to estimate quantitatively the risks due to the IT infrastructure failures for companies that use the outsourcing model. The IT infrastructure includes client premise (e.g., customer application, Web browser interface), the physical transport network (e.g., the fiber cable and the routers/switches), and the customer site (e.g., the ASP site with Web servers, back-end applications and databases). These failures may happen due to outages, errors/mistakes or malicious attacks.

For purpose of discussion, the e-commerce (EC) activities used by companies fall into the following stages of evolution (see Figure 2-16):

- Stage 1: A Web site that allows external customers to get information about the company products and services (through a Web site) and get in touch with appropriate people through email and/or phone.
- Stage 2: An e-commerce site that will allow the users to directly order and purchase the company products and services through the Web site. This implies that the internal flow of online purchase information has to be supported.

- Stage 3: An “advanced” e-commerce site that allows the company to participate in a virtual enterprise by using B2B trade. This implies a great deal of activity in the network with complex interfaces, protocols, and middleware components.
- Stage 4: In this case, the B2B trade is conducted through an emarket. In addition, features of realtime business activity monitoring, mobility and self serve customers are included.

Naturally, the risk depends on which stage a company is operating at present. In addition, *estimation* of risk becomes increasingly complex as you proceed from stage 1 to stage 4. The companies in stage 3 and stage 4 are more at risk, especially if the services have been outsourced. The following discussion presents an outline of a methodology that can be used in estimating risks of outsourced businesses.

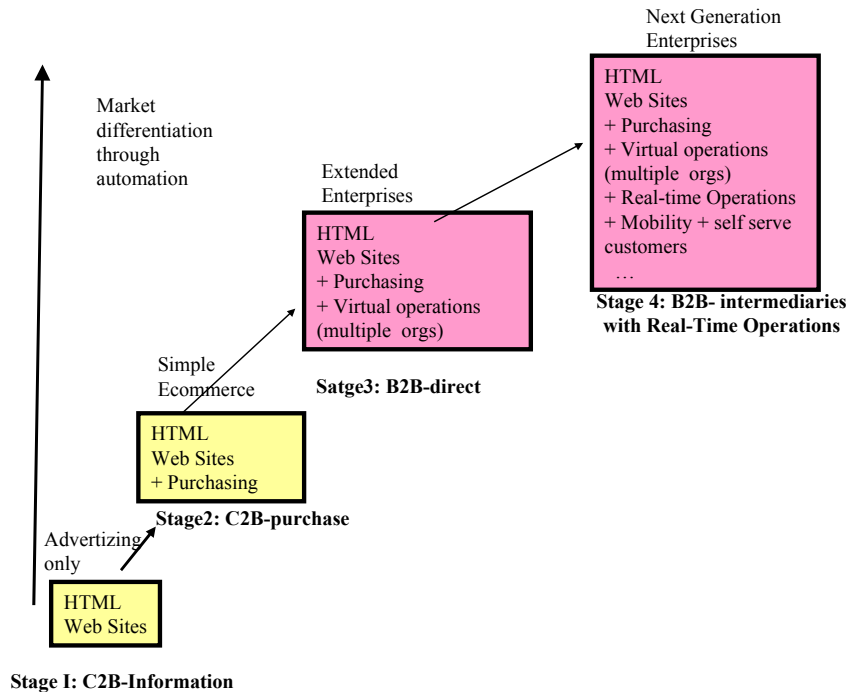


Figure 2-16: Stages of e-Commerce/e-Business

2.7.2 An Approach to Estimate Risks

There are numerous ways IT infrastructure can fail. Each of these failures could have a different effect on the business. Insurance companies need to estimate the risks of these types of failures for an e-commerce business. Here are some key ideas:

Establish a Checklist. A checklist is needed to elicit information from clients about the infrastructure that supports e-commerce. In addition, the checklist will form the basis for a qualitative assessment of the risks of a failure on e-commerce for a business. The checklist covers network architecture, network equipment, network implementation, and the databases/applications. The checklist attempts to identify major single points of failure and gives a qualitative assessment of the impact to the insured. The checklist covers:

- Access portions as well as common portions of the network
- Type of information (volume and value) that is being transmitted over the network
- Implications of wireless versus wireline operations
- Impact of practices and policies

This information can be used to assess how various types of services over networks are affected by different types of failures (i.e., stage 1 through stage 3) under different types of outsourcing models (e.g., ISPs and ASPs).

Develop Methods to Quantify Risks. Software tools and methodologies for quantitative analyses of IT risks based on the checklist produced earlier are needed. This includes the following activities:

- Review of network, service, security, and reliability data collected as appropriate for the selected network or service, as inputs to the analysis. This data includes availability, security and reliability data on e-commerce infrastructure failures.
- Identify failure scenarios for an e-commerce system that will result in loss of service for that business's customers. These failure scenarios may be single points of failure or combinations of failure events in the network. Most failure scenarios will be combinations of individual failures. These scenarios include failures of hardware, software, procedural and security causes.
- Develop models that can be used to estimate the frequency that each failure scenario will occur in that e-commerce network. The models should be able to provide information for small networks as well as typical large generic networks.
- Develop models to estimate the anticipated effects of each type of failure on e-commerce. These effects or impacts depend on the services for which the system is being used. These models should be scalable, i.e., they should be able to provide information for small networks as well as typical large generic networks.
- Adapt models to estimate the overall risks in the business's system. You need to quantify those locations that contribute to overall risk in an ISP or a business network.

The models should be documented with a description of the various failure scenarios; a description of the major models used to estimate frequency of failures present in the network or service architecture; and a description of how the impact of various failures is estimated and used.

Application of Methods for an e-commerce System. You finally apply the methods developed previously to an actual e-commerce implementation. This may be a stage 1, stage 2, or stage 3/4 implementation. In particular, you need to:

- Review network, service, security, and reliability data collected on the particular network being analyzed.
- Identify failure scenarios for the actual e-commerce implementation. This may differ from the scenarios identified previously. Most failure scenarios will be combinations of individual failures. These failure scenarios will cover hardware, software, procedures and security.
- Estimate the frequency that each failure scenario will occur in the e-commerce network.
- Calculate anticipated effects of each type of failure on e-commerce.
- Estimate the overall risks in the e-commerce network. You will quantify those locations that contribute to overall risk in the e-commerce network.
- Come up with recommendations on how the network could be improved to reduce large risks.

2.7.3 Review Questions

- Extend the four stages of e-commerce by using additional business patterns.
- Cast the approach given above into an expert system. What will be the questions asked and what will be the rules used in the expert system? Give an example.

Hints: The chapters in the “Applications” Module have many business patterns and information on outsourcing that can help with this case study.

2.8 An Integrated Manufacturing System

2.8.1 Overview

Manufacturing companies have a major integration task ahead of them. Simply stated, manufacturing companies need to extend and integrate the applications that support three types of processes:

- Business processes such as payroll, accounts receivable/accounts payable, order processing, marketing information systems, online-purchasing, and computerized checkout systems,
- Engineering processes such as computer-aided design, computer-aided engineering, computer-aided process planning.
- Manufacturing processes such as material requirement planning, production scheduling and flexible manufacturing systems.

Xbuild is a medium sized manufacturing company that is integrating and automating the order processing, inventory control, CAD/CAM (computer-aided design/computer-aided manufacturing) and the manufacturing processes of the company products that involve building IBM PC desktops, laptops, and network devices. This system, referred to as the ***Integrated Computer Integrated Manufacturing (ICIM)*** will receive a customer order and assemble and pack a product for shipping within an hour of order reception.

Figure 2-17 shows a very high level view of ICIM. The first stage in this system is an order processing system which processes orders for a product. If the specified product is in stock and the customer's credit is acceptable, the product is shipped to the customer from the finished product inventory. The product inventory is adjusted to show products shipped. For an out of stock product, a CAD/CAE system produces the design based on the customer specification. The design is then downloaded to a Computer Aided Process Planning (CAPP) system where the manufacturing program is automatically created which shows how the product will be assembled. The CAPP system uses the information about available assembly equipment to generate the process plans. An MRP (Material Requirement Planning) system determines the materials needed for the product. MRP systems use sophisticated algorithms to take into account quantity discounts, vendor preferences, various capacity constraints and factory status. The manufacturing program is downloaded to a flexible manufacturing system (FMS) which consists of an area controller, two cells and several manufacturing devices. FMS also receives a production schedule (how many products to manufacture) and needed raw materials. Because FMS is a realtime system, it must conform to the constraints of realtime control on factory floors.

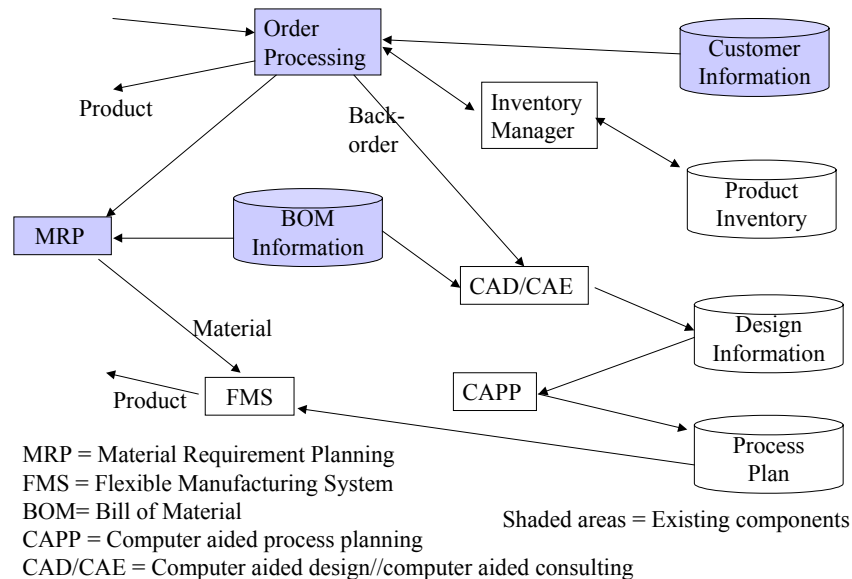


Figure 2-17: Advanced Integrated Control System (AICS)

As can be seen, this system has a combination of application types such as operational support (e.g., inventory manager, order processing), decision support (MRP) and realtime (FMS). At one point, some applications and associated databases are built by the company, others are packaged applications that have

been purchased, and yet others need to be either built, purchased, or outsourced. Interlinking of these applications is a major challenge facing many companies.

2.8.2 Review and Discussion Questions

- Extend to include an online purchasing system. In other words, integrate the ICIM system with an online purchasing system so that a customer can order a product online that is built by using ICIM and shipped to the customer.
- Develop a business component model of ICIM. Does this help in integration?
- Develop a list of standards that are playing a key role in computer integrated manufacturing systems.
- Survey, evaluate and pick COTS technologies and platforms that can support ICIM system. Keep the standards and interoperability issues in mind while making these decisions.
- Translate Figure 2-9 into a physical architecture ("solution architecture") that shows the middleware and network. You should also include security considerations and map existing COTS technologies to this architecture.
- Develop a detailed network design for this system. In particular, what network design issues will you need to consider to support realtime operations such as FMS?

Hints: For integrating with online purchasing, you should review the purchasing systems represented by Figure 2-8 and Figure 2-9. A good place to start for manufacturing standards is the Society of Manufacturing Engineers Web site (www.sme.com). For manufacturing related COTS products for MRP, BOM, and others, search for ERP (Enterprise Resource Planning) Systems for manufacturing from companies such as SAP (www.SAP.com) and BAAN (www.BAAN.com). Discussion of object models, architectures, and network design issues can be found in the "Architectures" and "Networks" modules.

2.9 A Customer Relationship Management Portal – A Technical View

2.9.1 Overview

This case study describes a customer relationship management (CRM) portal that combines several individual CRMs by using several emerging technologies such as XML, EJBs, WAP, VML, EAI, etc. The purpose of this CRM portal, conceptually shown in Figure 2-18, is to provide a common integrated view of an international corporation that has acquired several companies, each with its own CRM and customer contact center. The acquired companies are run as "Providers" that collectively provide the overall services of the corporation. Examples of the Providers are an e-shop, a financial institution that issues credit cards, a customer contact center, and several manufacturing plants. Each Provider currently has its own CRM -- thus the customers cannot take advantage of any cross-over functionality. For example, if a customer buys something from the e-shop, she does not know anything about the financing options available to her from the affiliated finance Provider. Development of this portal includes:

- Front-end integration with customers using Web browsers, cellular phones (WAP), and voice recognition systems (Voice markup language)
- Back-end integration with existing partner CRM systems (e.g., e-shop CRM, delivery CRM, credit card CRM, etc.).
- Business processing with EJBs (Enterprise Java Beans) for maximum flexibility

The system is accessed through 2 media: Voice and Web. Voice access is through 3 types of devices: Wireline phones, Wireless phones, IP phones. These 3 devices are accessed through 3 different types of networks: IP networks, Wireless phone networks, and the good old fashioned PSTN (public switched telephone network)

Even though these access technologies are different from each other, the system will provide the same functionality when being accessed by voice. The functionality will be provided through VoiceXML to implement the dialogs and voice menus. The VoiceXML server will be attached at the back-end to the main

application server, while at the frontend, it will interface with the 3 kinds of networks mentioned above through different gateways.

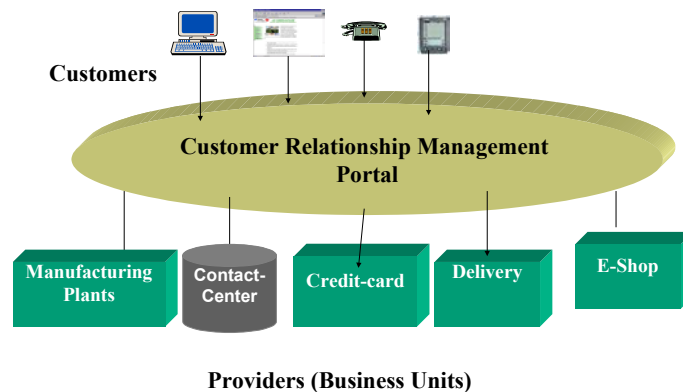


Figure 2-18: Conceptual View of a CRM Portal

This case study is based on an extensive prototype that included the following considerations:

- The Web access to the system is through 2 types of devices: Web browsers running on workstations, Wireless Phones and Wireless PDA's that run micro-browsers over WAP.
- The data served by the Web server is in XML. The user is presented this data in 3 different formats: XML/XSL for XML-enabled browsers, HTML for browsers that are not XML-enabled, WML for Wireless phones and PDA's.
- The system translates the XML data to HTML or WML before serving it to the users of non XML enabled browsers or wireless devices.
- The functionality accessed through the different media are consistent. The functionality accessed through voice and WML are subsets of the functionality provided by the Website.
- Business logic includes Object Classes, Business Objects, Customer Data, Usage Log, Work Flow, and message between objects.

The various usage scenarios and the detailed architecture of this prototype are discussed here for insights.

2.9.2 Usage Scenarios

A few scenarios are included in this section to clarify the functionality of the CRM Portal and to illustrate how the various back-end systems interplay to satisfy customer needs.

Scenario 1: Buying a product from an e-shop

This scenario works in coordination with a personal portal site like my.cnn.com or my.yahoo.com, a public service site, an e-shop, a delivery service provider, a call center and a personal application -- all provided through the contact center.

- All customers start from the customer's personal front page. The page gives the customer an entrance to her subscription site. The contact center manages her authentication info to login to other personal sites. Therefore she does not need to re-enter her ID nor password to enter any new site.
- The customer (Pat) buys a PC from the e-shop, checks its price and possible delivery days. She purchases it by using her credit card number, delivery destination and other necessary info (this may be pre-registered). After order completion, a button to track the delivery is added to her "watch list". The PC is registered in Pat's personal profile as "ordered"
- On the day when the delivery was scheduled, the PC does not arrive. Pat checks the status of delivery by using her cell-phone.
- The tracking info on the watch list claims that the delivery will be late.
- Pat wants to complain, she pushes the "call" button on the tracking page. Immediately her cell phone makes a call to the call center.

- At first, an operator apologizes for the delayed delivery and gives a new delivery date to Pat. Then the operator offers that the e-shop will give Pat additional service points at the next purchase.
- Pat calms down, then hangs up.
- Next day, the PC is delivered and automatically status of the PC in Pat's profile is updated

Scenario 2: Buying a problematic dishwasher

Maintenance process is illustrated in this scenario.

- Tom's family bought a dishwasher from one of the providers.
- After a while, an email message is sent to Tom's family. The message says "welcome to appliance warranty program", and an application form is attached in the email indicating that if Tom signs up for this warranty in two days, he will get two years free maintenance. This email is sent by a back-office marketing system of the shop that has connectivity to the customer contact center.
- The family subscribes to the program. A little later, the dishwasher appears in the family's appliances list automatically. Each item in the list has anchor points for CRM info managed by a manufacturer, dealer or participating store.
- The dishwasher arrives and does not work properly
- Tom's wife opens the technical support Web pages of the dishwasher from the family's appliances list. The page leads her to more detailed info about the dishwasher
- She can not find out how to fix the problem by using the Web site. Irritated, she pushes "call me" button (we assume that her phone is IP enabled)
- A repair specialist calls and asks her to check several points. These points are displayed on Web pages (the contact center knows that she has navigated many pages, so the call request is transferred to the specialist with this information)
- The repair specialist decides that an engineer dispatch is needed. He then arranges for a repair visit.
- In Tom's family calendar, date and time for the engineer visits are added.
- The repair fee is paid by the warranty program.

Scenario 3: Storing reports in Contact Center repository

- The Credit Card Company sends a message to the customer's (Sam's) message inbox. The message contains a bill with a usage log.
- Sam (at home) retrieves the message into his secure repository at the Customer Contact Center in order to access the message from his cell phone later. He chooses a secure mode of storing data.
- A few days later, Sam checks the log by accessing the repository through his cell phone. The repository is guarded by user ID and password.

Scenario 4: Subscribing to the Customer Center

A customer (Joe) buys a refrigerator from a provider .

- A salesperson recommends that Joe subscribe to the "Ultimate Customer Service". Joining this service gives Joe free Internet access time and cell phone/long-distance phone time.
- Joe is glad to join the service and he agrees that his cell phone bill and credit card bill are sent by the e-service so that he can get more free access time
- For first time access to the service, he is asked to fill some personal preference information that is used in later interactions with the customer.

2.9.3 Detailed Design – Solution Architecture

Figure 2-19 shows a high level architectural view of the CRM portal. The architecture is organized in the typical front-end, business, and back-end integration layers. This architecture uses WAP (Wireless Application Protocol) and VXML (Voice XML) at front-end, EJBs (Enterprise Java Beans) at the middle, and JMS (Java Message Services) and XML at the back-end integration layers. The system is designed to be loosely coupled so that new players can be added into this architecture. Figure 2-20 shows a more detailed

implementation view of the CRM portal. A few of the implementation details are discussed in the following sections for illustrative purposes.

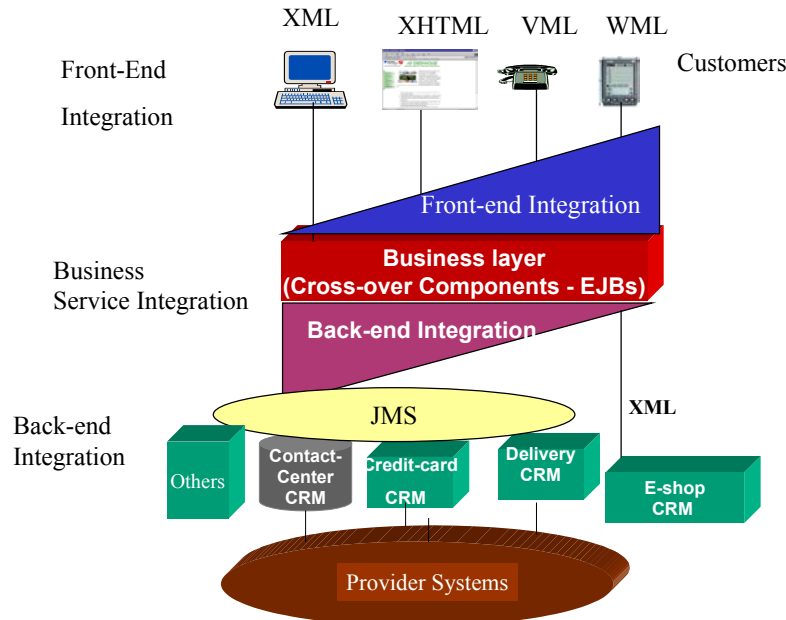


Figure 2-19: CRM Portal - High Level Architecture

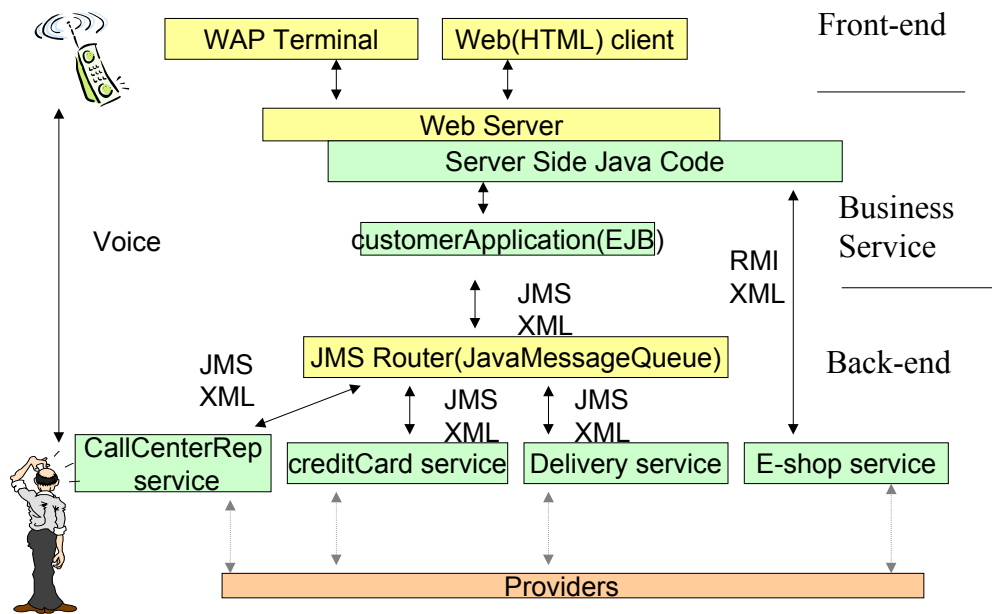


Figure 2-20: CRM Portal -- Implementation View

2.9.3.1 Front-End Integration

The front-end of this system provides integrated access through Web browsers and WAP phones. For WAP access, the WML (Wireless Markup Language) and Phone.com UP.Link Platform were used. The UP.Link (Unwired Plane Link) platform from Phone.com offers a free SDK (Software Development Toolkit). The VoiceXML infrastructure was also used to support voice users. The Voice eXtensible Markup Language

(VoiceXML) is an XML-based standardized voice markup language that has been proposed by an industry forum founded by AT&T, IBM, Lucent and Motorola and supported by more than 150 organizations. The goals of VoiceXML are to enable Internet accessibility via voice and phone (e.g., voice activated Web browsing), make development of interactive speech-enabled applications easier, and provide easy integration of voice and data services.

2.9.3.2 Business Services Layer Design

The business services support the scenarios described above. They handle user requests, perform appropriate processing and invoke needed back-end services. These services are implemented by using Java technologies. In particular, EJB's are used to develop these services as business components. The customer database uses Oracle DB that is accessed by the EJBs that are invoked from the Web browser by using Java servlets and Java Server Pages (JSP's). This is shown in Figure 2-20. This means that the Web server must support servlets. Examples of such Web servers are Java Web server (<http://java.sun.com/products/jsp/download.html>) and the Apache Web server (<http://httpd.apache.org/>). XML is used heavily in this layer. The XML documents are processed with a Java XML parser such as Sun's XML parser (<http://java.sun.com/xml/download.html>).

The business services invoke the back-end services by using one of the following approaches:

- The system interfaces to the back-end databases using JDBC.
- Remote objects representing back-end systems such as e-shop are invoked through Java RMI (Remote Method Invocation). This approach is used for tightly coupled systems.
- A publish-subscribe model that uses the Java Message Service (JMS). This is discussed in more detail in the next section

2.9.3.3 Back-end Layer Design

The back-end layer consists of a variety of existing services that are invoked largely by using the Java Message Service (JMS). The Java Message Service API is part of the Sun J2EE architecture specification. It supports the following two main models of messaging systems:

- The Publish/Subscribe model. The publish/subscribe messaging system uses an event driven model where a publisher (or message producer) publishes an event, while a consumer that has subscribed to the event receives the event message. The messages are routed to subscribers based on their topics, which the consumers have registered to.
- The Point-to-point model. In the point-to-point model, each consumer has its own queue for incoming messages. The producer sends a message to the message queue of the subscriber directly.

JMS is very suitable for this application because it allows you to invoke existing provider services (e.g. credit card CRM, contact center CRM) asynchronously. In particular, the publish/subscribe is a very powerful model for loosely coupling a wide range of services. JMS is not an implementation -- it is a specification for vendors of enterprise messaging systems to implement a Java API to their systems.

2.9.4 Review Questions

- What key technologies are used in this prototype and what value do they provide (suggest alternatives if possible).
- Translate the scenarios discussed above into UML sequence diagrams
- Study Figure 2-20 and describe the XML documents that represent the messages exchanged between various components.
- Give pseudo code of the various EJBs that are used in this system.

Hints: As a general comment, this case study is very detailed and involves many implementation details. The "Architecture Implementation" chapter in the "Architecture" Module provides a great deal of information in this area and should be consulted.

Specifically, the following technologies are being used with these possible business values (you should extend this discussion).

- **XML:** XML can be used at different layers. At the back-end layer, the communication between the back-end database and the application server is in XML. At the presentation layer, XML is used to present data to the user, through the Web or through VoiceXML.
- **Business Components:** Business components (e.g. EJB's) can be used in designing highly flexible E-business applications.
- **Mobility:** Support of system access through wireless devices running WAP.
- **Integration of the presentation media:** The system demonstrates how the different presentation media offer similar functionality using the same server logic.

2.10 Migration of a Project Management System

2.10.1 Overview

In an electronic component corporation, a company-wide project management system (PMS) was migrated from mainframe to PC-based platforms. The PMS facilitates 3 functions: project creation, project tracking, and electronic transmission of the project information to project managers for status review. Typically, there are 3 categories of data associated with projects:

- Administrative data that contains information such as project number, organization responsible, the manager name and phone number, funding source, funding level, major customer(s), planned completion dates and various other project administration indicators.
- Text data that typically contains descriptions of the project work, its basic assumptions and expected benefits.
- Status information that describes the financial status (e.g., the amount of money spent), and progress status (e.g., milestones completed, open issues, etc.).

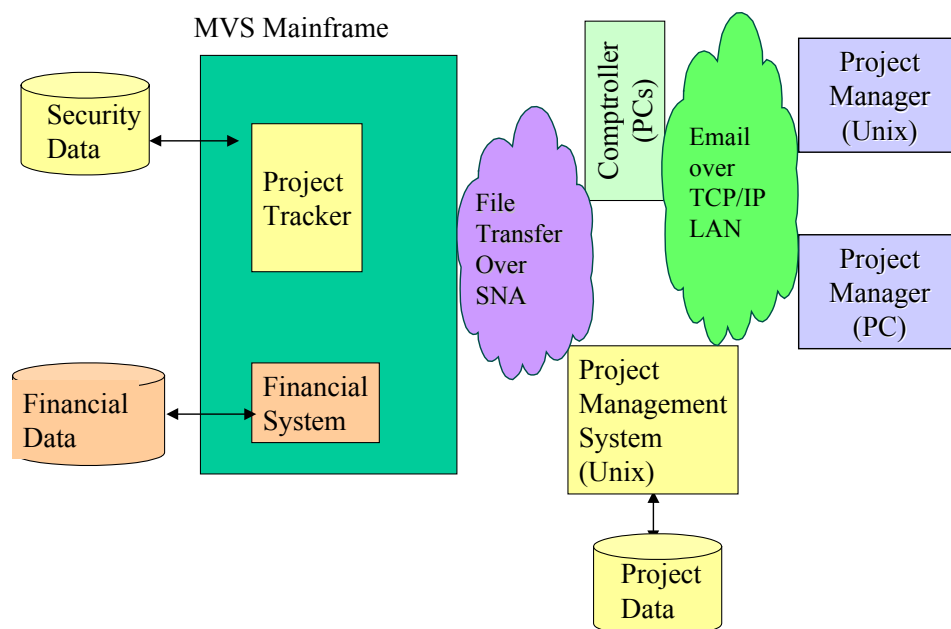


Figure 2-21: Initial Project Management System

The initial PMS was an IBM mainframe MVS-DB2 application which provided displaying, updating, and printing capability of project information over an SNA network. The project number was used for validation within other internal financial systems on MVS. The initial PMS system maintained a table of access security (project security table). Support staff could only access their authorized project information through this security table. Figure 2-21 gives a high level functional view of the initial system. The PMS

system accessed the project data and security tables stored on MVS. The main users of this system are the company comptrollers who send the information to the project managers. Initially, project managers did not have a direct way to review and/or update project information. The initial process was to download project information to comptroller department computers (PCs). Then the project information was transferred to the project managers through e-mail (some project managers use PCs, the other use UNIX platforms). After a project manager had reviewed and updated the project information associated with the project, the comptroller support staff received a copy through email and then updated the MVS PMS for the official information and printing.

2.10.2 Step 1: Business Drivers and Requirement Definition

With increasing responsibilities of the project managers and reduction in the number of comptroller support staff, there was a need to provide a mechanism that allows all the project managers to track their project information on-line directly. There were about 300+ project managers who resided in 10 regions of the company. Each region had a comptroller. The main requirement was to support the 300+ project managers directly with minimal or no intervention from the comptrollers. The following statements provided additional information:

- Security and confidentiality of project information was a crucial requirement. In the initial PMS system, support staff could only access their authorized project information through a security table. It was up to the support staffs to distribute the project information to the responsible project managers. An equivalent security system would be needed, i.e., as a policy, the project security table would provide the security for auditing of any future system.
- Size of the project information database was about 800 MB.
- Each region had different local software packages (e.g. Microsoft Word, etc.) to manage their project information. Some of these packages ran on UNIX based word processors, some ran on Mac based word processors, and others ran on PC based word processors. Before updating the MVS PMS system, all word processor macros had to be filtered out to first obtain a plain data file. This was a very time consuming task for the support staff. This task had to be eliminated.
- Organizational and strategic pressures forced reduction of the operating cost by 20%. This could be accomplished if mainframe CPU and DASD usage was reduced by migrating to cheaper platforms. The comptroller support staff was part of the company overhead and this overhead was being reduced. Thus any solutions had to consider reduction of the comptroller staff. The organization feels that the future of the business systems will move from expensive mainframes to cheaper micro computers without losing performance.

2.10.3 Step 2: Development of a Solution Architecture

Based on an initial analysis of the problem, a task force identified the following viable options:

- Keep the initial configuration intact (i.e., do not change anything).
- Keep the project data and PMS system on MVS and provide access to it through terminal emulation software from project manager desktops.
- Keep project data and PMS on MVS and provide access to project data through client tools located on project manager desktops.
- Move most of the project data and PMS to a UNIX platform and provide access to project data through client tools located on project manager desktops.

These options were evaluated against the key business drivers and requirements stated in Step 1. Table 2-1 summarizes the results of this analysis.

Table 2-1: Intuitive Analysis of Migration Strategies

| Evaluation Factors (Information) | Keep the initial Configuration | Project Data and PMS on MVS, | Project Data and PMS on MVS, | Move most of the Project Data and |
|-------------------------------------|-----------------------------------|---------------------------------|---------------------------------|--------------------------------------|
|-------------------------------------|-----------------------------------|---------------------------------|---------------------------------|--------------------------------------|

MODULE (OVERVIEW)

| Requirements) | | direct access from desktops through terminal emulations | direct access from desktops clients | PMS functionality to UNIX and access UNIX from desktop Clients |
|---|----|---|-------------------------------------|--|
| Give more control to project manager | -3 | -1 | 2 | 3 |
| Eliminate intervention by comptrollers | -3 | 3 | 3 | 3 |
| Support 300+ users directly | -3 | -2 | -1 | 2 |
| Flexibility and growth for each project manager needs | -3 | -1 | 2 | 3 |
| Reduce operating costs | 3 | 2 | 1 | 0 |
| Enforce security | 3 | 3 | 2 | 1 |

Legend: +3 means that the requirement is satisfied very well by this configuration

-3 means that the requirement is not satisfied very well by this configuration

The task force proposed to move most of the Project Data and PMS software from MVS to the UNIX environment and use the client/server architecture between the UNIX server and desktops (configuration 3). This solution was chosen even though it increased initial operating cost (had to buy a UNIX machine and C/S middleware software) and raised some security issues. However, the proposed solution provided maximum flexibility to the project managers so that they could run their own tools, bypassed the comptrollers and made the PMS system available directly to the project managers.

The main motivation for migrating the PMS data and programs away from MVS was to reduce MVS workload. For example, if the 300+ users were allowed to logon directly to MVS PMS system through terminal emulations or C/S middleware (instead of the 10 comptrollers who initially logged on to MVS PMS), it could cause response time problems for PMS users as well as other users of MVS (the MVS system supports many other production systems). In addition, setting up 300+ users to access the SNA network required additional hardware and software. With technical staff reductions at the central mainframe site, the increasing number of users could have created additional support bottlenecks.

Figure 2-22 shows a conceptual view of the proposed solution. According to the proposed architecture, a project manager can directly display, update, or print the official project information. The client software may run on UNIX workstations, PCs, MACs, and LAN servers. The administrative data, text data, and status information is migrated to a UNIX machine. A copy of the administrative data will be maintained on MVS to provide interface with other MVS-based financial systems through a very simple Project Tracker on MVS (this system only supplies project administration data to other systems). Due to the complexity of maintaining dual database synchronization, the MVS database for the text description and status information

will not be maintained. The UNIX-based PMS will interface with MVS for project number validation. The following processing steps were proposed:

- All users will invoke a client that will access the PMS system on UNIX. The UNIX PMS database will have the entire project description. The MVS PMS database only keeps valid project administration information (especially project numbers).
- If a new project is to be initiated, then the UNIX PMS will access the MVS project database for creation of a new project with associated administrative data (e.g., project number). This administrative data is sent back to the UNIX server and stored in the PMS UNIX database.
- If an existing project description needs to be reviewed or updated, then only the UNIX PMS database is accessed. There is no need for MVS database access.
- If a project is to be deleted, then the UNIX server will access the MVS database and delete the project administration data from MVS. The project is then deleted from the UNIX PMS database.

The proposed solution satisfies the migration requirements and also attempts to eliminate the involvement of the comptrollers and puts the project information directly in the hands of project managers. The initial cost to set up the new UNIX client-server application requires a one time upfront charge which involves hardware, software, and the database management system. The operational cost savings will take place thereafter.

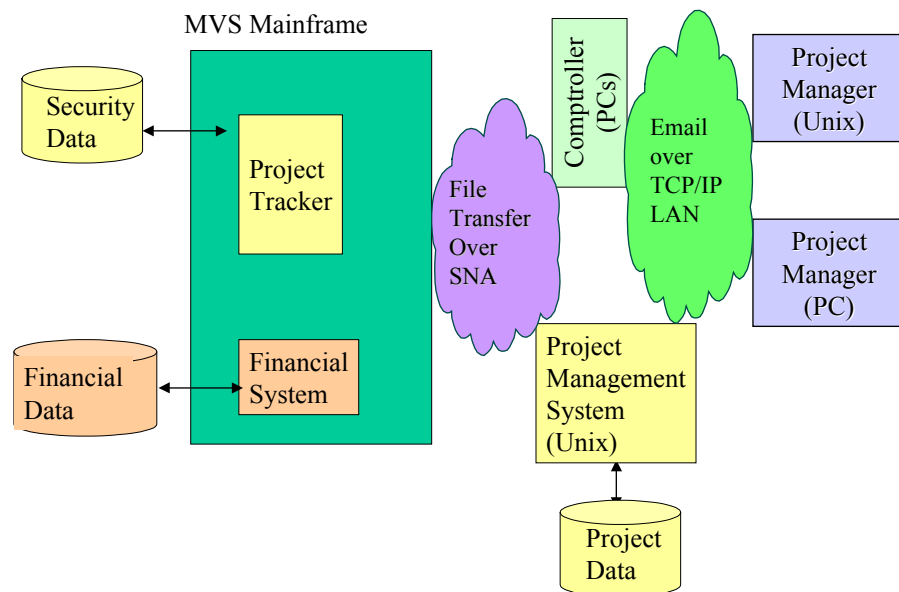


Figure 2-22: Proposed Project Management System

2.10.4 Step 3: Detailed Design and Implementation

Rough data partitioning and clustering was done before allocation (see Chapter 5 for a review of data partitioning and clustering). The data to be allocated was partitioned by regions because very few projects cross regions. Thus, 10 data partitions were created. The need for clustering of data between two regions arose because these two regions had several common projects. Thus the 10 partitions were reduced to 9 datasets (d1,d2,,,d9), where d9 is the clustered dataset from two regions. Although there are many computers in each region, each region designated a database server for the regional project database.

The data allocation problem is to allocate the datasets d1, d2,,,d9 to regions r1,r2,r3,,,r10. The following issues must be addressed:

- Should all datasets be allocated to one UNIX machine dedicated to project management?
- Should the datasets d1,d2,,,d8 be allocated uniquely to the regions r1, r2,,,r8? Is there a need for some data redundancy among these regions?
- Where should d9 be allocated? Should it be allocated to r9, r10 or both?

After some quick paper and pencil analysis, the first strategy was used because one UNIX machine could easily handle 300+ users plus house the 800 MB of project information. The UNIX machine is dedicated to this purpose and is henceforth referred to as the Project Management Server. The response time of each project manager is primarily a function of the data transmission time (the Project Management Server is a very powerful machine and is very lightly loaded). The Project Management Server is accessed through a TCP/IP Ethernet LAN (the transmission rate is 10 MBit per second at maximum). This would satisfy most performance requirements. However, if the TCP/IP LAN has significant network traffic and the Project Management Server is used more frequently, then the Server hardware should be upgraded and the network should be segmented for improved performance.

The PMS database will maintain the security access table and will be transparent to the MVS PMS system security. There will be additional security to authenticate each client. Since most of the users are linked to the TCP/IP LAN, the C/S middleware needed to connect the end-user tools to the remotely located PMS database must use the TCP/IP communication protocol. TCP/IP connection will need to be installed for those users who do not have TCP/IP connection. The Project Management Server must also interact with MVS Project Tracker for validation since the project number is created on MVS, and the project number interfaces with other existing MVS financial systems for validation.

The key point of this case study is that an MVS project database is migrated to UNIX based on the systematic methodology presented in this chapter. This case study shows how the business drivers were documented and how the decisions were made to conform to the business drivers. The result of this analysis produced a solution in which the project data is migrated to a UNIX machine with a copy of project administration information kept on the MVS machine for interfacing with other mainframe-based systems. The project managers could review and update the project status data by using their own desktop tools. However, they could not create new projects or delete existing projects without coordination with the mainframe system.

2.10.5 Review Questions

This case study is based on work that was done in the mid 1990s. Take the solution shown in Figure 2-22 as a starting point and migrate it to the latest MS Dot Net and Sun J2EE environments. Follow the same three-step migration procedure described above.

Hint: The procedure used in this case study is explained in detail in the "Migration" Chapter of the "Integration" Module.

2.11 Case Studies and Examples of Networks

2.11.1 ANX- Example of an Extranet

Extranets are IP networks that are jointly owned by corporations for conducting secure business processing. These networks use the same Internet technologies, however, the physical network is collectively owned by corporations to meet the security and reliability requirements imposed by the owners. An example of Extranet is the Automotive Network eXchange (ANX) network initially formed by manufacturing corporations (GM, Ford, Chrysler, and others). ANX was formed by the Automotive Industry Action Group (AIAG) to provide a common communication infrastructure among automotive Trading Partners. ANX is intended for North America initially with plans to expand worldwide to over 3000 trading partners. The drivers for ANX are control of communications cost with predictable service quality, support of a common set of applications (e.g., EDI, database lookup, Web, email, and Computer-Aided-Design (CAD) file transfer), and smooth introduction of new applications (e.g., videoconferencing, interactive CAD) using the Internet technology.

ANX was initially designed by Telcordia Technologies and is currently operated as a business by Telcordia owners (SAIC). The business, called ANXe-business Corp (<http://www.anxo.com/>), provides the ANX Network for mission critical business transactions for aerospace, automotive, chemical, electronics, financial services, healthcare, logistics, manufacturing, transportation and related industries. In July, 2001, Mitsubishi Corp. purchased a 20% stake in ANXe-business and decided to use the ANX exchange network for its global trading strategy.

From a user (subscriber) point of view, ANX is a Virtual Private Network (VPN) that is operated and maintained by a Certified Exchange Point Operator (CEPO). The CEPO has business relationships with many ANX Certified Service Providers (CSPs) that are the entry points for the subscribers (trading partners). The public Internet users are connected to the ANX through a special CSP. The architecture of ANX is shown in Figure 2-23.

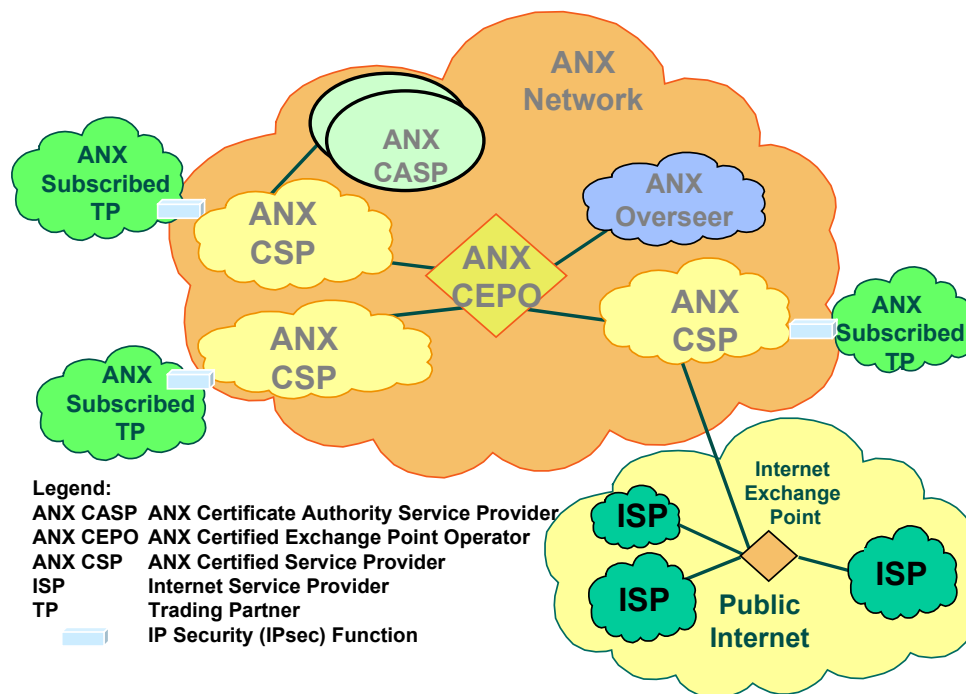


Figure 2-23: ANX Extranet Architecture

The ANX Network, at the time of this writing connects more than 900 companies and delivers multi-provider solutions for fast and secure e-business. The network reportedly exceeds 99.93% end-to-end availability rate, with network latency averaging less than 50 milliseconds. ANX employs Asynchronous Transfer Mode (ATM) broadband switching and transmission technology to ensure reliable, high-speed video, voice and data communications. The ANX Certified Exchange Point Operator (CEPO) operates and maintains the ATM network, and acts as a virtual switchboard, routing incoming subscriber transmissions to the designated trading partners.

As shown in Figure 2-23, subscribers connect to the ANX CEPO via ANX Certified Service Providers (CSPs). ANX CSPs are connected to the ANX CEPO's primary ATM switch via T3 lines and to the back-up switch via T1 lines. ANX CSPs are fully interoperable with each other and offer dial-up, DSL, ISDN and dedicated connectivity. Each subscriber to ANX is provided with an Internet Protocol Security device that resides on the subscriber's network. These devices create secure encrypted tunnels -- Virtual Private Networks (VPNs) -- that subscribers use to communicate securely. VPNs ensure both the identity of the sender and the integrity of the data. The network is intended to support a Public Key Infrastructure service to further enhance security via digital certificates.

How is an extranet like ANX being actually used? Here are some examples, extracted from the ANX site (<http://www.anxo.com/>):

- Siemens Automotive, a Tier-one supplier of automotive and electrical/electronic systems and components, had only dial-up electronic communication capabilities -- in most cases to a maximum of 28.8kbps -- to some of its customers. The company was sending CAD (Computer Aided Design) data tapes via postal mail and courier to customers where electronic connections did not exist. Additionally, Siemens' branch locations across Canada and the United States had connectivity solutions to individual customers, yet shared customers would require separate communication links. This required ongoing support and extra costs at the local level. This approach was not adequate for a global company such as Siemens that employs more than 447,000 people in 193 countries. For example, an engineer would cue up CAD files to send in the evenings, (to save on long distance charges as transfers would take hours), only to find in the morning that the transfer(s) had failed and needed to be re-sent. Due to these problems, Siemens subscribed to ANX for file transfers. In addition, Siemens now has access to multiple applications at its customer sites, which allow for realtime collaboration online. One of the first applications that Siemens deployed over the ANX Network was "IMI", Ford's design development collaboration solution.
- Dofasco, Inc., producer of flat rolled steel products, is using ANX for a faster, more reliable exchange of EDI transmissions with reduced total operating cost. An IP-based VPN, similar to ANX, was installed to exchange files directly via FTP but was replaced with the ANX network in 1999, which is now used to exchange 60% of all EDI transactions. Due to the extranet (jointly owned) as compared to public Internet, special pricing and quality of service (QoS) arrangements can be made.
- Ricardo Inc., an engineering technology provider for the automotive manufacturers, needed to facilitate the exchange of large volumes of data with its suppliers and customers located around the world. Computer Aided Design (CAD) files, running as large as 5 or 6 gigabytes in size, were required to be delivered and returned on a routine basis by using various media types (tapes, CDROMS, etc.) through courier and express delivery services. These services averaged more than two days in turnaround. Ricardo proceeded with the installation and testing of a low speed dedicated ISDN line direct to their supplier in Mexico, but encountered major labor-intensive and cost-prohibitive maintenance issues in the administration of a router in a different language and country. Additionally, even when the line was functioning satisfactorily, it was unable to accommodate the larger data files, which still needed hard-copy courier or express service delivery. Turnaround became such an issue to Ricardo that the full-time employment of on-staff couriers was seriously considered. Due to these problems, Ricardo became one of the first suppliers to join the ANX community and installed a T3 with ANX in the fall of 1999. At present, Ricardo uses the global ANX Network for connectivity to customers and suppliers based in Mexico, Germany, Italy, Sweden, Australia and Canada.

Review Questions

- What type of services are provided by ANX?
- What is the value of these services to the businesses?
- How can ANX be used for e-commerce applications, instead of the file transfers? For example, how could on-line purchasing and integrated manufacturing be supported over the ANX network?
- Determine and evaluate some of the ANX services such as ANXVelocity and IMI.
- Produce a detailed network architecture of ANX that shows the ATM backbone and the connections from trading partners (subscribers) to the CSPs.

Hints: The ANX Website has a large amount of information that can be used to answer these questions.

2.11.2 An Enterprise Network Example

Enterprise networks consist of many technologies, devices, protocols, and interfaces supplied by a multitude of vendors. Network architecture standards are needed for compatibility of inter-vendor and intra-vendor network products and services. Figure 2-24 shows enterprise network of a telecom company. In this network, like many other enterprise networks, a corporate WAN (wide area network) is connected to many

subnets through routers and gateways. This enterprise network consists of three types of offices: a corporate business office that uses IBM's mainframes and IBM SNA (System Network Architecture) for business partners, one or more engineering sites that use TCP/IP over Ethernet and a few marketing offices that use Novell and Token Ring. The WAN uses TCP/IP (i.e., TCP/IP is the native protocol stack). The subnets use routers to access the corporate WAN. If a subnet uses TCP/IP also, no protocol converters (gateways) are needed. However, a Novell IPX/SPX subnet will need to use a TCP/IP to IPX/SPX gateway, in addition to a router, to communicate with the TCP/IP WAN. The routers are important in this network because they can recognize the destination and route the messages to the destination. Most of the routers in this configuration would support multiple protocols. The gateways would convert different protocols (e.g., Novell SPX/IPX to TCP/IP, SNA to TCP/IP). It may be possible to purchase one "box" which provides routing as well as gateway functions, or to install router/gateway software on an NT or SUN workstation. The subnets may utilize FDDI, Ethernet or token ring communications technologies. The corporate backbone is typically a wide area network that may utilize frame relay, ATM or X.25 communications technologies.

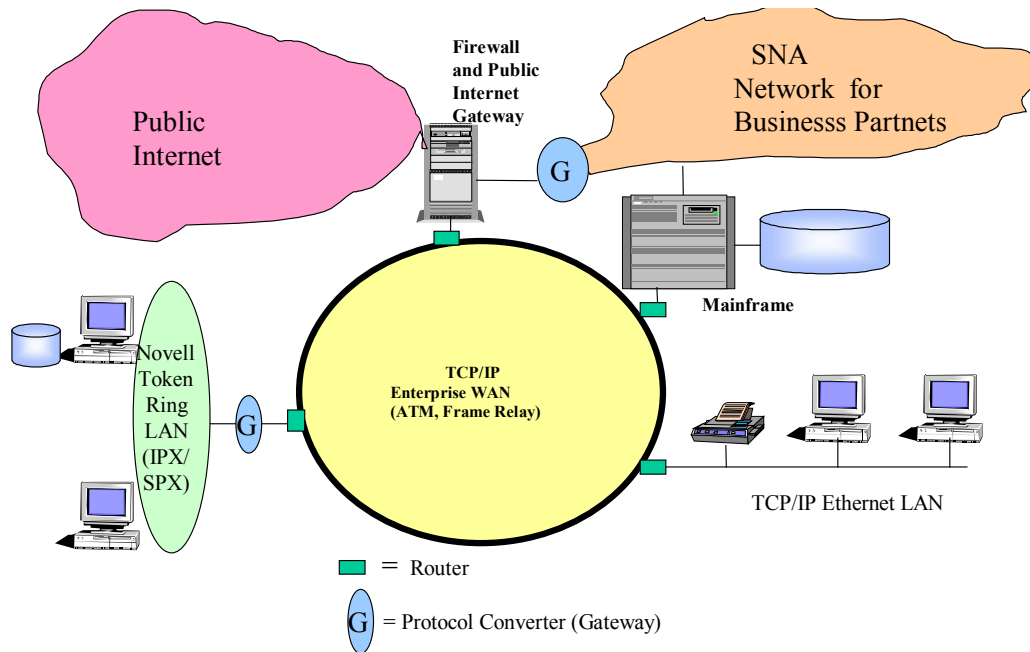


Figure 2-24: A Typical Enterprise Network

Note that the mainframe is connected to SNA as well as the corporate TCP/IP WAN. The SNA network is connected directly to the TCP/IP WAN through network gateways that convert protocols between SNA and TCP/IP. In addition, access to public Internet is provided through an "internet gateway" that may contain firewall software. These gateways can become security, availability and performance bottlenecks. As mentioned previously, many network gateways use UNIX or NT platforms. These gateways must be kept in secure environments because a gateway user can see everything being transmitted through it. Alternate gateways must be available if one gateway crashes. Many network gateways have trouble sustaining more than 100 sessions, thus many gateways may be needed for improved performance and load balancing.

This example illustrates the device to network and the network to network interconnectivity scenarios. Application interconnectivity is handled by the higher level layers at the end systems. Two applications at two different end systems, independent of the network interconnectivity option chosen, need to agree on the format and rules of message exchange. When this is not easily done, application gateways are introduced in the network. For example, two subnets may use different Email systems. However, to exchange mail from one subnet to another, a mail gateway may be needed with security features.

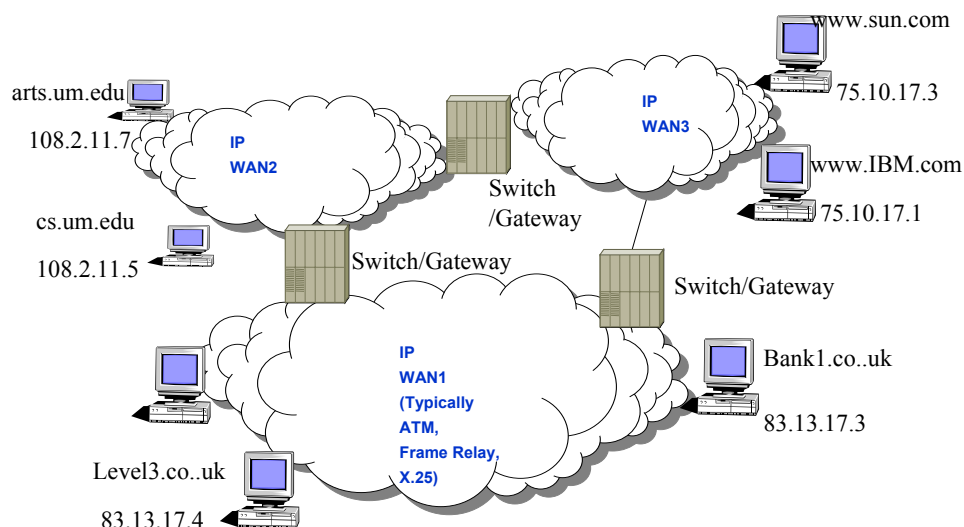
Review Questions

- What would this enterprise network look like if all networks used TCP/IP (i.e., no SNA or Novell IPX/SPX)?
- What would this network look like if the Ethernet LAN and the Novell IPX/SPX LANs were replaced with wireless LANs (Wireless Ethernet and Bluetooth, respectively)?
- Supposing this company is planning to use ANX instead of the SNA partner network. What would the network look like in this situation?

Hints: These questions can be easily answered by reviewing the "Network Architecture" Chapter in the "Networks" Module.

2.11.3 Example of Public Internet

Figure 2-25 shows a conceptual and partial view of Public Internet. This Internet shows three networks (a university network with two computers, a commercial company network, and a network in UK). Each computer ("host") on this network has an IP address and also has been assigned a domain name. Internet is very heterogeneous (i.e., different computers, different physical networks.) However, to the users of this network, it provides a set of uniform TCP/IP services (TCP/IP hides many details). We will use this simple Internet to illustrate the key Internet capabilities.



- DNS (Domain Name Services) translates cs.um.edu to 108.2.11.5
- Telnet cs.um.edu = Telnet 108.2.11.5
- FTP cs.um.edu = FTP 108.2.11.5

Figure 2-25: Partial View of Public Internet

Since the Internet is based on TCP/IP, the applications and services provided by TCP are also available on the Internet. From an end-user point of view, the following services have been, and still are, used very heavily on the Internet:

- Email
- Telnet
- FTP

Electronic mail on the Internet is based on the **Simple Mail Transfer Protocol (SMTP)**. This TCP based protocol is the Internet electronic mail exchange mechanism. Email is still one of the most heavily used services in the Internet. Users on the Internet have email addresses such as johnm@cs.um.edu, hevner@sun.com and howard@bank1.co.uk.

Terminal emulation is used to remotely logon to other machines. **Telnet** is used to provide terminal access to hosts and runs on top of TCP. Let us assume that a user "joe" on cs.um.edu needs to remotely logon to the bank1.co.uk machine to run a program "directory". The user would use the following steps (the steps are explained through comments in /* */):

```
> telnet bank1.co.uk      /* invoke Telnet. Could have typed "telnet 85.13.17.3" */
bank1> enter login: joe    /* prompt from bank1 for login ID. joe is ID */
bank1> password: xxxx     /* prompt from bank1 for password */
bank1> directory          /* run the program "directory" */
bank1> exit /* quit telnet */
```

File transfer is used for the bulk of data transfer over the Internet. The **File Transfer Protocol (FTP)** provides a way to transfer files between hosts on the Internet. Let us assume that a user "garner" on "sun.com" needs to transfer a file from the host arts.um.edu. The following steps would be used (the steps are explained through comments in /* */):

```
> ftp arts.um.edu        /* invoke FTP. Could have typed "ftp 102.52..10.7" */
arts> enter login: garner /* prompt from arts.um for login ID. garner is ID */
arts> password: xxxx     /* prompt from arts.um for password */
arts> get file1 file2     /* FTP file transfer command */
arts> exit (or quit) /* quit FTP */
```

For many years, Internet had been used mainly by researchers, teachers, scientists, students, and programmers to transfer files and send/receive electronic mail. These users relied on text-based commands to do their job. WWW is a set of services that run on top of the Internet. The two main features of WWW are use of GUI and hypertext to make the life of Internet users easy and fun. We will discuss WWW in more detail in the next section.

We should mention that the users access the Internet either directly or indirectly. **Direct Internet users** reside on the machines that have IP addresses while **indirect Internet users** remotely logon to the machines that have IP addresses. For example, America Online is an Internet Access Provider that actually has machines with IP addresses (direct access). If you subscribe to America Online, then you dial into an America Online machine (i.e., you are indirectly accessing the Internet).

Figure 2-26 shows an example of how a student from Drexel university in Philadelphia can exchange email with another student in Izmir (Turkey). It shows the various networks, gateways, and email servers that participate in email exchange.

Review Questions

- Look at the network of a network service provider such as UUNet and draw a network configuration diagram for that network.
- Look at the network of an ISP (Internet Service Provider) and draw a network configuration diagram of that network.
- Show the network configuration of a VPN (Virtual Private Network).

Hints: These questions can be answered by visiting Web sites of ISPs/NSPs and by reviewing the "IP Networks and the Internet" Chapter in the "Networks" Module

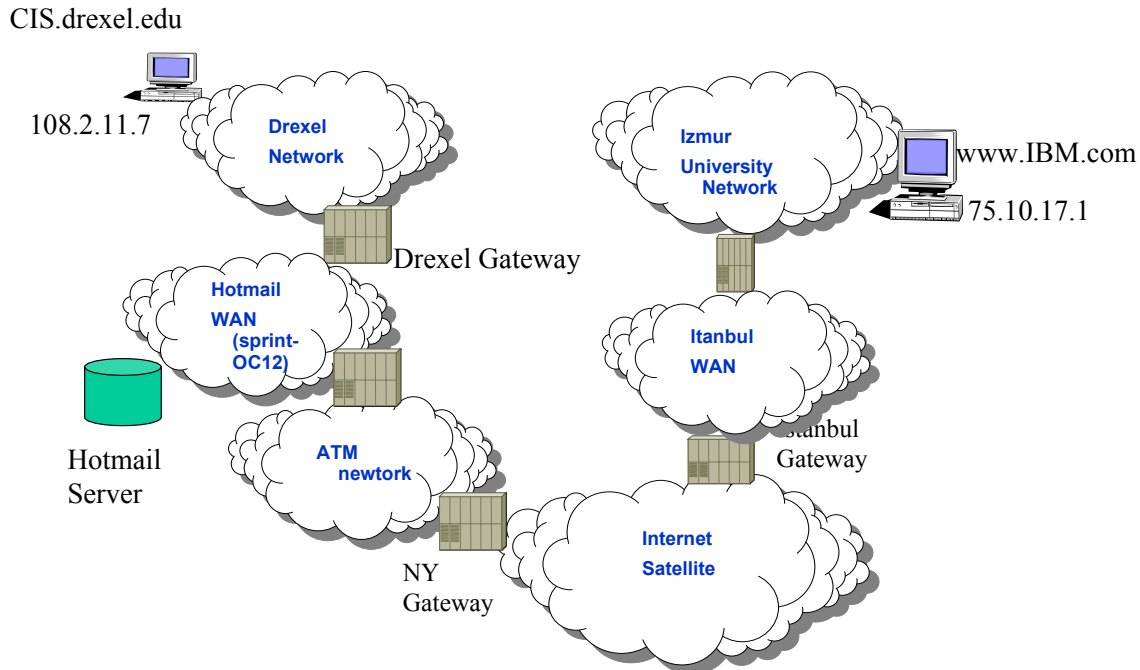


Figure 2-26: An example of IP connectivity from Drexel (Philadelphia) to Izmir (Turkey)

2.12 Older Case Studies – A Fresh Look

A large number of case studies that describe different approaches to deal with legacy applications were published in the mid 1990s because at that time the client/server paradigm challenged the old mainframe systems. But all these solutions are legacies at the time of this writing (circa 2002). It is interesting to revisit some of these case studies at this time and decide how we would deal with these issues *now*.

2.12.1 Hewlett Packard's Migration to Client/Server Architecture

Hewlett Packard (HP) is an international corporation with more than 90,000 employees and annual revenues exceeding \$16 billion. HP initiated an effort to move its terminal-host legacy information systems to client/server applications in 1989. This move was driven by the following business drivers:

- The rapidly changing business requirements required flexibility and end-user control which was not possible by the terminal-host information systems (host systems required lengthy procedures to produce new reports and/or to add new capabilities)
- The HP management needed realtime access to operational data (e.g., information about the status of business). This data was embedded in "vertical" applications which made it very difficult to access cross-application data.
- The global customers expected HP to act and look the same anywhere in the world, while the local customers in different countries needed different "local" views to support local legal and competition requirements. This two-level view is very difficult to maintain with terminal-host systems.

Figure 2-27 shows the application architecture proposed to satisfy the stated requirements. The architecture consists of user interfaces, user task logic, data services, and business transaction logic. The process/task clients contain the logic that performs the processes/tasks needed by the user. This client contains user task logic in addition to the user interface handlers. The data servers provide the clients with access to one or more databases. Ideally, a database contains a single data subject -- a collection of closely related data such as customer, product and order. Business transactions, defined in the business data model and managed in

the data dictionary, are the means for communications between clients and servers. HP clients communicate with HP servers by using business transactions. The clients essentially invoke these business transactions through messages which are sent over the HP network.

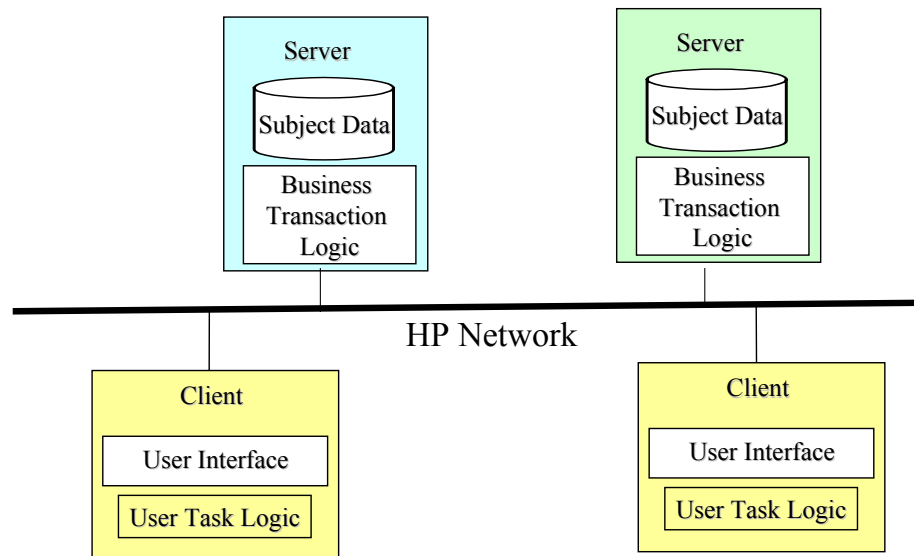


Figure 2-27: Hewlett-Packard's Client/Server Architecture

The migration from legacy terminal-host applications to this C/S application architecture took two forms: leverage and design from scratch. HP decided to leverage the operational applications and redo the obsolete applications (the literature does not clarify the exact business decisions).

To leverage terminal-host applications, the application structure and logic were carefully analyzed. Typical applications consisted of several hundred thousands lines of COBOL code which incorporated program logic, screen handling and database access calls in large monolithic applications. The leveraging activity consists of the following steps:

- The existing database is retained but a server is put around it.
- The server logic is built by reviewing the legacy application code and translating the useful part of this code into business transactions. The useful portion of the code is "cut" from the monolithic application code and used to build the server. HP claims to reuse up to 60% of the old code by using this technique.
- The screen handling code is scrapped and rebuilt for GUI client interfaces. The client code is typically 30% of the C/S application code. The clients take advantage of 4GLs, data browsers, and GUI tools.

The design from scratch decision was made for those applications which could not be leveraged. HP used the "information engineering" methodology for the new development. According to this framework, the following activities are performed in sequence: an information strategy is planned, a data model is constructed, a data flow diagram is developed, database schemas and software design are built, and the database as well as software modules are implemented. CASE tools and a repository were used to develop these applications. The following rules were used to cast this model into a C/S application architecture: a) the client application logic automates the processes from the DFD, b) the business transaction messages are the data flows in the data flow diagram, c) the business transaction logic is built from the data relationships in the ERA diagrams, and d) the data manager is constructed by using the database schema. This transition spanned three years (1989-1992). The main steps of this transition are:

- At the start of the project in early 1989, about six people were assigned for six weeks to experiment with and learn the C/S technology. No specific deliverables were assigned to them.
- After the experimentation, a task force was formed to develop a C/S application and platform architecture.

- A structured pilot was conducted to test the C/S architecture developed by the task force. Another team of six people was assigned to this task for six weeks. This team built three clients and three servers of different types (PC and UNIX workstations accessing relational and non-relational data servers).
- In February of 1990, a major terminal-host application was "leveraged" to C/S. This was accomplished in eight months by three teams: a client team, a server team and a network team. After initial interactions needed to describe business transactions, the three teams worked somewhat independently and reportedly finished this conversion on time.
- In the summer of 1990, a C/S application was designed from scratch in six months.
- In the spring of 1991, the C/S architecture was standardized for company-wide deployment after a few modifications.
- In May of 1992, applications were developed to demonstrate the reusability of the client/server components to build new applications.

During this migration, HP invested in hardware, software, telecommunications, and staff training. New skills for developers, as well as data center staff, were needed. In addition, standards and guidelines were developed for development methodologies and platforms.

Review Question. What could this solution look like if it was done today. What issues and solution approaches are fundamentally the same and what are different? In particular, translate Figure 2-27 into the latest IT infrastructure (wireless networks, mobile users, Web-XML, Dot Net, J2EE).

2.12.2 Off-Shore Reengineering

Dedene and DeVreese [Dedene 1995] describe how two large scale legacy applications from Sidmar Steel and Catholic University of Leuven, both in Belgium, were reengineered by using off-shore software houses. The results of this experience are positive. We describe the experience with Sidmar Steel. The interested reader should review the original source for additional information.

Sidmar Steel had a large scale mainframe-based application consisting of more than 7000 programs that managed everything from production to finance and sales. The application originated in 1965 and had evolved into a complex system that utilized IBM 370 assembler code, PLI programs, and IMS and DB2 databases. The developers estimated that it would require approximately 145 staff years to reengineer this application. The size of the effort eliminated the in-house conversion as a viable option (the company had only 120 qualified staff who could perform the conversion). Outsourcing the job to a European, preferably Belgian, software house was eliminated due to prohibitive costs (these houses typically use time and material-based contracts which are expensive and "open-ended"). A Philippine software house was selected, after reviewing the bids from many off-shore software houses, due to its cost, availability of skilled talent (the software house was associated with a nearby university computer science department), convenient time differences (Philippines staff could work on conversion while the Belgian staff slept and vice versa), and language (English).

A satellite link was established between the Philippine software house and Sidmar. The Philippine software house's computers acted as terminals thus eliminating need for any additional computing resources on either side. This arrangement eliminated many logistic hurdles (e.g., the software was converted and tested on the machines and computing environments where it was intended to operate). The project ended on time and within budget; the working relationship was quite satisfactory; and the conversion quality was quite high. Mainly, the computer link was key to the project success and the Philippine software house saved Sidmar 35% compared with US or European software houses.

Review Question. How would this re-engineering work today? Make specific changes to the solution approach by using the latest IT infrastructure. Are there approaches and technologies available today that make outsourcing easier? What are they?

2.12.3 University of Florida Reengineers via the World Wide Web

The University of Florida changed its registration process in the mid 1990s by moving away from the old central registration system to a self-service system that uses voice response units and the World Wide Web. In the new system, the students do not have to wait in line at the start of each semester to register for courses, pay tuition, and buy books. Instead, they use the Web technology in a self-service mode by using their desktops.

The University of Florida (a 40,000 student and 20,000 employee university) was running all applications on an IBM4300 using COBOL. Students and administrators accessed the system through dumb, text-based terminals. The university decided to first adopt a client/server strategy and started developing applications in INTERSOLV's tools that ran on mainframes as well as on PCs. Student registration and many other administrative applications are being developed using C/S model.

The student registration system generates student ID cards by using a client/server model (the student data and business logic are kept at the mainframe and the students' digitized pictures reside in the workstations). The system currently uses the Web technology and voice response units (where needed) to allow students to register for courses and pay tuition without having to stand in line. The Web browsers especially eliminate the user interface problems for on and off campus registration. The University hands out free Netscape software to students for installation on their home PCs to promote the use of Web technology.

Review Question. What could this solution look like if it was done today? Show a possible architecture by using the latest Web and Internet technologies.

2.12.4 3Com's Data Warehouse

3Com Corporation is a California-based manufacturer of global data networking products. Networking is a fast moving, very competitive industry requiring fast response to varying global market conditions. Like many other similar industries, 3Com has experienced problems in using its data. Here are some examples:

- Top level executives demanded that reliable sales and marketing information be available to make production decisions.
- Monthly sales reports obtained from different departments did not match.
- Reports were more suitable for operational day-to-day support instead of trend analysis and decision support.
- The business analysts needed an informational environment that integrated well with the desktop tools they were already using - spreadsheets, word processors, and data browsers.
- IS professionals were spending most of their time on writing one-time reports of sales and marketing for business analysts.
- Many reports were too general and larger than the analyst needed for any single analysis.

3Com developed an information architecture that consisted of separating the operational systems (currently running on HP platforms) from a data warehouse (a new system developed on DB2). The data warehouse is accessed from IBM PCs, Macs, and Sun workstations through SQL browsers, spreadsheets, and an EIS.

An initial prototype of the data warehouse was established for proof of concept as well as internal marketing of the concept. The prototype tested the technical as well as administrative aspects of the new environment and measured the impact of the data warehouse traffic on the corporate backbone. The prototype was accepted for production after four months. A production methodology was followed that included a team of IS and business analysts; hiring of an experienced data modeler to lead the data modeling effort; rewriting of the extract/load program; extensive data warehouse design effort to build summaries and views; implementation of on-line help; and development of a training program.

The data warehouse was implemented in 5 months. The data warehouse consisted of 5.5 GB of data. Due to the data warehouse, top managers can access recent sales information, business analysts have discovered

some shifts in business practices, and requests for paper reports have decreased significantly. Several direct quotes from 3Com management are reported in the case study to indicate management satisfaction. It is noted that the data modeling effort and the summarization process could have been initiated during the prototyping process.

Comments: This case study represents a very common approach taken in developing data warehouses. The lessons learned from this undertaking are typical of many other data warehousing projects (e.g., complete separation of operational and decision support systems is essential, starting with a small prototype is a good idea, development of a sound data model is essential, and proper hardware/software estimates are important before embarking on large scale production).

Review Question. What could this solution look like if it was done today? Show a possible architecture by using the latest IT infrastructure. Also show how this data warehouse can be used for data mining.

2.12.5 British Telecom's Data Warehouse

British Telecom (BT) has established a data warehouse for its Group Logistics and Group Procurement Divisions. The warehouse is a single centralized database that contains all the data needed to support the management and delivery of BT's supplies in the UK.

The primary business objectives of this data warehouse were to provide stock visibility to all parts of the organization and thus reduce stock, reduce lead times, and increase efficiency by reducing paperwork. To meet these objectives, BT embarked on a three year task of separating data from operational systems, and developing a data model for the logistics and procurement business functions. The project required an investment of roughly \$4.5 million and required involvement of 124 BT staff members.

The implemented data warehouse is not used as an extract database -- it involves realtime updating of the database. The data warehouse resides on an MVS-DB2 mainframe. It handles 850,000 orders per month, holds information on 73,000 item records and 100,000 contracts with 26,000 suppliers, and dispatches goods to more than 10,000 customer delivery points. The database is divided into three categories: reference databases (for item, customer, supplier); transaction databases (order and purchase order); and application databases (stock, contracts). Outside suppliers are connected to the data warehouse through Electronic Data Interchange (EDI) -- this allows switching of orders to outside suppliers if a needed item is not available in BT inventory.

This project has reportedly caused a \$600 million reduction of stocks, a reduced requirement for inventory warehouses, reduction of repetitive clerical activities, better cost control, and reduced lead time for some products (e.g., delivery time cut from 10 days to 3.5 days for chloride batteries). The biggest benefit to BT is that the data is held in one central site and not in multiple locations. This provides total stock visibility, facilitates switching of orders to the most appropriate stock holding location, and enforces standardization to allow systems accessing the data to be organizationally independent.

Comments: This case study shows implementation of a centralized data warehouse. Unique features of this case study are:

- More business benefits due to centralization of information for total stock visibility. The benefits of centralization are not usually mentioned nowadays.
- Long range (3 year) effort as compared to short (3 to 4 months) data warehousing efforts.
- The data is updated (most data warehouses are retrieval only).

Review Question. What could this solution look like if it was done today? Show a possible architecture by using the latest IT infrastructure. Will you still use a centralized data warehouse? If not, what has changed to make a different architecture more appealing? Also show how this data warehouse can be used for data mining.

2.13 References

- [1] Huff, S., et al, "Cases in Electronic Commerce", Irwin-McGraw Hill, 2000.
- [2] Harmon, P. and Morrissey, W., "The Object Technology Casebook -- Lessons from Award-Winning Business Applications", John Wiley, 1996.
- [3] Kalakota, R. and Robinson, M., "e-Business 2.0", Addison Wesley, 2001.
- [4] Severance, D. and Passino, J., "Making I/T Work", Jossey-Bass, 2002.
- [5] Telcordia Technologies Handbook, "E-commerce Case Studies", 1999.

MODULE INDEX(visit www.amjadumar.com for index of the entire handbook)

- Accounting Management, 1-44
- Amazon.Com, 2-1, 2-8
- Amp Inc, 2-1, 2-8, 2-9
- ANX-, 2-2, 2-42
- Application, 1-7, 1-8, 1-28, 1-31, 1-38, 1-39, 1-40, 1-43, 1-46, 1-48, 1-49, 2-1, 2-3, 2-11, 2-28, 2-29, 2-31, 2-45
- Application Architecture, 1-28, 1-36, 1-38, 1-39, 1-40, 1-47, 2-1, 2-6, 2-28, 2-48, 2-49
- Application Reengineering, 2-2, 2-50
- Application Servers, 1-12, 1-29, 1-31, 1-39, 1-43, 2-28, 2-33, 2-37
- Architectures, 1-2, 1-12, 1-17, 1-18, 1-19, 1-23, 1-31, 1-32, 1-37, 1-38, 1-39, 1-40, 1-41, 1-44, 1-45, 1-46, 1-47, 1-49, 2-2, 2-6, 2-7, 2-9, 2-10, 2-11, 2-12, 2-13, 2-17, 2-20, 2-24, 2-25, 2-26, 2-27, 2-28, 2-30, 2-31, 2-32, 2-33, 2-34, 2-35, 2-36, 2-37, 2-40, 2-43, 2-44, 2-45, 2-48, 2-49, 2-50, 2-51, 2-52
- Authentication, 1-29, 2-20, 2-21, 2-22, 2-25, 2-26, 2-27, 2-34, 2-42
- Auto-by-Tel, 2-1, 2-5
- B2B, 1-19, 1-22, 1-25, 1-26, 1-27, 1-28, 1-29, 1-31, 1-32, 1-34, 1-35, 1-39, 1-40, 1-43, 1-47, 2-1, 2-17, 2-18, 2-20, 2-21, 2-22, 2-24, 2-27, 2-29
- B2B Integration, 1-19, 1-22, 1-25, 1-26, 1-27, 1-28, 1-29, 1-31, 1-32, 1-34, 1-35, 1-39, 1-40, 1-43, 1-47, 2-1, 2-17, 2-18, 2-20, 2-21, 2-22, 2-24, 2-27, 2-29
- B2B Servers, 1-39, 1-43
- B2N, 1-23, 1-25, 1-27, 1-28, 1-32
- Bluetooth, 2-45
- Boo, 1-14, 1-15
- Boo.com, 1-14, 1-15
- Book Reference Model, 1-2, 1-5, 1-36, 1-37, 1-38, 1-41, 1-42
- British Telecom, 2-2, 2-52
- Business Component, 1-3, 1-31, 1-39, 2-32, 2-37, 2-38
- C2B, 1-19, 1-22, 1-25, 1-26, 1-27, 1-31, 1-34, 1-35, 1-39, 1-47, 2-1, 2-8, 2-15, 2-17, 2-20
- Catalogs, 1-12, 1-19, 1-33, 1-43, 1-48, 2-8, 2-10, 2-11, 2-15, 2-16, 2-17, 2-18, 2-19, 2-20
- CDMA, 2-28
- Cellular Network, 1-41, 2-28
- Charles Schwab, 2-1, 2-5
- Cisco -, 2-1, 2-11
- Client/Server, 1-43, 1-49, 2-2, 2-40, 2-42, 2-48, 2-49, 2-50, 2-51
- COBOL, 2-49, 2-51
- Collaboration Servers, 1-39
- Commerce Servers, 1-39, 1-43
- Components, 1-5, 1-9, 1-31, 1-38, 1-40, 2-38
- Connectivity, 1-42, 2-28, 2-29, 2-34, 2-43, 2-44, 2-48
- Control, 1-6, 1-14, 1-23, 1-25, 1-26, 1-37, 1-44, 2-16, 2-17, 2-18, 2-19, 2-31, 2-32, 2-39, 2-42, 2-48, 2-52
- CORBA, 1-31, 1-39, 1-43, 1-45, 2-11
- Cryptography, 1-44
- Customer Relationship Management (CRM), 1-26, 1-27, 1-48, 2-2, 2-33, 2-34, 2-35, 2-36, 2-37
- Customer Support, 1-9, 1-10, 1-14, 1-45, 2-16
- Cybercash, 2-14
- DBMS, 2-9
- DCOM, 1-31
- Dell Computers, 1-3, 1-4, 1-8, 1-9, 1-10, 1-23, 1-25, 1-35, 2-1, 2-5
- Digital Certificates, 2-13, 2-43
- Digital Signatures, 2-21, 2-25
- DISCUS, 2-1, 2-11
- Distributed Operating Systems, 1-43
- Distributed system, 1-2, 1-3, 1-5, 1-6, 1-7, 1-22, 1-29, 1-30, 1-31, 1-32, 1-49
- eAI, 1-38, 1-40, 2-4
- EAI, 1-38, 1-40, 2-11
- EAI Platform, 2-11
- e-Business, 1-1, 1-2, 1-3, 1-4, 1-5, 1-7, 1-8, 1-9, 1-14, 1-22, 1-23, 1-31, 1-32, 1-37, 1-38, 1-39, 1-39, 1-47, 1-48, 1-49, 2-1, 2-3, 2-4, 2-6, 2-30, 2-53
- e-Business application integration, 1-38, 1-40, 2-4
- e-Business Applications, 1-2, 1-38, 1-39
- e-Business Evolution, 1-2, 1-22
- e-Business Management, 1-39
- e-Business Patterns, 2-31
- e-commerce, 1-3, 1-4, 1-9, 1-10, 1-11, 1-12, 1-19, 1-22, 1-24, 1-31, 1-43, 1-47, 2-1, 2-14, 2-15, 2-17, 2-29, 2-30, 2-31, 2-44
- EDI, 1-25, 2-9, 2-10, 2-12, 2-20, 2-42, 2-44, 2-52
- EJB, 1-39, 2-37, 2-38
- eMarkets, 1-8, 1-27, 1-35, 2-20, 2-25, 2-26, 2-27, 2-29
- Enterprise application integration, 1-38, 1-40, 2-11
- ERP, 1-5, 1-25, 1-26, 1-27, 1-39, 2-3, 2-4, 2-6, 2-33
- Fault Management, 1-7, 1-37, 1-44
- Furniture.com., 1-13
- Gateway, 2-44, 2-45
- GE, 1-2, 1-3, 1-8, 1-23, 1-25, 1-26, 1-27, 1-33, 1-34, 1-35, 1-47
- GSM, 2-28

MODULE (OVERVIEW)

GUESS Jeans, 1-16, 1-19, 1-36
Hewlett Packard, 2-2, 2-48
HTML, 1-40, 2-15, 2-34
HTTP, 1-31, 1-43, 2-15, 2-20, 2-24
IBM, 1-3, 1-4, 1-8, 1-9, 1-15, 1-31, 1-44, 1-45, 2-4, 2-5, 2-7, 2-15, 2-31, 2-36, 2-38, 2-44, 2-50, 2-51
I-Mode, 1-43
IMS, 1-45, 2-50
Integration, 1-2, 1-5, 1-7, 1-15, 1-16, 1-18, 1-19, 1-22, 1-23, 1-27, 1-28, 1-31, 1-32, 1-34, 1-36, 1-37, 1-38, 1-39, 1-40, 1-41, 1-46, 1-47, 2-2, 2-3, 2-4, 2-5, 2-6, 2-11, 2-12, 2-20, 2-31, 2-32, 2-33, 2-34, 2-35, 2-36, 2-37, 2-38, 2-42
Internet, 1-3, 1-4, 1-5, 1-6, 1-7, 1-8, 1-9, 1-10, 1-11, 1-12, 1-14, 1-15, 1-19, 1-22, 1-23, 1-24, 1-25, 1-26, 1-29, 1-31, 1-32, 1-33, 1-34, 1-37, 1-38, 1-39, 1-40, 1-41, 1-45, 1-47, 1-48, 1-49, 2-2, 2-5, 2-9, 2-12, 2-13, 2-14, 2-15, 2-17, 2-20, 2-24, 2-26, 2-29, 2-35, 2-42, 2-43, 2-44, 2-45, 2-46, 2-47, 2-51
Intrusion Threats, 1-44
IP, 1-31, 1-39, 1-41, 1-42, 2-24, 2-28, 2-33, 2-35, 2-41, 2-42, 2-44, 2-45, 2-46, 2-47, 2-48
ISO/OSI, 1-42
IT Infrastructure, 1-2, 1-6, 1-7, 1-15, 1-16, 1-19, 1-23, 1-25, 1-26, 1-29, 1-32, 1-35, 1-36, 1-37, 1-38, 1-39, 1-40, 1-44, 1-45, 1-47, 2-3, 2-5, 2-6, 2-8, 2-29, 2-30, 2-50, 2-52
J2EE, 1-31, 1-39, 1-43, 2-37, 2-42, 2-50
Java, 1-3, 1-39, 1-45, 2-22, 2-33, 2-35, 2-37
JDBC, 2-37
JMS, 2-35, 2-37
JSP, 2-37
Legacy applications and systems, 1-38, 1-40, 2-6
Management, 1-2, 1-7, 1-15, 1-15, 1-18, 1-19, 1-23, 1-29, 1-35, 1-37, 1-39, 1-44, 1-47, 1-48, 2-1, 2-2, 2-3, 2-6, 2-10, 2-13, 2-16, 2-19, 2-33, 2-38, 2-41, 2-42
Management Platforms, 1-29, 1-39, 1-44, 1-47
Middleware, 1-5, 1-7, 1-12, 1-15, 1-18, 1-19, 1-21, 1-23, 1-25, 1-26, 1-27, 1-29, 1-31, 1-32, 1-35, 1-36, 1-37, 1-39, 1-40, 1-42, 1-43, 1-44, 1-47, 1-48, 2-2, 2-9, 2-11, 2-12, 2-17, 2-20, 2-27, 2-28, 2-29, 2-32, 2-40, 2-42
Middleware Platforms, 1-43
Migration, 1-9, 1-32, 1-38, 1-40, 1-46, 2-2, 2-38, 2-39, 2-41, 2-42, 2-48, 2-49, 2-50
Miller Industries, 2-1, 2-6
Mitre Corporation, 2-1, 2-11
Mobile Agents, 1-32, 2-20
Mobile Application Servers, 1-39, 2-29
MOM, 1-43
Monitoring, 1-3, 1-8, 1-12, 1-23, 1-25, 1-26, 1-27, 1-28, 1-29, 1-32, 1-33, 1-34, 1-36, 1-44, 2-16, 2-19, 2-29
Multimedia, 1-15, 1-39, 1-41
Nestle Corporation, 2-1, 2-4
Networks, 1-3, 1-4, 1-5, 1-6, 1-7, 1-9, 1-16, 1-19, 1-23, 1-25, 1-26, 1-27, 1-28, 1-29, 1-30, 1-31, 1-35, 1-36, 1-37, 1-38, 1-39, 1-40, 1-41, 1-42, 1-43, 1-44, 1-45, 1-47, 1-48, 2-1, 2-2, 2-5, 2-7, 2-9, 2-11, 2-12, 2-13, 2-17, 2-20, 2-21, 2-22, 2-24, 2-25, 2-26, 2-27, 2-28, 2-29, 2-30, 2-31, 2-32, 2-33, 2-38, 2-40, 2-42, 2-43, 2-44, 2-45, 2-46, 2-47, 2-49, 2-50
Network Management, 1-26, 2-11
Network Security, 2-13
Next Generation Enterprise (NGE), 1-2, 1-23, 1-24, 1-25, 1-26, 1-28, 1-29, 1-32, 1-33, 1-35, 1-46, 1-47, 1-49
Northern Telecom, 2-1, 2-6
Open Market, 2-10
Operating Systems, 1-5, 1-6, 1-7, 1-12, 1-42, 1-45
Organizational Structures and Design, 1-8, 1-44
Outsourcing, 1-8, 1-26, 1-46, 2-5, 2-6, 2-14, 2-15, 2-29, 2-30, 2-31, 2-50
Off-Shore, 2-2, 2-50
Payment Systems, 1-5, 1-12, 1-43, 2-14, 2-17, 2-22
Performance Management, 1-7, 1-37, 1-44
e-Business, 1-2, 1-11
Planning, 1-3, 1-7, 1-10, 1-12, 1-14, 1-15, 1-17, 1-18, 1-28, 1-35, 1-36, 1-38, 1-39, 1-46, 1-47, 1-48, 2-3, 2-26, 2-27, 2-32, 2-33, 2-42
Portals, 2-2, 2-33, 2-34, 2-35, 2-36
e-Business, 1-2, 1-7
Public Key (Asymmetric) Cryptography, 2-43
Public Key Infrastructure (also see PKI), 2-43
Publish-subscribe, 1-29, 1-43, 2-27, 2-37
Realtime enterprises, 1-2, 1-25, 1-27
RMI, 2-37
SAP, 1-25, 2-33
Security, 1-2, 1-12, 1-28, 1-39, 1-44, 2-1, 2-13, 2-39, 2-43
Semantic Web, 1-39, 1-43
SET, 2-10
Shopping Carts, 1-15, 2-9, 2-14, 2-15, 2-16, 2-18
SOAP, 1-39, 1-43
Solution Architecture, 1-17, 1-18, 1-29, 1-38, 1-39, 1-46, 2-2, 2-9, 2-11, 2-12, 2-17, 2-20, 2-32, 2-35, 2-39
SQL, 1-39, 1-42, 1-43, 1-45, 2-51
Standards, 1-39, 1-42
Strategies, 1-1, 1-2, 1-7, 1-11, 1-15, 1-16, 1-17, 1-18, 1-19, 1-20, 1-21, 1-22, 1-27, 1-34, 1-35, 1-36, 1-38, 1-39, 1-40, 1-45, 1-46, 1-47, 2-1, 2-2, 2-3, 2-4, 2-5, 2-6, 2-39, 2-41, 2-42, 2-49, 2-51

Supply Chain Management, 1-4, 1-5, 1-8, 1-9, 1-19,
1-25, 1-26, 1-27, 1-32, 1-38, 1-39, 2-1, 2-4, 2-6,
2-10

Tiered Architectures, 1-32, 2-43

Tivoli, 1-44

Training Issues, 1-34, 1-44, 1-45, 1-47, 2-50, 2-51

Transaction Management, 1-39

Visteon, 2-1, 2-4

Voice XML, 2-36

VPN, 2-42, 2-44, 2-47

WAP (Wireless application Protocol), 1-43, 2-33, 2-
34, 2-35, 2-36, 2-38

Web, 1-3, 1-5, 1-7, 1-8, 1-10, 1-11, 1-12, 1-13, 1-14,
1-19, 1-22, 1-23, 1-24, 1-25, 1-27, 1-29, 1-31, 1-
32, 1-33, 1-34, 1-35, 1-37, 1-38, 1-39, 1-39, 1-
40, 1-41, 1-43, 1-45, 1-46, 1-47, 1-49, 2-1, 2-2,
2-4, 2-5, 2-6, 2-8, 2-13, 2-14, 2-15, 2-16, 2-19,
2-24, 2-27, 2-29, 2-33, 2-34, 2-35, 2-36, 2-37, 2-
42, 2-47, 2-50, 2-51

Web Security, 2-1, 2-13

Web services, 1-31, 1-39, 1-40, 1-43

Web Services, 1-31, 1-39, 1-40, 1-43

Websphere, 2-15

Wired Wellington, 2-1, 2-5

Wireless, 1-3, 1-6, 1-16, 1-26, 1-27, 1-31, 1-32, 1-37,
1-39, 1-40, 1-41, 1-43, 1-47, 2-1, 2-27, 2-28, 2-
29, 2-30, 2-33, 2-34, 2-36, 2-38, 2-45, 2-50

Wireless Ethernet, 2-45

Wireless Java, 1-43

WML, 2-34, 2-36

Working Solutions, 1-1, 1-2, 1-7, 1-15, 1-16, 1-19, 1-
35, 1-38

Xerox, 2-1, 2-6, 2-7

XML, 1-3, 1-31, 1-32, 1-38, 1-39, 1-39, 1-40, 1-43, 1-
45, 2-4, 2-8, 2-9, 2-10, 2-12, 2-17, 2-20, 2-24, 2-
25, 2-26, 2-27, 2-33, 2-34, 2-35, 2-36, 2-37, 2-
50

XSL, 2-34

MODULE (OVERVIEW)



- *Preface*
- *Suggested Usage in Courses*
- *Acronyms and Glossary of Terms*
- *Text Material*
- **Handbook Detailed Table of Contents**
 - Overview Module
 - Applications Module
 - Architecture Module
 - Integration Module
 - Networks Module
 - Middleware Module
 - Platforms Module
 - Management Module
 - Tutorials Module