Enhancing Collaboration through Application Service Providers

by

Kaissar E. Gemayel

Bachelor of Science in Mechanical Engineering
American University of Beirut, 1998.

Submitted to the Department of Civil and Environmental Engineering in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Civil and Environmental Engineering

at the

Massachusetts Institute of Technology

June 2000

Signature of Author______________________________________
Department of Civil and Environmental Engineering
May 17, 2000

Certified by______________________________________
Feniosky Peña-Mora
Associate Professor, Department of Civil and Environmental Engineering
Thesis Supervisor

Accepted by______________________________________
Daniele Veneziano
Chairman, Departmental Committee on Graduate Studies

Enhancing Collaboration through Application Service Providers

by

Kaissar E. Gemayel

Submitted to the Department of Civil and Environmental Engineering
on May 17, 2000 in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Civil and Environmental Engineering

ABSTRACT

Information technology has become the strategic weapon of market leaders everywhere for supporting; influencing and transforming the way companies do business. Applications Service Providers are capable of giving organizations of any size fast, predictable, affordable access to virtually any application. As a result, organizations are free to focus on the speed and competitiveness of their operations, unfettered by hardware, software, database, network and peopleware constraints. Under the ASP computing model, even the smallest, most resource-constrained companies can access best-of-breed technology solutions to level the competitive playing field in today's global, networked economy. Fundamentally, the ASP movement is about business transformation.

This thesis intends to determine the perfect traits for an ASP to apply them to a software package developed for collaboration purposes between geographically distributed teams.

Thesis Supervisor: Feniosky Peña-Mora

Title: Associate Professor, Department of Civil and Environmental Engineering

Acknowledgements

I would like to thank Professor Feniosky Peña-Mora and the entire ieCollab 1999-2000 team who made this great experience possible. To all the friends in the ieCollab team, with whom I have gone through many enriching and fun times with this year, I wish you people the best in everything.

I would like to thank my parents and my brother, who have given me love, inspiration, and support throughout every moment in my life, for without them this wouldn't have been possible. Finally, I would also like to share my gratitude and love for my fiancée, who have made life fun, interesting, and worthwhile.

I thank you all for everything throughout the years.

Table of Contents

List of Figures

List of Tables

Chapter 1

Market Background

The ASP market is young but growing fast, taking advantage of technologies developed by a number of IT vendors. This chapter looks at the development of the market and of those technologies. It then gives an overview of some of the key characteristics that providers must develop in order to successfully deliver application services.

1.1 Thesis Organization

The aim of this thesis is to reach a conclusion or find a way that will help ieCollab team to find the most suitable and reliable Application Service Provider. In order to help achieve that, I will start my first chapter by describing the market and the importance of ASP in facilitating collaboration between teams. Next, in chapter two, I will discuss the features that a company has to consider when choosing a specific provider. For instance, what sort of application support should it look for? Chapter three will get more into the details of the offerings of Application Service Providers. I will accomplish that by talking about the features of four different ASP providers and trying to show what they provide, their architectures, their objectives and most importantly their goals. Chapter four deals with the importance of collaboration and how it is enhanced through applying ASPs, as well as a description of the nine-month project that was developed in collaboration between three geographically distributed teams (MIT, PUC and CICESE). And finally, the last chapter will denote the key elements that are most important for the ieCollab package to perform optimally. Following that is a glimpse to the future of collaboration and communication.

1.2 Four Types of Application Services

Application services have emerged from four separate developments in IT in the late 1990s. Although these four phenomena have very different origins, it seems they are all converging to a single model. Each of them offers online application services on a subscription basis. This is either as part of a lease arrangement that spreads the initial cost across monthly payments for a three- or five-year period, or as a monthly pay-as-you-go rental scheme. In the coming sections, I will talk those four strands more thoroughly

1.2.1 Application Outsourcing

Outsourcing was considered as a case where a company hands its entire information technology infrastructure to an outside provider. But today, with all the sophisticated Internetworking technologies and high-speed communications, an organization is capable to selectively outsource certain specific, individual elements only. All this idea showed up in the beginning when companies wanted to outsource their systems management. This meant that an outside facilities manager might take responsibility for just the desktop computers in an organization, or for administration and maintenance of important servers for example.

What is happening today is that companies are able to outsource individual core business applications such as enterprise resource planning (ERP), customer relationship management (CRM), or email and groupware. This is now possible due to the advancement in the practice of selective outsourcing in the information technology field.

In this way, services of well-known e-business vendors such as SAP, Peoplesoft, JD Edwards [Graham, 1999] and others can be offered for a fixed per-user, per-month subscription payment basis across the life of the contract. This allows them to avoid the high upfront commitment required for a conventional purchase, which facilitates their business a lot. I should note that setting the payment as a per-user figure has the extra benefit of relating the monthly sum to actual consumption in a transparent and predictable manner.

1.2.2 Application Hosting

In their early days, Internet service providers (ISPs) provided simply Internet access; whereas today, many of them specialize in providing servers and support for their customers to host web sites. Internet service providers are finding themselves, nowadays, hosting applications on their web server farms; since most pages and web sites are moving towards more interactive functions such as electronic commerce and customer self-service, rather than having the traditional static pages. While some of those ISP's insist on customers retaining complete responsibility for operating those applications, others find it more as an opportunity to offer by themselves the applications as managed services to their customers.

In the enterprise market, hosting providers have been drawn into providing increasingly complex Intranet and Internet servers for their customers, often taking care of the provision, implementation and management of the server rather than simply providing data center space. A number have extended the principle to other networked applications such as Exchange and Notes email and messaging systems. Today, such providers are responsible for complex, mission critical e-commerce and e-business applications on behalf of their customers [Wainewright, 1999].

Another type of providers, known as ASP infrastructure providers, have preferred to evolve their web hosting services to concentrate on managing server farms for dot-com enterprises and other application service providers.

In the always-connected online world of the Internet economy, partnership is becoming increasingly important, and few ASP offerings are the product of a single entity that owns every element of the solution. In many cases, separate providers take care of implementation, application management, subscriber management, server hosting and connectivity; but each works closely with the others to ensure the customer experience a seamless, reliable service.

1.2.3 Web Sourcing

Internet portals extend their online activities beyond content into interactive applications and they are considered as the largest volume host of applications delivered online today. Portals have been developed from the ground up as Internet-based offerings and are provided either free or at a very low subscription cost. Examples include email and collaborative applications, file and document storage, web and e-commerce site builders, simple desktop productivity tools, and others tailored to meet more specialist needs. Often, users can sign up online using a credit card or other electronic payment method and begin using them straight away.

Providers who target the business market are beginning to offer application services for instant online rental, either as a bundle or as individual choices from an online catalogue. They aim at being able to provide all sorts of applications for customers ranging from electronic purchasing to more specific professional services.

At the present stage, there are very few examples of necessary enterprise applications that are being offered in this model. This is mainly due to the fact that already established software products have to be redesigned and reengineered for the Internet delivery, while other emerging startup brands are still building up their offerings. Furthermore, customers expect a significant degree of individual configuration of enterprise applications. ASPs must spend time preparing them for delivery, imposing an upfront cost that has to be either paid in advance or recovered over an extended contract period.

1.2.4 ASP Aggregation

As more and more applications become available as online services, a new type of ASP is beginning to emerge, one that integrates multiple applications into a coherent aggregated offering. This type of ASP is very much of use to collaborative applications that can share information and interact collaboratively with each other. It must be noted that high-end ASPs are starting to build up portfolios of complementary enterprise applications, while at the entry-level, online portals are gathering together collections of applications and business services to meet a particular market need. Even some banks and telecommunication companies are becoming ASP aggregators to offer services to their small business customers.

In some cases, end users will want to develop the ability of connecting applications by themselves. This requires a very high level of technical skills. This is why a majority will prefer to leave the integration headaches to specialists, taking to heart the ASP message that promises to relieve them from the headaches of technology.

1.3 Market Potential

Although the application services market is relatively young and undefined, and thus all but impossible to size with any precision, there is a consensus among analysts that it is growing at high speed. I will try to describe its size and potential by giving some samples of reports published for this reason.

- Forrester Research projected rapid growth for outsourcing of enterprise applications. It still stands by those figures today, which projected a total market more than doubling each year to reach $21bn in 2001. Within that figure, it projected over 400% average annual growth for what it called apps rental, to reach $6 billion in 2001. This is often interpreted as the total size of the application services market, but in fact ASPs will also take share of the remaining $15bn [Forrester, 1997].

- Another report published by IDC sized the market for high-end ASPs, which it defines as those who deliver complex applications with sophisticated supporting services. It estimated average annual growth at

91%, and sized this part of the market at $150 million this year, rising to $2 billion in 2003 [IDC, 1999].

- No analysts have separated out application services activities by ISPs, but value-added services as a whole, including web hosting, already add up to a $3.9 billion business this year. IDC believes application services will be the main contributor to an average annual growth rate of around 40% for this element of the ISP business, taking it to a $12.9 billion value in 2003 (out of a total market value of $37.4 billion) [IDC, 2000].

- Analysts studying the portal market have focused almost exclusively on revenues from advertising and electronic commerce; neglecting the role that application rental is likely to play in the newly emerging business-to-business portal sector. Thus no projections are currently available that can help estimate the likely value of application services within this hot new growth area.

"Until researchers become more familiar with the ASP phenomenon, precise figures will remain elusive (today, there is not even broad agreement on what is meant by "ASP"). There has been no attempt to measure the total size of the market, but the consensus view seems to converge at an average annual growth rate over the next few years of somewhere between 100% and 400%. IDC's sizing of a narrow subset that has a relatively long sales cycle at $130 million suggests that it would not be unreasonable to estimate the total size of the worldwide application services market for the year 2000 in the region of $0.5 billion, growing to reach $1.5 - 2 billion next year, the bulk of it in North America." [IDC, 2000]

1.4 ASP Market Trends

Several separate market trends have adopted the idea of application services as a more flexible tool for enterprise computing and collaboration.

1.4.1 Acceptance of Outsourcing

The concept of outsourcing is increasingly accepted in business today. Management professionals and investors are very much interested in the merits of concentrating on core competence alone. Instead of getting into the specifics of information technology, they are now capable of bringing outside specialists to perform all non-essential functions.

Evolution in technology has enabled outsourcing to become much more selective. In a manner that allows many business now to outsource specific elements of their total IT infrastructure to an outside service provider, including the provision and operation of the data network, the monitoring of service levels experienced by users, and increasingly the provision of specific applications themselves.

The spectrum of application services now available as a managed service ranges from simple website design packages, passing through hosted email and messaging, right up to high-end enterprise resource planning (ERP) applications. This new wave of outsourced application services is typically priced on a per-user, per-month subscription basis as mentioned earlier.

Such developments have been encouraged by an increasing trend among IT departments to view themselves as internal service organizations, that wants to deal with understanding key business issues and finding solutions to them. Outsourcing allows such groups to concentrate on delivering strategic business value, leaving everyday operational issues in the capable hands of an external provider. For smaller enterprises, the issue is a lack of knowledge and resources to deploy and operate the new technologies they need to retain their competitiveness. Outside providers can bridge that gap between the possibilities business owners perceive and the practical inaccessibility of today's business applications.

1.4.2 Need for Speed

The fast-moving pace of the business and technology today puts increasing pressure on scarce skills and resources. Applications are growing more complex, the business environment is more competitive, skilled staff become more difficult to train and retain, while costs continue to be under tight control.

The IT industry's response to these factors has been to increase the extent to which it packages its products ready for use. Even high-end enterprise applications are now packaged and delivered with templates that provide an immediate starting point for implementation.

Application service providers aim to devise methodologies and templates that short-cut implementation even further, down to weeks and days. This process drives applications towards an off-the-shelf, easy-configuration state that lends itself to online delivery as an Internet-based service.

1.4.3 Commerce Gets Wired

Businesses today fully accept the need to "get wired". The need of e-commerce and e-business capabilities within enterprises, already well established in North America, is now spreading into Europe and the Asia Pacific region.

This is not just about companies having a website from which they represent static information or passively sell to consumers. Web servers have become dynamic, reacting to user input and providing an increasingly personalized experience, and in the process they have proven their ability to host a wide range of sophisticated applications.

As a result, today many aspects of enterprise computing are moving to an Internet-centric model. I will briefly describe those aspects:

- ***Electronic commerce***

  Electronic trading and communication with suppliers and customers up and down the value chain is allowing businesses to realize huge economies and efficiency gains in applications such as electronic procurement, customer relationship management and supply chain collaboration.

- ***Enterprise portals***

  Business applications are moving towards an enterprise portal concept, in which the user has access to relevant information feeds and other services as subsidiary applications running in a browser window alongside the core product.

- ***Intranets***

  The company Intranet is becoming the default platform for delivering enterprise computing. Many organizations are moving their entire computing infrastructure onto Intranet architecture, using Internet technologies to distribute applications and information within the enterprise.

- ***Extranets***

  Similarly, the move towards giving customers, partners and suppliers access to company applications via a browser interface is making web-based delivery a growing feature of external communications. In this context, larger enterprises are themselves becoming application service providers to their suppliers and customers.

Many businesses rely on outside providers to operate the infrastructure for these new computing architectures. Internet service providers and web server hosting providers who started out hosting static web servers have grown in expertise and scale as their customers' needs expanded. They are often better equipped that their customers to handle the delivery of applications across a wide area network as a 7x24 operation.

1.5 ASP Technology Trends

Various emerging technologies over the past two or three years have combined to make application service a cost-effective and accessible option today.

1.5.1 Universal IP Networking

The advent of the Internet has demonstrated the viability of a simple, open, platform-independent network protocol that delivers both content and applications to users cheaply and easily. The Internet's underlying Internetwork protocol (IP) is becoming the default standard architecture for all forms of telecommunications. Meanwhile, the laying of high-capacity fiber optic networks throughout the developed world has created an oversupply of bandwidth that telecommunications providers are eager to fill. These two trends together have converged to create an all-purpose, global telecommunications infrastructure based on IP technology, for which the cost of access is falling steeply.

The benefits of IP are not limited to cost and availability. Developers have had many years to create additional services to sit on top of the IP infrastructure, with the result that functions such as security, user profiling and network management are now highly advanced. Security is a particularly important consideration when users are accessing applications and data across the open Internet. Technologies such as the secure socket layer (SSL) in the web browser, public key infrastructure (PKI) to authenticate users and virtual private networks (VPNs) to secure transmissions, each provide varying levels of security to meet stringent user requirements. These technologies are now reliable, proven and affordable [McKie, 1999].

1.5.2 Development of Universal IP Networking

Cheaper, and more plentiful telecommunications have in encouraged the development of server-based computing architectures, which are designed for environments where client computers access centralized servers across a telecommunication link. In a server-based computing architecture, most of the data storage and application processing takes place on the central servers, while the client computers are mainly concerned with accepting instructions from the user and displaying the results. A welcome side effect of server-based computing architectures is that they tend to be easier to manage and less wasteful of resources than PC-centric approaches.

The most widespread example of server-based computing is the worldwide web itself, in which central servers hold all the content and applications, while clients access them using a web browser. Various technologies have been evolved to run applications centrally on the server such as CGI, Perl and Java, as well as Microsoft's Windows DNA (Distributed Internetwork Architecture) and a wide range of application server platforms [McKie, 1999].

Another variation on server-based computing is Windows terminal technology. This allows Windows applications to run on a centralized server, sending just the user interface across the network either to a dedicated Windows terminal or to a general-purpose we browser.

1.5.3 Distributed Systems Management

Easier connectivity has revolutionized systems management, enabling computer systems to be monitored, managed and even repaired across a phone line. This has been a major contributor to the emergence of selective outsourcing, enabling reliable remote management from a centralized enterprise operations center. Using today's application management solutions, it is now possible to track the application performance actually being experienced by a user from the opposite side of the globe, and to drill down and examine the performance of any of the individual routers, servers or applications that affect the overall experience perceived by the user.

1.6 Summary

ASP therefore is enabling companies to concentrate more and more on their core businesses and forget all about the problems or difficulties of maintaining a powerful, reliable information technology infrastructure. The next chapter will get more into the depth of what companies should be looking at in their quest for the perfect ASP.

Chapter 2

An ASP Menu for the Enterprise

Enterprises today must add online application services to the list of options they examine when considering the acquisition, new or replacement computing resources [Wainewright, 1999]. The ASP model is technically stable; it is more reasonable commercially and is increasingly accepted as a mainstream business practice. It will not always be the most appropriate route, but experience has shown that there are a number of specific applications and business environments where it is increasingly likely to provide the most effective solution.

This chapter reviews the factors that make application services most relevant to an enterprise computing strategy, and provides an overview of the range of application types currently offered by ASPs.

2.1 ASP Convenience

The importance of the ASP relies in its ability to provides a means of supplementing existing resources without significant delay or the need to commit large amounts of advance funding. It provides access to skills and technology investments that may otherwise be out of reach for small and medium type of businesses. Another issue that facilitates the usage of ASP is the fact that it is uniquely disposable, meaning that what has been added can easily be taken away again once it has served its purpose. These basic features have relevance to some very specific issues in enterprise computing today.

2.1.1 Economies of Skill

One of the most pressing issues of CIO's today is the extreme difficulty of recruiting, training and retaining staff. The headache is intensified by the accelerating complexity and pace of change in information technology.

The use of online computing services lessens the skill shortage by trying to move into external expertise and economies of scale in systems management. Most ASPs can outperform in-house solutions because of several inherent advantages:

- They can employ an advanced, automated systems management infrastructure, spreading the cost of the investment across all of their customers

- The application is already implemented and operational within their infrastructure so any teething problems have been tackled and resolved
- Since they serve the same applications to multiple customers, they have already encountered and solved most of the common setup and operational quirks of the software and can utilize proven best practices
- The cost of expensive, highly trained staff is similarly shared across many customers, and this staff has more incentives to stay with an ASP than the average enterprise is in a position to offer. They get to deal with the latest applications, a large-scale implementation and a broad variety of customer types, plus they can look forward to a career path within the organization
- As a high-volume user of the application with a recognized high level of technical expertise, an ASP can expect to have close relationships with its supplying vendors and good access to their expert technicians.

All of this resource is included in the price when applications are delivered from an ASP. In most cases, they are delivered to pre-set quality of service standards laid down in a service level agreement. The only management overhead for the customer is in monitoring that service quality to make sure that it measures up to the pre-agreed standards. No other in-house resource is required to operate the solution once it is up and running, and most application services have rapid or even instant implementation cycles [Citrix Systems, 1999].

2.1.2 Budget Friendly

Application services introduce the phenomenon of predictable costs into enterprise computing. With in-house solutions, budgeting has tended to be more of an art than a science. Although the cost of acquisition is fairly predictable, the cost of operating and managing an application is extremely difficult to estimate before building up some experience of operating the system. In the ASP model, the cost is fixed in advance, according to the number of users and the features being used. If either of those subsequently varies, the resulting change in the cost is also known in advance.

Since the payment covers both the application itself and its operation, there are no hidden extras for unforeseen glitches or breakdowns - that too is covered by the price and the service level agreement. Equally, the cost remains the same even if the user's environment is undergoing rapid change. It does not matter if the user is relocated to a different network or building, or installs new software on the client machine; whatever the case is, the application will continue to be available from the provider's remote data center in exactly the same way as before.

Implementation too is less costly than is the case with conventional methods that acquire computing. Since there is no initial payment to acquire the software or the server, it costs less to get started. There are fewer technical issues to overcome, since by definition, an online application is network ready. It can therefore be accessed without modification in any standard network ready. So the only implementation work that may be necessary is to provide integration between the new application and the existing applications that are run internally, or to configure the application to the specific business processes of the customer organization.

These implementation and other cost of acquisition factors combine to make new or specialist applications more affordable and easier to implement using ASPs compared to traditional methods.

2.1.3 A Flexible Friend

Another important point is that although more advanced applications are typically supplied by ASPs on three-year contracts, some online applications give enterprises the freedom to use them on an as-needed basis. This feature can be deployed to great effect when moving into a new business area or taking advantage of a new application capability, since the move can be tested without committing a large investment in a resource that may only be used for a short time.

In the case of more sophisticated applications, there is still scope to simply "switch on" or "switch off" new users or new functionality almost instantly. Similarly, upgrade revisions and provisional releases of the software can generally be implemented without disruption to the service that users experience. A further boon is a trend among software vendors to use the ASP model to demonstrate and configure applications prior to implementation, giving enterprises much more of an insight into the suitability and scope of the software.

2.1.4 Network Ready

Internet-based applications come into their own when the function is one that is to be accessed by users across several locations or organizations. Deploying and maintaining applications on multiple sites is complex and costly using traditionally installed software. But on the other hand, if the applications are hosted on Internet-based servers, they are immediately accessible from any web browser. The browser does not have to be on a PC or a network terminal. It could be a handheld computer, or even a mobile phone or other wireless appliance. Distance becomes no object, since the Internet is designed to allow access to any server irrespective of physical location. Likewise there are no incompatibility issues when sharing an application across separate organizations, even if each party uses completely different and incompatible local network systems, application suites and client environments. This allows better and more flexible means for future improvement and collaboration.

Enterprises can make the most of these benefits by setting up and running Internet-based applications themselves. But at the same time, delivering applications across a wide area network is a complex skill, and in the Internet environment, the additional overhead of maintaining 7x24 operation is the accepted norm. Turning to an ASP is the final step in taking advantage of the network-based architecture of the web server model. Having the application managed by an ASP is simply a matter of changing the hosting location from an internal data center to an external provider's, in the process moving it closer to the appropriate skills and resources.

2.2 ASP -Ready Business Environment

The characteristics of adaptability, flexibility and network readiness associated with application services makes them highly suited to certain types of business environments. The market experience of providers has demonstrated that ASP delivery offers concrete benefits to customers where any of the following characteristics are present in significant numbers.

- ***High growth***

  No type of business is more in need of the ability to quickly access computing resources than a business that is growing fast, particularly small businesses and those in industries with a high proportion of startups, such as technology and Internet. There is little time available to find staff, evaluate alternatives and gain expertise. Growing businesses just want to access a resource that will get the job done so they can forge ahead to the next milestone.

- ***Mobile workforces***

Better communications and advances in portable computing have increased the use of automated solutions in field sales and customer service. But in order to operate such solutions, an organization must become an expert in the provision of applications to a dial-in user base across the wide area network or Internet. Many customers have found it makes sense to use an outside provider to deploy and operate applications involving collaboration such as sales force automation and customer relationship management, using infrastructure and expertise that is dedicated to implementing and operating such applications.

- ***Changing headcounts***

When the IT organization needs to react fast to rapid changes in the user base of an organization, being able to call in the services of an ASP provides valuable extra resources. Corporate reorganization, mergers and acquisitions and expansion into new geographies or business areas all place extreme demands on an IT infrastructure, at a time when IT staff are under pressure to adapt existing systems to the new corporate landscape. Using an ASP to deploy applications relieves the pressure on internal IT staff and retains a level of adaptability to further change that purchased solutions lack.

- ***Highly distributed branch networks***

Rolling out applications to an extensive branch network is a complex, costly exercise using traditional client-server computing. Many organizations are seeking to ease this overhead by moving to server-based architectures in which branch staff access applications using a web browser interface or other thin client. The organization's central data center thus becomes a remote service provider to the outlying branches, but without the expertise in handling wide area network communications that a specialist provider possesses. Having made this move, it is a short, simple and logical step to use a third-party provider to fulfill that role, taking advantage of the provider's greater expertise and economies of scale.

- ***Extranets***

Unlike purely internal deployments, extranets require significant security preparations that further increase their cost and difficulty. An

ASP will have built the sophisticated firewall and network technologies needed to securely connect with partners and customers, making deployment that much easier.

- ***Virtual teams***

  Today's business environment is giving rise to a large number of unplanned teams and "virtual corporations", where participants are scattered either geographically or across multiple organizations, with members often joining for short periods to contribute specific components and then leaving again. ASP delivery of Internet-based shared applications overcomes these hurdles and allows such teams to work together flawlessly and cost-effectively [Wainewright, 1999].

2.3 ASP Applications for the Enterprise

The range of applications available to enterprises from ASPs grows by the day, as more and more vendors and providers enter the market. Keeping in mind that most companies are interested in E-commerce, CRM, and ERP type of applications, other categories of applications are also supplied by ASP's nowadays. The most well defined ones are messaging and collaboration, distance learning and instant applications.

2.3.1 Messaging and collaboration

It is hardly surprising that messaging should have been one of the first applications to be hosted, since its purpose is to shrink distances and thus inevitably it has been deployed among distributed groups of users that are natural candidates for ASP delivery. In the largest organizations, it is also highly infrastructure intensive and vitally mission critical. If an email server goes down, messages get lost. Few enterprises have the data center resources available to keep servers running every minute of the day on a 24x7 basis.

Early hosting of messaging has been a proving ground for application sharing. One of the challenges of the ASP model often overlooked by outsiders is the fact that conventional client-server applications are designed to be owned, not shared. Most providers have only been able to offer it on dedicated servers, unable to realize the economies of scale that are available when a single server system can be shared by multiple customers.

Despite ambitions to universal, persistent connectivity, in the real world high-quality telecommunications lines are not always available, and even where they are, they sometimes go down. Replication to a local server is often a more cost-effective means of providing for continued productivity than adding sufficient connectivity to guarantee the communications link. The ability to support satellite servers, and indeed desktops, is likely to become a necessary part of an enterprise ASP's offering.

Messaging and collaboration are obvious complementary applications to e-commerce and CRM, and the ability to offer integrated solutions will be another benefit of ASP aggregators. Indeed, one of the evaluation criteria that potential customers of ASPs should take into account ought to be the range of applications available.

2.3.2 Distance learning

The extent to which education and training is already delivered as an application service is often overlooked. It is an interesting example in that a hosting provider will typically provide its service to an education or training organization, which will use it to deliver content to trainees or students. Hence the hosted organization is the provider of the main service, with the original computing simply a delivery mechanism. However hosting providers also might have a direct relationship with the course recipients since part of its service is to register users and collect course subscriptions. The example illustrates that end users will interact with ASPs in different ways, sometimes directly, sometimes as an infrastructure provider or subscription management agent.

2.3.3 Instant applications

A growing number of entry-level and point solutions are available in the form of online rental applications. They are as useful to meet specific needs in large enterprises, as they are to fulfill core requirements in smaller organizations.

Provided ready to configure and use online, and accessed through a browser without any local install, the most common examples include:

- Website and e-commerce store designers
- File sharing and document management
- Project management
- Business administration
- Electronic marketing
- Email managers
- Teleconferencing
- Unified messaging
- Specialist point solutions

2.4 Summary

As described in this chapter, there are certain features and basic tools that should be available by the application service provider to be able to attract companies towards its services. Next chapter will layout the features of some well-established application servers in the market, this way we can visualize more specifically what those features will look like and how effective they are.

Chapter 3

Technologies Involved in Supporting Collaborative ASPs

The previous two chapters dealt with the business and enterprise side of the ASP model. In this chapter, I intend to talk more about the platforms and architectures supporting this type of technology. I will also give an overview of the most known packages in the market, as well as a brief description of the features that each provider is supplying.

3.1 Evaluating the Provider

Turning to an application service provider puts an enterprise in a position of complete dependence on a third party. This is not unusual in business, consider for instance enterprises that depend on telephone companies for their telephone service, and on banks to process financial transactions for them. But before entering into a service relationship with a phone company or a bank, a certain amount of research is prudent to ensure the provider is capable of delivering the required services to a competent and consistent level. The same disciplines apply with even more stringency to ASPs, since it is an industry where best practice has not yet been established and generally accepted. There are a number of areas where a potential client must satisfy themselves of the provider's capabilities.

3.1.1 Fitness for Purpose

The first step is to assure that the application has been properly prepared for ASP deployment. The provider should be able to show some experience and expertise in operating application services, and the service itself should be professionally presented as an ASP proposition.

It is important to establish the financial stability of the provider. It will inevitably have made substantial upfront investments in data center facilities, staff and equipment in order to be in a position to offer ASP solutions to multiple customers. That financial stability must remain secure for the lifetime of the contract in order to assure continued service.

The provider should demonstrate an understanding of the product and its use in ASP deployments. It is also important to determine that the provider has a secure relationship with the application vendor that will be sustained throughout the intended contract period.

3.1.2 Subscriber Management Infrastructure

Service level agreements (SLAs) are a vital component of any application services relationship. They determine what the client can expect in return for the subscription or rental payment, and provide a basis for resolving any disputes or interruptions to service. It is up to the client to verify that quality of service guaranteed by the SLA measures up to the needs of users. The provider must take responsibility for monitoring its performance against the SLA, and it must make sure sufficient resources are available to correct any problems before they threaten to breach the SLA limits.

If a provider does not demonstrate a professionally prepared SLA and an infrastructure that is able to support it, then the client should find an alternative supplier.

Delivering on the promised flexibility and responsiveness of the ASP model places special demands on a provider's administrative resources. Providers must demonstrate a robust, preferably automated, system for tracking subscriber accounts, with an auditable record of each change and reconfiguration. This should link into an efficient billing system that provides customers a clear and detailed invoice every month.

Where the relationship with the provider will be a long-term one, then it is important to give some thought to upgrade paths and scalability, particularly in the case of entry-level rental solutions. Many rental applications in the market today have been created to reach a volume market, but with no thought given to any upgrade path once their limits have been reached. Likewise, some providers are not set up to host applications beyond a certain size or reach. Some assessment of the provider and the application's ability to scale to the furthest potential extent of the requirement is always advisable, in order to weigh up the risk of reaching the limits at a critical point. This is particularly prudent in the case of e-commerce trials, where success can lead to sudden rapid expansion. Being forced to move to a new platform in the midst of such success could fatally undermine this type of project [NaviSite, 1999].

3.1.3 Technology Infrastructure

Since one of the reasons for going to an outside provider is in order to achieve higher standards than are possible using internal resources, customers expect high standards from third-party providers. The provider's technology infrastructure underpins the reliability and performance of any ASP solution, and should be expected to surpass the customer's own in-house setup. If a provider's facilities do not impress, it is a sure sign that the delivery will also be a disappointment [SCO, 1999]. The influential factors are:

- ***Performance***

  The data center should incorporate technology to maximize performance, with load balancing across servers, high-bandwidth links to external networks and high-specification server platforms. There should be dual redundancy in servers, network connections, power supplies and other points to ensure that, in case of a failure of one component, service can be continued.

- ***Data safeguarding***

  The provider should be able to show state-of-the-art data protection capabilities, with Raid storage or similar, frequent backup, a disaster recovery plan and high standards of physical security. There should also be a clear procedure for the client to recover data from the provider at the termination of the relationship or contract.

- ***Security***

  Various measures are necessary to protect client data. The data center should have sophisticated firewall protection against unauthorized access to servers. There must be clear procedures for authenticating user access to the applications. The provider should also operate stringent in-house security procedures to protect against infringements by employees or physical intruders.

- ***Systems management***

  A sophisticated infrastructure is required to oversee the operation of the data center and ensure that service is maintained within SLA parameters at all times. The system must be managed at the network, data center and applications levels, with the objective of ensuring users always receive a satisfactory experience when working in the application.

3.1.4 Support infrastructure

Most important of all, support infrastructure is all about determining whether the service provider can cope with something going wrong and be able to manage his way out. Even in the case of a single rental application that progresses without technically hitches for six months, if there is an invoice error that goes unresolved or the application goes down for several hours without any explanation, it poisons the entire experience.

It is essential to verify that any provider has proper procedures for dealing with support problems, including those originating from subcontractors or which might require technical support from the application vendor. Users are also entitled to expect predetermined response times to be set and met.

Provision of an effective level of service depends on the tight integration of in-house and partner resources, across sales, implementation, operations, application development, customer service and technical support.

3.2 ASP Delivery Channels

During the first year or so that the ASP industry has been in existence, the interaction was directly with the user. The industry now is shifting to a channel model, in which ASPs form partnerships with ISVs, resellers and systems integrators to achieve more extensive market penetration. Therefore an enterprise may increasingly find that its access to application services is through resellers rather than an ASP provider.

In many cases enterprise applications, which require significant implementation work, have traditionally been delivered under the application purchase model by systems integrators. While there are some exceptions, it is becoming evident that the larger ASPs are generally falling in with this channel structure, and forming channel partnerships with solution providers who will take care of consulting and implementation of the application. Once implemented, the customer relationship is directly with the ASP, but the channel partner remains available to help with implementation support issues and modifications.

On the other hand, mid-market applications that have in the past been delivered through a reseller channel are adopting a similar model, in which the reseller acts as the sales and implementation agent, and then continues to provide local support after service provision begins. A variation on this scenario is where an ASP hosts an application for an ISV, and provides the application service to the ISV's customers on its behalf.

What this means for customers is that in many instances they can continue to deal with resellers and integrators they have always dealt with for their computing needs, but can now look to those providers to act as agents for the supply of application services as well as installed solutions. They must make sure to establish that a strong, contractual relationship exists between the agent and the service provider, sufficient to assure continued service throughout the term of the contract.

As ASP provision becomes more prevalent, resellers and integrators may find a new role as ASP aggregators. It is unlikely that one provider will ever be able to satisfy all of the computing needs of a typical mid-sized business, unless it specializes in a specific industry sector. Thus IT solution providers will have a new role to play in helping clients to integrate together different application suites delivered from different service providers.

3.3 ASP Technology Platforms

The arrival of application services has been enabled by the development of server-based network computing. This allows Windows PCs, terminals and web browser to access today's sophisticated business applications running on high-powered network servers at the provider's data center.

There are various components to the technology. The ability to run applications so that they continue to perform well in a wide area network environment is vital. Under-the-cover capabilities that deal with issues such as security, manageability and system performance are also crucially important. One of the reasons customers turn to ASPs is to benefit from service levels that are higher than those they can achieve on their own resources. ASPs have to operate an infrastructure that meets high standards of security and reliability.

There are currently four leading platforms for application services in the market.

3.3.1 Citrix

Competitive advantage in the application hosting market will be won or lost by an ASP's ability to meet demand for the broadest range of applications that are both "thin" for deployability and "rich" for usability, and with the performance and reliability customers expect - all at the lowest possible cost. While other approaches for deploying, managing and supporting business-critical applications across the extended enterprise have been introduced, only the server-based computing model developed and refined by Citrix over the past decade provides today's emerging application service providers with the tools and capabilities they need to be successful. This innovative software enables application service providers to experience [Citrix, 2000].

- ***Top-line Growth***

Citrix enables ASPs to reach the broadest addressable market by delivering any application, to any client, over any network, anywhere, in the fastest time possible. Citrix allows ASPs to serve the richest range of business and productivity applications, including the latest Windows-based, client/server, mainframe, Internet, HTML and Java-based applications. Users can access these applications from virtually any device, including Intel x86- and Pentium-based PCs, Macintosh computers, Windows-based terminals, UNIX workstations, wireless devices and information appliances. And with Citrix Application Launching and Embedding (ALE) and new Program Neighborhood application browser, you can extend virtually any application instantly across the web to any standard Web browser, without rewriting a single line of code.

- ***Bottom-line Profitability***

Citrix provides robust enterprise-scale management, giving ASPs the full-range of tools needed to deliver the highest, most predictable service levels at lowest cost. Whether ASPs need to serve hundreds of enterprise customers or tens of thousands of users over the Internet, Citrix application server software solutions provide the scalability, reliability and security needed. Citrix provides centralized, single-point deployment and management of applications. Load Balancing Services extends this capability across multiple MetaFrame servers throughout the entire data center. Additional servers can simply be added to these groups for unlimited scalability, while Installation Management Services enables new applications to be deployed and replicated quickly and easily across the server farm from a single point. Server-based computing addresses the issue of security by ensuring sensitive corporate applications and data remain on the server, rather than traveling over the network. Resource Management Services provides the vital information needed to track system usage for accurate account management and billing. And Session Shadowing allows ASP helpdesks to easily troubleshoot applications and train users remotely, rather than traveling to company sites.

- ***Maximum Brand Impact***

Citrix software enables ASPs to deliver highest customer satisfaction by providing digital independence from technology deltas and complexities. Citrix application server software provides seamless desktop integration of the user's local and remote resources and applications with exceptional performance. Citrix Program Neighborhood gives users their own desktop "application portal," so they can easily browse and access all published applications hosted on Citrix server farms from a single window from their desktop or browser. Citrix software delivers LAN-like performance even when accessing the "fattest" applications over dial-up Internet connections allowing applications running remotely from the ASP data center look and feel as if they were running locally with access to local printers, peripherals and audio.

Following is a picture of how Citrix is capable of connecting all sorts of business terminals, platforms, pc's whether desktops or laptops and even enabling wireless connectivity to its services.

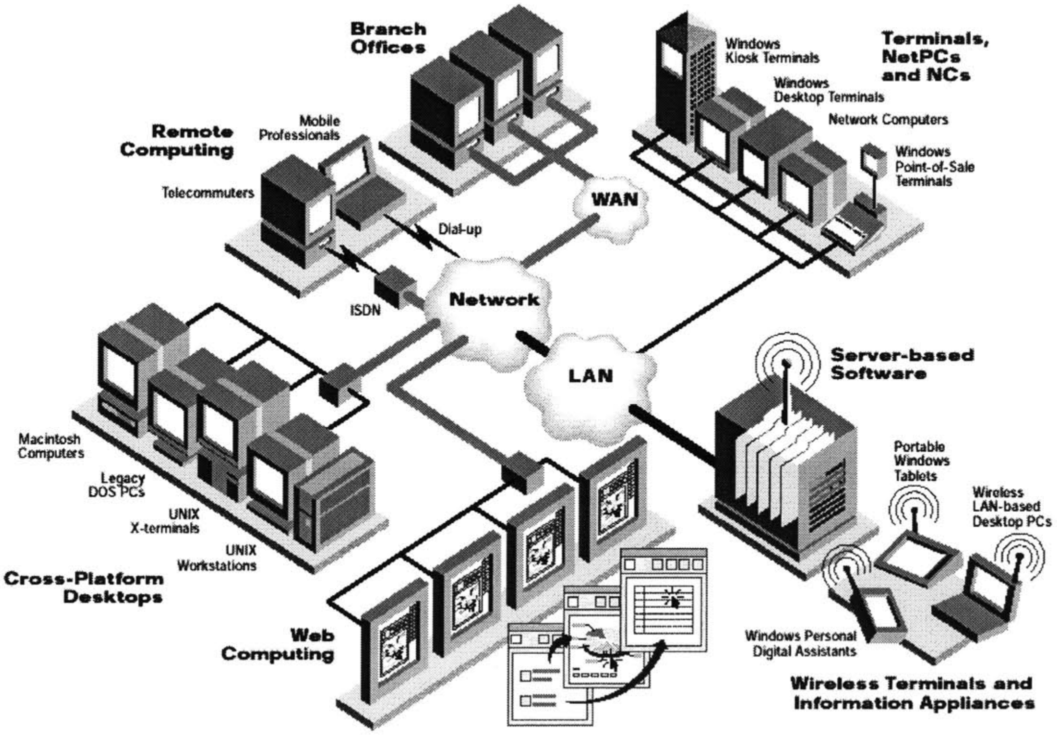

Figure 1. The architecture of the Citrix platform [Citrix, 2000]

3.3.2 Sun/Netscape

Sun's Java is a powerful environment for developing and running server-based applications. It has also introduced Jini; an intelligent network architecture for distributed applications. But both technologies are still maturing to develop the robust support infrastructure required for enterprise-class application services.

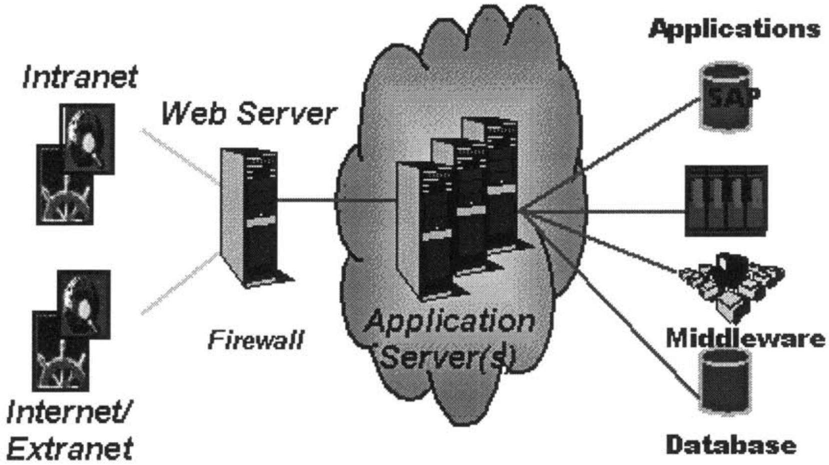

Figure 2. The architecture of the iPlanet platform [iPlanet, 2000]

The alliance between Sun and Netscape is currently known as iPlanet. The aim of iPlanet is to provide infrastructure solutions, integration solutions, e-commerce solutions and industry solutions. Our interest relies in the e-commerce infrastructure domain where messaging and collaboration is a main aspect as well as the security services provided by such an alliance. In the coming paragraphs, I will try to identify some of the features that those aspects have.

IPlanet helps an enterprise manage users and security inside and outside the firewall, and they provide a channel for communications and e-commerce transactions. Global directory services are accessible and manageable from anywhere. A standards-based architecture consolidates naming, directory, and authentication protocols from multiple sources into a single, logical database.

Messaging and collaboration software offered by the Sun-Netscape Alliance enables an enterprise's employees, partners, suppliers, and customers to efficiently share, manage, and coordinate information. Email and calendaring applications offer universal access and scalable, reliable support for all platforms via a web browser and component-based architecture scales to the needs of large enterprises. IPlanet messaging and collaboration comprises of Netscape's Messaging Server, Sun's Internet Mail Server, and iPlanet's Calendar and Wireless Server.

The iPlanet Portal Server enables community creation and management; it is also capable of achieving multi-tiered portal personalization that facilitates delivery of integrated content, applications, and services through customizable portal channels. Most important is that it allows secure extranet access to portal by mobile/remote employees, suppliers, and partners that require no additional client software to install and maintain. Another feature includes its ability to integrate native non-web-based Windows NT, Unix X-Server and Mainframe applications. And of course, like all other servers, it incorporates a highly scalable and reliable architecture [DHBA, 2000].

3.3.3 Oracle

The Oracle Internet Platform is built on three key pieces: browser-based clients, application servers, and databases. The browser-based client processes presentation, while the application servers and databases process business logic and data. The figure below shows how these key pieces interact with each other.

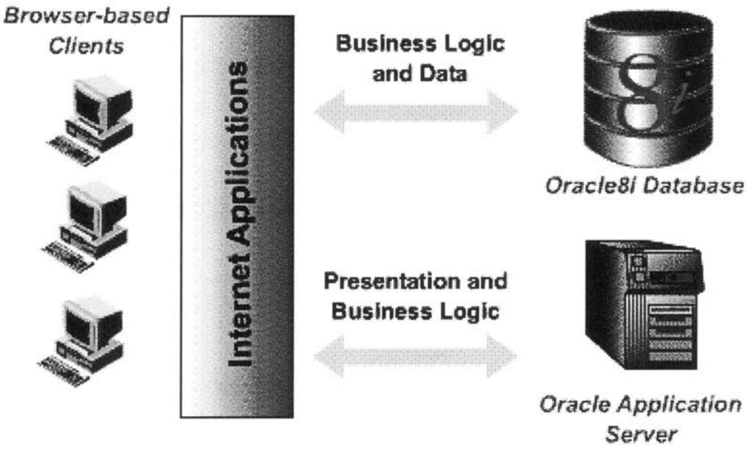

Figure 3. The architecture of the Oracle platform [Oracle, 2000]

Along with clients, application servers and databases, an important element of the infrastructure is network services. Network services such as security and directory provide standard protocols and interfaces for operations over the network.

Some examples of these protocols and interfaces are:

- SSL (secure sockets layer) and X.509 certificates. These are industry standards that are used to implement single sign-on (SSO) across the network

- LDAP (Lightweight Directory Access Protocol) for accessing the Oracle Internet Directory

- TCP/IP, HTTP, and CORBA IIOP for network communication protocols

Oracle Internet Platform contains two additional parts known as System management and Development & Decision support tools. System Management provides services for managing users, applications, and the Internet platform itself. Development tools enable customers to build Internet applications, using component technologies like Enterprise Java Beans and Java Servlets, which can then be deployed with HTML or Java clients. Decision support tools aid in publishing reports and performing ad hoc query and analysis on data. There's one other interesting element that the Oracle Internet Platform provides which is the flexibility to be able to support for an e-business application framework.

Oracle Business Components for Java is argued to be the only Java application framework to radically enhance developer productivity and ensure scalability for enterprise applications.

Today, developers typically spend a significant amount of time writing business logic. Business Components for Java enhances productivity by enabling developers to reuse and customize the same business logic across multiple applications. With the industry shift towards purchasing packaged applications and away from completely custom development, the ability to tailor packaged applications becomes essential for customers to achieve an exact fit to their business needs. Business Components for Java exploits XML's powerful ability to describe data in a structured way and leverage it to manage an application's metadata. As a result, customizing an application becomes as easy as editing XML information.

Oracle Business Components for Java promises to increase software development ROI by shielding the developer from some of the low-level details, allowing them to focus on developing business rules for an application.

Another important feature of the Oracle Internet Platform is the ability to decouple applications from proprietary operating systems. On the Internet, applications are written to the Internet programming interfaces, operating system APIs. Operating systems manage system and network hardware. Platform management tools such as Oracle Enterprise Manager manage the whole stack. The benefits are better application portability, more application scalability and less proprietary lock-in.

To maximize performance on the Internet, business logic must be partitionable on any of the three tiers whether it is the client, the application server or the database. To do this, business logic is written using a single language, Java, and uses business objects that can rely on a consistent set of services offered by the Internet platform across all tiers. Applications written directly to an operating system:

- Typically must be written in different languages for each tier, depending on the operating system
- Are not easily repartitioned among tiers
- Cannot depend on a consistent set of application services from every operating system

Oracle is a vendor that can provide comprehensive, integrated products for e-business from the applications through the Internet platform itself. This means customers can quickly develop their e-business solutions and collaborative applications using off-the-shelf technology [Oracle, 2000].

All of this is built in an open systems framework that enables IT managers to customize their solutions with specialized software, when desired [Oracle, 2000].

3.3.4 Lotus

Lotus was the earliest vendor to begin supporting hosted applications on its Notes and Domino server platforms, but its focus at the beginning was on entry-level rented applications. It then managed to attain a position in the world of ASP's.

Notes R5 is the cornerstone of Lotus' client family. Very good e-mail, calendaring, group scheduling, Web access and information management; all integrated in an easy-to-use and customizable environment. It has the ability to integrate with enterprise systems, it is optimized for collaboration by providing comprehensive application services like workflow and messaging, so you can easily build and manage integrated, collaborative solutions. Deployment and maintenance made easy by server-to-server replication, maintenance and rollback applications. It is open for a lot of choices such as HTML authoring tools, Java IDEs and scripting tools to create Domino applications. Lotus also uses Microsoft IIS as the HTTP engine for Domino. And finally, it allows the ability to keep track, report and analyze system usage for billing, charge back and capacity planning purposes.

The Lotus Domino Application Server is an open, secure platform optimized to support rapid delivery of collaborative Web applications that integrate your enterprise systems with dynamic business processes.

Domino Application Server enables you to further leverage current Web investments. Domino Enterprise Connection Services (DECS) provides rapid connectivity to enterprise data. CORBA/IIOP support lets you integrate Domino with your applications architecture. Support for Microsoft IIS brings Domino's rich application services to NT Web environments.

A comprehensive development environment, the Domino Application Server lets you move beyond static Web sites -- to create high-value business solutions that include workflow, content management and highly flexible security.

The Domino Application Server is a well-established Web application server due to the following reasons:

- ***CORBA/IIOP support***

  Extends Domino application services to Web clients, for integration with your existing applications architecture. Serve Lotus Notes clients and Web browsers with the same applications.

- ***Flexible, pervasive security***

  Personalize access to data and applications based on individual and group roles. Extend Domino security to HTML files and other data, for pervasive security no matter how or where Web content is stored.

- ***Integrated X.509 support***

  Register new users with Notes and/or X.509 certificates. S/MIME support ensures message integrity for all client types. SSL V3 for IIOP and LDAP clients. Authentication via trusted third-party directories reduces complexity and duplication of information.

- ***Enhanced HTTP stack***

  The Domino R5 HTTP engine delivers outstanding performance and Java servlet support.

- ***Integration with Microsoft IIS***

  Use IIS as the HTTP engine for Domino, to dramatically enhance IIS security and bring Domino's rich Web application services to your NT-based Web environment.

Domino Application Server includes Domino Enterprise Connection Services (DECS), for live access to enterprise systems. It supports a wide range of enterprise systems, including DB2, Oracle, Sybase, ODBC, EDA/SQL, SAP, PeopleSoft, JD Edwards, Oracle Applications, MQSeries, CICS, and more. The Domino Application Server also delivers powerful administration and Internet messaging functionality found in the Domino Mail Server such as e-mail, calendaring and group scheduling, bulletin board and newsgroups.

Another emerging Lotus product aimed for collaboration is iNotes. It delivers the powerful messaging, e-business applications and mobility of the Domino server to web browsers and extends Domino's scalable, reliable messaging capabilities to Microsoft Outlook users. iNotes also includes off-line support for better mobility and a simple deployment model with a no-touch install. iNotes lets you standardize your server infrastructure on Domino while giving your users the flexibility they demand.

Mobile Notes delivers Domino's leading messaging and collaborative applications to mobile and wireless devices. Users will benefit from anytime access to key information including e-mail, calendar and directory. Mobile Notes is the client access license used to access the Mobile Services for Domino (MSD) server and will be available in the first half of 2000 when MSD 1.1 ships. With MSD 1.1, users will get broad coverage and support for Internet/WAP phones.

3.4 Data Storage

Transactions occur whenever customers order products and materials, send or receive shipments, and so on. Therefore, business computing is highly considered transaction-oriented. The larger the number of transactions implies that the more critical the business computing solution becomes. The main key technology behind efficiently managing transactions is the database engine and the data storage value. To help maintain data integrity, RAID (Redundant Array of Inexpensive Disks) technology is commonly used to prevent loss of data due to a simple failure of the hard disk drive. In the case of a significant disaster (fire, flood, stolen server, etc.), backups are also critical to the solution.

The server is the critical tool by which users access the data. As databases grow over time, it's important to select a server that will scale up to meet the needs of a business solution both today and in the future. The solution designer must provide a system that can scale up to handle additional loads as the organization grows. An adequate system for today may not even come close to what is required a year later, after an organization has grown. Therefore, data storage becomes a critical factor.

One of the main suppliers for Internet data storage is EMC Corporation. EMC Storage Solutions are in control of today's Internet info-structure, and is considered as one of the leading storage systems for application sharing in the market [EMC, 2000].

3.5 Summary

After describing all different aspects of the ASP model and discussing few of the most well known ASPs in the market, I am now ready to discuss the nine-month project that took place through collaboration between three geographically distributed teams. I will describe the project in detail in chapter 4 and will conclude in chapter 5 by giving a recommendation regarding adopting ASP technology and which would better fit for ieCollab.

Chapter 4

Case Study: *ie*Collab

Collaboration on different tasks between geographically dispersed teams has been a challenge for all sorts and scales of projects. Nevertheless, communication and interaction among team members can be facilitated by careful implementation of a Computer-Mediated Communications System (CMCS) [Stasser et al. 1989]. In this chapter, I intend to discuss the importance and the need for efficient collaboration and I will try to give a description of the 9-month project at MIT that was devoted to solving the problems of collaboration within geographically distributed systems.

4.1 Collaboration

The best way to define collaboration is by describing the concept of the virtual organization. The virtual organization can be defined as a group of companies forming an alliance through the use of Information and Communication Technologies (ICT) to collaborate in the production of a joint product; the location of the group of companies may be distributed locally or globally but they appear to function as single organization.

For a team to work effectively they must be provided with a set of common tools so that they can collaborate effectively. This can be through the Internet, through a closed company Bulletin Board System (BBS) set up for a particular event, or through an in-house network system with remote login capabilities. Most companies do not possess the necessary specialist skills needed to create their own "virtual working environment software." Therefore, provision of "off-the shelf" groupware, video conferencing equipment/software are important factors that decided whether a company can "go virtual" [Stasser et al. 1985, 1987].

The main criteria of the virtual organization is perhaps to have an organization with many locations (within or outside the national boundaries) and the need to communicate between those locations, to share information and to work collaboratively on that shared information to produce joint products, with the use of ICT as the facilitator.

4.2 Importance of Information Exchange

In order to be effective, groups must exchange information effectively. Members of a group will have a larger pool of information (for making decisions and for supporting other group processes) than lone individuals. Not only can group members help one another recall information that an individual member may have overlooked, they may also bring unique information previously unknown to others in the group. Thus, groups are potentially able to make more informed decisions than individuals if they effectively exchange information. However, studies of information exchange in groups suggest that both virtual and face-to-face groups fail to take advantage of their information advantage [Hightower et al. 1995, 1996; Stasser et al. 1989; Stasser et al. 1985, 1987]. Groups tend to concentrate on common information and regard group discussions as a means to negotiate consensus rather than exchange information.

4.3 Effects of Collaboration on Decision Making

A group decision results from interpersonal communication among its members. This interaction serves a variety of functions but one of the most important is information exchange. The information exchanged in a group is drawn from a pool composed of two elements: unique information known to only one or a subset of the group's members, and common information known to all group members. Effective information exchange depends on sharing unique information. For this sharing to take place, an individual must recall the information, have the opportunity to mention the information and be willing to mention it.

4.3.1 Information Recall

One factor affecting the group's ability to recall information is the frequency with which a person has been exposed to an information item [Stasser et al. 1989]. More exposure results in easier recall. Other factors are individual preference, familiarity with the information and amount of information. Preferred consistent information is more easily recalled [Stasser et al. 1987]. Similarly, familiar information items are more likely to be remembered. Finally, large amounts of information reduce the likelihood of any particular item being recalled [Stasser et al. 1989].

4.3.2 Opportunity to Contribute

Opportunities to contribute information are affected by time constraints and information load, unreliable communication channels, and social influences. Time constraints or a high information load may reduce the opportunities to mention information. Group discussions tend to follow themes, "rehashing" the same information until a new theme arises [Lamm et al., 1973]. This results in an unproductive use of time and reduces the amount of information that can be contributed. Additionally only one person can talk at a time further limiting the information exchanged [Diehl et al., 1987]. When time becomes too constrained or information loads too high, some of the information may never be contributed.

Unreliable communication channels can limit a group member's opportunity to communicate on a timely basis. Classic examples are asynchronous communications systems such as e-mail. When using e-mail, the sender relies on the receiver to read their e-mail. In this case feedback is not immediately provided and the sender is not certain if the message was received and understood.

Social status may also limit a group member's ability to mention information. Often, lower status members are not given the same opportunity to speak and may find themselves bystanders to a discussion among higher ranking group members [Weisband et al. 1995].

4.3.3 Motivation to Contribute

Even after information has been recalled and an opportunity to contribute is available, a group member must choose to mention it. An individual may choose not to mention information for a variety of reasons. First, people are reluctant to mention information that counters prevailing group's sentiment [Hartwick et al. 1982]. Comments may be biased to match the perceived preferences of the audience. Generally, the first person to speak has an inordinate influence on the group's discussion and ultimate decision. Secondly, people tend to take an advocacy role with respect to their own preferences and are less likely to mention information that undermines those preferences [Gigone et al. 1984; Stasser et al., 1985]. The third factor that may affect an individual's motivation to contribute relates to the member's level of commitment to the group. Individuals with little interest in the group are usually less willing to contribute.

4.3.4 Effectiveness of Information Exchange

The difficulty arising in achieving complete information recall and having enough opportunity and motivation to contribute when being involved in group work results in a situation where unique information may be mentioned less frequently than common information. Stasser et al. (1992) found if a group member mentioned a common information item during discussion, the item would probably be mentioned again during the group's discussion. In contrast, even when previously mentioned, unique information was rarely repeated. Stasser et al. speculated that by mentioning common information, a group member increases the item's salience to other members since they were previously exposed to it. However when a unique information item was mentioned, most group members had not been previously exposed to it. Therefore even after being mentioned, unique information is less salient than common information and is less likely to be recalled later. As a result, common information dominates discussion and has more influence on group decisions than a unique information [Gigone et al., 1993]. This means groups can fail to capitalize on their potential information resources.

4.4 CMCS and Information Exchange in Virtual Groups

Virtual groups may use synchronous or asynchronous CMCS to facilitate information exchange [Stasser et al. 1989]. Synchronous computer conferencing systems (SCCS) permit group members to communicate by typing their comments into the computer and then broadcasting the comments to other group members. The system is interactive in the sense that the group members "meet" at the same time although they are physically separated. SCCS are one of the more restrictive forms of the CMCS in terms of the communication modalities it allows. Three characteristics of synchronous CMCS affect information exchange in virtual groups: reduced communication modalities, more uninhibited communication, and more equal participation [McGarth et al., 1994].

4.4.1 Reduced Communication Modalities

Many of the cues present in face-to-face discussion that help regulate the flow of conversation, provide feedback and convey subtle meanings that are not present in virtual groups. The absence of such cues reduces the amount and richness of the information discussed [Hightower et al., 1995, 1996; McGarth et al., 1994].

A number of studies have shown that the total amount of information discussed in virtual groups is less than that in face-to-face groups. Without para-verbal and nonverbal cues, virtual group members are not able to duplicate the normal give-and-take of face-to-face discussions. Comments appear to be sometimes out of context or the conversation may appear to lack focus because multiple group members are "talking" at once. This is exacerbated by the fact that people type and read at different rates. Virtual team members who type slowly or edit more thoroughly may find their comments are no longer relevant when they are ready to transmit them, Moreover, because every group member can transmit their comments simultaneously, group members may be required to process a large number of comments in a short period of time.

Another factor, which reduces the amount of information that a virtual group can discuss, is simply that members must type their comments, Siegel et al. (1986) attributed part of the inefficiency they observed to the fact that using a keyboard to enter comments takes longer and requires more effort than speaking.

The lack of nonverbal and para-verbal cues also reduces the richness of the information transmitted by virtual group members. Daft et al. (1986) define media richness as the ability of information to change understanding within a time interval. The richness of a medium is determined primarily by the communication modalities it allows. Rich media allow multiple feedback and information cues, for example the words spoken, the tone of voice, and body language. It takes more effort by virtual group members to achieve the same level of mutual understanding in a lean medium, such as CMCS, than a rich one such as face-to-face communication.

The effect of reduced amount of information exchange and lower media richness is that virtual groups may find it more difficult to communicate than face-to-face groups. Virtual groups usually take longer to complete tasks and are less likely than face-to-face groups to reach a consensus [McGarth et al., 1994]. Virtual groups have also been found to spend more time coordinating their activities relative to face-to-face groups [Hightower et al., 1995]. These factors may make it difficult for virtual groups to exchange information effectively.

4.5 Increased Inhibited Communication

Research has shown that virtual group members are more likely to express their opinions and engage in extreme behavior towards others in virtual groups than in face-to-face groups. Siegal et al. (1986) suggest that "anonymity and lowered salience of social controls...leads to feelings of loss of identity and uninhibited behavior." Group members are more likely to respond to others emotionally in virtual groups because they concentrate more on the content of messages than on the message senders.

One result of uninhibited communication is that conflicts are typically more pronounced in virtual groups than in face-to-face groups. In face-to-face groups when the first member speaks, succeeding speakers tend to agree with the first person. This is because group members are often reluctant to contradict the prevailing sentiment in the group [Hartwick et al., 1982]. However, this is not the case in virtual groups. Virtual groups contradict one another more readily because they feel freer to express opinions [Weisband, 1992]. A willingness to contradict one another and respond emotionally often leads to deeper conflicts in virtual groups than in face-to-face groups.

The effect of uninhibited communication on information exchange is probably positive within certain bounds. A lively debate would presumably increase information exchange but name-calling may reduce task-oriented information exchange.

4.5.1 Increased Participation by Group Members

A third CMCS characteristic that may promote information exchange is that CMCS have been found to promote more equal participation by virtual group members. In face-to-face conversation when one person speaks, others must listen. When one person or few people dominate the discussion others don't get the opportunity to speak. People who are more talkative or who have higher social status tend to dominate group discussions [Weisband et al. 1995]. With virtual groups, members can "talk" at the same time so every group member has equal opportunity. CMCS may also reduce the effect of differing social status because they mask physical and social cues by which people judge others.

The extent to which these three CMCS characteristics affect information exchange probably depends on the task faced by the virtual group and the experience the group has with the medium and with each other. The communication difficulty that virtual groups experience because of reduced communication modalities can have a detrimental effect on information exchange. Siegel et al. (1986) suggest that virtual group members may sometimes get frustrated trying to support their viewpoint and will simply state their preference with minimal supporting information. This type of behavior would tend to reduce the effectiveness of information exchange. Previous studies suggest that information exchange is less effective in virtual groups using a CMCS [Highertower et al., 1995]. However, Hollingshead et al. (1993) found differences between face-to-face and virtual groups may depend more on experience with the CMCS and with group membership than on the task facing the group. Those authors were not concerned with information exchange but their findings suggest that differences in information exchange between virtual and face-to-face groups may narrow as the group members become more familiar with the CMCS and with one another.

4.6 Project Description

As a means to try to solve the above-mentioned problems, the idea of the Intelligent Electronic COLLABoration (ieCollab) was very evident as a necessity. The intention was to develop the software through collaboration in distributed teams. The teams comprised of 24 students of various backgrounds from Massachusetts Institute of Technology (MIT), 6 computer science students from Centro de Investigacion Cientifica y de Educacion Superior de Ensenade (CICESE) which is a graduate university in Mexico, and 4 computer science students from Pontifica Universidad Catolica de Chile (PUC) which is a university in Chile.

The project was undergone through coordination between all three geographically distributed teams. The first semester had meetings that were controlled by three professors and three teaching assistants, the objective of those meetings was to familiarize us with the different roles involved in the software development cycle. But at a later stage, mostly the second semester, the students were in control of the amount and purpose of meetings required.

4.7 Project Objectives

Two goals were sought to be achieved from ieCollab project. It was intended to teach the students about the various roles that interact within each other throughout the whole software development cycle. This included having students face the difficulties of collaboration with other teammates that lived in different time zones and that had different cultural backgrounds. The second objective was to come out with a Java-based collaborative software architecture that allows conducting real time meetings for virtual teams that can enable document and application sharing facilities. The software aimed at stressing on social interaction and casual contact features which differentiated it as a new technology for enhancing human interaction in virtual meetings.

4.8 Project Process

To achieve a specific, definite and reliable package, a well-knit process has to be designed according to the several factors. Considering that ieCollab is a unique package, it is best to portray the whole process by describing the first the available resources, then defining accurately the lifecycle model chosen, and finally the roles that denoted each stage of the cycle.

4.8.1 Resources Available

Resources for ieCollab are divided into three main categories

- ***Human resources***

  The project, as mentioned earlier, 34 students. Most of these students had a primary and a secondary role. Below is a listing of the people involved in the development along with their corresponding primary and secondary roles. It is worth mentioning that almost everyone also contributed in the coding process of ieCollab since most of us lacked a solid technical background.

Table 1a. ieCollab team members

Student	University	Primary Role	Secondary Role
Alan Ng	MIT	System Analyst	Programmer
Anup Mantena	MIT	Configuration Manager	Programmer
Bharath Krishnan	MIT	System Analyst	Programmer
Blanca Roman	CICESE	Quality Assurance	Knowledge Manager
Cesar Guerra	CICESE	Tester	Programmer
Chang Kuang	MIT	Tester	Programmer
Erik Abbott	MIT	Project Manager	Knowledge manager
Eswar Vemulapalli	MIT	Business Manager	Programmer
Gyanesh Dwivedi	MIT	Programmer	------------
Hao Chen	MIT	Designer	Programmer
Hermawan Kamili	MIT	Tester	Programmer
Ivan Limansky	MIT	Project Manager	Programmer
Jaime Tirado	CICESE	Business Manager	Programmer
Joao Arantes	MIT	Project Manager	Quality Assurance
Justin Mills	MIT	Business Manager	Programmer

Table 1b. ieCollab team members

Student	University	Primary Role	Secondary Role
Kaissar Gemayel	MIT	Quality Assurance	Programmer
Kenward Ma	MIT	Tester	Programmer
Leopoldo Moran	CICESE	System Analyst	Programmer
Li-Wei Lehman	MIT	System Analyst	Programmer
Luis Alberto Rojas	PUC	Designer	Programmer
Manuel Alba	CICESE	Configuration Manager	Designer
Maria Elena Ruiz-Tagle	PUC	System Analyst	Programmer
Nhi Tan	MIT	Quality Assurance	Designer
Octavio Garcia	CICESE	Project Manager	Programmer
Paul Koon Po Wong	MIT	Knowledge Manager	Programmer
Pudubu Wariyapola	MIT	Marketing Manager	Programmer
Rafael Fuentes	PUC	Designer	Programmer
Roberto Machorro	CICESE	Designer	Programmer
Rosa Alarcon	PUC	System Analyst	Programmer
Saeyoon Kim	MIT	Quality Assurance	Programmer
Steven Kyauk	MIT	Marketing Manager	Programmer

Table 1c. ieCollab team members

Student	University	Primary Role	Secondary Role
Sugata Sen	MIT	Programmer	------------
Theresa Liu	MIT	Configuration Manager	Programmer
Wassim El-Solh	MIT	Designer	Programmer

- ***Hardware resources***

  Since ieCollab is designed from the start to be platform independent through the use of Java technology, a lot of machines were available for the team members to develop the package further. Mainly, the machines running Microsoft Windows NT in the Design studio of the Future (DSOF) at MIT, Athena clusters running on Unix platforms at MIT also, as well as machines at the computer science labs in CICESE and PUC were used.

- ***Software resources***

  The software was developed through collaboration of teams situated in three different countries. This requires handy communication tools to enable efficient interaction. Throughout the two semesters, Microsoft's NetMeeting and ICQ were used as well as WebEx to force presentations. We also made use of the CAIRO software that was developed the previous years at MIT.

Coding was being performed using Sun's Java Development Kit v1.2, and OMG's (Object Management Group) CORBA (Common Object Request Broker Architecture). For design purposes and flexibility, both Rational rose and Visio were used.

To support the whole process of development, we were using several tools for producing documents, presentations, and refining the schedule as we move forward. The tools used were Microsoft's Word, Excel, PowerPoint and Project along with some implementation of Primavera.

4.8.2 Model Adopted

At the start of the project, the Waterfall lifecycle model was being used to organize the movement of the project. Although it is a very traditional model to be used in an untraditional environment, it was chosen because it constituted the basis for more effective lifecycle models. In the waterfall model, the project progresses through an orderly sequence of steps from the initial software concept through the system testing. Usually at the end of each phase, the project holds a review to determine whether the project is ready to advance to the next phase. But if the review determines that the project isn't ready yet to move forward, it stays in the current phase until it is ready. I should note that the waterfall model is document driven, which means that the main work products that are carried from one phase to the other are documents. So it is obvious now that the phases are continuous, and there's no overlapping. This type of model suits very much an inexperienced or weak technical staff because it provides the project with a structure that helps to minimize wasted effort. But one of its disadvantages is the difficulty to fully specify the whole requirements for a product that is novel and targeted to hit a market instead of being a product developed for a specific customer at the beginning of the project, before any design has been done or any code has been written.

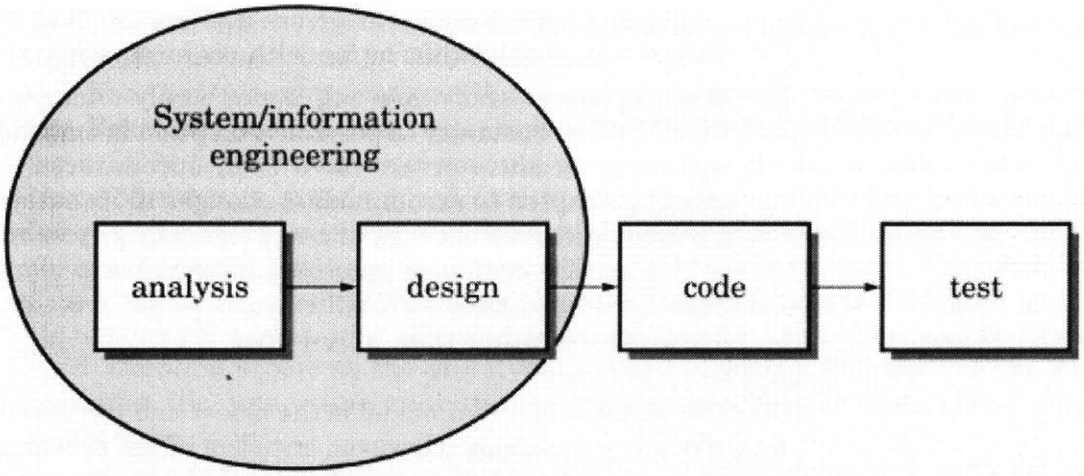

Figure 4. The Waterfall model [ieCollab, 2000]

What we ended up doing at a later stage was to use the incremental model due to the fact that we changed some of the requirements. We wanted to develop two subprojects simultaneously instead of proceeding with one big project.

Knowing that the incremental model is very close by definition to the waterfall model, and that it also allows the decomposition of the project into smaller subprojects, the transition was quite easy. The incremental model allows risk reduction and an increase in the progress visibility of the project by providing finished, operational pieces of the system long before the whole system becomes operational.

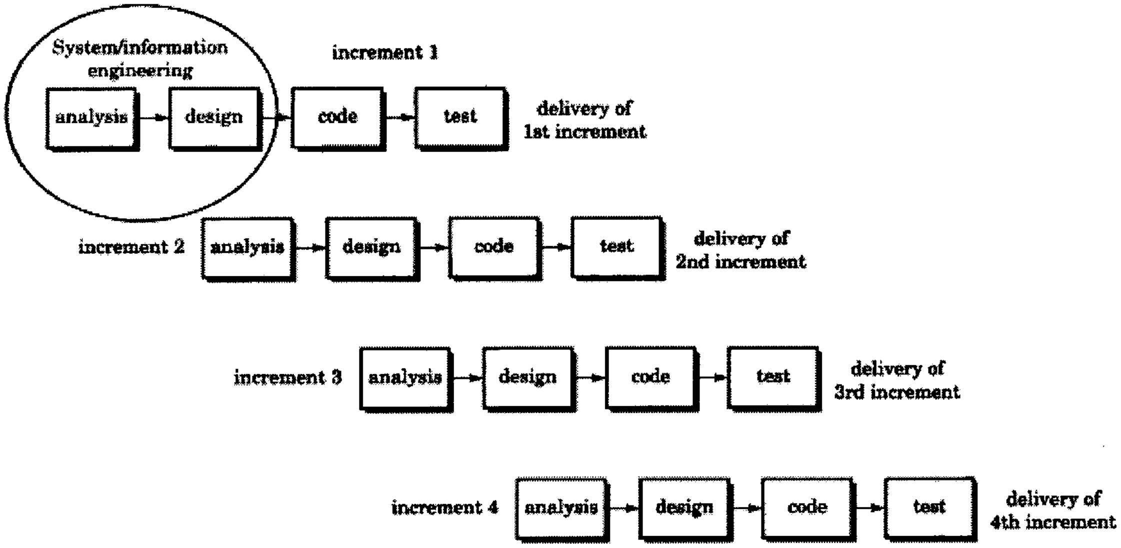

Figure 5. The Incremental model [ieCollab, 2000]

4.8.3 Roles

For each member of the ieCollab project there was a specific role tailored to suit his abilities. Personnel who had previous experience in management were distributed among several managerial position, people with very strong technical backgrounds were assigned programming or testing positions and so on. I will describe those roles very briefly in the coming section.

- ***Project Manager***

  The software project manager team has to plan and organize activities for all other team members in such a way that there is confidence that the product will be delivered on time, within the resources projected and within the quality expected. The project managers accomplished that by assembling a staff that is qualified to perform the tasks required, by directing and redirecting the staff potential as necessary, and by putting the controls in place that will provide the information necessary to direct the staff to its completed assignment. In addition to that, the project managers were responsible for intangibles such as alleviating the morale of other team members, creating harmony between team members, as well as interfacing with market advisors. The project manager team was keeping track of all necessary information through weekly reports submitted by each sub-team concerning its progress.

- ***Business and Marketing Manager***

  I will divide this role into two different ones, the first is the role of market advisors, and the second is the role of the ieCollab business and marketing sub-team.

The market advisors or mentors are basically the professors who already have a wide market experience in both the collaboration field and the software design and development field. They are known as mentors, because they act as leaders or as guidance for the team towards the final destination. Those mentors are basically Prof. Feniosky Pena-Mora from MIT, Prof. Jesus Favela from CICESE and Prof. David Fuller from PUC.

The second part or role is that of the business and marketing sub-team. They are mainly responsible for aligning customer needs with product requirements, formulating acquisition strategies, recruiting capable personnel for the business development team and finally delivering a detailed business and marketing plan discussing the steps to be performed by the team during the extent of the project. I have to add that the marketing strategy tackles virtual teams in automotive, aerospace, construction, defense, and software industries. Typical buyers will be project managers in charge of cross-functional teams working across geographical, temporal, organizational, and cultural boundaries. The first customers of ieCollab's offering have been identified and, in some cases, have been working with the core MIT research group in the development of first version of collaborative tools.

- ***Requirement Analyst***

The focus of the requirement analysis sub-team is on determining what the software does and how it should do it. The main objective of the team is to achieve agreement on the requirements presented. The documents presented by the requirement analyst team provide a basis for the designers and a reference for the testers. So what they did in brief was to gather the business plan and search for ideas, then structure and analyze the information in hand, and accordingly generate use cases to reach agreement between team members to finally come up with the requirement specifications.

- ***Designer***

The job of the designers was to build a complete model of the solution without getting into the details of implementation. The final documents presented by the requirement analysts are given to the design team. The designers in turn have to translate the requirements into a "blueprint" of the system moving from the system architecture to the individual design of each component. From the beginning of the design phase, the team decided to do its best to provide the best object-oriented design, meaning that they tried to show the system as a collection of objects rather than a collection of functions. The most common tool that was used by the designers was the Unified Modeling Language (UML) that enabled them to represent the structure of the entities and the flow of data among them.

- ***Programmer***

Similar to the designers when they used the requirement analysis documents as a basis for their work, the programming team in its turn uses the design to achieve all their requirements and functionality. The programming team had its work divided into five different aspects. The first was to do the programming for the client side, meaning the graphical user interface which is what the usual user interacts with. The second was to do the coding for the server, which describes how the server acts when certain operations are required from it. The third aspect is the creation of the database that will enable storage and retrieval of all sorts of information such as special attributes for users, etc... Both the server and the database are called back-end programming whereas the programming for the client side is known as front-end programming. The two last aspects were known as the integration phase, they consisted of connecting the functional graphical user interface with the server code, and connecting the server code to that of the database. The integration is meant to take all those small functional parts that were developed in parallel and connect them so that the whole system can interact with each other to provide a single, reliable package. As mentioned before in this chapter, the main coding language used was Java along with the use of the Java Database Connectivity (JDBC) to connect the server to the database, and CORBA to connect the client to the server.

- ***Tester***

  The testing is a crucial part of the software development cycle, it is often underestimated but it is the last stage for the ultimate review of specifications, design and coding. Testing is considered as the final fortress from which errors can be detected. The main purpose of the testers was to demonstrate or prove that the program does not work by using all sorts of destructive acts that aim at making the program fail. But at the same time they wanted to ensure the integrity of the software by assuring that it meets all the requirements and specifications, and by showing that the package can perform satisfactorily under all conditions. The main flow of activities for the testing team comprised of planning the test cases to be performed, designing the test cases, executing the tests and finally reporting the test results.

- ***Quality Assurance Manager***

  The concept of assurance of software quality is based on the principle of establishing good software engineering practices and monitoring adherence to those practices throughout the software development life cycle. This results to a large extent, in giving control of the software development process priority over control of the software product. It must be understood that quality cannot be the assigned function of any one person in the organization; rather, it must be the primary responsibility of every person involved in the software development process of a product. The primary role of the quality assurance team, then, was to influence everyone to perform his or her function in a quality manner. The basis for this philosophy is that the consistent use of a quality process will result in a quality product. For this purpose, the quality assurance team checked the product at every stage in the cycle. They were responsible for conducting inspections, walkthroughs, and audits to assure the quality at each stage as well as recording any non-compliances and reporting them to the project managers. The quality assurance team had to also submit a plan at the beginning of the project that has all desired specifications to be used for the development process.

- ***Configuration Manager***

Configuration management is an integral part of the software development process across all phases of the life cycle. It functions as a controlling discipline, enabling changes to be made to existing documentation and products in such a way as not to destroy integrity of the software. The configuration management team has to keep track of all version of the product at all times of the development process. They should be able to prevent simultaneous updates, the possibility of uninformed team members, and most importantly dismiss any version confusion. The configuration management team was responsible to create a web repository that includes the final frozen documents submitted by all teams.

- ***Knowledge Manager***

  The knowledge manager was in charge of creating and maintaining the project memory. In that function, the team also had the responsibility of maintaining the project web site and keeping track of all project documents and preparing the documentation report. The knowledge manager also plays the lead role in putting together the users' manual and setting forth the standards for report formats. Another deliverable was saving every single document related to the project on a compact disk (CD) for future software development.

4.9 Project Technology

Collaboration has been the main theme of our nine-month project. The basic intended features for ieCollab are the ability of sharing ideas, models, calculations, drawings and even simulations. This should enable a virtual desktop that allows users to easily edit the same document in real time and at the same time.

4.9.1 Why ASP?

The ieCollab team decided on adopting the ASP technology into its product. The decision came not only from the fact that ASP technology is emerging rapidly and becoming widely used, but also because of the fact that this technology facilitates the use of ieCollab and enables a broader functionality of the system. First of all the user is spared the headaches of installing the new software and configuring his system to react according to ieCollab's demands, and at the same time we are allowing the delivery of our package to all sorts of PC users whether they are developers or the usual internet browsing type of person. Second, all upgrades or future versions of ieCollab are transparent to the user. By transparent, I mean that the user has to know nothing about it and all his settings would still be the same. Although the intention is to serve ieCollab as an application, using the ASP, the user is able to have access to a broader number of applications other than one single software package and for a reasonable cost that is much lower than buying those packages. And since everything is installed and maintained by the server side people, the user doesn't have to care about maintenances and upgrades, but rather he would be able to log on to his ASP provider and attend all his business meetings and schedules.

4.9.2 Architecture

As described in the previous section, ieCollab would be available on a collaborative Application Service Provider, instead of having the client-server-database usual connection. The collaborative ASP will enable ieCollab to connect to other ASP providers in the back end and retrieve information. So picture that you have a scheduled meeting, but before attending this meeting you want to check your schedule and maybe set up a second meeting for the afternoon. All you have to do is log on to the ASP of your choice, and then start the ieCollab application. Once you are inside the ieCollab application, you click on the schedule button. In a matter of seconds, you are able to browse through your schedule, check your meetings, add some and even alter some. What the user witnessed was usual web browsing, but what ieCollab did at the back end was connect to another ASP and retrieve the information requested by the user. The whole idea behind that is that we are able to incorporate features from several other already existing applications. From now on, ieCollab can be made as flexible as can be by connecting to numerous numbers of ASP's and retrieving the information requested.

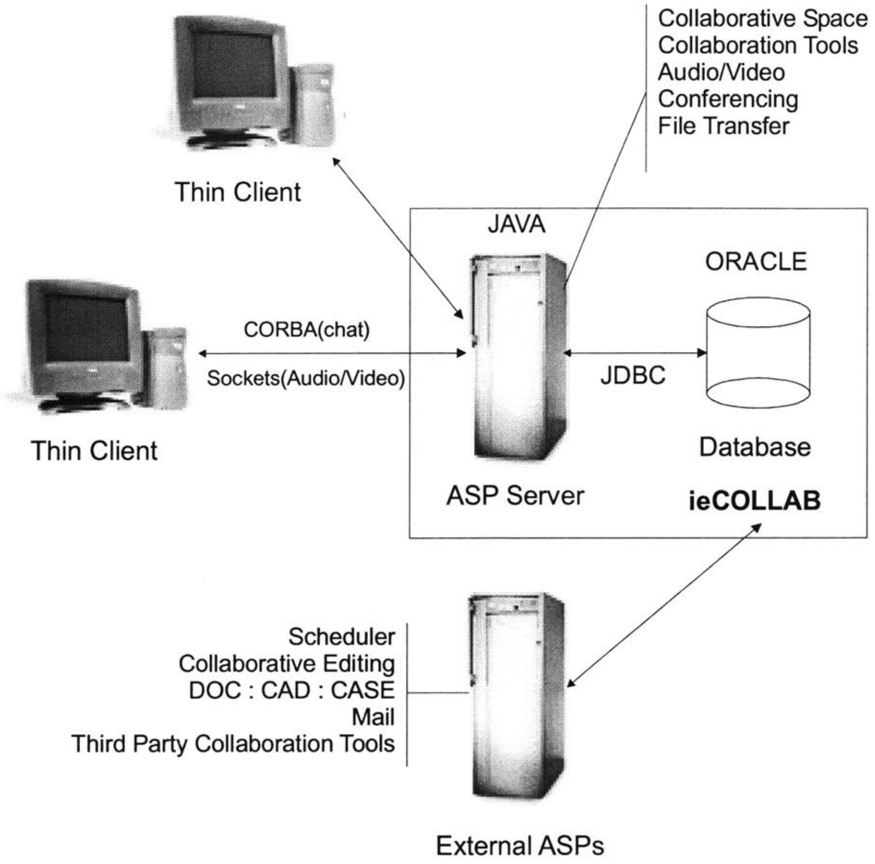

Figure 6. The ieCollab Architecture [Limansky, 2000]

To be able to accomplish this flexibility, a lot of work had to be done from the design side. The above picture describes a simple model of interaction between two people having a meeting. When both thin clients try to connect to an ASP that is providing ieCollab services, ieCollab instructs the server to retrieve the data of both members from the database. To be able to log on to a meeting, both users will request their personal schedule, so ieCollab again instructs the ASP server to connect to an external ASP and retrieve the information from it. The external ASP might also be connecting to one or several databases or any other external server that provides information. IeCollab then will allow both users to simultaneously attend the meeting, share ideas through forcing presentations or displaying some sketches in real time through a simple application, or even developing a drawing while interacting through audio or video at the same time. I should note that it is not necessary for a client to log on to the same ASP provider to be able to connect to a meeting. Through ieCollab, we can join users connecting through different ASP's. So we just look at it as a community of ASP's working in a very intense collaborative environment, where anyone can get anything from anywhere while still using one application.

4.9.3 Why Java?

Due to the diversity of people who will be using ieCollab for the purpose of collaboration, ieCollab was designed to be compatible with all available systems and platforms. What technology to deliver this feature other than Sun's Java? The Graphical User Interface is totally independent from the surrounding pc environment; it is an applet that could be run from any possible pc architecture. Other than having platform independent software, the application is also accessed from a traditional web browser that is available on most personal machines nowadays. As for the development of ieCollab, Java was a very flexible ground to start from. It allowed more visibility of the project's coding progress and a faster development time.

4.10 The Future as we see it

Everybody is moving into wireless technology, this is the hottest topic in the market nowadays. The ability to make decisions from remote locations has always fascinated people. Previous focus to facilitate the wireless technology was on making it cheaper, faster, smaller and most importantly lighter. Future improvement is concentrating on making the technology more comfortable, more fashionable, and even more wearable. The future market is estimated to become over $110 billion with a triple in size. People would be able to get all the features in one package; mobile phone, organizer, email and browser. So if browser technology is enabled, then immediately you can have access to ieCollab's capabilities.

The future will go more into the depths of virtual collaboration by enhancing communication tools to be able to deliver more reliable synchronous interaction. Technology will enable you to be able to whisper to another person sitting in the virtual meeting, ignore another and even share your ideas by voting to decisions. Users will be able to move in a 3D environment where they can pass by blackboards that contain announcements, they might choose to look at it or explore further to attend a meeting as listeners or as participants. Tools that can measure body perspiration or facial movements could be used to indicate the different states of the participant.

4.11 Summary

As we move further and further forward, we are sensing the need of more enabling technologies. For this technology to improve, there has to be a flexible background that can adapt to its ever-changing face. The next chapter will discuss the basic features needed to withhold the pace.

Chapter 5

Conclusion and the Future

As described in chapter 4, ieCollab relies a lot on the services provided by ASP's whether they are internal or external ones. For a collaboration tool, as complex as ieCollab, there has to be some basic standard traits that will allow it to perform well. This final chapter is meant to recommend most of the features that are required from an application service provider in order for ieCollab to function optimally.

5.1 Selecting an Application Server

As businesses continue to move to web-based computing architectures, application servers are becoming increasingly strategic pieces of the IT puzzle. Application servers are, in fact, what allow IT departments to fully leverage all of their back-end resources - including legacy applications, distributed systems databases, and even other web-based content - and make them available to end-users of every type over Intranets, extranets, and/or the public Internet [Rofail, 1999].

leCollab's choice of application server therefore sits at a critical "crossroads" in the flow of information-based business processes. If the application server can't securely deliver any data or application to any required internal or external user, then it becomes an obstacle to success instead of a competitive advantage. So, given the importance of the app server's role in today's IT strategies, how do you pick one?

Basically, application server requirements can be divided into three overall categories:

- ***Capabilities***

  What the app server can do for you when you have it in place

- ***Implementation***

  How easily and quickly it lets you get to and manage those capabilities

- ***Business***

  What it will cost you to gain all these wonderful abilities, and whom you will have to deal with to get them

Without weighting these categories, the following sections will give a brief overview of how to evaluate each area of consideration.

5.1.1 Capabilities

Since application servers essentially act as integration engines, it's obviously important that they be able to support the full range of resources a company wants to deploy. On the back end, these may include mainframe CICS systems, applications with industry-standard CORBA or COM "hooks", proprietary ERP software, and any transaction- or messaging-based middleware already in use across the enterprise. For ieCollab, which should have already begun a search for web-based computing architectures, the application server must already be able to incorporate Java technology deployments such as Enterprise Java Beans (EJB) and/or Java Server Pages (JSP).

To effectively deliver any and all of this "content" to end-users, the application server should be able to sense and accommodate the various types of clients it will be asked to support (i.e. different browser types). This is also why a "100% pure" Java implementation is preferable since that helps ensure that the application will perform properly and consistently on Java Virtual Machines (JVMs) running on different platforms.

XML support has also become a key consideration, since it has rapidly gained support as a standard for sharing data models between different companies' applications. Thus, XML can act as both a source and a client for the application server. In some cases, the XML model may be attractive as an application integration interface for multiple applications within the enterprise itself. Thus, an application server's XML implementation should be a matter for close scrutiny by any prospective buyer [Ceponkus & Hoodbhoy, 1999].

In addition to being able to link diverse sources and clients over the web, an application server must also be able to do so efficiently. Performance is always an important consideration, especially when your users are customers whose tolerance for delays or lost connections may be minimal. And, in the case of a collaborative tool like ieCollab, which would be available on a 7x24 basis, the number of simultaneous users can often be several orders of magnitude greater than the typical enterprise application. Scalability is therefore a must.

There are a few ways to check an application server's scalability. One is to evaluate its performance features. These can include a variety of caching and load-balancing functions. However, much of an application server's scalability and performance are based on the elegance and efficiency of its internal processes, which are impossible to evaluate on paper. That's why it's important to stress test an application server in a load-intensive testing environment. Reference accounts with high-volume deployments and/or published comparative analyses are also useful in evaluating competitive products for performance [iPlanet, 2000].

Another important factor to consider in ensuring reliability and positive user experience with web applications is "state." By preserving or maintaining "state", an application server retains awareness of whatever is happening with a given user executing a given transaction at a given time. This is important on the net, since the connectivity between a user and a server over an IP network can de-materialize without warning. An application server must therefore be able to cope with such an occurrence and allow the user to pick up wherever he or she left off when they return.

Security is obviously another key concern for applications that use shared networks. In such shared network environments, the company deploying the application doesn't own or control the entire infrastructure over which it runs. Security therefore has to be performed at the application level, rather than at the network level. Since the application server controls client access to back-end resources, it makes sense to deploy security measures there. So ieCollab should consider an application server's authentication and encryption facilities carefully.

5.1.2 Implementation

No matter how great an application server may perform, it won't do ieCollab users much good if it takes several months to get new applications and upgrades up and running - or if it's problematic to make modifications to an application once it's running. After all, today's digital business environment is highly competitive, and there is tremendous value in being first-to-market.

An application server should therefore provide a very easy-to-assimilate development environment. Intuitive, drag-and-drop tools are essential to eliminate potentially steep learning curves for IT staffs. So is broad support for the development technologies that IT staff is most accustomed to using - which may include Java, C++, scripting languages, and/or XML.

It also helps if there is a rich library of objects already available for use. This object library should be easily extensible, so that any investments in the writing of code can be leveraged in other ways.

Ultimately, what ieCollab future developers should be looking for is a complete, integrated development environment (IDE) that includes robust code editors and controls, remote debugging tools, and other code management facilities. We, as a team, should look for a good match with our particular development team's "personality".

Once a particular application is up and running on the application server, it's essential to be able to monitor its performance and instantly pinpoint any problems with particular components or processes. That's why ieCollab team should closely examine the application server's event logging and reporting functions. Think of it this way. If you have an application server where only one thing goes wrong in the first few days - but you don't find out about it - you're in much worse shape than you are with a server where several things go wrong, but which you can quickly tweak into shape.

In large environments, it may be necessary to consider how well these management facilities integrate with the third-party systems and application management solutions already in use across the enterprise. Without such integration, troubleshooting processes can get overly complex - resulting in slower time-to-fix and higher overall ownership costs [DHBA, 2000].

"Tweakability" isn't just important for fixing problems. It's also important for responding to today's super-dynamic business and technology environments. That's why it should be relatively easy to make adjustments in the way any given application runs on an app server - especially when it comes to adding, deleting, or modifying objects. Otherwise, such an application will rapidly become just another "legacy" system. Checking with a reference account that has had long-term experience with the product can be very helpful in this regard.

5.1.3 Business

Naturally, all these capabilities and features come at a price. The application server market is still maturing, so pricing models vary greatly in terms of servers, clients, additional platforms, add-on modules, and bundled support service. Some vendors are offering more innovative pricing models based on overall usage, rather than a specific number of CPUs. The appropriateness of these various models varies greatly, depending on how the application server is being deployed.

Services are definitely of particular interest for ieCollab. Because there is often tremendous pressure from business units to get specific applications up and running - and because those objectives may be well above and beyond the current capacity of the IT department - it is often appropriate to bring in significant outside development help over a short period of time. Application server vendors who can deliver that capacity, either through their own deployment staffs or authorized systems integration partners, can deliver significant value while allowing IT managers to avoid making long-term hiring commitments in an effort to meet near-term deadlines [NaviSite, 2000].

When trying to determine how good a business partner an application server vendor may be, it also helps to spend a lot of time snooping around their web site. Look at who their partners are, how they have handled upgrade pricing and migration, and how quickly they have been able to respond to the emergence of new standards and/or technologies. The company's support/technical bulletins content is often particularly telling. If we should find that the site developers' area isn't open to the public, we should demand at least temporary access to it before we buy. That way we can see how the company handles bugs and fixes. The more detailed and forthcoming they are, the better.

We may also want to drill down for more technical detail about which features are supported under which platforms. Application server vendors often make claims about features that are only supported on specific web servers or for certain types of back-end applications. Verify that the features we need are supported for the platforms we have, and we'll avoid much unpleasantness further down the road.

5.2 What the Future Will Look Like

At home, you'll use simple screen phones to access home banking and ATMs from a central server farm residing in the ASP data center. You'll be able to download electronic cash to your credit card, access your financial information, pay bills, and make stock transactions. In addition, home network terminals and television sets will be connected to extensive information services. Communication, education, shopping, and entertainment will be available, at your fingertips, 24-hours a day. The home will be "appliance-enabled," with single-purpose information appliances to serve the needs of anyone, anywhere.

Mobile service workers will use multi-functional, wireless information appliances to communicate with other service personnel. They will be able to access repair manuals, review service records, and check parts inventories using applications delivered via the ASP data center. Customers will get better, faster service, and their service providers will get a competitive edge.

Educators will be able to focus on teaching again, instead of managing computing environments. Students, teachers and administrators will all be connected to application service centers that offer the latest software programs, education information and learning resources, whether at school, on a field trip or right at home. The power of computing will at last be available to everyone - regardless of socio-economic status or school location.

Information appliances in your automobile will update you on traffic conditions, help you navigate, allow you to review entertainment options, restaurants, shopping, hotels and more - all while you are on the road. Every new function will be delivered, managed, monitored and billed from the server [Citrix, 2000].

5.3 Conclusion

Application servers offer IT departments a tremendous opportunity to create business-enriching applications that can be flexibly applied to internal users, customers, suppliers, and other trading partners. The next generation of XML-compliant application server technology is likely to revolutionize how companies leverage the Internet to accelerate and streamline inter-enterprise business processes. Companies that don't rapidly capitalize on these opportunities will be left behind. The rapid embrace of application server technology is therefore vital to ensure ieCollab's long-term competitive position. At the same time, the business team should be very careful about who they choose as their application server vendors, since it's a choice that can affect IT's effectiveness for years to come.

References

Citrix Systems, "Achieving Business Transformation Through Application Service Providers." In: *Citrix iBusiness,* 1999, pp. 3-15.

DiSEL, "Final Presentation." In: *CAIRO project*, 1998-1999.

EMC, "EMC Celerra File Server." In: www page, http://www.emc.com/products/networking/celerra.jsp?openfolder=storage_networking , 2000.

Lotus Development Corporation, "iNotes." In: www page, http://www.lotus.com/home.nsf/welcome/inotes, 2000.

Lotus Development Corporation, "The Domino Server Family." In: www page, http://www.lotus.com/home.nsf/welcome/domino, 2000.

Forrester Research, "Packaged Apps Outsourcing." In: www page, http://www.forrester.com, Cambridge, MA: January 1997.

Forrester Research, "Worldwide Application Service Provider Forecast 1998-2003." In: www page, http://www.forrester.com, Framingham, MA: March 1999.

Forrester Research, "Internet Service Provider Market Review and Forecast, 1998-2003." In: www page, http://www.forrester.com, Framingham, MA: March 1999.

R. T. Hightower, and L. Sayeed, “Effects of Communication Mode and Perdition Information Distribution Characteristics on Information Exchange in Groups.” In: *Information Systems Research,* 1996 vol. 7, pp. 451-465.

R. T. Hightower, and L. Sayeed, “The Impact of Computer Mediated Communication Systems on Biased Group Discussion.” In: *Computer in Human Behavior,* 1995 vol. 11, pp. 33-44.

R. T. Hightower, L. Sayeed, M. Warkentin, and R. McHaney, “Information Exchange.” In: M. Igbaria and M. Tan (Ed.) *The Virtual Workplace,* Hershey, PA: Idea Group Publishing, 1998.

IeCollab, “Final Presentation v9.0.” In: *ieCollab project*, 1999-2000.

IPlanet, “iPlanet Application Infrastructure.” In: www page, http://www.iplanet.com/products/infrastructure/index.html, 2000.

I. Limansky, “ieCollab Architecture.” In: *ieCollab project*, 1999-2000.

C. Manasseh, “Project Management of Geographically Distributed Software Teams.” In: *CAIRO project,* 1998.

S. McKie, “Outsourcing with ASPs in the Internet Age.” In: *Business Finance Magazine,* November 1999.

NaviSite, “Internet Applications Outsourcing.” In: www page, http://www.navisite.com, version 2.0, September 1999.

NaviSite, "Five Tiers of Data Center Redundancy." In: www page, http://www.navisite.com, version 1.3, April 1998.

NaviSite, "Outsourcing E-business Infrastructure." In: www page, http://www.navisite.com, version 1.0, September 1999.

Oracle Corporation, "The Oracle Internet Platform for e-Business." In: www page, http://www.oracle.com/iplatform/index.html?architecture.html, 2000.

A. Rofail, "Optimize MTS Components." In: *Expert Visual Programming,* July 1999, pp. 1-3.

SCO, "Enterprise Scalability." In: www page, http://www.tarantella.com/info/wps/broker.html, *Tarantella Web-Enabling Software,* 1997-1999, pp. 4-6.

SCO, "Extending the Reach of Your Application Without Compromising Security." In: www page, http://www.tarantella.com/info/wps/security.html, *Tarantella Security*, 1997-1999, pp. 6-8.

SCO, "One World, One Network, One Answer." In: www page, http://www.sco.com/telephony/white/tta.html, *Tarantella Web-Enabling Software,* 1997-1999, pp. 5-9.

SCO, "The Adaptive Internet Protocol." In: www page, http://www.tarantella.com/info/wps/aip.html, *Tarantella Web-Enabling Software,* 1997-1999, pp.7-8.

G. Stasser, and D. Stewart, "Discovery of Hidden Profiles by Decision-Making Groups: Solving a Problem Versus Making a Judgment." In: *Journal of Personality and Social Psychology,* 1992 vol. 63, pp. 426-434.

G. Stasser, L. A. Taylor, and C. Hanno, "Information Sampling in Structured and Unstructured Discussion of Three- and Six-Person Groups." In: *Journal of Personality and Social Psychology,* 1989 vol. 57, pp. 67-78.

G. Stasser and W. Titus, "Effects of Information Load and Percentage of Shared Information on the Dissemination of Unshared Information During Group Discussion." In: *Journal of Personality and Social Psychology,* 1987 vol. 53, pp. 81-93.

G. Stasser, and W. Titus, "Pooling of Unshared Information in Group Decision Making: Biased Information Sampling During Group Discussion." In: *Journal of Personality and Social Psychology,* 1985 vol. 48, pp. 1467-1478.

P. Wainewright, "Interliant: ASP fusion for the enterprise." In: *ASP case studies,* Ed 1.0, September 1999, pp. 3-17.

P. Wainewright, "Futurelink: Building the Information Utility." In: *ASP case studies,* Ed 1.0, May 1999, pp. 4-12.