An Architecture for Integrating Multi-Vendor Catalogs in Electronic Commerce

Priya Parimelalagan

A Major Report
in
The Department
of
Computer Science

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Computer Science at
Concordia University
Montréal, Québec, Canada

April 2000

ABSTRACT

An Architecture for Integrating Multi-Vendor Catalogs in Electronic Commerce

Priya Parimelalagan

Internet based commerce is evolving at a rapid pace with companies vying to set up a Web front for shopping by their existing and prospective clients. However there is a problem, when an individual user wants to gather information from competing online stores. In the current electronic commerce world, stores are not connected in a seamless fashion, except in a few cases where the stores belong to the same vendor. Each online marketplace requires different log on procedures, has a separate set of rules for fulfilling orders and, more than likely, has a different business model for charging for its services. In this major report, we study the evolution of the Web based electronic commerce and the Consumer's perspective on such systems. We identify two major shortcomings in the current generation electronic commerce systems: (a) lack of infrastructure for buyers to quickly and easily obtain product information from multiple vendors and (b) lack of consistent user interface across data retrieved from multiple sources. We present in this report a Data Integration System to resolve these shortcomings. The proposed System architecture uses a three fold solution: the Web, which makes electronic commerce accessible to anyone with a browser; eXtensible Markup Language (XML), which allows information on disparate systems to be shared; and industry standards like common ontology and Document Type Definition (DTD) that define protocols for different transactions. The system feasibility is demonstrated by developing a prototype using VisualAge for Java.

ACKNOWLEDGEMENTS

First and foremost my sincere thanks to my advisor Dr. T. Radhakrishnan, for his guidance and support throughout my studies at Concordia. Without his talks providing a spur to my confidence, I am sure I would not have been able to produce what I finally did.

My gratitude to Dr. Lixin Tao for agreeing to be my project examiner and evaluating my work in a short time so that I can return to my family. I would also like to thank the Canadian Institute for Telecommunications Research (CITR) for funding the resources for my project.

A very unique appreciation goes to my friend Caroline, who put up with my stress and seclusion (and use of her home), even when both of us were in the crazy boat of completing our Master's together. Thanks for being there when I needed the most and also knowing when to let me be Me!

To Sathya, Baskar and my parents, I probably could not thank them enough for keeping me in touch with the real world, when I came back to Montreal to complete this project. My special gratitude to my parents for teaching me the power of learning. And Sathya, what can I say? It was the "driveway"! Without Baskar's involved assistance every step of the way, I would not have been able to construct the report you now see here. Baskar, I appreciate you taking the roller-coaster ride and hanging in there with me from the day I came up with my "final project topic", to the day I completed this project. Not to mention all the ungodly hour telephone calls from me, which you always answered with a smile. See you soon.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

Introduction to Electronic Commerce

Internet and electronic commerce are changing the very foundation of many businesses. Over 200 million users and 60,000 commercial websites traded over US$8 billion in 1999. It is predicted that Internet electronic commerce will reach 1.4 trillion US dollars by the year 2003 [URL1, URL2]. Businesses are turning to electronic commerce to help lower the costs, improve customer relationship and increase productivity. Customers are shopping online for increased choice, convenience, better information, and lower price.

Electronic commerce began three decades ago with the introduction of Electronic Data Interchange (EDI) between companies, and Automatic Teller Machines (ATM) for consumers banking [URL3]. EDI and ATM, however, operate in a closed system between the parties allowed in. The introduction of the Web and the Web Browser in the early 1990's opened up a new age by combining the open Internet and the easy user interface. The Web has hastened the convergence of content and transmission, so that stored data can be quickly and easily accessed through the open networks. The Web and the Internet has brought electronic commerce within the reach of thousands of businesses and millions of consumers, fueling rapid evolution of electronic commerce.

This chapter introduces what electronic commerce means, cites a few examples, lists different categories, and surveys the evolution of the various ecommerce constructs.

1.1. What is Electronic Commerce?

Electronic commerce, at its most basic level, is about two parties – one wishing to purchase information, goods, or services in a digital format, and the other offering to sell

it. There are several subjective views ranging from as confined as "selling products and services on the Web", to as broad as "network enabled business practice, including fax, email, Electronic Data Interchange (EDI), and the Web" [1].

Electronic Commerce encompasses a broad range of activities, including trading of electronic material, and of physical goods and services. The trading of electronic contents such as software, music, images, news, and games, represent a revolutionary new way of trading, for which the entire commercial transaction cycle can be automated. On the contrary, the electronic trading of physical goods and services represents an evolution in the present ways of trading. As shown in Table 1, ecommerce not only involves order transactions, but also permeates many other phases of business. Potentially, electronic commerce can provide comprehensive business support by integrating many shared processes.

Phase	Sample Applications
Research / Selection	Electronic catalogs, on-line reviews, multimedia kiosks, video conferencing
Order	Air-line reservations, E-mail, Electronic Contracts, EDI, exchange confirmation
Delivery	Order tracking, bar code labeling
After-sales Service	Electronic Maintenance Manual, Spare parts ordering, Merchandise return, and insurance claims.
Payment	Credit card transaction and electronic cash transaction.

Table 1 Various phases of business activities, and sample applications.

1.2. Examples of Ecommerce Companies

There are many well-established examples of electronic commerce in a wide range of industry sectors, and a wide range of application areas. A few of these are mentioned here to illustrate the scope and scale of current ecommerce activities.

1.2.1 Retail

Amazon.com [URL4] was started in 1995, and quickly became one of the largest booksellers in the world. This Internet retail store exists only as a site on the Web – it has no physical outlets. With a choice of over 3 million titles, and effective search tools, book reviews, brief descriptions, and simple checkout process, Amazon makes book-buying process easy and convenient. These tools combined with discounts of up to 40 percent help convert window shoppers into customers. Amazon believes that it has an inherent cost advantage versus traditional booksellers. Because Amazon does not support a physical store infrastructure, it benefits from lower rent and depreciation and lower labor costs relative to traditional booksellers. Today, Amazon.com has revenue over US$1.6 billion. In addition to books Amazon now sells music CDs, audio books, videos, DVDs, and computer games.

1.2.2 Finance

Wells Fargo [URL5], one of the largest US banks, has physical locations in 10 US states and manages roughly US$109 billion in assets. To meet changing customer references and to lower operating costs, the bank introduced online banking in 1989, with Prodigy and a proprietary direct dial-up. The offering generated little interest in the customer base. Later in 1994, The bank launched an information-only Web site. Then by 1995, Wells Fargo became the first bank to offer its customers access to account balances online. Today, customers can access account balances and transaction history, transfer

funds between accounts, pay all of their bills, order travelers checks, and exchange foreign currency online. Internet banking is offered free to the customers. By the end of 2000, Wells Fargo expects to have more than 1 million online customers.

1.2.3 Distribution

A number of delivery and logistics companies, including Federal Express (FedEx), the United Parcel Service (UPS), the U.S. Postal Service and others are using the Internet in key business processes. The example of Federal Express [URL6] illustrates the role played by the ecommerce in improving efficiency, customer satisfaction and reducing the operating cost. Federal Express delivers 2.5 million packages daily to 211 countries around the world with an on-time delivery rate of 99 percent. Electronic commerce has been at the heart of FedEx's operations. In the mid 80's, the company introduced a program called FedEx PowerShip that gave its major customers a window into FedEx's computer systems with a proprietary link and a terminal. In 1995, FedEx introduced FedEx Ship, a free software program that would work on any personal computer with a modem connection. In July 1996, FedEx launched FedEx InterNetShip, extending the company's proprietary online capabilities to the Internet. Within two years over 100,00 customers started using the service. A fedex.com customer can request a parcel pickup or find the nearest drop-off point, print packing labels, request invoice adjustments, and track the status of their deliveries without leaving the Web site.

1.2.4 Customer Support

Boeing [URL7], the largest Airplane manufacturer, established EDI for its large customers over Value Added Networks (VAN) for spare parts ordering. It took nearly two decades to connect 10% of the largest customers. In the mid 1990's Boeing invested in the Internet to encourage more of its customers to order electronically. Customers

around the world could check parts availability and pricing, order parts, and track order status. Within a year of introduction over 50% of the spare parts customers placed orders via the Internet. The Web site also provides many customer service functions, such as maintenance questions, drawing reference, guidance and recommendations. Boeing's online strategy is to provide customers with one-stop shopping for online maintenance information to maintain and operate airplanes, regardless of whether the data is from the airframe builder, component supplier, engine manufacturer, or the airline itself.

1.2.5 Business support

Started as an electric motor distribution company in the 1920's, Grainger added more products through the years as industry's needs grew for a quick and convenient supply of maintenance, repair and operating (MRO) supplies. Today, Grainger [URL8] is the leading distributor of MRO supplies serving the diverse MRO needs of more than 1.3 million customers. Grainger operates a network of 350 distribution centers with 1,600 person sales force using 4,000-page catalog as their primary marketing tool. Grainger launched its Website in 1995, giving small and medium-sized businesses the convenience of selecting products, checking accounts and determining product availability online. Revenues from the Web site have been growing 100 percent every quarter. The Internet, Intranet, extranet and private networks allow Grainger to leverage information to reach new customers, and improve its operations.

1.2.6 Publishing

Reuters [URL9], a $5 billion global financial news and information company, traces its roots back to 1851, to a small service firm delivering stock prices. The business outgrew and the company relied on the telegraph, and later a computerized quotes display

system. Through the early 1990's, many securities brokers and traders depended on Reuters' terminals to keep current on stock prices. Today, Reuters online offers a variety of services designed for the Internet, including news reports and brokerage products. Users of the online service can tailor the newspaper to their own specific interests or perform a search for past articles. For brokerages, Reuters offers subscription to quotes, pricing charts, company news, market snapshot, etc. Reuters also provides turnkey Internet brokerage solution, including hosting, maintenance and updating. Reuters delivers service through traditional proprietary networks, as well as through the Internet. As the business shifts to the public Internet over time, Reuters plans to get out of the business of proprietary networks, and move fully into the Internet based services [URL10].

As listed above, there are many established companies who are structuring new ecommerce strategy to take advantage of the Internet by introducing new services or by reconstructing their existing arrangements. There are also many new businesses forming to fill the opportunities created by the Internet ecommerce. In addition, the examples help to illustrate various levels at which electronic commerce can be conducted, ranging from a simple network presence to electronic support for processes that are jointly enacted by two or more companies.

1.3. Categories of Electronic Commerce

The different ecommerce models in existence can be categorized as follows:

- Business-to-Business (B2B),
- Business-to-Consumer (B2C),
- Consumer-to-Consumer (C2C), and
- Business-to-Employee (B2E)

B2B commerce model or the inter-organizational ecommerce is between two or more business entities linked together to facilitate transactions. An example in the B2B would be a company that uses a network for ordering from its suppliers, receiving invoices, and making payments. This category of ecommerce has well established over the past several years, particularly using EDI over private or Value Added Networks (VANs). These traditional EDI systems are being supplemented or supplanted by open, less expensive, and wider reach Internet. B2B is the largest and fastest growing ecommerce category in term of revenues. Analysts predict that B2B transactions will be as much as $1.2 trillion by 2002 [URL1].

The business-consumer category largely equates to electronic retailing. As mentioned earlier B2C can be further subcategorized into selling digital content and selling tangible goods. This category has expanded greatly with the advent of the Web. With over 300 million consumers connected to the Internet, B2C creates immense opportunity for existing and new businesses. There are now shopping malls all over the Internet offering all kinds of consumer goods, from books and wine to computers and cars. Web retail sales, as some suggest, could reach over US$100 billion by 2002 [URL2]. To make the most of the potential of the Internet, retailers will have to overcome a number of challenges. Among others, they will need to address the questions of trust, security, privacy and payment. These issues are discussed in more detail in Chapter 2.

The Auction sites fall under the C2C or consumer-to-consumer commerce model. For perishable, scarce, and end-of-life goods, sellers and bidders link to a Web auction. All players have equal visibility into lot descriptions, asking price, last offer, and time remaining – while still retaining their anonymity. The bidders will be able to react instantly to each other's moves – making bidding frenzies possible. Auction sites are of

two types, a standard auction in which the price increases as buyers outbid each other [URL12] or a reverse auction, in which the buyer posts a need and suppliers undercut each other's bid to meet the need, ultimately lowering the final price [URL11].

The B2E or intra-organizational ecommerce occurs within a business entity. B2E links different sections within a company to increase the flow of information. For example, intra-organizational human resources web site providing policies and benefits information is a typical application of B2E.

There are also other categories like Business-to-Government and Consumer-to-Government. These categories are not developed yet. Figure 1 pictorially shows the categories discussed.

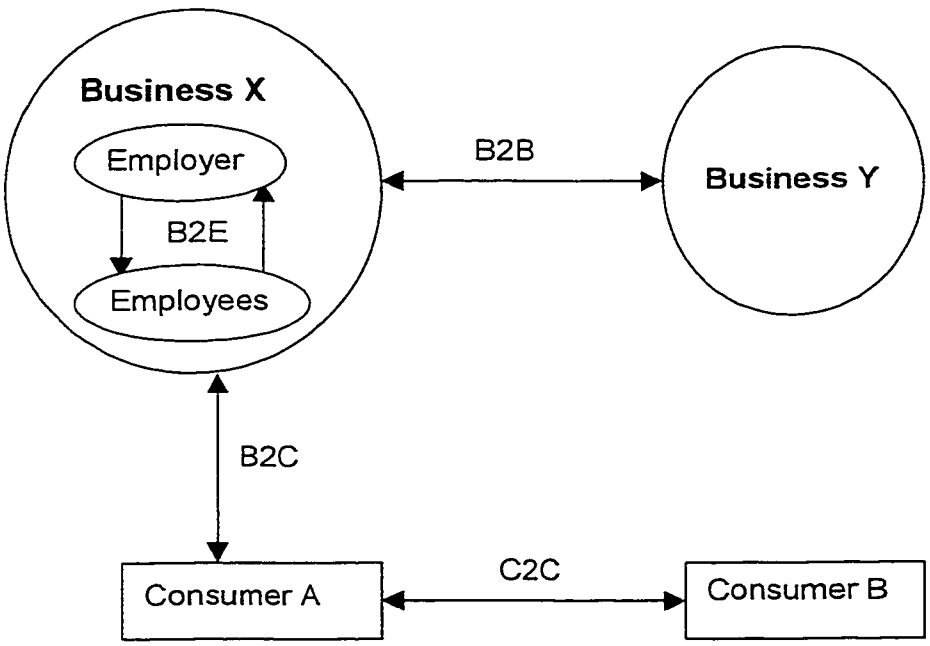


Figure 1 Categories of Electronic Commerce Models

1.4. Evolution of Ecommerce

As mentioned earlier, the electronic commerce in the form of Electronic Data Interchange (EDI) has been used for the past 30 years, especially among large corporations. The Internet and the Web, in just a few years, changed the face of electronic commerce. The evolution of ecommerce is surveyed in this section with a brief coverage of EDI, followed by discussion on various generations of the Internet ecommerce.

1.4.1 EDI

Traditionally organizations conducted business using preprinted forms to exchange information with each other. Over the years, the number of these paper-based exchanges and the associated data increased dramatically. Many of these papers were indeed created and stored in computers. Yet, companies had to re-enter and store the data received from other companies. EDI emerged as a technique to exchange the data electronically between trade partners. EDI made economic sense to many companies as it reduced labor costs by eliminating document entry, reduced document management cost and eliminated mailing cost. In addition, transactions are instantaneous, and less prone to errors. There are 3 components to EDI: Standards, Software, and Communications. EDI standards define the methods for structuring data elements, such as product code, price, name, address, etc, for various trading documents. The most common standards are ANSI X12 in North America, and EDIFACT elsewhere. EDI software is designed to translate messages from standard format to internal format and vice versa. The communications represent the infrastructure, which enables messages to flow between trading partners [URL3, URL13]. The EDI process is depicted in Figure 2. Traditionally companies used Value Added Networks (VAN) or direct connection to carry out these data exchanges. Use of Internet for communications instead of VAN's is

increasing. Today, there are almost 100,000 North American companies who use traditional EDI. Eventhough expensive and complex EDI is challenged by open and inexpensive Internet, many believe, for the short term the Internet would augment existing EDI infrastructure instead of replacing [URL14].

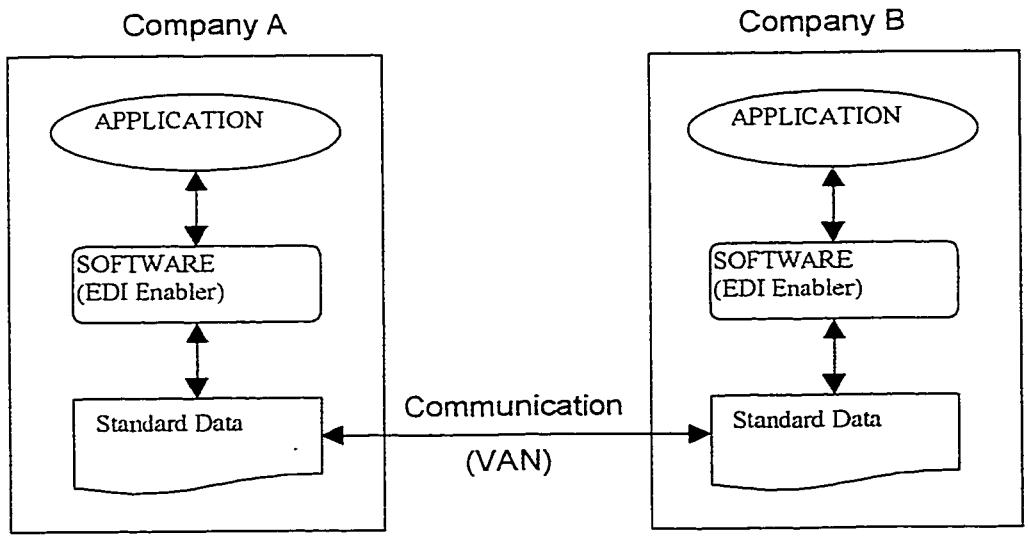


Figure 2 EDI Process

1.4.2 Early Web Ecommerce Models

In the early Web years of 1993 and 1994, many businesses launched informational web sites. Some of these early ecommerce sites included a sampling from their product inventory. The product information was encoded in HTML markup language with images and displayed as static pages. The site provided contact phone number or email address for the customers to receive more information or to order products. These early models did not use the interactive client-server features. The front-end browser displayed a static HTML page from the back-end Web server. These display-only ecommerce models may be considered a mere reproduction of the paper catalogs the companies traditionally maintained.

1.4.3 *First Generation Model*

Soon web businesses adapted to the changing technological trends. The front-end HTML forms interacted to a database in the back-end server via Common Gateway Interface (CGI) programs, as denoted in Figure 3. These interactive forms based models are referred here as the first generation models. In first generation models, a shopper can perform interactive search, and choose a product. Then the shopper fills out a form and clicks a button, which appends the product to the web site's shopping cart. At the check out time, the shopper fills another form with the shipping and the payment details, which is sent to the order-processing department via a CGI email. With advanced scripting languages such as Microsoft Active Server Pages, the merchants could link their product catalogs to the Web, which allowed dynamic access to their product catalogs. The online catalogs were indexed and made computer searchable. A secure transaction system is used for payment.

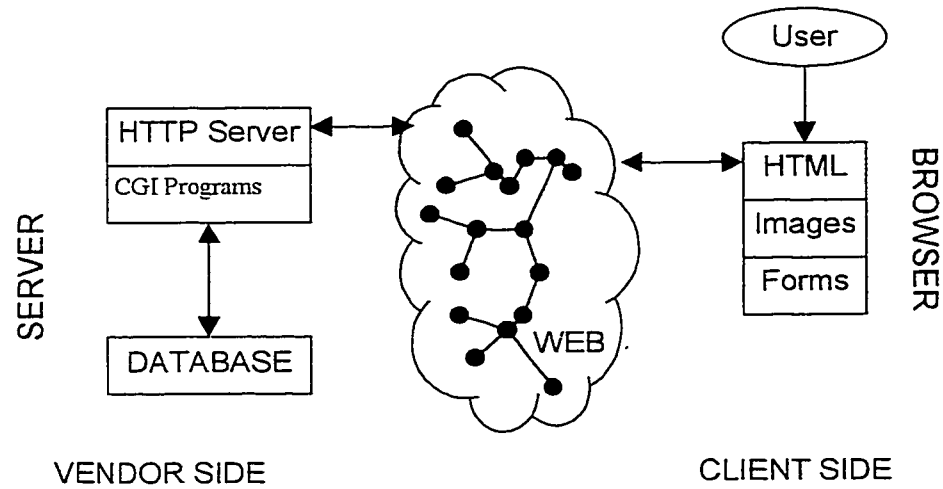


Figure 3 First Generation Ecommerce Model

Eventhough the first generation is a significant improvement over earlier static models, it still had a few distinct limitations for the shoppers:

- Generic: The sites provide no customization or personalization. Generic look for all visitors at all times.
- Multiple Browsing: Comparing price or features of similar products is cumbersome. Shopper needs to browse several vendors sites one after another, and compare them manually.
- Interface: Since each site uses different design, interface, and terminology, the user has to learn and get familiar as he moves from one site to another.

1.4.4 Second Generation Model

Introduction of agents in ecommerce marks the second generation model. The agents allow customization and personalization of the pages displayed. Although the theory behind agents has been around for some time, agents have become more prominent with the recent growth of the Internet.

A software agent is a computing entity that performs user delegated tasks autonomously. In general, agents are based on the idea that an user needs to specify only a high-level goal instead of issuing explicit instructions, leaving the 'how' and 'when' decisions to the agent [URL16]. A basic software agent has three essential properties: autonomy, reactivity, and communication. The notion of autonomy means that an agent operates without direct intervention to the extent of the user's specified delegation. Reactivity means sensing or perceiving change in their environment and responding through effectors. Communication denotes agent's ability to interact with the user and other agents to receive task delegation instructions, and inform task status and completion through an agent user interface or through an agent communication language [URL17]. Table 2 gives a thumbnail view of the broad functional benefits of agents.

Feature	Advantage	Benefit
Automation	Perform repetitive tasks	Increased productivity
Customization	Customize information interaction	Reduced overload
Notification	Notify user of events of significance	Reduced workload
Learning	Learn user(s) behavior	Proactive assistance
Tutoring	Coach user in context	Reduced training
Messaging	Perform tasks remotely	Off-line work

Table 2 Thumbnail View of Agents Functional Benefits.

Internet agents are computer programs that reside on the Web servers, as indicated in Figure 4. The Internet agents serve as information brokers between information suppliers (i.e. Vendor Websites) and information consumers (i.e. Web users). Internet agents match the information needs of the users against the attributes of information suppliers, kind of information, and information content. There are many Internet agents programmed to perform various functions as shown in Table 3 [2].

Function	Description	Example
Information filtering	Filter the information according to the personal interests of the user	NewsHound NewsPage Direct
Off-line delivery	Deliver a personalized package of on-line information to a user desktop according to user preferences	PointCast Freeloader
Search	Search the server database on behalf of the users to provide intelligent search services	Any major ebusiness site, such as Amazon
Notification	Inform a user of events of personal significance to a user	URL-Minder
Miscellaneous Service	Act as an information broker by matching the attributes of context providers against the interests of their service members	Many such as: Book Agent in Amazon, Wine Agent in wine.com
Web site host	Serve as an electronic host to visitors.	Avatar, MUD

Table 3 Function, Description, and Examples of Internet Agents.

Adding agent to a web site provides customization, personalization, and several other functions. "Generic limitation" mentioned in the first generation models is solved in the second generation models using Agents. However, multiple browsing, differing interface, and consumer awareness limitations remain unanswered in the second generation ecommerce models.

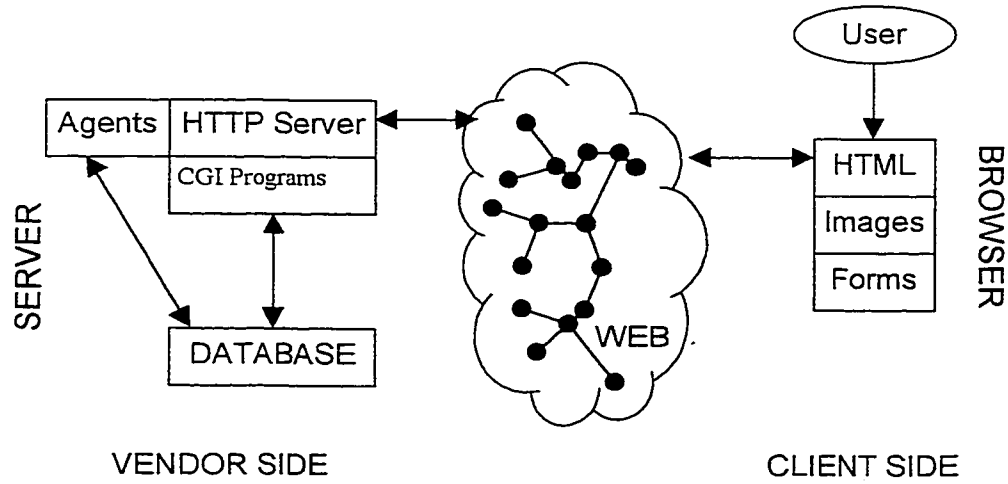


Figure 4 Second Generation Ecommerce Model

1.4.5 Third Generation Model

There are many types of agents serving varying needs of ecommerce, as listed in Table 3. In general, an agent can be measured in terms of agency, intelligence and mobility. Agency is the degree of authority and autonomy given to the agent as it interacts with its user and other agents in an environment. Intelligence is the degree of reasoning and independent learning abilities of the agent.

Mobility is the ability to reach, and communicate to another machine across different system architectures and platforms. The second generation agents have varying

degrees of agency, and intelligence. However, they are limited to the server they reside on. For applications like comparison shopping, user needs to compare price and features of one or more products from many vendors. This task is handled by a special mobile agent, which marks the third generation ecommerce models. Figure 5 portrays the third generation model.

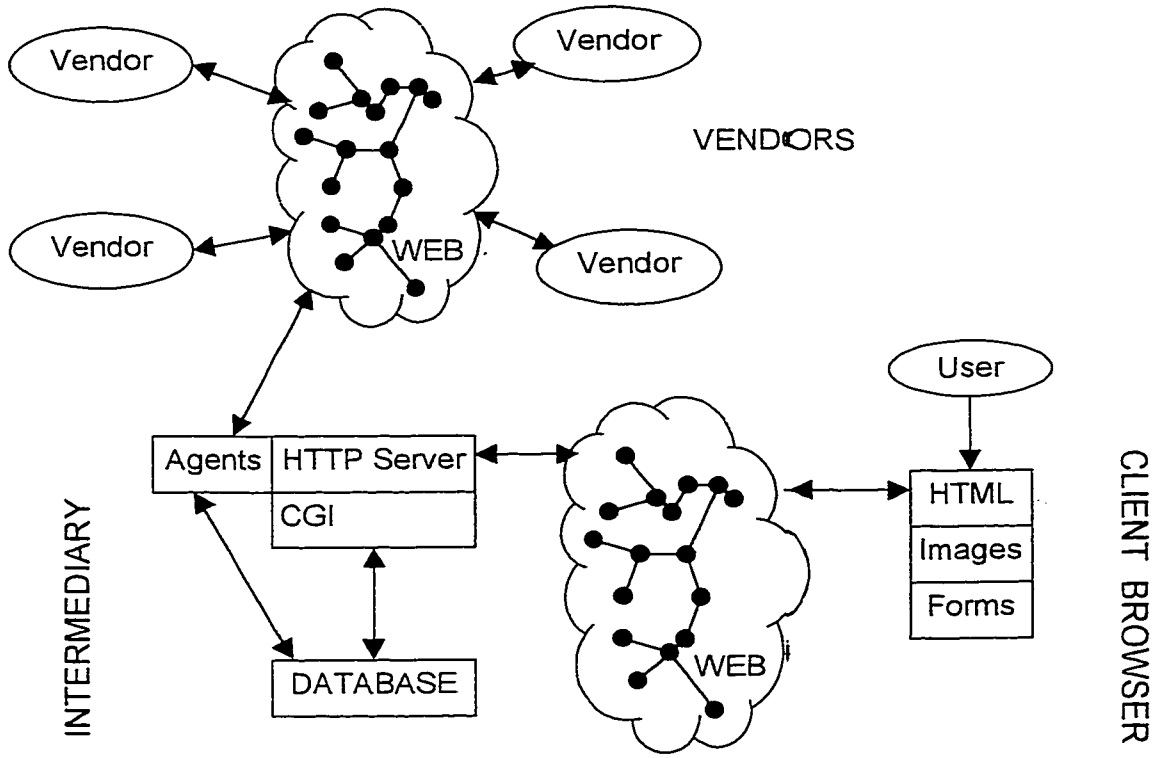


Figure 5 Third Generation Ecommerce Model

The third generation models such as pricescan.com [URL40] and dealpilot.com [URL39] connect to many websites and collects the information requested by the user. The comparative result obtained is then displayed in a tabular format in the user's browser. If the customer wants more info about a product or decides to buy a product from that list, he clicks the link provided, which disconnects the user from comparison site and connects him to the seller site. Multiple browsing and inconsistent user interfaces are

reduced from many to a few. Though this scenario is much better than the manual multiple site comparison, user still needs to face inconsistent user interfaces when he is taken from comparison site to actual purchasing site. Overall, though third generation models simplify comparison shopping, it is still deficient in providing an integrated buying experience for the shopper.

1.5. Conclusion

This chapter introduced ecommerce, listed different categories and presented a few examples. Then in the evolution of ecommerce EDI, early ecommerce models, first second, and third generation ecommerce models were discussed. It was pointed out that the first generation models have three limitations: Generic, Require Multiple Browsing, and Inconsistent Interfaces. The first limitation, Generic user page, is addressed by the second generation ecommerce models with the use of intra-site agents. Other limitations, Multiple Browsing and Inconsistent Interfaces, are significantly reduced by the third generation ecommerce models, but not eliminated. Though comparison shopping, the key focus of this project, is significantly simpler in the third generation model, it is still fragmented.

Chapter 2 examines ecommerce from the consumers point of view, while Chapter 3 discusses the issues and challenges with the present ecommerce model, and examines various trends toward solving these issues. In Chapter 4 an architecture is proposed to resolve the main issues identified in the current generation ecommerce systems. Finally in Chapter 5, a prototype of a subsystem for the proposed architecture is designed.

CHAPTER 2
Consumer Oriented Electronic Commerce

This chapter examines ecommerce from consumer's point of view. The discussion focuses on why consumers shop online, and how the buying process is realized. Also addressed are a few key requirements for the growth and expansion of the consumer ecommerce.

2.1. Benefits for the Consumers

Consumers shop online because they find their choices dramatically increased. They have access to much more information when making a purchase decision. Busy consumers can save time and find shopping more convenient as merchants serve the consumer needs individually. Better information, greater selection, faster and convenient access, combined with lower price draws consumers online and drives ecommerce growth. The following sub-sections inspect these key attributes that attract consumers to shop online.

2.1.1 Choice

The sheer number of stores that can be visited online far exceeds even the most densely populated retail areas in the country. Online, customers can shop at stores in other states, in other countries, and at stores that do not exist in traditional formats. Online bookstores provide a vivid example of this new opportunity. The largest bookstore chains carry an inventory of about 150,000 different books. On the Web, readers can choose from over 3 million from one site, Amazon.com, covering both in-print and out-of-print books [URL4]. In addition to general-purpose books, Amazon carries books on antiques, books written in foreign languages, rare editions, and other

specialty books that would otherwise require extensive phone calls and physical trips to obtain.

2.1.2 Convenience

Consumers cite convenience as the number one reason for making a purchase online [URL10]. Shopping on the Internet can save time. A consumer does not have to travel to a store site or adjust his schedule around the store's hours. No longer does a consumer have to wait on hold for a customer service representative to answer a phone call. Product, service, technical support, and other more information are readily and instantly available over the Web. Consumers can also compare pricing and features of related products, read third party reviews. The example of Garden Escape [URL18], an ecommerce gardening company, shows how combining products and services in a virtual store can save consumers a great deal of time and effort in addition to the convenience of shopping at any time and from anywhere.

2.1.3 More Complete and Upto-Date Information

Online consumers are often better informed than their offline counterparts. For example, shopping for a car can be a very complex process. It involves choosing a particular make and model of car, evaluating different accessories, choosing financing options, purchasing an auto insurance policy, and negotiating a fair price. Prior to the Internet, gathering all these information could take a lot of time, and many consumers went to dealers ill prepared. The Web has changed the dynamics. Web shoppers can view pictures of different cars, learn the technical details, read extensive reviews, compare features with other models, and find the dealer invoice price and manufacturer rebate [URL19, URL20].

2.1.4 Lower Prices

On an average, books sold on-line are about 40% discounted over typical bookstore prices. Average on-line stock transaction charge is US$12 vs. US$80 for traditional brokerage charges [URL21]. This pattern of lower prices is not universal. Some retailers have determined that their current Internet customers buy products from them primarily because of convenience, wider selection or better quality of service. In the short term, these vendors do not feel that lowering prices would lead to any additional sales. As ecommerce continues to grow, competition and the favorable economics are likely to translate into lower prices for the average consumer [URL22].

2.1.5 Customization of Services

The Internet offers the potential for increased customization. Some e-businesses, particularly those who sell music CD and personal computers find the combination of innovation and economics is encouraging increased customization. The Internet and ecommerce technologies may encourage businesses to explore the feasibility of mass customization. Whether and how extensively retailers and manufacturers start to customize the products based on individual customer specifications will ultimately depend on market demand [3].

2.2. Shopping process

Figure 6 shows the e-business process model. Consumer (Buyer) activities are listed on the left-hand side, and corresponding seller activities are shown on the right-hand side. From consumer's perspective, the shopping process can be divided into three stages – selection, ordering and after sales service, as shown in Figure 6:

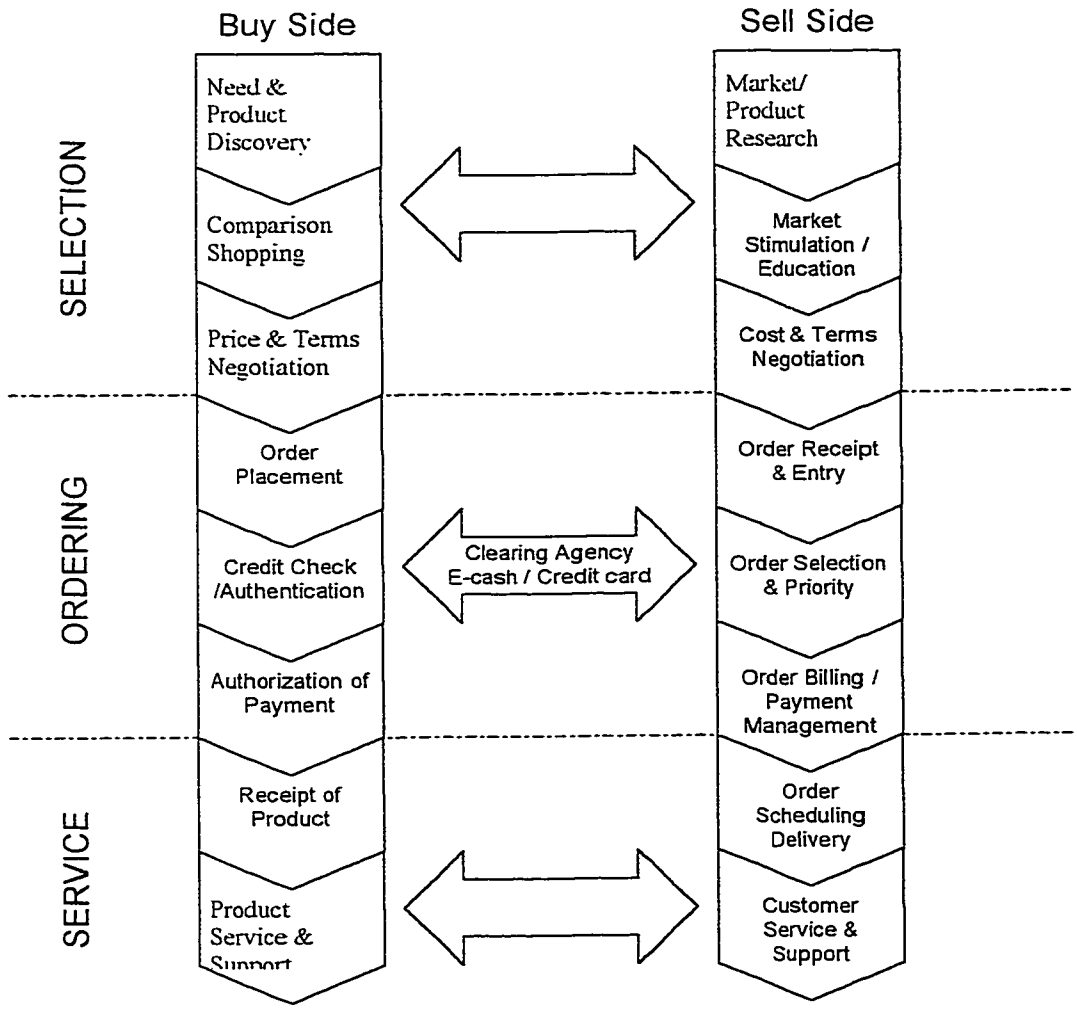


Figure 6 Ecommerce Buying Process

2.2.1 Selection

Any major purchase can be assumed to involve some amount of pre-purchase deliberation, the extent of which is likely to vary across individuals, products, and purchase situations. Selection or purchase deliberation is the elapsed time between consumers first thinking about buying and the actual purchase itself. Information search typically constitutes the major part of the choosing stage. In addition, product comparison and price negotiations are also important part of the pre-sales selection stage. Using an intermediary, i.e. information broker, is the emerging trend in selection. These intermediaries fill the role of integrating the search and helping consumers with

effective comparison shopping. In-store influence of the conventional physical store on a buyer's deliberation is well documented. A similar study of e-store will reveal how these traditional roles vary in an online store [URL10].

2.2.2 Ordering

After identifying the products to be purchased, the buyer and seller interact in some way to actually carry out the transaction. Depending on the payment mode mutually agreed upon, they may interact by exchanging electronic currency that is backed by a third party (e.g. a clearing agency) or by transferring authorizations from credit billing organizations (e.g. Visa). There are many important issues, such as security, confidentiality, identity, payment systems, etc. that are interconnected to the order transaction stage. These issues are discussed in Section 2.3.

2.2.3 Service

As long as there is payment for services, there will be refunds, disputes, and other customer service issues that need to be considered. Returns and claims are an important part of purchasing process that impact administrative cost, transportation expenses, and customer relations. Unless the logistics are well designed, the merchandise reverse-flow could snarl transactions, and result in very dissatisfied customers [4].

There are other complex customer service challenges that arise in retailing, like inventory issues, special requirements or questions about products, shipping, and myriad of other things. These challenges and more will be addressed as the field matures [5].

2.3. Requirements of Consumers

Businesses want to leverage the Internet to be more profitable and more competitive. They are attracted by the advantages of ecommerce – convenience, speed, error elimination, global reach, and cheaper processing. Nevertheless, the ecommerce survey results from customers are revealing the real picture of the current consumer ecommerce. In general, customers don't trust ecommerce, can't find what they're looking for, don't think it is safe nor secure, and can't easily pay for things. The following are some of the main requirements identified by consumer studies for a successful ecommerce implementation: (a) user interface, (b) security, (c) privacy and identity, (d) electronic payment systems. The following sub-sections elaborate on these requirements.

2.3.1 User Interface

The promise of the ecommerce will depend to a large extent on the user interface and how a user interacts with the computer. Online shopping incorporates many of the same characteristics as real shopping. The marketing literatures have identified attributes that shoppers consider when patronizing a retail store. They can be categorized into Merchandise, Service, Promotion, Convenience, Checkout and Navigation [6].

Merchandise represents product selection, assortment, quality, guarantees, and pricing. Consumers infer information about quantity, quality, and variety of products from brand names or reputation of the physical store. Unfortunately, most online stores do not have all the products that are available in the merchants catalog or real store. Customers in general prefer a large product selection. However only 5% of the online stores have more than 500 products [6]. One reason was the complexity of navigation increases non-linearly as the offerings increase. There are other factors like screen size, picture

size, and picture quality that are to be considered as well for the user interface. in addition to the product listing, e-catalog should provide links to extensive product information, product reviews, and product demonstrations to help buyer make an informed decision.

Service includes general service in the online store, credit policies, and after-sales merchandise return, after-sales support etc. Interactive customer services are vital for on-line stores. FAQ section, gift services, company information, product information, links to extended product descriptions, feedback, payment and return policies, phone support are some of the service information online customers commonly expect.

Promotion and advertisements attract customers. Store promotions include frequent buyer schemes, magazines with product and related items, glossaries, product related tips, lottery games, links to other sites and auctions. Advertising has many forms on the Web. Banner ads are small rectangular ads on the side that link to a target site. Spotlight ads feature icons or images that link directly to products. It has been studied that the animation and moving text have overpowering effect on human peripheral vision making it difficult to process information elsewhere in the page [6].

Convenience includes store layout and organization features, as well as ease of use. General help functions might assist users in error recovery or find a particular function in the document. Help also includes information about the store's navigation or the use of metaphors like shopping cart, and convenience features like status indicators. It is noted that short page, succinct text, multilevel headings, and use of colors make an effective impression [URL25].

Checkout should be easy and fast. Presently the checkout process is different for every store. This causes confusion and repetitive data entering. Universally adopted standards would alleviate this concern. Shopping cart management, such as viewing, undo, cancel, etc. form an important part of checkout design. Other checkout process items would include information about shipping date, order number, and confirmation.

Navigation refers to the simplicity and efficiency of finding the desired item. Store navigation features include product search functions, site maps, product indices, consistent and context-specific navigational links, browse forward and browse backward buttons, grouping and sorting functions, and the overall site design and organization. Search engines should be mandatory for all large sites. In addition to Boolean keyword search, incorporating category pick lists, and radio buttons would help focus searches.

2.3.2 Security

Studies have shown that security is the number one concern keeping customers away from doing business on the Internet [URL26]. As online commerce thrives, the physical safeguards of software and consumer information will become a necessity to ensure the integrity of any economic transaction. Computer networks are the central nervous systems of e-business. The very nature of Internet, openness and lack of regulation poses a number of threats to the security of these networks

Table 4 lists the security threats and the solutions. The degree and focus of security architectures will vary from business to business and from system to system depending on what is at stake and what the threats are. Establishing a secure environment requires a comprehensive approach that includes policies, education, physical protection,

security software, and manual security procedures. Further, the security system must routinely be monitored, tested, and validated.

Security Threat	Consequences	Solution	Remarks
Access Control. Individuals gain unauthorized physical access to a computer	Service disruption Data loss.	Password Firewalls	Filters and prevents certain traffic from entering the network.
Bugs. Badly written programs or privileged software are compromised into doing things and creating security holes.	Security holes.	Rigorous Software testing.	Hard to isolate.
Virus. Malicious data or code.	Data loss	Virus Prevention & containment Programs	Scanning before download, as well as regular scanning.
Spoofing & password sniffers	Unauthorized access to the system.	Proper System Configuration. Proxies	
Service overloading (filling up memory or disk) Message overloading (Flooding with bogus requests). Killing the source/server	Service Disruption Degradation of service levels Customer dissatisfaction	Application Level Firewalls	

Table 4 Security Threats, Consequences, and Solutions

The common approach to ensure suitable physical security control is by the use of firewalls and proxy servers.

Firewalls are an essential element of ecommerce defense architecture. The purpose is to protect the internal information and systems from external attack. Firewalls check incoming and outgoing traffic in order to keep intruders out, while giving insiders access to the Internet in accordance with corporate policy. Firewalls range from simple and

inexpensive low-ends to highly configurable and expensive high-ends [URL27]. There are two kinds of firewall type: Network level firewalls and Application level firewalls.

Network level firewalls work by screening and filtering data packets. As each packet arrives, the firewall looks at where it has come from and where it is going and determines from programmed rules whether to permit or block the packet's passage. Network level firewalls are fast and transparent to users but are prone to "spoofing" attacks, where an intruder manipulates packet addresses in order to impersonate someone with privileges.

Application level firewalls allow for much stricter security. Rather than just check packet addresses, they also apply rules to deal with application-specific threats. They generally involve proxy servers between the corporate network and the Internet. All interactions with the Internet pass through a proxy server, which ensures that the details of the corporate network are hidden from outsiders and that all traffic is in accordance with application-specific policies. For example a user types in an URL in the browser. The request goes to the company's http server. The server knows all about http applications, including what the company does and doesn't allow. Having cleared the request, the server sends it over the Internet to the Web site. When the site responds, the proxy server validates the data packets, attaches the user's true address, and forwards them to the user. In a similar vein, e-mail and other common Internet applications are handled by proxy servers set up for those purposes. Further, proxies can be used to track network traffic information, including date and time, client IP address, and file size, etc.

It should be noted that firewalls do not provide much protection against viruses and other data-driven attacks. These must be prevented and contained by other organization-wide control measures.

2.3.3 Privacy and Identity

Without assured privacy, effective electronic commerce is not possible. It is also critical that individuals and organizations can be positively identified over the Internet so that parties know with whom they are doing business. Encryption, digital signature, and digital certificates are the basic building blocks for protecting information [URL28]. Specifically they are used to ensure confidentiality, integrity, authenticity, and non-repudiability of the data transmitted in electronic transactions. Table 5 outlines the different assurances that promote trustworthiness of a participating source.

Data Attribute	Explanation	Solution	Remarks
Integrity	The message was not modified in transit.	Digital signature	Protective archiving to refer and restore in case of information loss.
Confidentiality	Someone else did not read the message.	Authentication Encryption	DES RSA RSA Digital Envelope
Authenticity	The message came from whom it says, not from an imposter. Also it is delivered to the right person.	Authentication Digital Certificates	Digital Signatures and Certificates to verify the identities of both sender and receiver
Non-repudiability	The sender can not deny that he sent the message.	Authentication	Digital Signatures to verify the identities of sender

Table 5 Privacy and Identity

The following paragraphs survey the different technologies available today for protecting the privacy and identity of ecommerce transactions.

Secret-key Cryptography involves a sender using a secret key to encrypt the message, and the receiver using the same key to decrypt it, as shown in Figure 7. The weakness of this system is the problem of key distribution; since shared keys must be securely distributed to each communicating party [URL29]. In a large distribution this becomes cumbersome. Although secret key method is feasible for single-user encryption and one-to-one interchange, it is impractical for multi-user e-business environments.

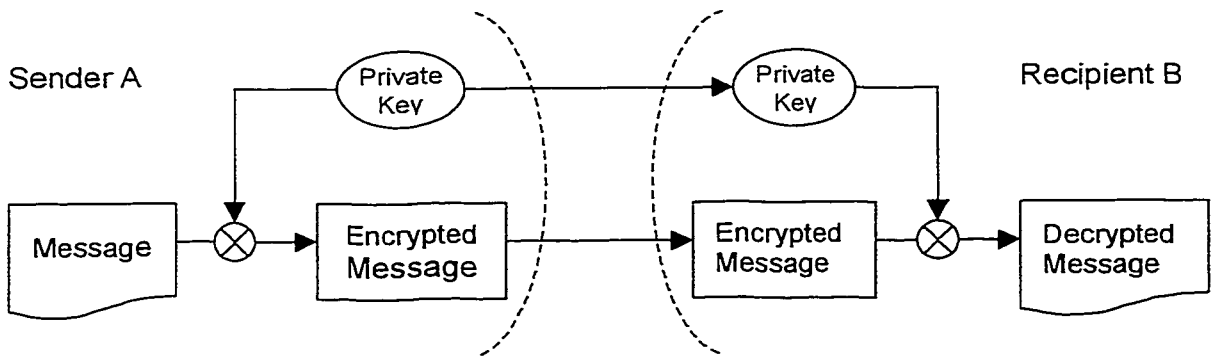


Figure 7 Private-key Cryptography

Data Encryption Standard (DES) is the widely adopted secret key cryptography. DES is extensively researched and studied over the last twenty years and is the most well known cryptography system in the world. DES operates on 64-bit blocks with a 56-bit secret key. Designed for hardware implementation, its operation is relatively fast and works well for large bulk documents. DES is virtually impossible to break with existing algorithms. Despite many new, faster encryption algorithms have been developed, DES remains the most frequently used encryption [URL30].

Public-key cryptography was introduced in late 1970s to solve the key management problem. In this method each person gets a pair of keys: a private-key and a public key. Public-keys are openly published for all to see, similar to a telephone directory. Private-keys are kept secret by the individuals and never transmitted or shared with anyone else. Keys are mathematically constructed and possess a property that messages encrypted with A's public key can be decrypted only with A's private-key, and vice versa, i.e. messages encrypted with A's private key can be decrypted only with A's public-key. When A wants to send message to B, A looks up B's public-key in a directory, A uses B's public-key to encrypt the message and transmits to B. B's private-key is the only one that can decrypt the message. The process of public-key cryptography is depicted in Figure 8.

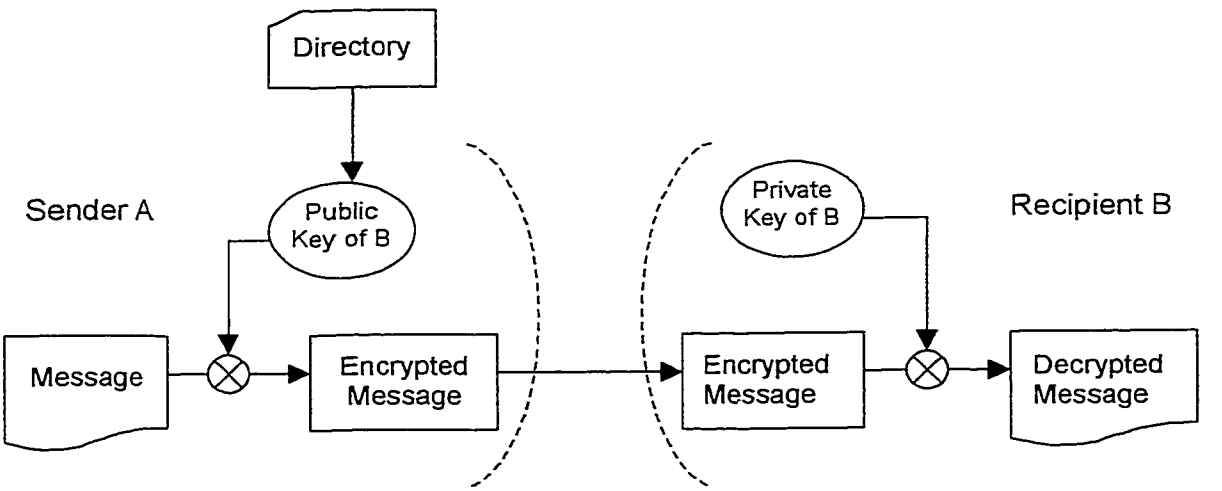


Figure 8 Public-key Cryptography

The system can be used in reverse to prove authorship. If A wants B to know for sure that a message is from A, A can encrypt the message with his private-key. When B successfully decrypts the message using A's public-key, B is assured that it came from

A. The later characteristic of public-key technique provides the basis for digital signature [URL31]. Public keys are used for all transactions and the private key never leaves the users premises. The security of public key encryption rests on the fact that, although public and private keys are mathematically related, it is computationally infeasible to calculate a private key of A from knowledge of the public key of A.

Among the several implementations of public key techniques that are currently in use, the *RSA implementation* (stands for the inventors Ron Rivest, Adi Shamir, and Leonard Adleman.) is the dominant, and considered very secure. Developed in 1977 RSA is undergoing a rapid expansion. It is currently used in wide variety of software products, platforms, and industries around the world. It is being incorporated into Web Browsers giving a wider audience. Combining RSA and DES is commonly used for transmitting large documents. First the message is encrypted with a random DES key and transmitted. Then DES key is encrypted with RSA and sent. Sending the DES-encrypted message and RSA-encrypted DES key is called RSA digital envelope [URL31].

Public key cryptography relies on individuals being correctly identified with their public keys. To verify individuals' correct identity trusted third parties, known as *certification authorities (CAs)* are used. CA's role is analogous to that of notaries in the world of traditional signatures. The CA issues a certificate, called digital certificate, to an applicant after checking the identity and background. Digital certificate contains applicant's name, a serial number, expiration date, the applicant's public key, and the CA's digital signature. The applicant company then can put the certificate on its site. CA and Digital certificate are the central parts of privacy and identity, guaranteeing that the two parties exchanging information are really who they claim to be.

2.3.4 Electronic Payment

Electronic commerce will not reach its full potential until there are simple, inexpensive, private, and secure ways to make payments over the Internet [URL31]. Credit cards are the principal means used today, but they have at least two drawbacks:

- The traditional dial-up credit card verification costs are relatively high – 81 cents per credit card transaction vs. 7 cents for cash transaction in the U.S. [4]. Dial-up access is more expensive in Europe and most other countries, which makes credit cards unsuitable for small purchases.
- The Internet offers inexpensive means to verify credit cards. However, many people are reluctant to send credit card information over the unsecured Internet, where it may be intercepted by a malicious hacker, or misused by unintended recipients.

The credit card was first introduced in the 1960s and has undergone considerable evolution. The current applications include everything from the ubiquitous credit card to advanced smart card applications. For ecommerce purposes, smart cards offer certain advantages in that they can store more information than a magnetic stripe card, and they can be used to conduct more complex tasks involving interactions with various types of terminals. Smart cards also offer security advantages in that the computer code is embedded in hardware, thus making them much more tamper proof than stripe cards. The simplest smart cards are the so-called "stored-value". These can be used as cash cards – "charged" and "re-charged" to a fixed value and used-up as each transaction amount is deducted. Applications of this kind are particularly attractive for small transactions in that each exchange requires no costly verification procedure. At a more complex level, smart card can combine full credit card and cash card applications, sometimes in several different currencies.

Another alternative is the money stored on a computer hard drive with special software to manage its receipt, storage, and expenditure over the Internet. Buyer downloads e-cash to his PC from issuing companies, through a secured channel. Buyer then buys a product and pays for the purchase by sending e-cash. Seller contacts clearing agency to verify the validity of the e-cash. The clearing agency gives okay signal to the seller after ensuring that the e-cash has not been duplicated or spent on other products. Seller delivers the product. Seller then tells bank to mark the e-cash as "used" currency [URL33]. The electronic payment using e-cash is illustrated in Figure 9.

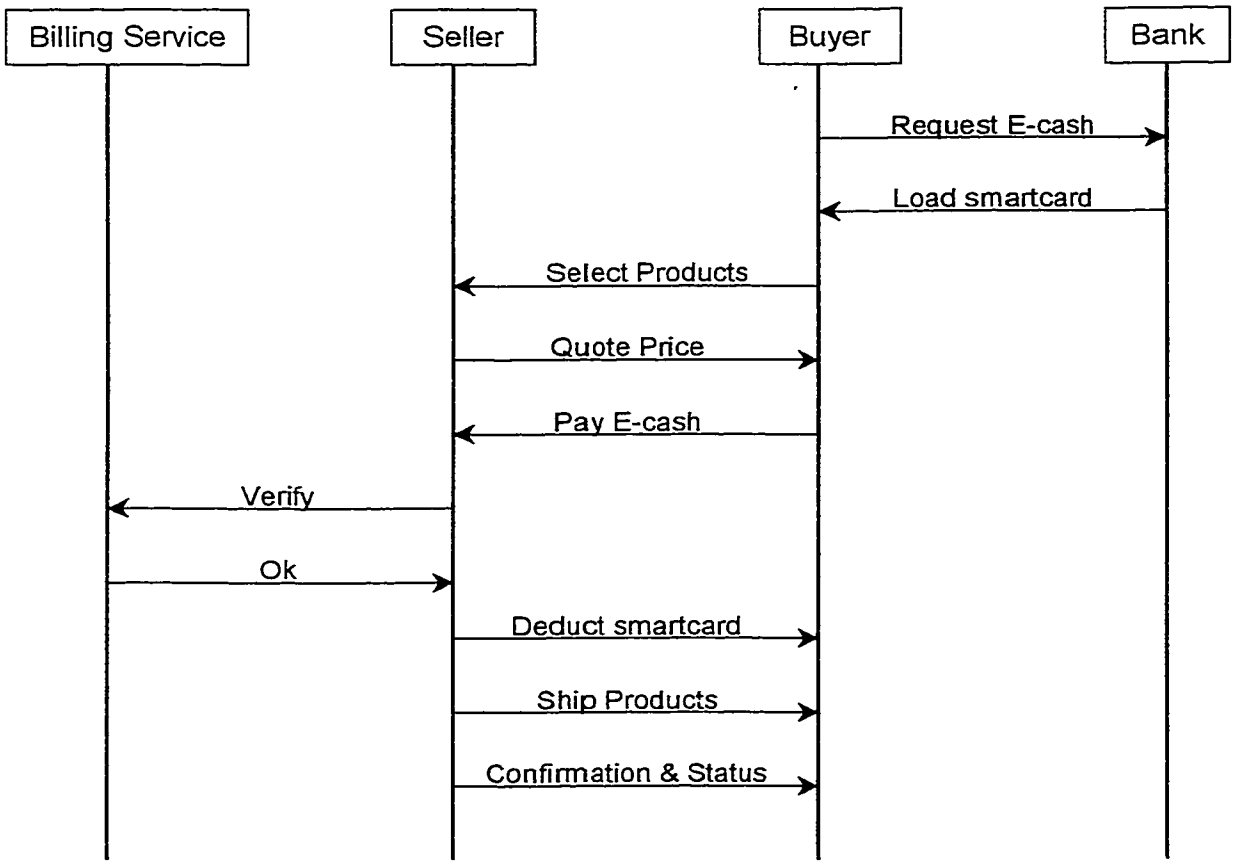


Figure 9 Electronic Payment Process Using e-Cash

Several methods for using credit card, cash(debit) card, smartcard, and local hard drive for online payment have been developed but no one method has yet gained widespread acceptance. These solutions offer at least an interim solution to some of the ecommerce

payment problems. Some believe that these cards will be supplanted eventually by new types of interactive real-time interfaces, which would eliminate the need for customers to carry special cards, relying instead on centralized storage. Compared to traditional payment systems, digital money is in its infancy, and a great deal of technological development and market consolidation must occur before the full potential is realized.

Secure Electronic Transaction (SET) is a standard that enables credit cards (and debit cards) to be used with confidence over the Internet. Backed by VISA, MasterCard, and a number of other partners, SET can be used in multiple hardware and software platforms. SET makes use of public key encryption, digital signatures, and digital certificates to accomplish the task [URL34].

The biggest challenge SET faces is that it requires a relatively complex "electronic wallet" functionality on the client end. E-wallets let customers establish credit card information with a third party service. When making an on-line purchase, the customer simply pulls the desired credit card out of his e-wallet to make payment. The data transmitted does not contain the actual credit card info, but the customer's authorization to the e-wallet, and the third party makes the payment to the vendor. Though popular browsers support some of SET's functionality, there is no agreement among the different browsers as to what the standard will be for trading secure financial transactions.

There are also indications that consumers and merchants are becoming more comfortable with simpler mechanisms that they regard as "secure enough" for the purpose. An example is the Secure Sockets Layer (SSL) that is built into Internet technology. There are several alternatives to SET that offer secure credit card payment

services. The best known of these is operated by CyberCash [URL35] and First Virtual [URL36].

Secure Sockets Layer (SSL) was originally developed by Netscape to address message encryption. SSL is the most widely used transaction security protocol used in the Web today. When a user accesses the Web with an SSL capable browser, SSL encrypts the HTTP transmissions. It is based on using digital certificates for server authentication, and encryption for transmission [URL37]. It does not offer digital signature, and it supports only point-to-point transactions. The shortcomings of SSL include shallow encryption, and the possibility of exposing credit information to unknown merchants [URL38].

2.4. Conclusion

Today, even though Internet retail represents only a fraction of total retail, it is expected to grow steadily in the years to come. The growth of consumer ecommerce will depend on the simplicity and ease of use, trust and security of the medium, confidentiality and identity of the data, and secure payment methods.

CHAPTER 3

Issues and Challenges in Current Ecommerce Systems

Chapter 1 and 2 presented the importance of ecommerce for both sellers and buyers. It was also pointed out that the first and second generation models are incapable of handling applications like comparison shopping. The third generation models can aid users on comparison shopping, but only to a limited extent. The prevailing third generation ecommerce models gather data from other sites. But the gathered data still requires human intervention to interpret, and any further transactions need to be carried out at the specific vendor site. This piecemeal shopping approach requires shoppers to browse multiple sites and learn disparate user interfaces. This chapter investigates the issues and challenges of the present ecommerce systems, and surveys the trends in solving them.

3.1. Comparison Shopping

To illustrate the deficiency of the existing ecommerce models, an example of book buying is reviewed. The shopper knows what book to buy, and he wants the best price overall. In first and second generation models, the shopper goes to online bookstores like Amazon.com and searches for the book. He notes down the price, availability, after sales service and other relevant details. Then the shopper visits another bookseller, BarnesandNoble.com, and types in the book name and finds price, availability, etc. The process of visiting and searching is repeated until the shopper feels he has enough information to make a purchase decision. It is evident that comparison shopping in first- and second-generation models quickly becomes cumbersome even for one book.

The third generation models, like dealpilot.com [URL39] and pricescan.com [URL40], deploy software agents that will search for the book in a list of online booksellers. A table of several vendor names with an access link, book price, delivery, and other relevant details are displayed in the user's browser. Shopper chooses a bookseller from the comparison table, and clicks the access link. The shopper is then taken to that vendor's Web site. There he learns the new interface, types in the order information, and places the order. If the shopper wishes to buy another book from a different vendor, he has to go through the entire sequence, including typing in his order information and placing order in yet another vendor web site, with yet another user interface. The process sequence of buying a book is shown in Figure 10 through Figure 12. Figure 13 contrasts how disparate user interfaces are from one site to another. While agent-automated comparison is better than the manual comparison, it is still a piecemeal solution, which becomes involved when multiple items are sought in multiple vendor sites.

It would be more convenient to a customer if he could compare multiple items from different vendors, in one web site and place order for all the multiple items in the same WebStore. This would allow a buyer to complete his entire buying process in one WebStore. Applications like comparison shopping underscores the need for querying and processing data from several vendors. This is not an easy task, mainly due to the fact that the majority of the data on the Web today is unstructured and non-standard.

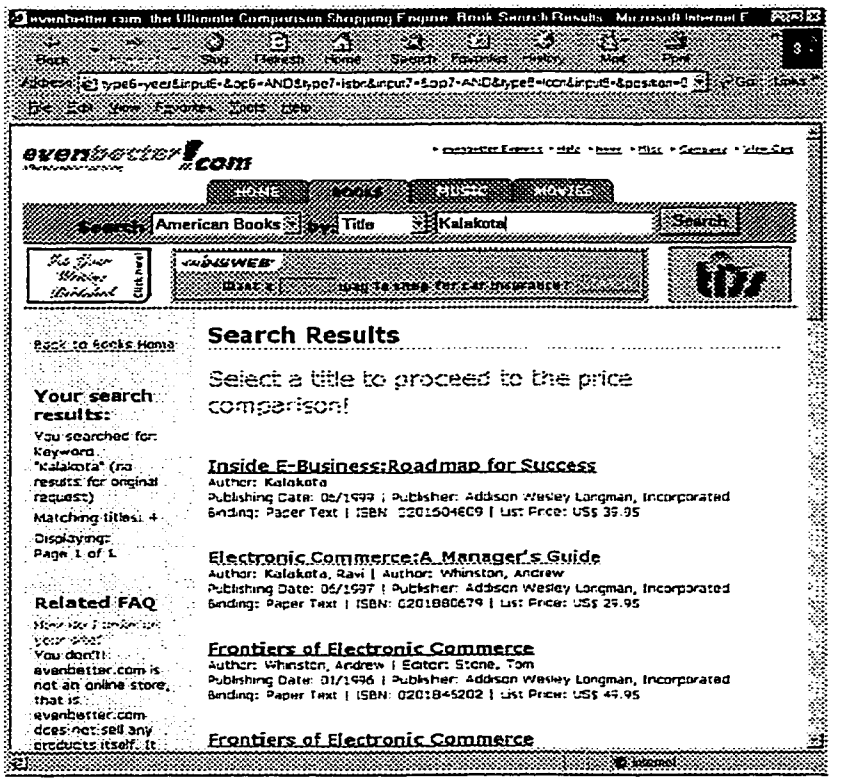


Figure 10 Result of a Sample Author Search

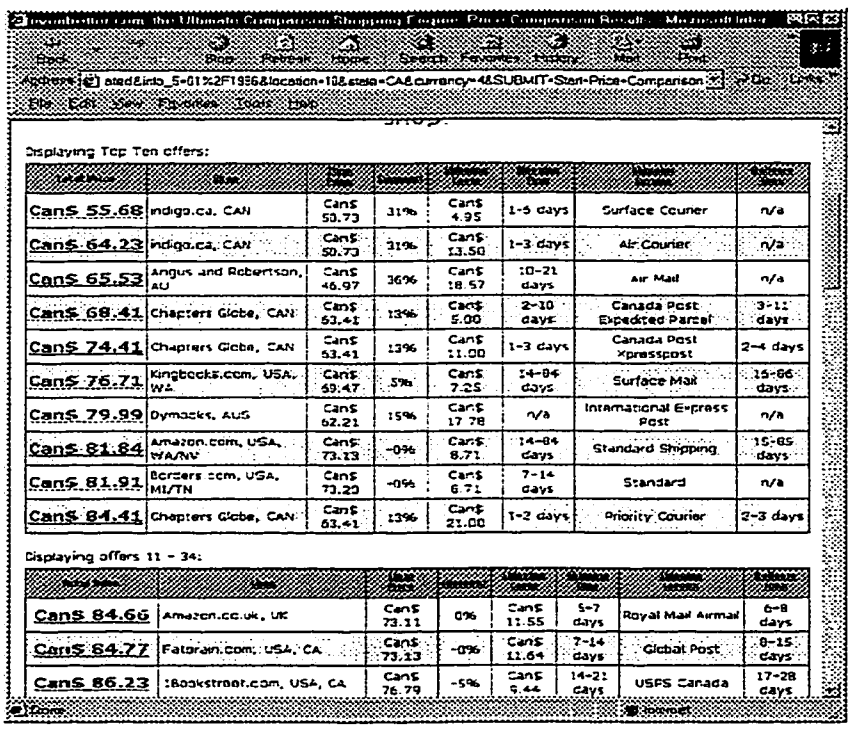


Figure 11 Tabular Comparison Results for a Specific Book Title

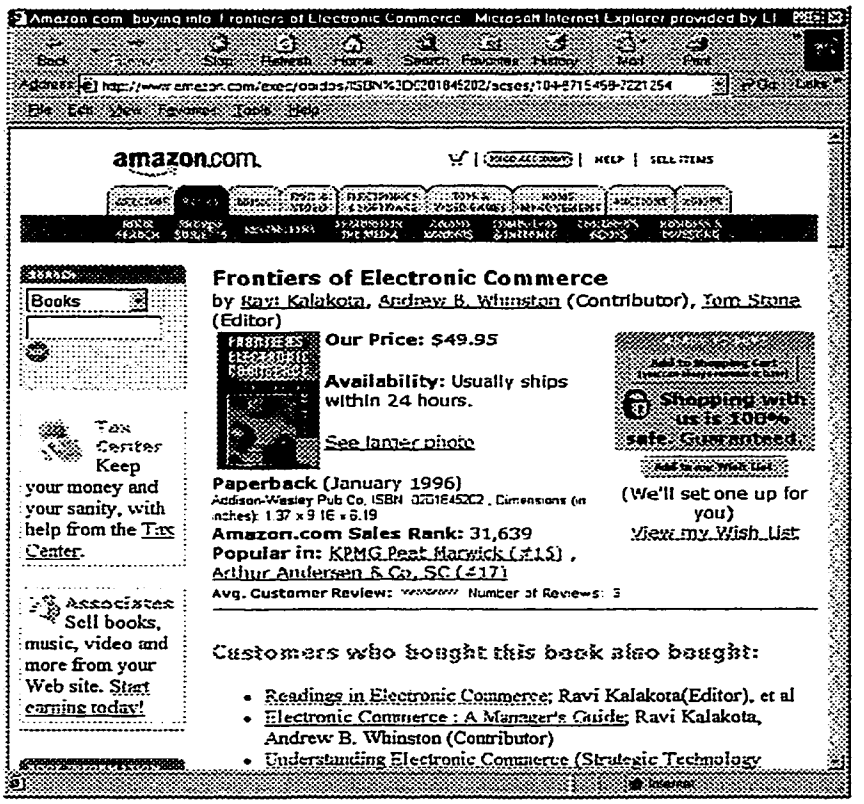


Figure 12 Detailed View of One of the Comparison Results Link

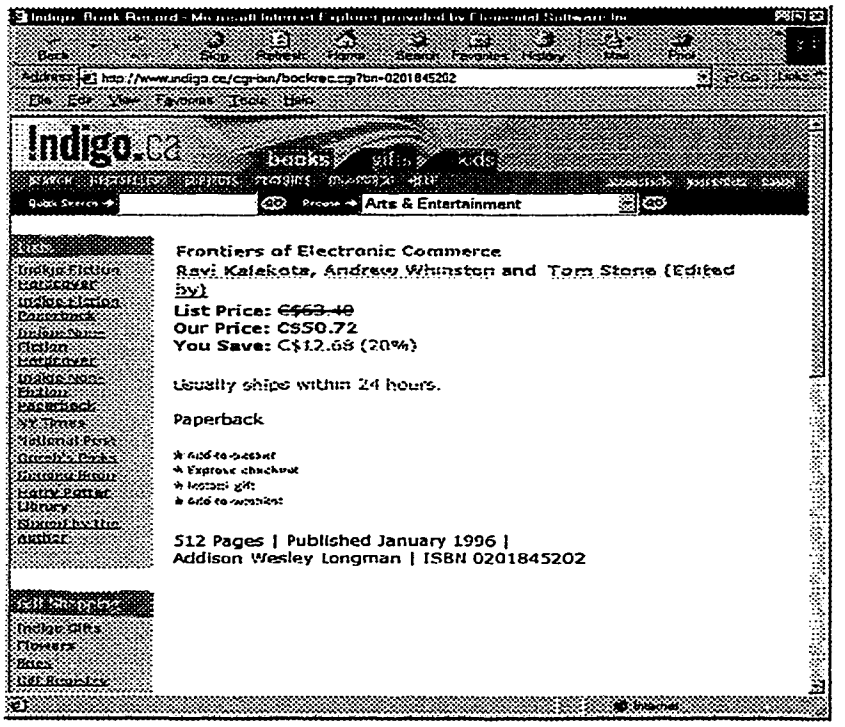


Figure 13 Detailed View of the Same Book with a Different Vendor

The key challenges in developing a fully integrated ecommerce systems can be grouped as follows:

- Lack of common data model among vendors
- Distributed and heterogeneous nature of data sources in different sites
- Dissimilar and inconsistent user interface between online stores

3.2. Lack of common information model

HTML, the immensely successful technology behind present day Web, has some significant structural limitations. HTML provides only a generic type of document structure with simple elements used to delineate headings, paragraphs, lists and tables. While these elements are useful in preparing generic documents, for more complex and structured documents this generic structure is severely limiting. For example, it might be appropriate to prepare a parts catalog using tags like <PARTNUM>, <DESCRIPTION>, and <COST> to delimit the sections of a parts data sheet. With improved structure, documents like a parts catalog could be indexed more accurately. Meaning could be determined by the elements used, rather than by some heuristics based on the number of times a word occurs in the document [URL41]. Besides improved search accuracy, an enhanced structure could ease the migration of data in and out of databases and help pave the way for exchange of structured information between organizations. As it stands now, HTML is not extensible and cannot provide means for users to define their own elements and add improved structure to documents.

Rather than using HTML for Web document markup, some suggest SGML instead. Standard Generalized Meta Language (SGML) is a Meta language, or a language used to define other languages. HTML itself is defined as an application of SGML. While HTML is the best known SGML-defined language, SGML has successfully been used to

define special document types ranging from aviation maintenance manuals to scholarly texts. SGML provides the facility to define whole languages in the form of a document type definition (DTD). SGML is a robust, mature technology and it has been an ISO standard since 1986. SGML is a platform neutral language that has been used successfully to archive huge document sets for governments and large corporations. The language can represent very complex information structures, and it scales well to accommodate enormous volumes of information.

Given all its positive points, one might even consider using SGML on the Web instead of HTML. The main problem with using SGML on the Web is that it is too complex for many applications. It really wasn't defined with Web problems in mind. Complexity is an important consideration and ease of use has enabled a wide range of people to author many documents for the Web. Few Web authors can write a document type definition for a custom language in SGML, and insuring conformance to any such definitions will be difficult given the fact that ad hoc approach to building documents is common on the Web. Even if complexity was not an issue, SGML is missing some important elements, such as a standard presentation. Its style sheet language, DSSSL (Document Style and Semantics Specification Language) lacks wide support. It also lacks base support for linking; relying instead on a technology called HyTime to provide linking facilities. Like DSSSL, HyTime is also not well supported, even within the SGML software community. Lastly, the major Web browser vendors have shown little interest in embracing SGML. This has kept the technology out of the hands of many users, even though its ability to create highly structured and reusable documents would be a boon. A simplified version of SGML for the Web would be one logical way to address the limitations of HTML. eXtensible Markup Language (XML) [URL42] was developed to meet this goal.

3.2.1 XML

eXtensible Markup Language (XML) is a simplified subset of the Standard Generalized Markup Language (SGML, ISO 8879) which provides a file format for representing data, a schema for describing data structure, and a mechanism for extending and annotating HTML with semantic information. The evolution of XML offers the prospect of creating a globally acceptable, cross-platform interchange of information. It can help companies integrate their legacy applications into Web-based systems. The World Wide Web Consortium (W3C) has already ratified the first version of XML specification [URL42], and groups within the consortium is working on associated standards. Major vendors like Sun, IBM, Microsoft, and Netscape have announced their intention to provide full support for these standards.

XML provides a way of defining structured data that is independent of the application that makes use of such data. The tags used in XML documents are not predefined. Any descriptive string can be used. In addition, the data defined by these tags can be displayed in any format that suits the purpose. XML adds a new, intermediate level of abstraction between the data source on one hand and the user interface on the other. This intermediate layer lets one access cross-platform data from any system that supports XML. Since the data is completely separate from the user interface, one can perform client-side processing before displaying the data.

Although one can use any tag to describe the data, it is expected that certain standards or traditions will develop over time. For example, there may be a standard for describing books in print. This will allow users to search across databases for books by a particular author. With the current technology (using HTML), there is no way to distinguish between books *about an author* versus *those by an author*.

While XML does not solve all the data interchange problems, it does standardize universal syntax for describing data. Standard universal syntax facilitates moving data between disparate systems. XML establishes a way to define tags and use them in documents, but it does not help to establish what these tags mean. XML doesn't standardize semantics; XML does not have a formal Data Type Definition (DTD) associated with them. For instance, it doesn't tell what the contents of a DATE element mean – date of publication, date due, date modified, and so on. In fact, XML doesn't even specify if the contents of DATE are strings or numbers or something else.

```
<!ENTITY % boolean "(true | false) 'false'">

<!-- top level element -->
<!ELEMENT  address-book (entry+)>

<!-- an entry is a name followed by addresses, phone numbers etc -->
<!ELEMENT  entry   (name,address*,tel*,fax*,email*)>
<!ELEMENT  name    (#PCDATA | fname | lname)*>
<!ELEMENT  fname   (#PCDATA)>
<!ELEMENT  lname   (#PCDATA)>

<!-- define the address structure. If several addresses exist, the preferred
     attribute signals the "default" address -->
<!ELEMENT  address      (street,region?,postal-code,locality,country)>
<!ATTLIST  address      preferred (true | false) "false">
<!ELEMENT  street       (#PCDATA)>
<!ELEMENT  region       (#PCDATA)>
<!ELEMENT  postal-code  (#PCDATA)>
<!ELEMENT  locality     (#PCDATA)>
<!ELEMENT  country      (#PCDATA)>

<!-- define the phone, fax and email structure -->
<!ELEMENT  tel     (#PCDATA)>
<!ATTLIST  tel     preferred (true | false) "false">
<!ELEMENT  fax     (#PCDATA)>
<!ATTLIST  fax     preferred (true | false) "false">
<!ELEMENT  email   EMPTY>
<!ATTLIST  email   href    CDATA        #REQUIRED
                   preferred (true | false) "false">
```

Figure 14 A DTD for the Address Book

Document type definition (DTD) or Schema to define the grammar for a specific application is described externally. DTD describes the structure that the elements must adhere to in order for the XML document to be a valid instance of that application. DTDs can be shared. For example, when a book DTD is available, everyone can understand all book XML files using this book DTD. Figure 14 illustrates a DTD for the Address book. First, the XML 1.0 specification has a few simple rules for the documents to follow. A DTD further directs the validity of a particular document instance, identifying particular tags and their attributes for any particular class of documents. Schemas, however, go beyond DTDs by further constraining the validity of the content between the tags, or the values of attributes. In general documents that need strong data typing, such as ecommerce business documents, use Schemas. Figure 15 shows the relationship between various segments of XML based e-business.

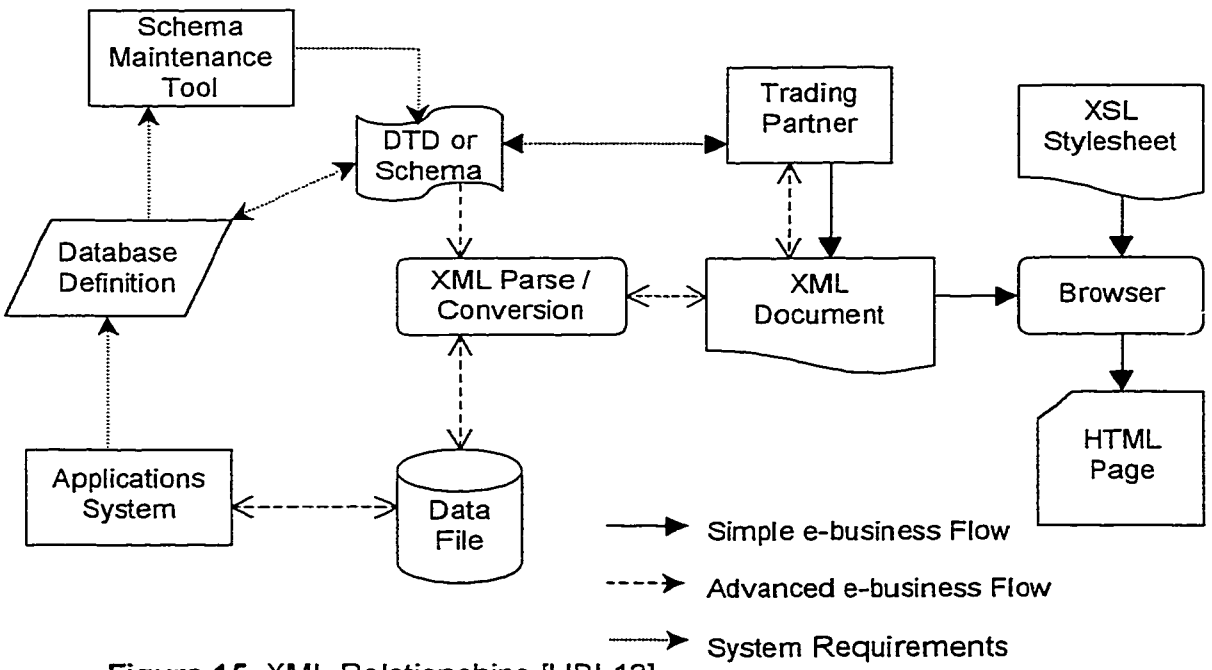


Figure 15 XML Relationships [URL13]

The important element of XML is its extensibility, i.e. the freedom it gives developers within its framework to define the messages they need. However, if everyone defines tags of his own, no benefit is gained. An application to handle the purchase order number will fail if one message uses <purchase_order_ no> tag and another uses <purch.ord.num> tag. Several organizations are developing standard schemas to describe the types of business documents that need to be exchanged so that trading partners can exchange goods and services. These standard Schemas may be held anywhere and shared over the Web. Some of the prominent standards are:

- Information and Content Exchange (ICE) [URL43]
- Ariba Commerce XML (cXML) [URL44]
- Commerce One Business Library (CBL) [URL45]
- Microsoft BizTalk Framework [URL46]

XML-based format comes closest to a universal data format. Using XML the data content can be made self-descriptive, and is easily validated through the use of a DTD or a Schema. This means that all applications can verify the structure and content of an XML file. Companies will use XML documents for publishing everything from product catalogs to stock reports. They will also use XML forms to place orders and to schedule appointments. In this ecommerce framework any agent with proper authorization will be able to obtain computer interpretable data sheets and price lists through the Web or email. XML will eliminate the need for custom interfaces with every customer and supplier, allowing buyers to compare products across many vendors and catalog formats.

3.3. Distributed and heterogeneous data sources

The strength of the Internet lies in its open and dynamic link to limitless networks all over the world. There are many databases in the Internet created at different times, for different purposes, by different companies, using different tools. Querying and processing these data sources collectively for applications like ecommerce poses several challenges. The merchant's database relations may be broken into fragments that are distributed among distinct databases. The products catalog can be distributed either by horizontal fragmentation where the rows of a database are split across multiple databases or by vertical fragmentation where the columns are split. This would require queries to obtain data from multiple databases instead of just one. In addition, the vendor catalogs may be heterogeneous, i.e. the catalogs are represented by different relational databases like SQL Server, Oracle, etc, or sometimes even by object databases like ObjectStore [URL23]. The distributed and heterogeneous data sources open up the following issues:

- How to identify the potential data sources to access for a given query?
- Which data source to access when more than one similar data source exists?
- How to integrate the query results from multiple sources? [7]
- How to handle variation in the database schema? [7]

One possible solution for identifying the potential data source will be to maintain a "Directory Service" (DS). A DS may be imagined as a super database or a knowledge base of the different public databases accessible via the Internet. To make this an effective solution, it is essential that each publicly available data source be registered with the Directory Service. The registered data source is then indexed into different categories and sub-categories. With a well-populated directory listing available, a query

to the DS will provide a list of possible contenders to pass the actual query for processing.

The DS can once again be utilized for minimizing the list of possible data sources for querying. By indexing the mirror sites[1] for a particular database, several optimizations can be achieved. Depending on the location from where the original query comes, the closest mirror site can be used. Again, by maintaining some sought of a rating index, only the databases with a threshold rate index are queried, thus minimizing the number of accessible databases. The results obtained from the individual data sources are combined by a simple join/union operation. These results can be further sorted based on data source, title, relevance and other similar criteria. Data Store is pictorially represented in Figure 16.

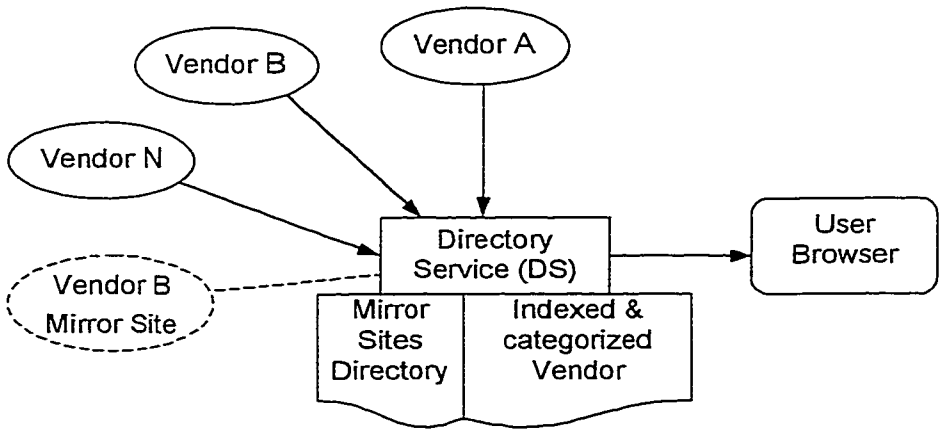


Figure 16 Directory Service (DS)

[1] A mirror site is a Web site or set of files on a computer server that has been copied to another computer server. The purpose is to reduce network traffic, to ensure better availability of the Web site or files, or to make access faster when the original site is geographically distant. A mirror site is an exact replica of the original site and is usually updated frequently.

Another significant issue in dealing with distributed and heterogeneous data sources is the variation in database schemas. Conceptual difference can be introduced by representing similar information in different database schemas. Also distinct databases may use different words to refer to the same concept, or they may use the same word to refer to different concepts. The common approach in dealing with variant schemas is to develop technologies for mapping different database schemas to a single, global model. Having once defined the mapping relationship between the database schema and the global model, the method allows queries that are written with respect to the global model to be translated into executable SQL queries at the distributed database level.

At present most companies still use proprietary terminology and data formats to represent their product information, and publish it through numerous channels including web, online databases, flat files, and XML documents. A global schema will enable users to acquire product information from any source, map it to a common taxonomy and schema, and automatically deploy powerful search applications.

XML can be used to glue the disparate databases. Various integration companies like Ariba, Commerce One and Microsoft are deploying standards for specific business needs. By specifying the standards from any one of the vendors, it is possible to use them as the global schema for mapping the local databases. At the database level the global schema chosen can be translated to match the local schema used. The translation between the different working schema can be accomplished using custom scripted programs or software modules called "Wrappers", as shown in Figure 17.

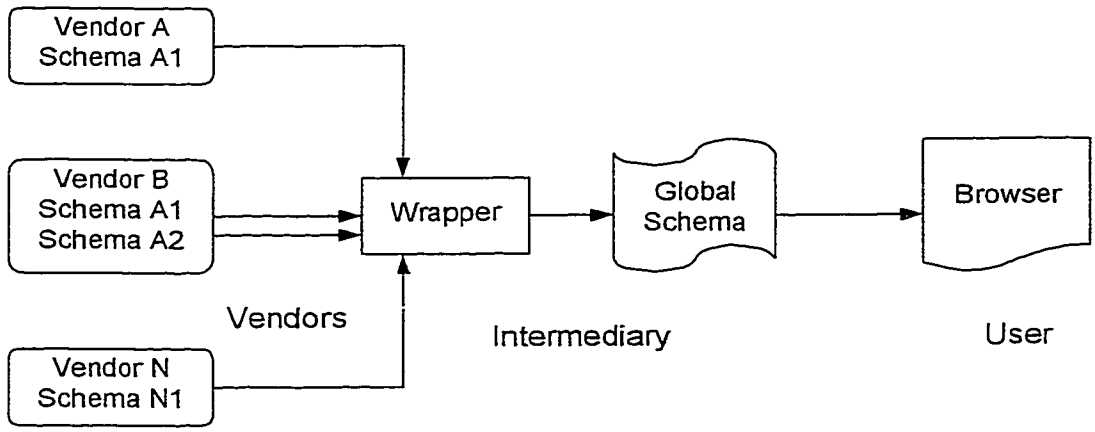


Figure 17 Mapping Local Schema to Global Schema using Wrapper

An effective way to create these wrappers is to model the underlying local database and develop objects. These objects are then manipulated and converted into the global schema. As a final output the Wrapper functions translate the local objects into the global schema and represent the local database as XML documents conforming to the global schema. The global schema may be defined by a document type definition (DTD).

3.4. Dissimilar and Inconsistent User Interface

A common scenario when accessing information from multiple data sources is the lack of consistent user interface across the sites. Each vendor site has its own look and feel. For example, suppose an application is deployed to search multiple vendor sites and provide a list of vendors selling a particular product with the sale price and other product details. In doing so, the agent application gathers information from participating vendors and displays the results in a tabular format. The results include the vendor name, product name, cost, availability etc. More information can be obtained by following the hyperlink provided for each displayed result. Upon selecting the hyperlink the user is taken to the actual vendor site with no further communication with the third party site that provides the comparison agent application. If another vendor needs to be checked the

user traverses back to the result list using the browser's "back" button and follows the link to the next vendor, who provides a different interface view. Different navigation links and disparate site design could disorient any typical user. Figure 10 through Figure 13 illustrate the dissimilarities typical in present day ecommerce sites.

The current trend in creating consistent view is to use content "Syndicators" and content "Subscribers" [URL43]. Content Syndicator is the source of information that controls the data provided for dissemination. In the consumer retail model, a Syndicator is the vendor publishing the product catalogs. The Content Subscriber receives information from many Syndicators and distributes them in their own format and layout. Subscriber is the third party or the intermediary site providing comparison shopping for the consumer. Thus the data received from different Syndicators are transformed and displayed in the Subscriber's layout template, as indicated in Figure 18. A Shopper can gather the necessary information about many vendors (Syndicators) by accessing Subscribers site. This avoids shoppers having to jump between several vendors, maneuvering through dissimilar sites.

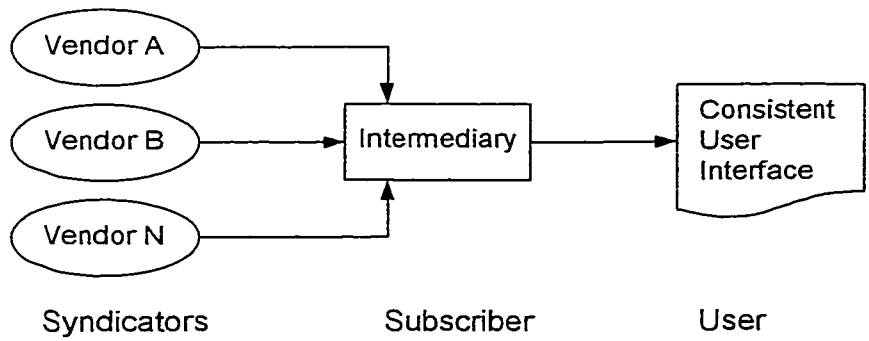


Figure 18 Syndicator / Subscriber Ecommerce Model

A prominent protocol adhering to this field is the Information and Content Exchange (ICE) standard proposed by Vignette and other vendors [URL43]. The ICE specification provides businesses with an XML-based common language and architecture that will facilitate the process of automatic exchanging, updating, supplying and controlling assets without manual packaging or knowledge of remote Web site structures.

3.5. Conclusion

This chapter illustrated the limitations of the current ecommerce models. The key challenges in developing fully integrated ecommerce systems are: lack of common data model among vendors, distributed and heterogeneous nature of data sources in different sites, and dissimilar and inconsistent user interface between online stores. The lack of common data model among vendors can effectively be resolved by XML. Directory Service and XML based Mapping are shown as possible answer to distributed and heterogeneous data sources problem. Syndicator and Subscriber Content Model offers potential solution for inconsistent user interface between online stores.

CHAPTER 4
Proposed Architecture for Data Integration

Chapter 3 discussed the key challenges in developing integrated ecommerce systems, namely, lack of common data model among vendors, distributed and heterogeneous nature of data sources in different sites, and dissimilar and inconsistent user interface between online stores. Chapter 4 proposes a new architecture for Web shopping which addresses these issues. Comparison shopping is the focus of this study. Comparison shopper is a buyer who wishes to compare price and features for a product sold by multiple vendors. For a given comparison search, a list of vendor sites are accessed. The product data from different vendors are gathered, consolidated, and displayed on the buyer's browser. In addition to the product selection, the shopper can also perform the purchase transactions in the same Web store. The proposed architecture renders a complete shopping experience for a consumer, by providing a unified, consistent and personalized view of the information supplied by different vendors.

4.1. Data Integration

Data Integration is the core of the integrated buying scenario pointed in the preceding discussion. Data integration is defined as application-to-application integration that crosses corporate boundaries. In other words, Data Integration allows the automated exchange of information across multiple, heterogeneous and distributed vendor catalogs.

EDI has been traditionally used for data integration between companies over private networks. Table 6 compares the features of EDI and XML [URL13]. Using the Internet,

HTTP, and XML for data exchange, instead of proprietary Value Added Networks (VANs) and proprietary protocols, is gaining wider acceptance.

EDI	XML
EDI 'marks up' data using Segments and Elements	XML 'marks up' data using Tags
EDI document structure described by a message definition that has to be programmed into the software	XML document structure described by a Schema, either included in the document or available via a URL
EDI Data Element types and formats well defined	XML Data (currently) restricted to strings without format
EDI Segment names typically 3 characters	XML Tag names have unrestricted length
EDI documents defined by 3 main standards bodies	XML document standards are rapidly evolving, albeit with many bodies
EDI business to user interface is awkward	XML business to user interface via a browser
EDI through VANs incurs a significant implementation effort	XML via the Internet takes advantage of universal connectivity
EDI uses cryptic codes to denote the segment structure of the message	XML message formats use readable elements
EDI is optimized for compressed messages	XML is optimized for easy programming

Table 6 Comparison of EDI and XML features

There are some weaknesses in XML, largely due to its relative immaturity, but steps are being taken to improve things rapidly, particularly in the area of emerging standards. XML is perceived to be a key enabling technology for exchange of information on the Web and within companies. As a consequence, integration of XML data from multiple external sources is becoming a critical task. The reality is that XML, being only syntax, only partially advances the prospects of data integration. However, with the arbitrary nature of the names and meanings of the tags used in XML documents; XML without an

agreed upon DTD i.e. the "mediated schema", does nothing to support data integration at the semantic level.

For successful data integration across multiple heterogeneous data sources, a mediated (global) schema is required to represent a particular application domain, and the individual data sources are mapped as views over the mediated schema. This way for any user query over the mediated schema, the integrated system reformulates it into a query over the individual data sources and executes it. Some of the issues that need to be addressed to enable true data integration are highlighted here.

- A key element for data integration is a language for describing the contents and capabilities of data sources. These descriptions provide the semantic mapping between the data in the source and the relations in the mediated schema.
- A need for algorithms to efficiently reformulate user queries posed on a mediated domain schema to queries that refer to the local data source.
- A need for translating data conforming to one DTD into an XML document conforming to a different DTD.

4.2. Integrated Shopping Process

A typical shopping scenario in an integrated ecommerce system is depicted in Figure 19. The shopper visits an "integrated" WebStore. In case 1, the shopper knows the exact product he wants to buy. In case 2, the shopper does not know the product he wants. To identify and select the product, he browses various product categories, and/or he runs product keyword search. In case 3, the shopper has no predetermined product to buy; he is merely window-shopping.

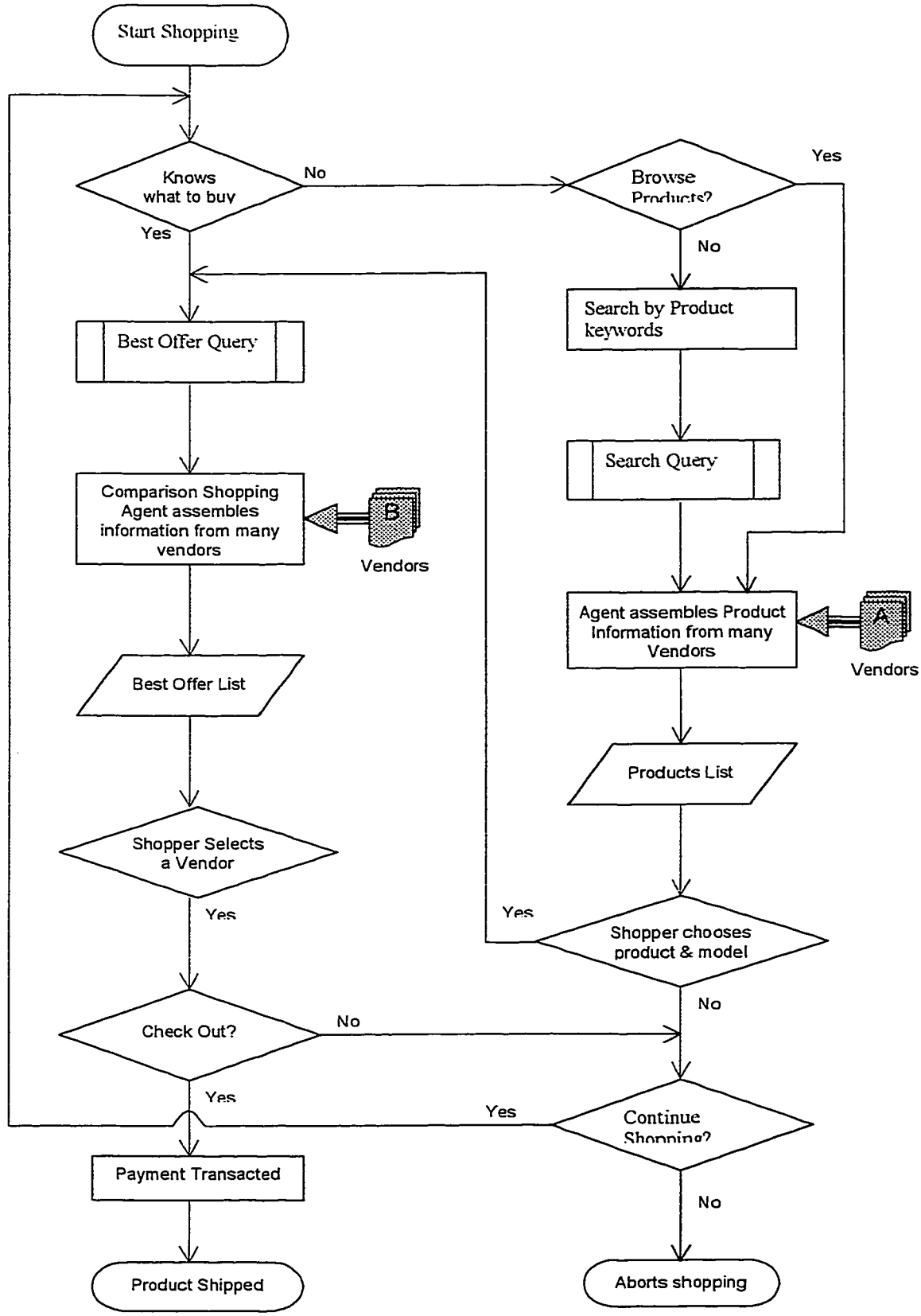


Figure 19 Shopping Process in the Proposed Architecture

All three cases utilize search – either category search or product search. The search ensues search queries, which are sent to a database that is in communication with various vendor sites. The search agent collects, compiles and displays a detailed list of products in the shopper's browser. The shopper then identifies and selects a product from this list. For the price quotes, another query is sent to the same database. Likewise, price quotes from many vendors for the selected product are assembled. The best offer list consists of quotes from several vendors passed to the user. Based on the information presented, the shopper accepts one offer. The vendor database is contacted. Purchase order is placed, payment is transacted and the product is shipped.

The multiple vendor information integration is represented by the gray links A and B in Figure 19. Multiple vendors catalog information is accessed and compiled for the shopper's benefit on two occasions: (A) for product search, when information from multiple suppliers selling similar products are integrated; (B) for best offer search, when data from different retail merchants selling the same product at different price and value-added services are integrated. In both situations, the WebStore contacts various vendors, aggregates data, and presents them to the shopper. Entire buying process, from search through purchase transaction to final confirmation, is completed in one integrated WebStore.

4.3. Overview of the Proposed Architecture

The objective of the model is to provide a complete shopping experience for the consumer, while providing a unified and consistent view of the information gathered from numerous distributed data sources. The integrated WebStore is the centerpiece of this model. The integrated WebStore is an intermediary who uses the data integration

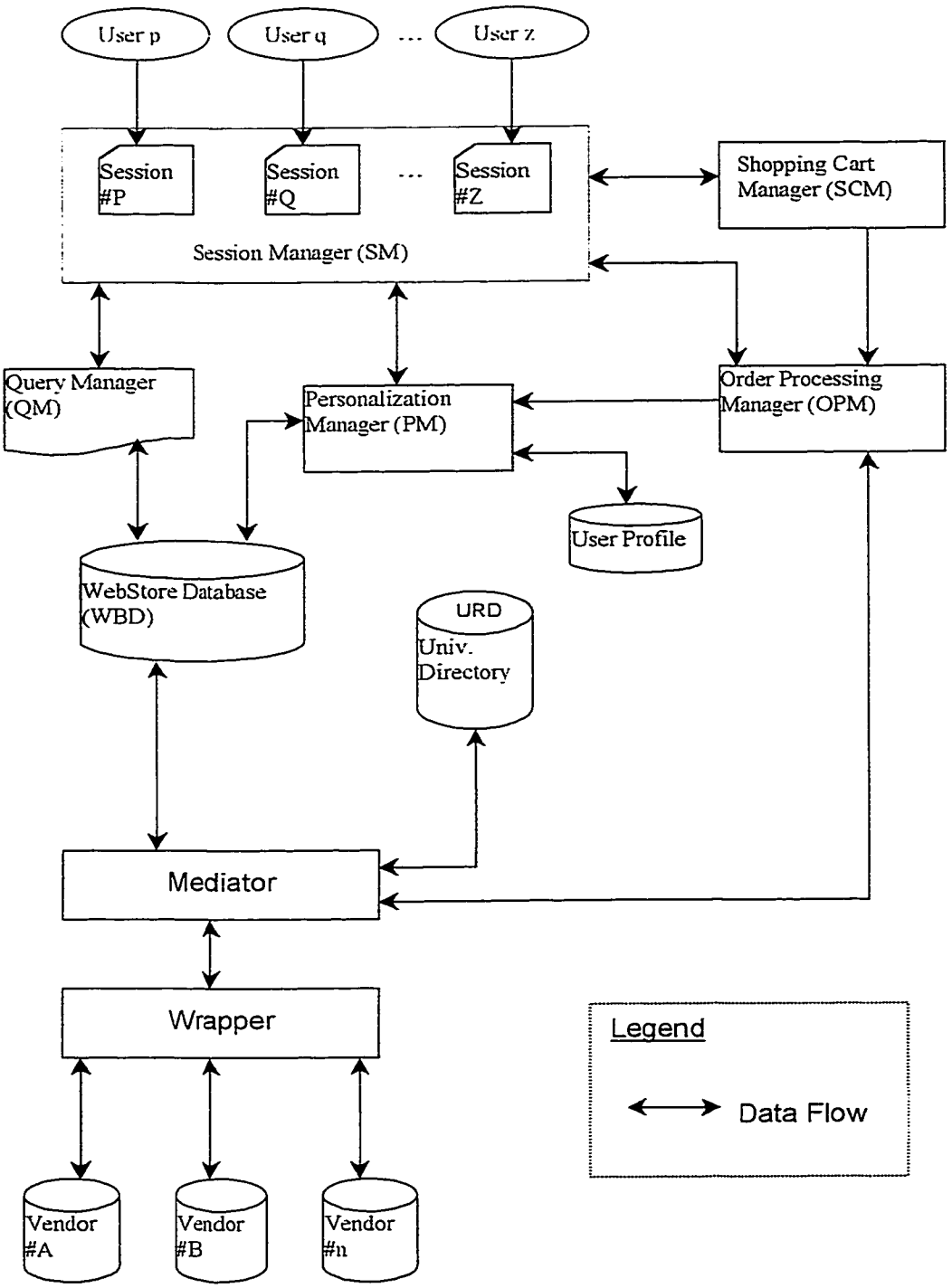


Figure 20 Architecture of the Proposed Data Integration Model

techniques to provide an enhanced and integrated shopping experience for the user. Data from several vendors are gathered, compiled and presented to a prospective buyer. The architecture of the integrated WebStore is shown in Figure 20. The outlines of the different modules are given below:

- **Session Manager (SM)** monitors the activities initiated by the shopper visiting the WebStore. Depending on the shopper's action, SM invokes other modules in the architecture. SM maintains an unique identifier called the *session identifier* (sessionID) for each shopper's session.
- **Query Manager (QM)** module handles all the queries initiated by the shopper's SM. There is one QM module for all the shoppers. The individual shopper's query request is tracked by the shopper's *session identifier.*
- **WebStore Database (WBD)** module is the local information repository of the WebStore site. The information from different vendors may be cached or stored by this repository for future data requests by other modules.
- **Universal Directory (URD)** is a resource center for information about the different vendor catalog definition, mediated schema definition for the current product domain, etc.
- **Mediator** is the control center for accessing heterogeneous and distributed vendor catalogs. The Mediator coordinates with WBD, URD and Wrapper modules to respond to a specific shopper's query.
- **Wrapper** is a set of translation rules to communicate between the WebStore site and a participating vendor catalog. There may be more than one individual Wrapper modules based on the vendor catalog definition.
- **Vendor** #A through #N are the external vendor catalogs accessed by the WebStore site. There are *n* distinct external catalogs to be accessed.

- **Personalization Manager (PM)** works closely with SM to provide personalized view for the visiting shopper. PM gathers relevant information from other modules in the architecture.
- **User Profile** is a repository of the shopper information collected by the WebStore site over time. This information may include data collected directly from the shopper via HTML forms when registering in the Web site or behind the scenes data collected based on the shopper's "in site" navigation and prior buying behavior.
- **Shopping Cart Manager (SCM)** manages the products added for purchase by the user. Each shopper has his own individual shopping cart to add products for future purchases. The different shopping cart's are managed based on the user's *session identifier.*
- **Order Processing Manager (OPM)** module completes the "buy" request from the shopper for all the products placed in the web site's shopping cart by transmitting the purchase request to the respective vendor via the Mediator.

The shopper completes the entire buying process in the proposed WebStore. The shown architecture offers buyers a complete shopping experience by providing a unified and consistent view of the information gathered from numerous distributed and heterogeneous data sources. The Mediator and Wrapper modules handle the multiple data source integration, while the WebStore Database acts as a local repository for the information gathered from different sources.

4.4. Description of the Modules in the Architecture

Architecture of the intermediary WebStore is illustrated in Figure 20. The various modules and their functions are described in the following sub-sections.

4.4.1 Session Manager (SM)

The Session Manager monitors browser activities and distributes messages to other modules accordingly. Each user session is given a unique sessionID, which is linked to a specific userID. Data validations using scripts are performed at the Browser/Session Manager. Then the data is passed to the Query Manger (QM). The results given by the QM is passed back to the browser.

For better sales and service support, SM gathers pertinent user and product data from the Personalization Manager, and provides a personalized custom view for the shopper. In addition, SM collects the user's actions and delivers them to Personalization Manager for updates as well as for future reference. When a shopper finishes his buying decision, SM also communicates with Shopping Cart Manager (SCM) and Order Processing Manger (OPM), to complete the buying process.

4.4.2 Query Manager (QM)

The Query Manager is initiated anytime a search is invoked. The keywords from the HTML/XML search form are converted into a valid query description in XML. The query description is later translated into a sequence of Java Database Connectivity (JDBC) or Open Database Connectivity (ODBC) calls for the WebStore Database (WBD) (see Figure 21). The query result from WBD is converted to a XML Document Object Model (DOM) structure and returned to the client browser either as a XML document or an HTML document.

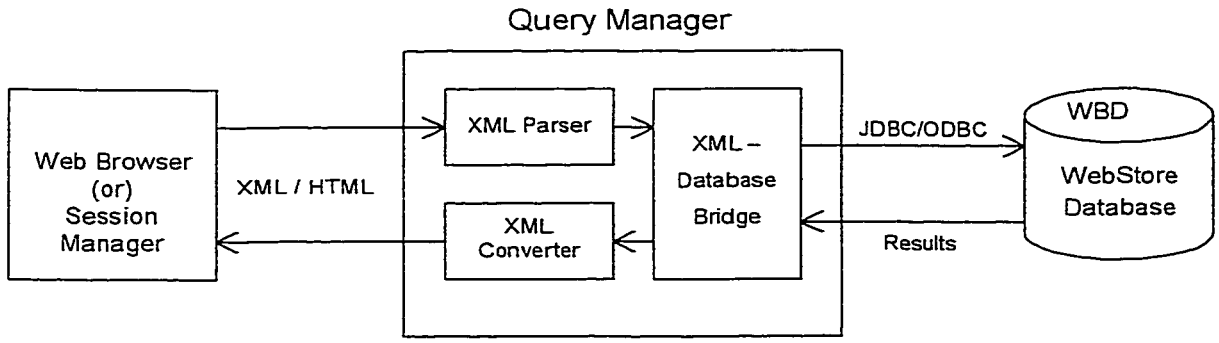


Figure 21 Interactions with the Query Manager Module

4.4.3 Universal Directory (URD)

The Universal Directory (URD) is a central repository of the product listing and the access details for all the registered vendors. URD can be a third party resource accessed by numerous independent entities in a multi-vendor framework. When the Mediator gets a request for product information, it queries the URD by passing the product category. As result the URD returns a list of vendor URLs, the vendor local products database schema, and the global schema (preferably as a XML DTD) for the given product category back to the Mediator.

4.4.4 WebStore Database (WBD)

WebStore Database (WBD) is a dynamic, virtual catalog [10] of product and vendor details. WBD stores the products and vendors' data, as well as the transactions between the shopper and the vendors. WebStore Database (WBD) is the key reference for product selection, source selection and description of terms and conditions. WBD plays an important role of accelerating the response by caching the recent and recurrent data.

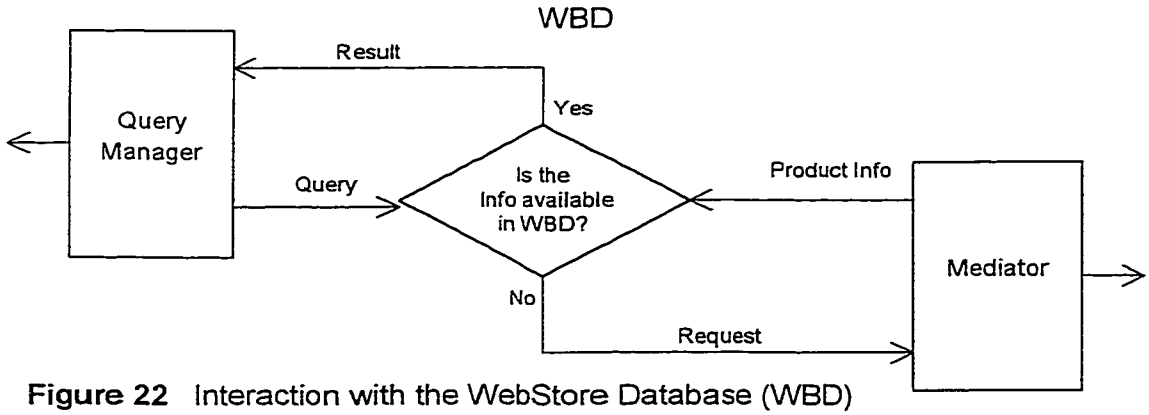


Figure 22 Interaction with the WebStore Database (WBD)

When the WBD receives a JDBC/ODBC query request from the Query Manager, one of the following two things happen: (a) the data required to fulfill the query results may already be available in the WBD database. In which case the query is simply processed and the results forwarded to the Query Manager; (b) on the other hand, the data available in WBD may be outdated[2] or incomplete. In this scenario, WBD requests the Mediator to resolve the case and provide appropriate results. The results from the Mediator will then be stored or cached by WBD before passing them to the Query Manager. Aiding Personalization Manager with product profiles for user customization is another function of WebStore Database.

4.4.5 Mediator

The Mediator is a software module, which identifies the data sources via Universal Directory (URD), and facilitates communication between the WebStore Database and the Wrapper. When the WebStore Database requests Mediator for product information, the Mediator, in turn, sends the information to the URD. URD returns to the Mediator a list of vendors who sell the desired product and their access details such as, their URLs,

[2] The data in the WebStore database may be collected via some subscription model, which may define the valid time frame for the subscribed data.

their global data source model, and their local data source model. Subsequently, the Mediator relays the who-to and how-to information to the Wrapper.

The Mediator is also activated by the Order Processing Manager, to dispatch the specific purchase orders to the corresponding vendor sites.

4.4.6 Wrapper

The Wrapper module communicates between the Mediator and the independent vendors. A Wrapper is a set of translation rules for converting the information stored in vendor's local data source model to the global data model, for the specific domain, and vice-versa. Typically, many Wrapper functions for translating various data source types would reside in the Wrapper module. For example, there may be an ODBC wrapper to communicate with SQL databases, an XML wrapper to interact with XML documents, and so on.

The Mediator may use standard protocols, like Information and Content Exchange (ICE) to communicate with its Syndicate members. The protocol standardizes the way information is communicated between the provider (Syndicator) and the requestor (Subscriber) via a Subscription. A subscription is a contract specification between two partners describing how the information should be transferred. The ICE standard makes it easier for Syndicators to deliver information in a controlled way to subscribers [URL43].

4.4.7 Personalization Manager (PM)

The Personalization Manager (PM) dynamically generates a customized HTML/XML code for the client browser. The Session Manager invokes PM and passes the userID.

PM retrieves data from user profile database, and customizes the browser view for the respective user. For a first time user, a new user profile will be created.

In addition to user profile, PM uses product profile and "nearest neighbor" profile for further customization. Nearest neighbor is an expert system profiling technique, which correlates patterns among the similar user-base, and predicts the probable items of interest for the current user. Some of the common customization techniques used to enhance the shopping experience include:

- Displaying related products, e.g. when a toy "train engine set" is displayed in online Toy store, the related accessory products like the batteries; extended product warranties or toys similar to the "train engine set" may be shown.
- Flash news about a new product. For the returning customers to a web site, a tailored list of new products in their desired category may be displayed e.g. the Lego Company has just released the latest MindStorm toy kit.
- FAQ listing for the listed product. This customization may include product descriptions, questions about the product features and terminology.
- Product ratings. This is a compilation of the criticisms and comments received from the customer's who have bought the same product. This kind of product rating helps a potential buyer to evaluate the product before actually purchasing it.

The Session Manager constantly polls the Personalization Manager to update the user interface based on the User's action. PM gathers the necessary information and data from the UserProfile database and the WebStore database to update this view. The Personalization Manager also communicates with the OPM to update the databases for future profiling needs.

4.4.8 Shopping Cart Manager (SCM)

The Shopping Cart Manager (SCM) keeps track of the items selected by the user during a shopping session. At anytime during shopping, the user may review the shopping cart status to update or delete selected merchandise. If the user changes or cancels his selection, the Session Manager will send out commands to SCM to perform appropriate actions. When the user clicks the "Check Out" button in the browser, Session Manager transmits the message to the SCM. SCM then empties the cart contents to the Order Processing Manager.

4.4.9 Order Processing Manager (OPM)

The Session Manager relays the "Check Out" signal to the Shopping Cart Manager (SCM), which in turn invokes the Order Processing Manager (OPM). OPM will interact with the SM and PM to gather additional information about the selected products, such as vendorID, price quoted, terms, shipping, and delivery. OPM passes on the purchase order data to the Mediator module. The Mediator dispatches electronic purchase orders to corresponding merchants for the products purchased by the user.

OPM also updates the User and Product Profile via the Personalization Manager for future profiling needs.

4.5. Conclusion

Chapter 4 proposed a model that integrates distributed and heterogeneous databases from different vendors. The architecture presents a novel approach to enhance the shopping experience of the user. Some of the issues resolved by this architecture include (a) integrating the multiple vendor catalog information and processing them as though the information came from a single source. The Mediator and Wrapper modules

converts dissimilar vendor catalogs to provide a unified view; (b) the information gathered from different vendors are presented to the shopper using the WebStore's own interface. This is achieved by using the virtual WebStore database; (c) finally, regardless of who the selling vendor is a prospective shopper can purchase products within a single site modeled on the proposed architecture.

CHAPTER 5

Software Design of the proposed "Data Integration" module

The main goal of this project is to illustrate the concepts of "Data Integration" when accessing information from multiple disparate merchant databases. Ecommerce applications like comparison-shopping require consolidating information from multiple sources for effective presentation of data to the web site users.

5.1. Project Objective

The objective of this project is to propose a prototype to illustrate the concepts of (a) information gathering and (b) providing consistent interface across data from multiple distributed vendor catalog databases. The six tasks identified to accomplish this objective are elaborated below.

5.1.1 Selection of the product domain

For illustration purposes, domains of *Toys* and *Music CD's* are used by this prototype. Various vendor websites contacted by the *Data Integration* prototype will offer product and price listing in one or both of the chosen product domains.

5.1.2 Design global schema for each selected domain

Most likely, the different vendor's contacted by the prototype will store their product and pricing information diversely to suit their own local data needs. When the prototype connects and receives information from the vendor repository, it must understand and adapt this received data.

To gather similar information from dissimilar vendor product catalogs and to standardize the information received from multiple vendor sites, a global data schema must be defined within the prototype. Each product domain will have an unique global schema. This way, all information from multiple vendors pertaining to a specific domain, e.g. Toys, can be consolidated and processed by the prototype. All global schema are defined as XML DTD's. Figure 23 defines the global schema for a Toy domain, while Figure 24 defines the global schema for the Music CD domain.

```
<!ENTITY % boolean        "(true | false)    'false'">
!-- top-level element, the toystore is a list of toy's -->
<!ELEMENT toystore        (toy+)>

<!-- a toy is a name followed by vendor, price, image, age, etc. -->
<!ELEMENT toy (toy_id,name,price,age,vendor,image?,description,other*)>

<!-- name is made of string -->
<!ELEMENT toy_id          (#PCDATA)>
<!ELEMENT name            (#PCDATA)>
<!ELEMENT price           (#PCDATA)>
<!ELEMENT age             (#PCDATA)>

<!-- definition of vendor structure -->
<!ELEMENT vendor          (vname,vurl,fax*,tel*,email*)>
<!ELEMENT vname           (#PCDATA)>
<!ELEMENT vURL            (#PCDATA)>

<!-- fax, tel, email, the preferred attribute signals the "default" one -->
<!ELEMENT fax             (#PCDATA)>
<!ATTLIST fax             preferred (true | false) "false">
<!ELEMENT tel             (#PCDATA)>
<!ATTLIST tel             preferred (true | false) "false">
<!ELEMENT email           (#PCDATA)>
<!ATTLIST email           preferred (true | false) "false">

<!ELEMENT image           (#PCDATA)>
<!ELEMENT description     (#PCDATA)>
<!ELEMENT other           (#PCDATA)>
```

Figure 23 The Global Schema (DTD) for the Toy Domain

```
<!-- top-level element, the musicstore is a list of CD's -->
<!ELEMENT musicstore (cd+)>

<!-- a CD is a title followed by label, genre, cost, num_disks, num_tracks, etc. -->
<!ELEMENT cd (cd_id,mtitle,label,genre,cost,num_disks?,num_tracks?,isbn,tracks*)>

<!-- made of string -->
<!ELEMENT mtitle (#PCDATA)>
<!ELEMENT label (#PCDATA)>
<!ELEMENT genre (#PCDATA)>
<!ELEMENT cost (#PCDATA)>
<!ELEMENT num_disks (#PCDATA)>
<!ELEMENT num_tracks (#PCDATA)>
<!ELEMENT isbn (#PCDATA)>

<!-- definition of tracks structure -->
<!ELEMENT tracks (track+)>
<!ELEMENT track (trackno,ttitle,sample?,artist*)>
<!ELEMENT trackno (#PCDATA)>
<!ELEMENT ttitle (#PCDATA)>
<!ELEMENT sample (#PCDATA)>
<!ELEMENT artist (#PCDATA)>
<!ATTLIST artist category (singer | composer | producer | lyrics |
                           instrument) "singer">
```

Figure 24 The Global Schema (DTD) for the Music CD Domain

5.1.3 Design database schema for participating vendors

In the prototype three vendor websites are contacted. One vendor caters to requests about toys, the second vendor website has information about music CD's and the third website caters to both the toys and music CD information requests. All vendor website's are XML-enabled, i.e. they can reply in XML to an HTTP request. Figures 25, Figure 26 and Figure 27 represents the product information from the vendor sites expressed as XML views.

Given a toy name, price and age, the Web site (www.acmetoys.ca) described by Figure 25 returns available toys as an XML document.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
    <toystock>
      <toy>
          <toyname sku="26012">Train Set (113 pcs)</toyname>
          <age>
            <age_from >5</age_from>
            <age_to>8</age_to>
          </age>
          <cost>$24.99</cost>
          <category>Power Toys</category>
          <brand>DanyToys</brand>
          <desc>
            <short>Kids can set their watches by this train. With 113 pieces
                   in all, this railway arrives with lots of track for the trains to
                   run on.</short>
            <long>Preschoolers might not be ready for an electric train set, but
                  they have the desire and the strong arms to pull the line. This
                  113-piece set includes several feet of straight and curved track to
                  build the Onion Pacific. Your little Brakeman Bill will enjoy
                  constructing a rail system and guiding the trains and cars on their
                  routes. This colorful set comes from Denmark-based DanToys,
                  specializing in molded plastic toys, all kid-tested for safety and
                  fun.</long>
          </desc>
          <image format="gif">
            <thumbnail>train_set_thumb.gif</thumbnail>
            <regular>train_set_regular.gif</regular>
            <zoomed>train_set_zoom.gif</zoomed>
          </image>
          <promo>Get 15 KidsClub points</promo>
      </toy>
    </toystock>
```

Figure 25 The Toys only Web site (http://www.acmetoys.ca)

The Web site (www.acmemusic.com) defined in Figure 26 returns the available CD's by the specified artist and song category.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
    <musicrack>
      <rackitem>
          <asid>167116454</asid>
          <album>MirrorBall</album>
          <label>BMG/ARISTA</label>
          <genre>Rock/Pop</genre>
          <duration>65:38 min</duration>
          <year>1999</year>
          <cover>mball.jpg</cover>
          <price>$19.99</price>
          <artist credit="vocals">Sarah Mclachlan</artist>
          <artist credit="guitar">Sean Ashby</artist>
          <artist creat="bass">Brian Minato</artist>
          <tracks total="16">
            <track>Building a mystery</troack>
            <track>Hold On</track>
            <track>Good Enough</track>
            <track>Adia</track>
          </tracks>
          <review> The Who's Live at Leeds not withstanding, history has shown
                   little use for live albums, a '70s dinosaur rock concept that has
                   largely gone the way of the eight-track tape. Mirrorball would
                   seem to have little purpose other than to showcase Sarah
                   McLachlan's ever-more-splendid voice, which is reason
                   enough. Mirrorball encompasses hits ("Angel," "Possession,"
                   "Building a Mystery"), rarities ("Path of Thorns," from the
                   pre-breakthrough Solace, "I Will Remember You"), and
                   crowd-pleasers ("Ice Cream"), none of which vary wildly from
                   their recorded counterparts. "Possession" is a little funkier, and
                   "I Will Remember You" a little more assured, while
                   Surfacing's frankly awful "I Love You" is little
                   improved.Mirrorball has minimal crowd noise and little to
                   suggest it isn't a studio album. It functions just as well as a
                   greatest hits package, or, given that McLachlan tends to
                   release records at a glacial pace, a placeholder in between
                   studio records. At this rate, her next one won't be due until
                   sometime in 2002.
          </review>
      </rackitem>
    <musicrack>
```


Figure 26 The Music CD only Web Site (www.acmemusic.com)

The Web site (www.acmegifts.com) described in Figure 27 returns either Toy information or Music CD information in XML form, based on the input query.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
    <giftshop>
      <gift type="CD">
            <caption graphic="mirror.gif">MirrorBall</caption>
            <prod_id>B000I4SU</prod_id>
            <prix  currency="CDN">12.59</prix>
            <brand>BMG/Arista</brand>
            <category>Roc</category>
            <details  len="long"> Grafted from McLachlan's supremely satisfying
1998 performances, Mirrorball is drawn almost equally from the multiplatinum Surfacing
and its superior predecessor, Fumbling Towards Ecstasy. (Included also is the lovely,
hard-to-come-by "I Will Remember You.")
            </details>
            <dimension>
                  <num_cd>1</num_cd>
                  <num_track>15</num_track>
            </dimension>
            <songs>
                  <song sample="mball/bam.wav">Building a Mystery</song>
                  <song sample="mball/dowhat.wav">Fumbling</song>
            </songs>
            <performers>
                  <perform>Sarah McLachlan</perform>
            </performers>
      </gift>
    <gift type="Toy">
            <caption graphic="tram.gif">Godern Ho Scale Train Set</caption>
            <prod_id>25595</prod_id>
            <prix  currency="CDN">29.59</prix>
            <brand>Life Products</brand>
            <category>Play Set</category>
            <details  len="short"> The working headlight of the diesel engine lights
your child's way into the hobby that built our country.
            </details>
            <details  len="long"> This four-unit starter train set is powered by a
mighty Union Pacific F40PH Diesel Locomotive with an operating headlight. Authentic
freight cars, including a refrigerator car, two-bay hopper and matching Union Pacific
caboose, follow the locomotive's lead. Comes with a U.L. approved power pack
providing forward and reverse mobility, plug-in terminal wires, extra couplers and a 32-
page beginner's guide to model railroading. Trac-Loc technology keeps the Golden Flyer
rails in place. As you expect from the model train experts of Life-Like, every component
in this set is backed by a lifetime warranty.
            <dimension>
                  <width>14"</width>
                  <height>1.9"</height>
                  <length>22.5"</length>
            </dimension>
      </gift>
```


Figure 27 The Toy and Music CD Web site (www.acmegifts.com)

5.1.4 Identify individual merchants for a specific domain

With numerous vendor site's available online, the prototype must know which vendor's to contact for any given product domain. To facilitate in the vendor site selection, a local knowledge base of all participating vendor's is required. The knowledge base will maintain (a) the available vendor URLs, (b) the product domains of the individual vendors, (c) the vendor product catalog XML view definition, (d) the global schema for any given product domain and (e) translation rules between the prototype and participating vendor site for communication purposes. This knowledge base will be replicated by the Universal Database in the prototype (Figure 20).

5.1.5 Define conversion rules

With the vendor databases represented in different formats, it was shown how a global data format is required within the prototype. Now to make the data flow feasible between (a) the prototype and the local vendors and (b) to gather information from multiple vendor catalogs – a set of predefined data conversion rules are necessary.

A conversion rule describes the mapping between two sets of documents in logically similar but syntactically different XML documents. In our prototype, the global schema for a domain is defined by a XML DTD and the participating vendor data is XML enabled. In Section 5.3.1 we define the XML Mapper (XML-M), a XML document to perform the actual conversion between documents.

5.1.6 Identify possible search queries

In the Toy domain, valid search product search queries are based on *toy name*, *price* and *age*. In the Music CD domain, the valid queries are based on *artist name* and *song category*. Some of the sample queries are shown in Figure 28.

Find toys with keyword *toyname='train'*.

Find me toys with *price<$20* and *age=10*

Find Music CD's by *artist= 'Celine Dion'*

Find Music CD's of *type='Jazz'*

Figure 28 Valid Sample Queries

5.2. Design Specification

Although one can access the individual merchant sites separately and get product information manually, the prototype will automatically pull information from the sites and create a consolidated list. The architecture of the data integration module in the Web application is shown in Figure 29. The different modules in the architecture are discussed below.

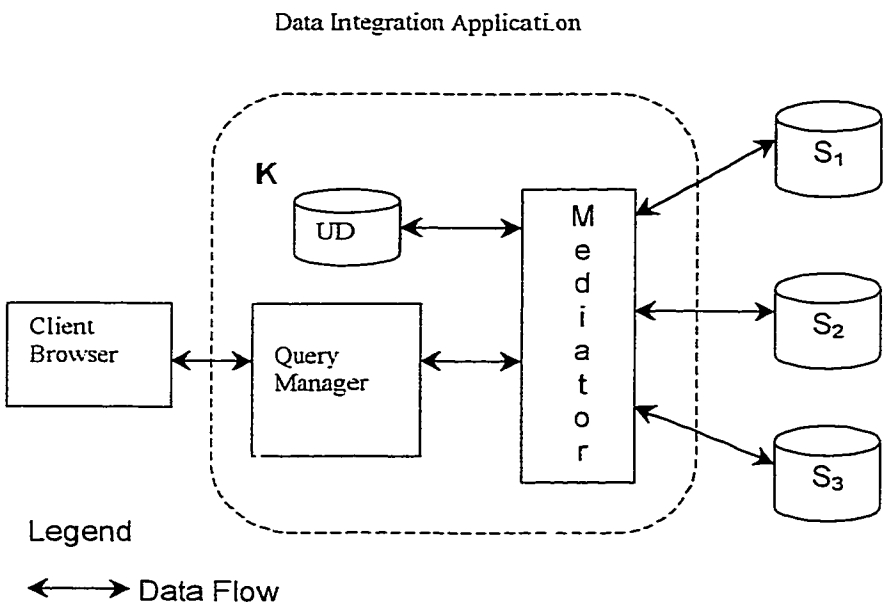


Figure 29 Architecture of the Data Integration Application

5.2.1 Client Browser

A typical user interacts with the system to gather product information from a preferred domain. The user interacts with the system K, by making HTTP requests via a client

browser. The primary interaction between the user and the application will be in the form of product search queries generated from fields in a HTML form. Anytime the user requires product information in a certain domain, he fills out a simple HTML form with values for the desired search fields and submits a request. The processed search results are presented to the user as an HTML table with optional hyperlinks to product details. The tabular results contains information regarding the selling merchant name, product name, price and availability.

The input/output dialog with the user are modeled by the following rules.

$\exists\ Q \in \{$ predefined queries $\}$

$\forall\ Q_i\ \exists\ P_i$

$P \rightarrow \{$ result set $\}$

5.2.2 Query Manager

When a search request is generated by the HTML form submission, the search field values are forwarded as parameters to the *Query Manager*. The Query Manager converts the search parameters into a XML Query document (Q_d) and forwards the request to the *Mediator* module. The search results are returned as a XML document. An XML-to-HTML conversion is performed by the Query Manager for display in the client's browser.

5.2.3 Universal Database

Universal Database (UD) is the knowledge base used to identify vendors catering to a particular domain along with the domain's global schema and conversion rules for proper data communications. UD passes the desired data to the Mediator.

5.2.4 Mediator

Mediator module takes the input search document and returns a consolidated search results from multiple vendor catalogs. The Mediator queries UD, to get a list of vendor catalogs to query. The Mediator module, identifies the vendor sites capable of responding to Q_d and translates the query document into a format understood by the local vendor site (S_k).

The system data representation acts as the global template for data communication with other vendor sites. The data translation is bi-directional in any communication between the Mediator and the sub system S_k. The results obtained from the individual vendor is converted to the global domain template. The results are joined and passed to the Query Manager.

5.2.5 Vendor Catalogs

The vendor catalogs reside in their respective Web servers and the product information are accessible using HTTP requests. The application has three vendor web sites to access information from, depending on the domain queried. The product information from the vendors is represented as XML views. The vendor catalogs represented as subsystems (**S_1**, **S_2**, and **S_3**) and their respective domains are defined as follows.

Vendor Catalog Subsystems → { S_1, S_2, S_3 } where S_k is the individual web site

D_k → { T, M } where *T* represents "Toys" and *M* represents "Music CDs" domains respectively

S_1 → { T }

S_2 → { T, M }

S_3 → { M }

5.3. Software and Technologies

The *Data Integration* application can be implemented as a three-tier *Web application*[3] using XML and Java. XML is used to extract product data from the vendor database's, while Java is used to write the application's business logic. Some of the Java technologies used in this application include:

1. Servlets, a Java alternative for invoking scripting language in the Web application. It is the server-side Java framework for Web applications that interacts with external clients and other Web applications. When a URL is requested, the associated Java class, i.e. the servlet, is executed. The servlet runs in a Java Virtual Machine (JVM) that resides in the same process space as the web server.
2. JavaBeans, the software component model for Java. JavaBeans allows the rapid development of an application by putting together existing components by using a visual builder like IBM Visual Age for Java.

The actual data integration of the different vendor catalogs (XML view) is done by the process of parsing. In XML based Web application, *parsing* is an important step and is done by an XML processor. XML processor is a software module that is used to read XML documents and provide application programs with access to their content and structure. Several XML processors written in Java are available and are classified as *validating*[4] processors, and *nonvalidating*[5] processors. In this project, we use IBM XML for Java [URL24], a validating XML processor. XML for Java is functionally very rich and

---

[3] A Web application is any application or system of applications that uses the hypertext transport protocol, or HTTP, as its primary transport protocol.
[4] A validating processor checks the validity constraints and the well-formedness constraints defined in the XML 1.0 recommendation and reports any violations.
[5] A nonvalidating processor checks only the well-formedness constraints.

has two major functions: (a) parsing and (b) generation. For ease of use within the application, we develop two wrapper beans for XML for Java.

- XMLParser bean, for parsing an input XML document
- XMLGenerator bean, for generating an output XML document

These beans provide appropriate properties, methods, and events to notify the generator that the Document Object Model (DOM) tree is ready, along with a way to pass the data. DOM is the object-based way to interface a parser with an application. Originally the W3C introduced DOM for browsers and XML implements Level 1 of the current recommendation [URL32]. XML for Java is one example of DOM implementation. Appendix A shows the interfaces defined in DOM (Core) Level 1 [URL15].

5.3.1 XML Mapper Definition

The XML Mapper (XML-M) is a set of Java class files capable of transforming one XML document into another using the DOM API proposed by W3C. The XML-M manipulates the internal DOM structure using XML for Java in three ways

- Parse the conversion rule file
- Parse the source document
- Generate a target document

The general syntax of XML-Mapper is defined in Figure 30. A "from" pattern is used to match with a part of the source document, while a "to" pattern is used to construct the target document.

```
<rules>
  <pattern>
    <from>...</from>
    <to>...</to>
  </pattern>

  <pattern>
    <from>...</from>
    <to>...</to>
  </pattern>

  ...

  <pattern>
    <from>...</from>
    <to>...</to>
  </pattern>
</rules>
```

Figure 30 XML Mapper conversion rule syntax

5.3.2 Software Requirements

Java Development Kit 1.1.7 or above

XML Processor – IBM XML4J Parser

IBM HTTP 1.3.3 Web Server for running the web application

IBM WebSphere 2.0 to provide the servlet environment

VisualAge for Java 2.0 for developing the different system modules as JavaBeans.

5.4. JavaBeans Definition

This section defines the various JavaBeans [URL15] developed for this project along with their properties, methods, and events.

5.4.1 XMLParser Bean

The XMLParser bean is a wrapper of the parsing function of XML for Java. It does parsing when its only method *parse()* is invoked. Table 7 shows the features of this bean.

Feature type	Feature name	Data type	Description
Property	fileName	java.lang.String	A property to hold the input filename. When parse() is called, the parser reads input from this file.
Property	inputStream	java.io.InputStream	A property to hold the input stream.
Event	handleXMLEvent (XMLEvent)	--	An event generated when parsing is finished. The resulting DOM tree is set in the XMLEvent.
Method	parse()	--	A method to start parsing.

Table 7 XMLParser Bean Features

5.4.2 XMLGenerator Bean

The XMLGenerator is a wrapper bean for the generation function of XML for Java. This bean has three properties and two methods and is defined in Table 8.

Feature type	Feature name	Data type	Description
Property	fileName	java.lang.String	A property to hold the output filename.
Property	outputStream	java.io.OutputStream	A property when not null redirects the output to this stream instead of a file.
Property	encoding	java.lang.String	A property to specify the character encoding of the output.
Method	generate (XMLEvent)	--	A method to start generation from the Document pointed to by the document property of the XMLEvent parameter
Method	generate (Document)	--	A method to start generation from the Document parameter.

Table 8 XMLGenerator Bean Features

5.4.3 XML-M Bean

This is a wrapper bean for XML Mapper. The bean takes two input documents – the conversion rule and the document to be converted. The features of this bean are given in Table 9.

Feature type	Feature name	Data type	Description
Method	convert (XMLEvent)	--	A method that starts the conversion of the document held by the XMLEvent.
Method	setRule (XMLEvent)	--	A method that sets the document held by the XMLEvent as the conversion rule.
Event	handleEvent (XMLEvent)	--	An event generated when conversion is finished. The resulting DOM tree is set in the XMLEvent.

Table 9 XML-M Bean Features

5.4.4 XMLServer Bean

The XMLServer bean encapsulates accesses to a Web site. The site's URL is given in the property URL. When the bean's *doIt()* method is called and parsing is finished, a DOMEvent is generated. The features of this bean are given in Table 10. Two methods, *clearParameters()* and *addParameters()*, are used to set the parameters attached to the HTTP request. The property method is used to specify the HTTP method.

Feature type	Feature name	Data type	Description
Property	URL	String	The URL of the XML document to be fetched.
Property	method	String	The HTTP method, either POST or GET, used to connect to the server
Method	clearParameters()	--	The method that clears the parameters to be sent along with the URL.
Method	addParameters (Name, Value)	String,String	The method that adds a name=value parameter
Method	doIt()	--	The method that fetches the page and parses it.
Event	handleXMLEvent (XMLEvent)	--	An event generated when parsing is finished. The resulting DOM tree is set in the XMLEvent.

Table 10 XMLServer Bean Features

5.4.5 DOMExpander Bean

The DOMExpander bean is used to consolidate query results from different vendors. Given a "skeleton" document with place holders and a set of "parts" documents, this

bean fills in the placeholders with the parts and returns the expanded tree as a separate copy. The DOMExpander bean is for replacing the placeholders in a skeleton output file. A skeleton file is an XML file with placeholders that have a XLink [URL46] like syntax:

<embed href="urn:javabean:toystore#child(2)"/>

The URN is interpreted as referring to a DOM tree by a name. The right hand side of the hash sign is an XPointer [URL47] that points to the element that is to be inserted as a replacement of this *embed* element. The example refers to the second child element of the root of the DOM tree named toystore. The features of DOMExpander are listed in Table 11.

Feature type	*Feature name*	*Data type*	*Description*
Property	skeleton	Document	The skeleton document.
Method	addEntry(evt,name)	XMLEvent, String	A method specifying a value with its name.
Event	handleEvent (XMLEvent)	--	An event generated when expansion is finished. The resulting DOM tree is set in the XMLEvent.

Table 11 DOMExpander Bean Features

5.5. Implementation

Figure 31 illustrates the system configuration of the Web application. The Data Integration application by itself, runs on a Web Server as a "Servlet". The Web server is also a client of the three existing Web applications for each of the vendor web sites. Using the different JavaBean components developed in Section 5.4 we build the *DataIntegrationServlet* using VisualAge for Java.

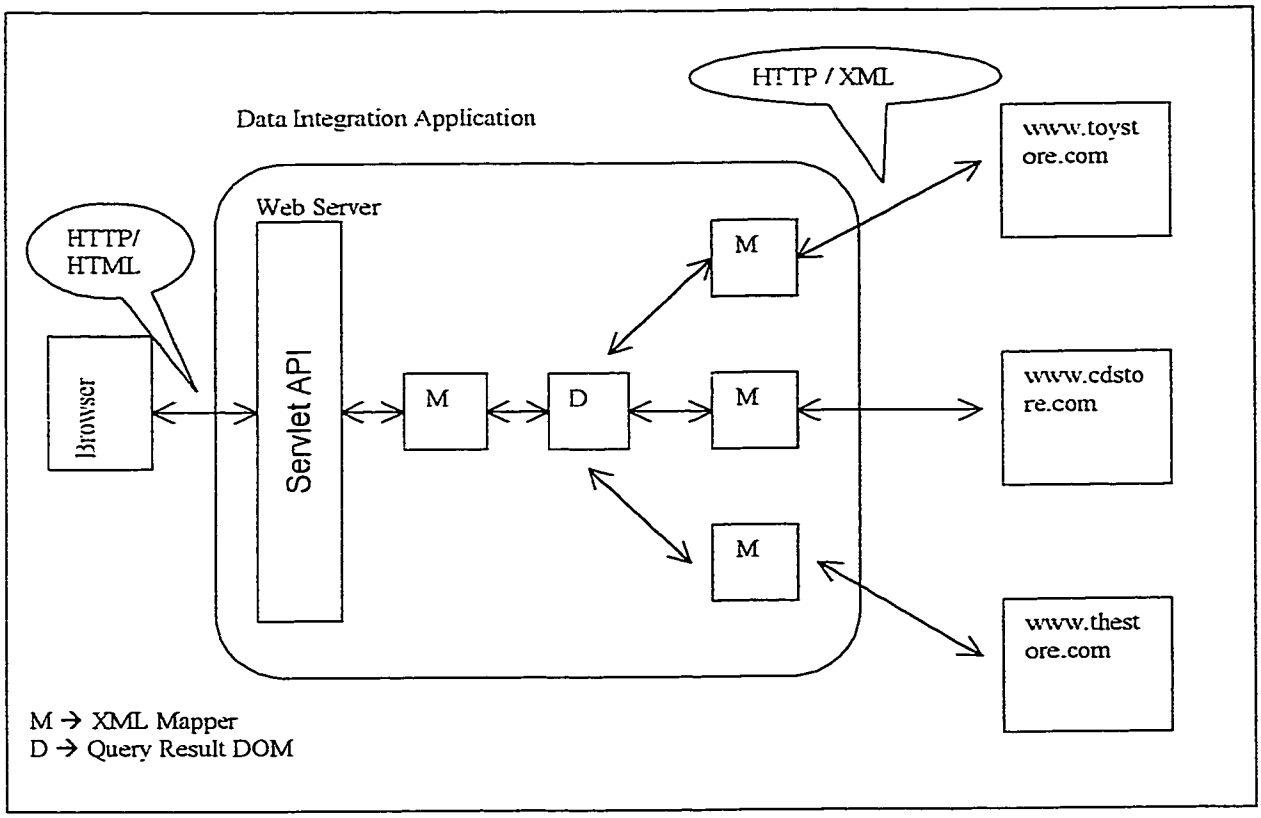


Figure 31 System Configuration of the Data Integration Application

Figure 32 represents the visual composition of the DataIntegrationServlet. When the servlet is initialized by submitting a search request from a HTML form, the following event propagation takes place.

1. The *skeletonFileName* property of the *DataIntegrationServlet* is set to the *filename* property of *SkeletonParser*.
2. The *xmlRuleFileName* property of the *DataIntegratoinServlet* is set to the *filename* property of *RuleParser*.
3. The *parse()* method of *SkeletonParser* is called, and the result is propagated to *DOMExpander1*.
4. The *parse()* method of *RuleParser* is called, and the result is propagated to *XMLConverter1*.

5. The servlet checks the domain queried, and gets a list of vendor sites to contact. For illustration purposes, the query domain is checked from the HTML POST parameters and the actual vendor sites to access and the *XMLMediator* parser rule files are hardwired. The *parse()* method of the appropriate *RuleParser* is invoked for each of the participating *XMLMediator*.

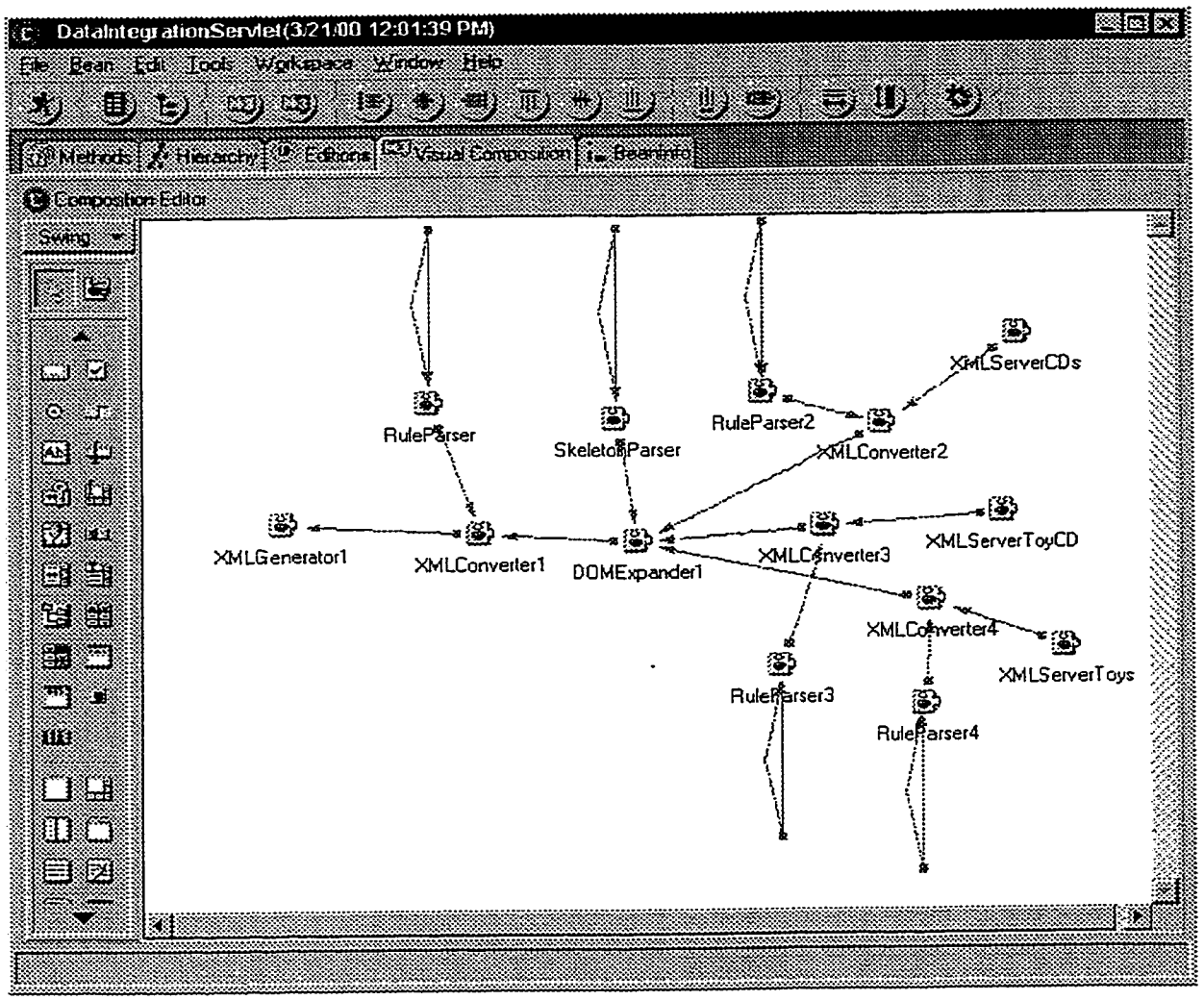


Figure 32 Visual Composition of the DataIntegrationServlet

The servlet is ready to process the POST requests. For each of the POST request, the following event propagation occurs.

a. When an HTTP POST request is received by the servlet, *doPOST()* is executed. This first sets *HTTPOutputStream* to the *outputStream* property [11] of *XMLGenerator1* and then in turn calls each of the *XMLServer* beans identified earlier in Step 5. The search fields are passed as parameters.

b. Each *XMLServer* bean establishes an HTTP connection to the specified server, retrieves the XML document, parses it, and generates an *XMLEvent* to send it to their respective $XMLMediator_k$.

c. $XMLMediator_k$ converts the received DOM to the global format using the specified rule file. When the conversion is complete an XMLEvent is generated and the formatted DOM is sent to *DOMExpander1*.

d. When *DOMExpander1* receives enough inputs (in our example, two), it makes a copy of the skeleton file by replacing the place holders with the values set in step c. Once the result is created, another XMLEvent is sent to the XMLConverter1.

e. XMLConverter1 converts the received DOM to an HTML document by using the specified rule.

f. XMLGenerator1 generates the HTML for the output stream and is displayed in the client's browser.

5.6. A Typical User Session

This section illustrates a mock up user session with a fully functional Data Integration Web application. A user accesses the DataIntegration application by making a HTTP request. Figure 33 shows a typical user *Welcome* page and Figure 34 shows a typical product information search in the Toys domain.

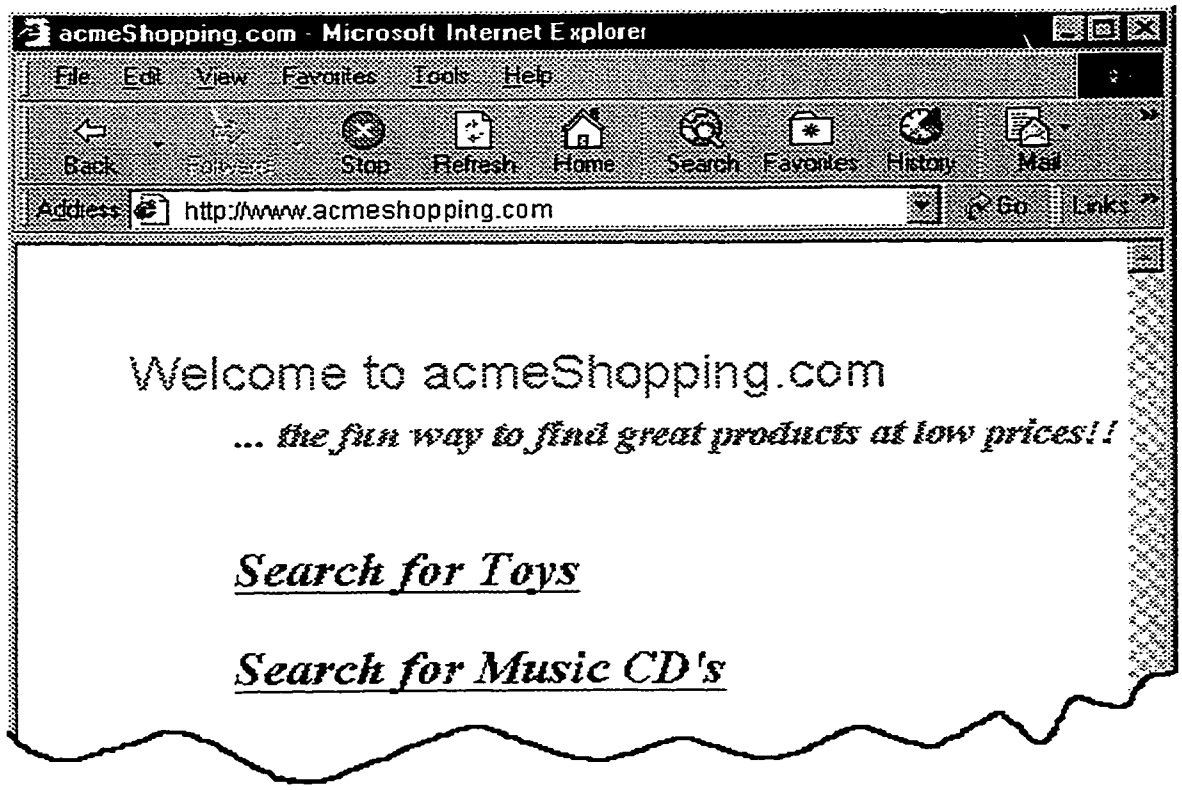


Figure 33 Welcome Page

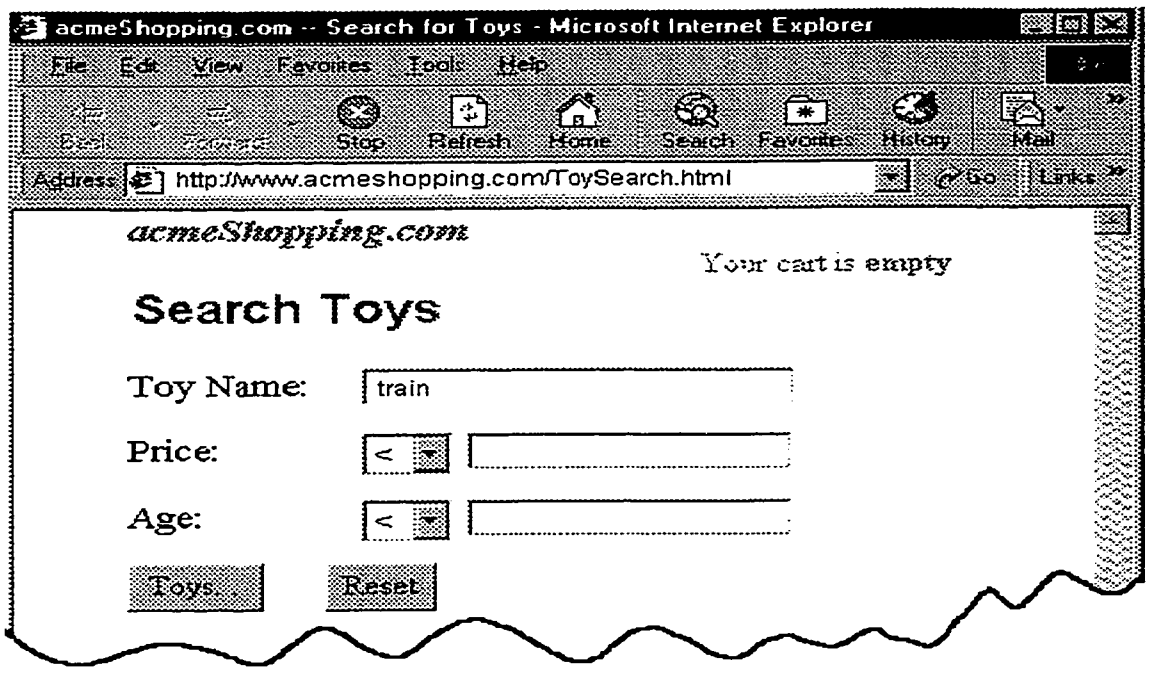


Figure 34 Toys Search Page

acmeShopping.com -- Product Details - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print

Address http://www.acmeshopping.com/detail.html?vuid=jW234joi3 Go Links

acmeShopping.com

Your cart is empty

Next >>

Train Set (113 pcs)

Product ID **26012**

$24.99

Preschoolers might not be ready for an electric train set, but they have the desire and the strong arms to pull the line. This 113-piece set includes several feet of straight and curved track to build the Onion Pacific. Your little Brakeman Bill will enjoy constructing a rail system and guiding the trains and cars on their routes. This colorful set comes from Denmark-based Dantoy, specializing in molded plastic toys, all kid-tested for safety and fun.

Qty:

Add to Cart

Sold By: acmeToys.ca

Product of: DanToys

Promotions: Get 15 KidsClub Points

Done

Figure 35 Product Details for a Toy sold at the Website acmeToys.ca

acmeShopping.com -- Product Details - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print

Address http://www.acmeshopping.com/detail.html?vuid=jW43nki Go Links

acmeShopping.com

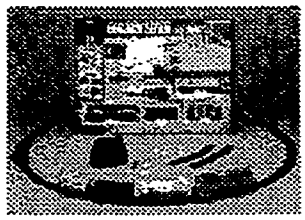
Your cart is **empty**

<< Prev

Goldern Ho Scale Train Set

Product ID 25595

$29.59



This four-unit starter train set is powered by a mighty Union Pacific F40PH Diesel Locomotive with an operating headlight. Authentic freight cars, including a refrigerator car, two-bay hopper and matching Union Pacific caboose, follow the locomotive's lead. Comes with a U.L. approved power pack providing forward and reverse mobility, plug-in terminal wires, extra couplers and a 32-page beginner's guide to model railroading. Trac-Loc technology keeps the Golden Flyer rails in place. As you expect from the model train experts of Life Products, every component in this set is backed by a lifetime warranty.

Qty:

Add to Cart

Sold By: acmeGifts.com

Product of: Life Products

Promotions: --

Done

Figure 36 Product Details of Another Toy Sold at the Website acmeGifts.com

Figure 37 shows the search query results. The results are shown as a HTML table, the product listing along with the vendor site name is displayed. The search results are compiled from multiple vendor product sources.

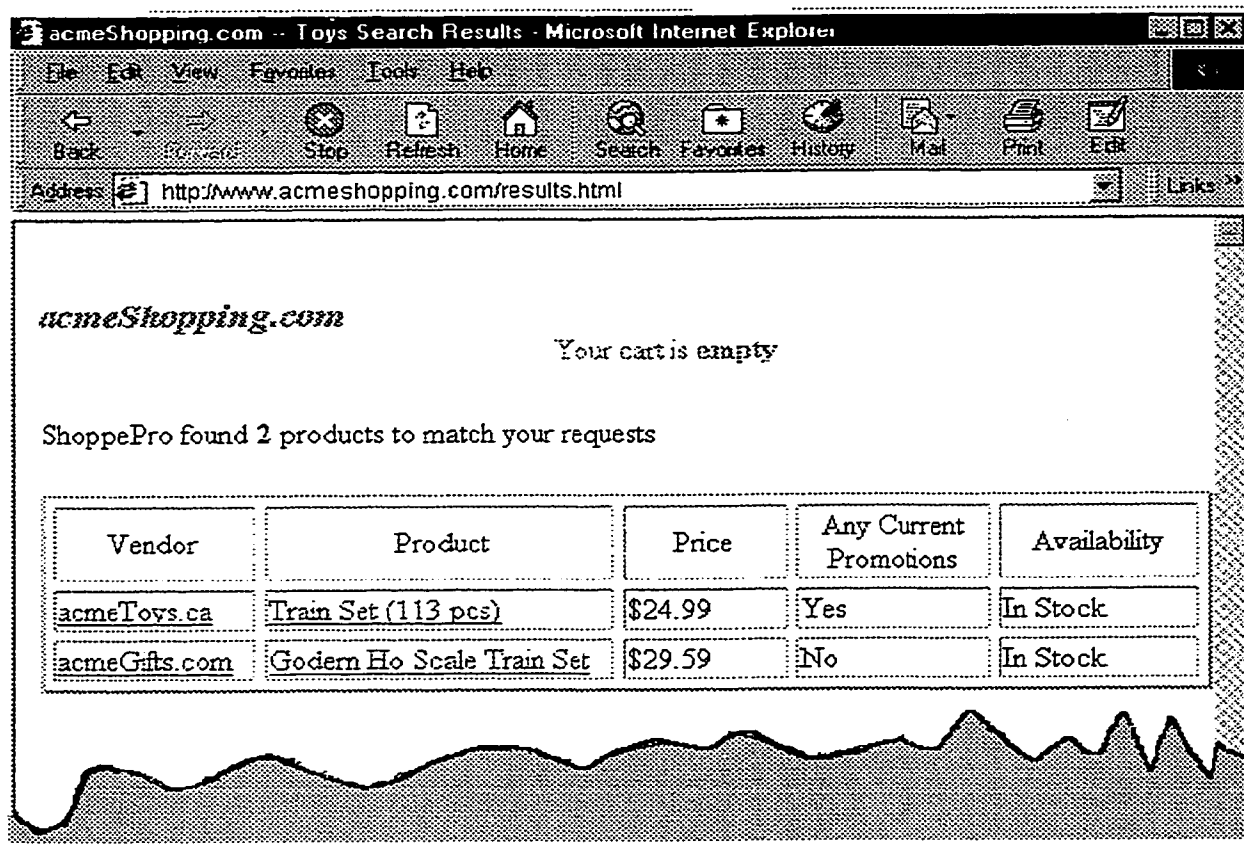


Figure 37 Typical Search Results

Figure 35 and Figure 36 shows the detailed product listing from the search result table. Even though the products are from different vendor sources, the information is displayed within a consistent and uniform interface.

5.7. Conclusion

One of the "stumbling blocks" in the world of electronic commerce, is the very nature of the underlying medium, namely the Web. Diverse marketplaces are being set up to cater

individual needs without much thought for inter-operability. As shown by the comparison shopping example, the wealth of information available on the Web quickly becomes difficult to use for an average user. This results in a fragmented shopping experience. To bridge this gap in the context of online shopping, a layered architecture for an intermediary "Data Integration System" is presented in this major report. The System integrates product information available from several distributed and heterogeneous sources or vendors.

Two primary objectives of the Data Integration system were (a) to gather product information from multiple sources or vendors and (b) to provide consistent user interface across data from these disparate sources so that the entire shopping can be transacted from a single Website. We use the emerging XML technology to gather information from different vendor websites and convert them to a common format. The problem of dissimilar interfaces among the different vendor websites is handled by presenting the gathered data using the intermediary website's interface. To illustrate these concepts a prototype was built using VisualAge for Java. The prototype handled two vendor categories used as case studies in the *Canadian Institute for TeleCommunications Research* (CITR) project.

The promise of the approach described in this project can be realized – in large scale if suitable industry standards for inter-catalog communication are employed. Specifically, the key areas where standards are imperative pertain to content description, access, and interoperability. An industry standard protocol for inter-catalog communication will build a bridge between many different database management systems used in the vendor catalogs on the Web. With the adoption of such a protocol, catalogs should be able to exchange standardized API-level catalog inquiries to integrate information from

diverse sources in spite of the many underlying differences in system platforms, operating environments, and data formats.

The report classified the evolution of electronic commerce systems into three generations based on the technology and software used. The First Generation systems based on HTML, CGI, and other scripting technologies helped tie the backend product repository with the front-end browser providing an interactive shopping experience. This generation had some inherent limitations, namely generic pages for shoppers, multiple browsing of vendor websites, and inconsistent interfaces. The Second Generation systems resolved the generic nature of pages by using advanced scripting languages like Dynamic HTML (DHTML), and intra-site agent technology. The intra-site agents provided customized interface for each shopper. The Third Generation systems employed inter-site agents to gather information from multiple vendor sites and provide data to a shopper. Facilities for browsing multiple vendor sites are still lacking and the shopper has to place distinct purchase orders for products sought from different vendors. Again because of the nature of the HTML language and the heterogeneity of the vendor information repository the agent deployment between different vendors is limited. The Data Integration System introduced in this project can be considered as the Fourth Generation electronic commerce system or the Next Generation ecommerce system. In this generation the system acts as an intermediary providing shoppers with information collected from different participating merchants. Using the intermediary web site a shopper can place a single purchase order for products procured from multiple vendors. This generation of ecommerce systems provide an enhanced and complete shopping experience to a prospective buyer.

References

1. Kalakota, Ravi and Whinston, Andrew B. "Electronic Commerce, A Manager's Guide", Redding: Addison-Wesley, 1997.

2. Colin Harrison, Alper Caglayan, "Agent Sourcebook : A Complete Guide to Desktop, Internet, and Intranet Agents", John Wiley & Sons, Inc., 1997

3 Kalakota, Ravi. "E-business: Roadmap for Success", Addison-Wesley, 1999.

4 Kalakota, Ravi, and Whinston, Andrew B. "Frontiers of Electronic Commerce", Addison-Wesley, 1996.

5 Jilovec, Nihid. "E-Business", 29th St. Press, 2000.

6. Lohse, Gerald, and Spiller, Peter. "Electronic Shopping." Communications of ACM July '98.

7. Ozsu, Tamer M. "Data Management Issues in Electronic Commerce", SIGMOD 99, Panel Description, June 1999.

8 Maes, Pattie, Guttman, Robert H. "Agents That Buy and Sell." Communications of ACM March '99.

9 Guttman, Robert H., Moukas, Alexandros G., and Maes, Pattie. "Agent Mediated Electronic Commerce: A Survey," Knowledge Engineering Review Journal, June 1998.

10. Keller, Arthur M., Genesereth, and Michael R. "Multivendor Catalogs: Smart Catalogs and Virtual Catalogs." In EDI Forum, The Journal of Electronic Commerce vol. 9, No. 3, September 1996.

11. Hunter, Jason, and Crawford, William. "Java Servlet Programming", O'Reilly Java, 1998.

URL1. Forrester Research, http://forrester.com

URL2. ePayments Resource Center, http://www.epaynews.com/statistics

URL3. EDI Aware, The Quarterly Online Magazine of the UK EDI Awareness Centres. http://www.edi.wales.org/ediaware.htm

URL4. Amazon, http://www.amazon.com

URL5. Wells Fargo Banks, http://www.wellsfargo.com

URL6. Federal Express, http://www.fedex.com

URL7. The Boeing PART page, http://www.boeing.com/assocproducts/bpart/partpage

URL8. Grainger.com, http://www.grainger.com

URL9. Reuters Group, http://www.reuters.com

URL10. The Emerging Digital Economy, http://www.ecommerce.gov/emerging.htm

URL11. Accompany Reverse Auction site, http://www.accompany.com

URL12. eBay Auctions, http://www.ebay.com

URL13. Electronic Data Interchange Overview,

http://www.saaconsultants.com/EDI/edi_overview.html

URL14. Sean Dugan, "Will the Internet kill EDI?", http://www.infoworld.com/cgi-bin/displayTC.pl?/980406sb4-big.htm

URL15. "The Power of XML: XML's Purpose and Use in Web Applications", http://www2.software.ibm.com/developer/education.nsf/xml-onlinecourse-bytitle/xmlpower/xmlpower-abstr.html

URL16. Fingar, Peter. "Intelligent Agents: The Key to Open eCommerce", http://home1.gte.net/pfingar/csAPR99.html

URL17. Hermans, Björn, "Intelligent Software Agents on the Internet: An Inventory of Currently offered Functionality in the Information Society and a Prediction of Future Developments", http://www.hermans.org/agents

URL18. Garden Escape, http://www.gardenescape.com

URL19. Autobuytel, http://www.autobuytel.com

URL20. Edmunds Online, http://www.edmunds.com

URL21. Krigel, Beth K. "Online Trading Up 150 Percent", CNET News.com,1998, http://news.cnet.com/news/0-1003-200-325600.html?tag=st.cn.1.

URL22. ECommerce and Web Marketing, http://www.wilsonweb.com

URL23. ObjectStore, http://www.excelon.com

URL24. IBM XML Parser for Java, http://www.alphaworks.ibm.com/tech/xml4j

URL25. Usable Information Technology, http://www.useit.com

URL26. "Dismantling the Barriers to Global Electronic Commerce", Discussion Paper, Turku Conference, http://www.oecd.org/dsti/sti/it/ec/index.htm.

URL27. "Internet Firewalls and Security", http://www.3com.com/nsc/500619.html

URL28. Truste, http://www.truste.org

URL29. "Data Encryption Standard Fact Sheet", http://csrc.nist.gov/cryptval/des/des.txt

URL30. RSA Security Inc., http://www.rsa.com/rsalabs/faq/

URL31. "The Canadian Electronic Commerce Strategy", Industry Canada, 1998, http://info.ic.gc.ca/icons/pdf/ecom_eng.pdf

URL32. "Document Object Model Level 1 Specification", http://www.w3.org/TR/REC-DOM-Level-1

URL33. "Electronic Payment Systems",

http://cism.bus.utexas.edu/resources/ecfaq/ecfaqd3.html

URL34. "Secure Electronic Transaction at Visa",

http://www.visa.com/nt/ecomm/set/main.html

URL35. CyberCash, http://www.cybercash.com.

URL36. First Virtual, http://www.firstvirtual.com

URL37. "The SSL Protocol", http://www.netscape.com/eng/ssl3/draft302.txt

URL38. "SSL vs. SET: Private Lives and Public Keys",

http://www.sanford.com/ism4480/notes/sslvsset.htm

URL39 DealPilot.com, http://www.dealpilot.com

URL40 PriceScan.com, http://www.pricescan.com

URL41 Powell, Thomas A. "XML: Bringing Structure to the Web,"

http://www.pint.com/Workshop/art_xml.htm

URL42 XML specification, http://www.w3.org/TR/1998/REC-xml-19980210

URL43 ICE specification, http://www.gca.org/ice

URL44 Ariba Commerce, http://www.ariba.com

URL45 Commerce One, http://www.commerceone.com

URL46 Microsoft BizTalk, http://www.biztalk.org

URL46 XML Linking Language, http://www.w3.org/TR/xlink

URL47 XML Pointer Language, http://www.w3.org/TR/xptr

Appendix A

From an object-oriented programming viewpoint, the Document Object Model (DOM) API is a set interfaces that should be implemented by a particular DOM implementation. XML for Java is one example of such a DOM implementation. Table shows the interfaces defined in DOM (Core) Level 1.

Class Interface Name	*Description*	*Implementation Classes in XML4J*
Node	The primary data type representing a single node in the document tree	Child or Parent
Document	The entire XML document	TXDocument
Element	An element and any contained nodes	TXElement
Attr	An attribute in an Element object	TXAttribute
ProcessingInstruction	A processing instruction	TXPI
CDATASection	A CDATASection	TXCDATASection
Document Fragment	A lightweight document object used to represent multiple subtrees or partial documents	TXDocumentFragment
Entity	An entity, parsed or unparsed, in a DocumentType object	EntityDecl.EntityImpl
EntityReference	An entity reference, as it appears in the document tree	GeneralReference
Document Type	A DTD, which contains a list of entities	DTD
Notation	A notation declared in the DTD	TXNotation.NotationImpl
CharacterData	A parent interface of Text and others, which require operations such as insert a string and delete a string	TXCharacterData
Comment	A comment	TXComment

Table 12 Core Class Interfaces of DOM Level 1

Class Interface Name	Description	Implementation Classes in XML4J
Text	Text	TXText
DOMException	An exception thrown when no further processing is possible	TXDOMException
DOMImplementation	A placeholder of methods that are not dependent on specific DOM implementations	--
NodeList	An ordered collection of nodes. Items in the nodeList are available	--
NamedNodeMap	A collection of nodes that can be accessed by name	--

Table 13 Core Class Interfaces of DOM Level 1 (contd....)