

ASP Report

ASPs: An Executive Report

*Are Application Service Providers
Ready for Prime Time?*

SPONSORED BY

California HealthCare Foundation

PREPARED BY

First Consulting Group

AUTHORS

Gregory LeGrow,
in collaboration with Jim Eschner,
Erica Drazen, Jon Soloman,
Roy Sailor

SEPTEMBER 2000

ACKNOWLEDGMENTS

First Consulting Group is a leading provider of information-based consulting, integration, and management services to health care, pharmaceutical, and other life sciences organizations in North America and Europe. Advancing from an early focus on health care systems, FCG has expanded its service offerings to meet the complex needs of all the major sectors in an increasingly sophisticated health care marketplace.

This report is the first in a series on Application Service Providers. To order additional copies of this report or others in the series, please call the California HealthCare Foundation's Publication Line at (510) 587-3199. Reports are also available in PDF format at ehealth.chcf.org.

Copyright 2000 California HealthCare Foundation
ISBN 1-929008-43-0

Contents

INTRODUCTION	5
<hr/>	
CHAPTER ONE	
<hr/>	
HOW DID ASPS EVOLVE?	6
WHAT ARE ASPS AND WHAT DO THEY DO?	6
ASPs Defined	7
Target Markets	9
<hr/>	
CHAPTER TWO	
<hr/>	
DIFFERENCES BETWEEN ASPS AND TIMESHARE OR OUTSOURCED IT OFFERINGS	10
Outsourcing vs. ASP	10
Timeshare vs. ASP	11
<hr/>	
CHAPTER THREE	
<hr/>	
WHEN ARE ASPS ADVANTAGEOUS?	12
Benefits and Drawbacks of ASPs	12
<hr/>	
CHAPTER FOUR	
<hr/>	
WHAT ARE THE DIFFERENT TYPES OF ASPS?	14
Application Categories	14
Service Tiers	15
ASP Players	15
Pricing Options	16
<hr/>	
CHAPTER FIVE	
<hr/>	
TODAY'S ASP MARKET	18
ASPs in Health Care	18
WHAT TO ASK AN ASP VENDOR	19
<hr/>	
RESOURCES	
<hr/>	
Recent Articles	22
Associations/Web Sites	22
<hr/>	
GLOSSARY	23
<hr/>	
CITATIONS	28
<hr/>	



Introduction

While the need for information technology (IT) is now recognized as critical to any health care enterprise, managing that technology remains a challenge for most organizations. Some common issues are the high capital costs associated with acquiring technology, high maintenance costs, and a competitive IT labor market with high salaries and more jobs than qualified candidates.

This Executive Report is an introduction to a growing trend in the health care marketplace that addresses these challenges; a new form of outsourcing – the use of Application Service Providers (ASPs).

Small and mid-sized enterprises are turning to ASPs as a means to avoid large capital outlays for hardware, software, and IT staff. Enterprises of all sizes rely on ASPs to accelerate software installation and quickly build e-business offerings. ASPs are even gaining acceptance by geographically dispersed businesses or merged organizations that seek a universal solution. However, the ASP market is in its infancy and, thus, poses some significant risks. This immature business model has yet to be time tested and the multitude of new entrants bidding for space make evaluating both vendors and the decision to use an ASP confusing.

It is important that health care executives understand this approach to acquiring software and support so that they can use it appropriately and to the benefit of their organization. To make an informed choice, executives need to understand what ASPs are and what they can offer, which ASP service offerings can best meet their needs, and the risks that are associated with contracting with an ASP. Toward that end, this report addresses the following:

- How did ASPs evolve?
- What are ASPs and what do they do?
- What is the difference between an ASP and a timeshare or outsourced IT offering?
- When are ASPs advantageous?
- What are the different types of ASP?
- What does the current ASP market look like?
- What do you need to ask an ASP vendor before signing up?

Chapter One

How Did ASPs Evolve?

ASPs are not a health care-specific IT solution. The market for ASPs developed in response to the needs of small and mid-sized enterprises that have limited resources to devote to IT investments and staffing, but need the capabilities to grow their businesses. By adopting the ASP model, both sides win. Vendors gain revenues by expanding their market to include organizations that previously could not afford their applications, and clients gain access to top-of-the-line applications at affordable prices.

Advancements in network technologies that permit high-speed connections from virtually anywhere have been one of the key enablers of this new form of application deployment. Centralized computing has been another major contributor. The combination of these two factors have allowed software vendors to move from a technique where the majority of an application's processing is done on an individual's personal computer to a technique where processing is done on a high-powered central computer. Add the acceptance of the Internet as an alternative network, scalable applications, and organization's dislike of installing and maintaining applications, and ASPs almost sell themselves.

In health care, vendors are clamoring to identify themselves as application service providers. Legacy health care vendors such as SMS, Millbrook, IDX, and others quickly recognized similarities between an ASP offering and the traditional application outsourcing services that they have provided for years. Traditional application hosting required the client to pay a licensing fee in addition to ongoing maintenance and support fees for the upkeep of the application. Therefore, it was a simple matter to adopt the ASP business model of financing and, using a broad interpretation of an ASP, declare themselves in a new line of business. Other vendors are adapting their technology to run over a network. The dot com start-ups are building applications from scratch to be used by multiple users over the Web.

What Are ASPs and What Do They Do?

Outsourcing and remotely running applications (also known as hosting) are not new concepts. Health care enterprises have outsourced entire IT departments as well as selected applications for years. What is new is the ASP business model of application delivery.

Despite the market hype, ASPs are a nascent business model and confusion abounds as to what can truly be considered an ASP. Offerings vary considerably and the industry is not yet using the term consistently. In order to distinguish hyperbole from reality, it helps for decision makers to begin with a fundamental understanding of how ASPs are defined.

ASPS DEFINED

The ASP Consortium, an international advocacy group formed in May of 1999 to promote the application service provider industry, offers the following: *“An ASP deploys, hosts and manages access to a packaged application to multiple parties from a centrally managed facility. The applications are delivered over networks on a subscription basis.”*

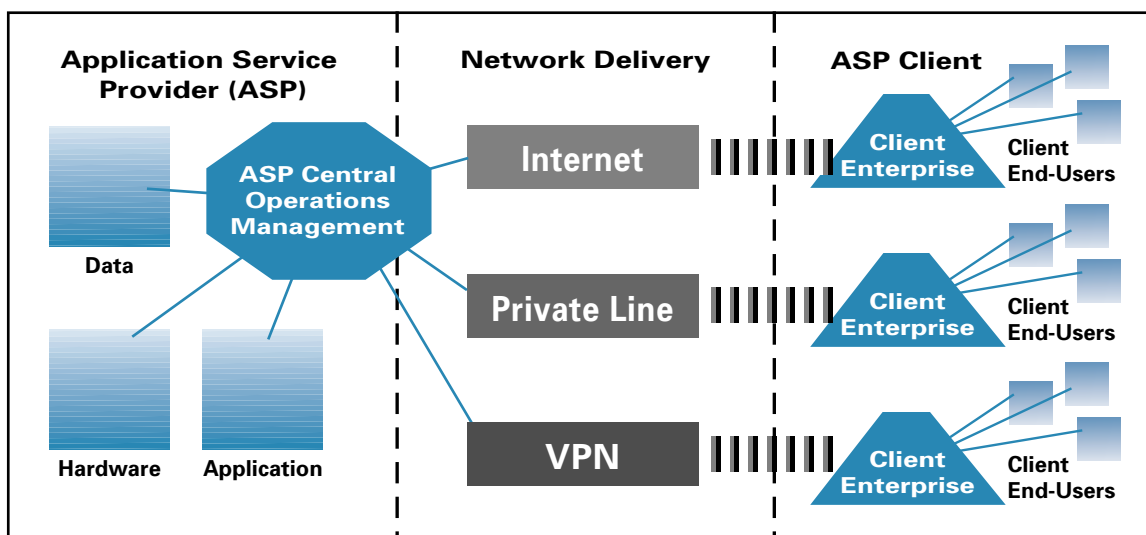
In effect, ASPs rent applications to organizations that use the software. They are businesses that either own or have purchased the rights to license the specific application software that they host in their own data center. Clients access the application from their remote location and the ASP operates and maintains the software and data center on the client's behalf.

In their purest sense, ASPs are merely service aggregators that act as the customer relationship manager and the single point of contact for all client interactions. They own or contract for all the services necessary to deliver an application into an organization. In order to deliver a product, ASPs must provide several components. (See Figure 1).

Coordinator of Services: Since few ASPs “own” the entire mix of services necessary to create an ASP offering, partnerships play a vital role in any offering. In many ways an ASP can be viewed as a virtual organization. Regardless of whether they own or subcontract for the components that make up an offering, it is ultimately their responsibility to respond to any client issues that arise. The ASP is the single point of contact for problem resolution.

Application Access: Vendor neutral (or pure-play) ASPs purchase the rights or establish contractual agreements with vendors to license access to their applications and then resell or rent that access. Some ASPs assemble entire suites of applications from multiple vendors and offer the customer a choice among vendors that deliver applications of the same type.

Figure 1: Structural Overview of ASPs



ASP offerings share access to an application where only a single copy of an application is installed and shared with multiple clients. Few health care applications, at least older legacy vendor offerings, are able to offer their products using this approach. For multiple clients to share a single copy of an application, the application must be designed and built for that purpose. Therefore, software that is capable of supporting multiple clients by means of a single copy is characteristically offered by new entrants. Alternatively, an application can be installed on a dedicated server devoted to a single client. The application is delivered remotely and is still considered an ASP offering. Most of the legacy health care products adapted for ASP delivery are deployed by this means.

Software vendors themselves may offer remote access to their application as an ASP offering or through a traditional application hosting relationship. Although similar in many ways, traditional application hosting is more apt to be a one-to-one solution tailored to meet the unique needs of an individual client.

Platforms and Data Center: In order to offer any application, ASPs must provide data center operations that include both the hardware and software the application is to run on. Hosted applications are installed at the ASP's data center and data are processed on site. Additionally, the data center usually physically hosts and stores client data, such as patient demographics and visit information.

Network Delivery: Customers access the software from their desktop computers via the Internet, virtual private network, or private secure lines. The use of the Internet is not mandatory in defining an ASP offering, although the ubiquitous nature of the Internet makes it an appealing option. In practice, most organizations' mission-critical applications do not use the Internet, as reliability and performance cannot be guaranteed via that medium.

User Interface: Applications are usually viewed through a Web browser located on the client's desktop. Since the application is transferred and all processing is done by the ASP computers, enterprises are able to use less powerful computer hardware (called "thin clients") without diminished performance. Under normal operating conditions, the software functions as if the application resides on site.

Preconfigured, Standardized Offering: ASPs offer standard packaged applications with minimal customization. To help solve this problem, vertical ASPs that offer industry-specific applications have sprung up and begun to tailor offerings to meet the needs of specific market segments. Additionally, some ASPs deliver limited customization by offering pre-configured options (based on templates) from which to choose. However, when substantial customization is required, costs go up and the benefits of using an ASP diminish and the offering begins to more closely resemble a traditional outsourcing arrangement.

Centrally Managed Operations: In addition to offering access to an application, ASP contracts include the management of the application. All maintenance – including application renewal, enhancements, error fixes, and new releases – is handled by the ASP from a central location. Since true ASPs run standard software without client-specific customization, the maintenance is easy and economical. When the software is updated, it is updated only once on the ASP's computer and does not need to be installed on each client's system.

One-to-Many Relationship: Since a standard application is essentially shared among ASP clients and delivered into client organizations via a network, the typical ASP model is a one-to-many relationship. Since the costs of maintaining an application are shared among all clients in a one-to-many relationship, they enjoy economies of scale and pay lower fees.

Pay-as-You-Go Pricing: ASPs use a variety of “pay-as-you-go” pricing models. Clients are charged a monthly fee for services. Thus, they avoid large initial licensing fees, hardware procurement costs, and other resource expenditures. Although most pricing is based on a subscription model, some ASPs offer rent-to-own options that allow clients to finance a software purchase and eventually bring the application in-house at their discretion.

Add-on Services: Although not necessarily part of the basic service, many ASPs offer, at extra cost, add-on services needed for application set up and integration. Others offer extended service agreements that include periodic visits and follow-up training. A few even offer to run entire business processes, such as billing in exchange for a percent of collections.

TARGET MARKETS

Start-up organizations are particularly attractive to ASPs and have embraced the ASP model to keep their infrastructure costs down while gaining quick time to market. As their organizations grow, they are able to scale applications and only pay for the services they use.

In the health care market, ASPs have tended to target small to mid-sized enterprises. Most of the large legacy vendors have focused on bigger health care organizations because they yield acceptable margins for their organizations. They have shied away from small enterprises where the cost of sales and ongoing support generate only marginal profits. Instead of redirecting their sales and support staffs, they license their products to independent ASPs whose business model permits profitable pursuit of the small to mid-sized market. The smaller enterprises that have fewer resources tend to find ASP offerings attractive.

Although there is evidence that larger enterprises such as university medical schools are beginning to use ASPs, organizations with large IT shops and mission-critical applications may find ASPs less desirable. Large enterprises often have multiple legacy systems that must be integrated with one another and customized to fit their needs. Because their more sophisticated business applications require advanced, customized functionality, a traditional vendor installation is often the preferred option.

Chapter Two

Differences Between ASPs and Timeshare or Outsourced IT Offerings

Sorting out ASP vendor offerings is not an easy task. In an attempt to capitalize on the ASP frenzy, many different types of companies claim to be an ASP. However, for a company to truly qualify as an ASP, it must deliver an application into an organization from a remote location. ASP, outsourcing, and timeshare arrangements are similar but important differences emerge in their delivery methods and application offerings, which distinguish the true ASPs. (See table below.)

OUTSOURCING VS. ASP

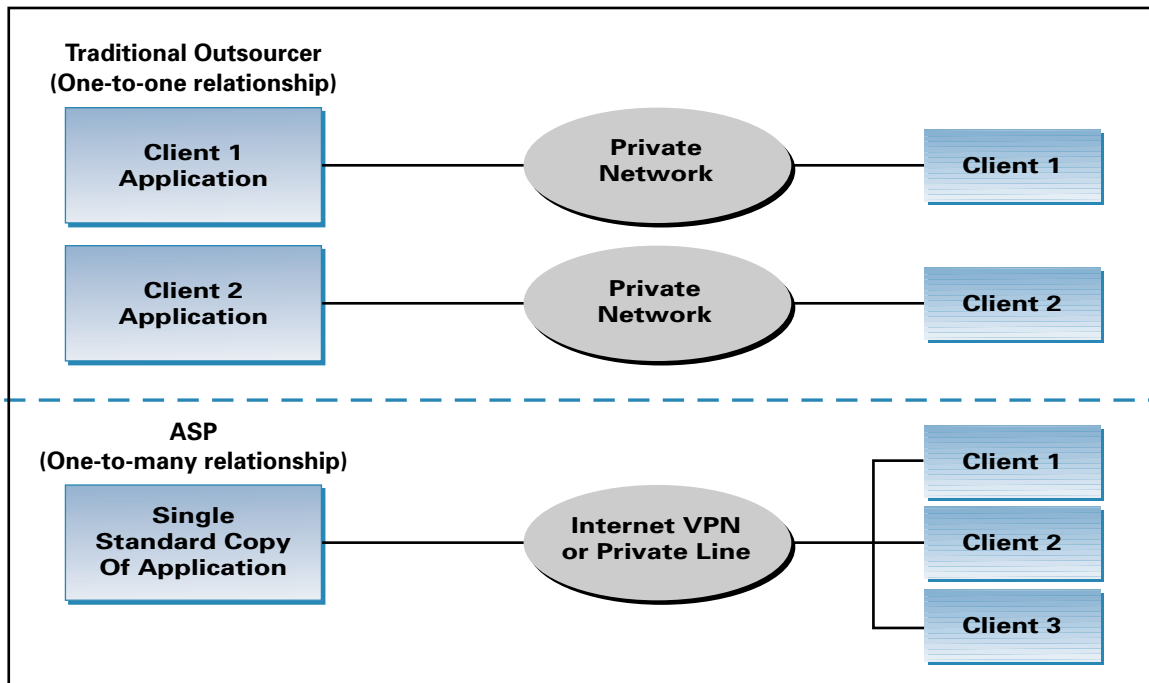
ASPs are a form of outsourcing. However, in a traditional outsourcing arrangement (also referred to as facilities management) full responsibility for an IT function is usually transferred to the outsourcer and often includes the transfer of staff. Essentially, part or all of a company's IT shop is shipped out and/or sold to an outsourcing company. In an ASP relationship, only the application is transferred. If the ASP provides a new application, the client can gain access to the new application for a slight incremental fee just as if they purchased and installed the software package.

Additionally, an outsourcer's data centers may or may not be located at the client site. Depending on the contract, the outsourcer takes over the operations of the client's data center or transfers existing data to their own. Some arrangements do not include the hardware to run the application software. One of the benefits of using an ASP is that clients gain access to both a data center and hardware as part of the ASP service package, thereby avoiding the need to invest in these elements themselves.

Table 1: Comparison of Outsourcing/Timeshare to ASP

Model Characteristic	Outsourcing/ Facility Management	Timeshare	ASP
Delivery Model	One to One	One to Many	One to Many
Delivery Method	On-site or Remote	Remote	Remote
Delivery Location	On-site = N/A Remote = Dedicated network	Dedicated network	VPN / Internet
Hardware Ownership	Client or Service Provider	Service Provider	Service Provider
Software Ownership	Client or Service Provider	Service Provider	Service Provider
Application Variety	Client Choice	Limited Choice	Limited Choice
Application Architecture	Client Choice	Client Choice	3 tier C/S; Web
Process Outsourcing	Add-on Option	Add-on Option	Add-on Option
Contracting/Financing	Various financing vehicles	Various financing vehicles	Various financing vehicles
Proprietary/ Customized Application	Yes	Yes	No, Standard Packaged Software

Figure 2: Comparison of Traditional Outsourcer to the ASP



Traditional outsourcers require initial software licensing fees and a one-to-one relationship that may include extensive software customization. Licensing software is equivalent to buying it. Since the software is owned by the client and merely run by the outsourcer on behalf of the client (a one-to-one relationship) the client can modify it to fit their specific needs. ASPs, on the other hand, rent access to applications and share a single copy of the software among numerous customers (a one-to-many relationship, as illustrated in Figure 2). Since the application is shared, clients trade off the ability to customize the software, but gain easier maintenance and lower capital investments.

Finally, traditional outsourcing arrangements are typically delivered over private networks. Although ASPs can also deliver services over private networks, they are able to use the Internet and Web browsers to run an application. Today, most health care ASPs are delivered over a virtual private network (VPN).

TIMESHARE VS. ASP

Some question if ASPs are just a new name for the timesharing model that was popular back in the 1960s. Although timesharing has many of the same characteristics as ASPs, it is more focused on accessing computing power than providing a usable application solution.

In timeshare arrangements, corporations and educational institutions would rent time on large mainframe computers that they could not afford to purchase. Unlike ASPs, which also rent computing as a part of their offering, timesharing systems typically ran client organizations' proprietary applications that were tailored for their specific needs. The system was accessed over a private network.

ASPs provide applications with wider appeal that multiple customers can use simultaneously. So in addition to the processing power gained by centralizing applications, ASPs provide the actual application and make it widely available by offering access over the Internet or VPN.

Chapter Three

When Are ASPs Advantageous?

Although some will argue that all organizations can benefit from using an ASP, small to mid-sized enterprises that are able to use an out-of-the-box standard application and larger enterprises looking for a tactical solution are ideal ASP candidates.

Small and mid-sized organizations are more willing to trade off extensive customization for the ability to get access to an application that would otherwise have been unaffordable. Additionally, with the current competitive labor market, smaller organizations avoid the difficulty in recruiting, hiring, training, and retaining qualified IT staff. Since most ASP applications rely on Web browsers, organizations with minimal technical expertise can train Internet-savvy users more quickly than would be possible using a new software application.

Organizations that have recently merged or operate in geographically dispersed locations benefit from sharing and centrally managing a single application under the ASP model. By not running duplicate copies of an application in multiple locations, organizations save both time and money. Since data is consolidated in one central location, better decision-making and analysis are possible. The key to getting this analysis is to ensure that the ASP provides the required reporting or a tool that allows users to query and build personalized reports.

For start-up organizations or organizations entering a new market, the ability to add new computing and data processing capabilities in weeks as opposed to months is a key benefit. As an organization grows, ASPs permit the addition of more users (i.e., scale the application) to meet new demand. Since a large IT department is not needed to use an ASP, the organization can focus its resources on strategy issues rather than on building and managing infrastructure.

BENEFITS AND DRAWBACKS OF ASPS

In addition to the aforementioned benefits of using an ASP, cost savings have been reported that range from 33%-53% (ASP consortium)¹ to 50%-70% (GartnerGroup)². However, some have reported no cost savings. Since the ASP market remains in the early stages of growth, as with any emerging market, risks may be high.

When asked to rate the level of concern relative to an ASP, 25 hospitals and integrated delivery networks (IDNs) surveyed in a Porter and Associates study rated security and integrity of data, reliability, and possible downtime among the highest concerns³.

The following table summarizes the potential benefits and shortcomings an enterprise might encounter when using an ASP.

Table 2: Pros and Cons of Using an ASP

Pros	Cons
<ul style="list-style-type: none"> ■ Reduced need for capital investment (no up-front licensing or implementation fees) ■ Guaranteed performance levels/ application up-time ■ Predictable costs and lower total cost of ownership ■ Access to expertise (knowledge of applications, data back-up, and recovery) ASPs able to focus on technical issues and best practices ■ Rapid implementation/time to market ■ Enables re-allocation of resources to grow business and focus on core competencies ■ Universal access via the Internet ■ Less powerful client computer need – greater processing power via ASP computers ■ ASP acts as single point of contact ■ Easier maintenance ■ Accounting implications of incurring an operational cost vs. depreciating a capital cost* 	<ul style="list-style-type: none"> ■ Customization limits (little available; risk of loss during upgrade) ■ Connectivity problems (ASPs may not guarantee performance in this area) Client needs strong connectivity infrastructure and access to a network ■ Internet security remains a concern (Can competitors access data?) ■ Business understanding (Can the ASP understand and satisfy the enterprise's specific business needs? Lack of health care application experience) ■ Data ownership issues (Can the ASP use and sell customer's data?) ■ Adequacy of business intelligence tools to query data ■ HIPAA compliance (client ultimately responsible for HIPAA compliance) ■ True ASP software may lack the industrial strength of some traditional health care applications ■ Client IT department may resist change (perceived loss of control) ■ ASPs have little experience integrating offering with legacy systems ■ Accounting implications of incurring an operational cost vs. depreciating a capital cost* ■ Immature business model/ market consolidation (vendor dependability; start-ups may drop out)

*depends on organization tax status

Chapter Four

What Are the Different Types of ASPs?

There are basically two methods of creating an application that will be run remotely by an ASP. The first is to design an ASP application from scratch. By designing the application from the ground up, vendors are able to build the application to support multiple client users using a single copy of the software. This approach takes advantage of new Web technology and leverages use of the Internet and other high-speed network connections.

The “ASP designed” approach is best suited for standard solutions that do not require unique client design features. An example of this development methodology is Microsoft’s Web-based Hotmail e-mail offering. Functionality is standard and all processing and data storage of clients’ e-mail is done on computers run and maintained by Microsoft.

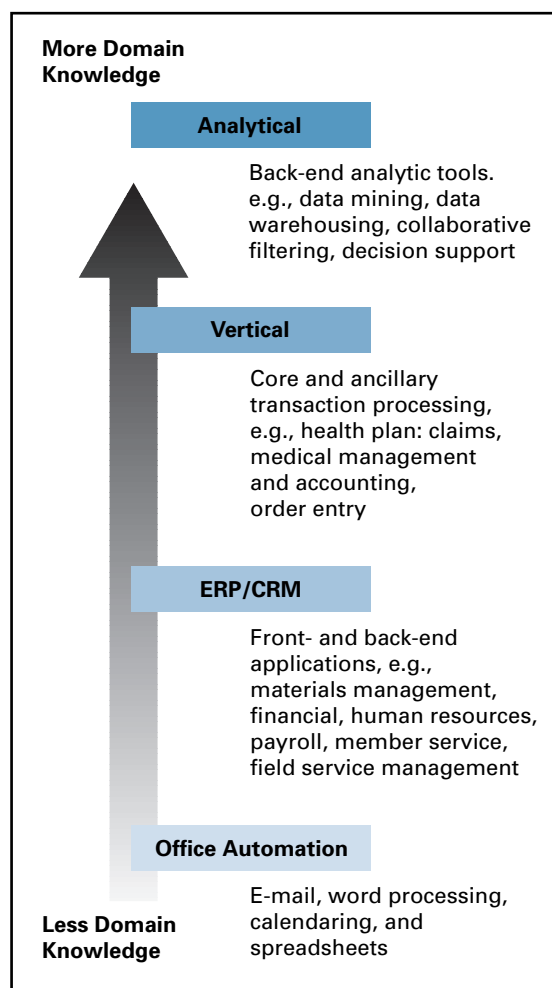
The second alternative for creating an ASP offering is to take an existing software application and “ASP-enable” it. This approach is common for older sophisticated applications that often must be customized to meet specific client needs. This approach takes an existing application and separates the visual components of the application from the computing components. Only the information that must be viewed is handled on the user’s computer, while all the heavy computing is done remotely on the ASP system. Since most legacy health care applications were written prior to the advent of the ASP model and require heavy computing power, this is the method they typically use to ASP-enable their products.

Although an ASP designed from scratch and running a single copy of the software is less complex and therefore less expensive to maintain, an “ASP-enabled” offering still offers the attractiveness of a low maintenance, low risk rental agreement.

APPLICATION CATEGORIES

The types of applications that are offered by ASPs fall along a continuum that ranges from those that require little industry specific knowledge to those that require significant knowledge of the user’s business processes and regulatory environment. Four categories have evolved.

Figure 3: Industry-specific Knowledge Needed for Four Types of ASP Applications



Office Automation: At the low end of the continuum are applications that are fairly generic and can be used by any organization regardless of the industry in which they compete. These include e-mail, word processing, staff calendars, and spreadsheet applications.

Enterprise Resource Planning and Customer Relationship Management: These front- and back-end applications are used by organizations regardless of industry but their needs and software requirements vary by industry segment. For example, unique requirements would pertain to materials management, financials, human resources, payroll, member services, and field service management.

Vertical: In the health care market, these applications include both core and ancillary transaction processing. For health plans they include claims and medical management. Health delivery applications include patient management/accounting, registration, scheduling, order entry, laboratory information, and physician practice management applications.

Analytical: Back-end analytical tools include applications such as data mining, data warehousing, collaborative filtering, and decision support.

SERVICE TIERS

ASPs offer various levels or tiers of service. Each tier is additive; without providing the service encompassed in the prior tier it is not possible to provide the next tier's service. At the low end of the service tiers, service is predominantly technical. At the high end of the spectrum, the ASP supports an entire business process.

Tier 1: The first tier of service is the base level needed to support an ASP offering. It includes the operations and management of the application environment including application monitoring, updates and upgrades, and basic customer support.

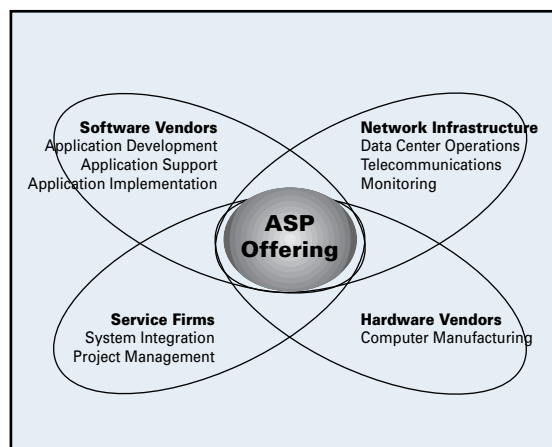
Tier 2: Tier two includes the services provided in level one plus some specialized offerings related to the configuration and integration of the applications, as well as guarantees regarding support, performance, security, and data back-up.

Tier 3: Tier three service is also known as business process outsourcing. In addition to providing the technical and additional management support, a tier-three agreement supplies services to perform an entire business process. In health care functions such as claims processing, billing, transcription, and reporting functions would be included.

ASP PLAYERS

In order to qualify as an ASP, the vendor must own or contract for multiple skill sets. Successful coordination of application software, computer hardware, network infrastructure, and services comprise an ASP.

Figure 4: ASP Players



Any one of the providers of these services can buy, partner, or contract to create an offering. Among them, the most likely candidates to deliver an ASP offering are software vendors that are seeking new channels for product sales, or service firms that are able to provide tier-two and tier-three service to add value to an ASP delivery.

Alternatively, a “pure-play ASP” can establish contractual agreements with vendors to license access to their applications and purchase or contract for the remaining elements. Although pure-play ASPs may lack deep product knowledge, they have the advantage of being vendor neutral and the capability to offer a broad selection of applications as a single source vendor.

In addition to having greater experience with and knowledge of their particular products, software vendors have the advantage of being able to offer their product through multiple channels. First they have the option of selling a traditional software license. If clients demand the application via an ASP arrangement, the vendor can either host the application or can sell the rights to a “pure-play ASP” to host the application.

PRICING OPTIONS

Regardless of the source of the solution, enterprises must choose from diverse, non-standardized pricing options. Some ASPs offer rent-to-own options, some offer just a monthly option, and others offer a low monthly subscription fee plus a transaction fee. Formulas for computing fees vary and can include variables such as volume, number of users, number of sites, revenues, and timeframes. Second-tier pricing fees for volume discounts and specific service and support options must also be considered. Finally, additional costs may accrue for implementation, integration, customization, and enhanced back-ups.

Contracting with an ASP can be as simple as accessing its Web site and clicking on the “enroll” button. Term lengths are set by the vendor and can range from a month-to-month commitment to a five-year, non-cancelable contract.

Since the ASP market is in its early development stages, many ASP vendors are flexible in negotiating payment terms. The key to picking the best terms is to understand the number of users your organization will have, the transaction volume, and expected growth rate.

Regardless of the pricing model used, an organization must understand what is included in an offering, how it is calculated, and how it will change as the organization grows. Modeling the projected cost of using an ASP vs. licensing an application will help ensure an informed decision and avoid the shock of unexpectedly high fees. Details such as who is responsible for desktop and network connectivity and whether back-ups and software upgrades are included in the monthly fee must be understood and spelled out in the contract.

To ensure high quality service, performance penalties should also be identified and documented in service level agreements (SLAs). SLAs should define how performance will be measured and, in the event of under-performance, how the penalties will be calculated and paid.

Here are some of the most common pricing methods.

- **Per-user Model:** This method can take one of two forms. First is a flat rate per user with a standard fee regardless of the customer's volume. The second method is a tiered fee structure where price varies based on the user's volume. Heavy users, defined by some number of hours per week or per month, are charged at one rate while other more occasional users are charged a different rate.
- **Per-Transaction:** This model offers low entry costs because organizations pay only for transactions as they need them. This approach is ideal for organizations that start with low volume and may have difficulty paying for a full service offering at the outset. The idea is that as capacity grows so does the ability to pay for that capacity. This model builds in volume discounts, so reaching higher volume levels affords savings and not just an additional cost.
- **Percent of Revenue:** Much like the per-transaction model, under the percent of revenue model the fees grow commensurate with the organization's level of business activity and ability to pay. However, instead of a per-transaction charge, the ASP charges a percent of revenues. This method is common for billing application ASPs that charge a percent of collected revenues billed.
- **Fixed Fee:** For organizations that prefer predictable costs, the fixed-fee model is appropriate. In this model the same fee is levied each month regardless of transaction volume. However, depending on the contract terms, a fixed cost may cover only certain blocks of users and/or a certain number of sites; additional fees may be charged for additions to either. The level of fixed cost also depends on the tier of service selected by an organization.
- **Other Fees:** As is true with traditional service providers, any of the pricing models may incorporate some add-on fees. Fees for set up, integration, training, and additional data back-up may be added at the beginning of a contract as a one-time charge. Organizations should also note whether periodic application management fees to cover the cost of maintaining application interfaces, computer hardware, and data centers are rolled into the monthly charge.

Chapter Five

Today's ASP Market

Analysts' predictions of the size of the ASP market vary considerably. International Data Corporation (IDC) projects the market will experience a five-year compound annual growth of 91% and reach \$2 billion by 2003⁴ while the GartnerGroup estimates an increase to \$23 billion during the same time period⁵. Giga, which recently adjusted its estimate downward, sizes the market somewhere between \$2 to \$6 billion by 2002⁶.

Part of the challenge of sizing the ASP market may stem from the ambiguity of the term. It's likely that the current market projections include Web and e-commerce hosting services, as well as existing outsourcing contracts that have been loosely interpreted as ASP deals. As a result, growth rates may appear much higher than they actually are.

ASPS IN HEALTH CARE

Fueled by articles in the media and an onslaught of vendor information, the market is buzzing about ASPs. However, ASPs are immature and health care organizations are not fully educated about the benefits and shortcomings of this mode of service provision. A July study released by Gartner Dataquest of 56 health care organizations revealed that only 46 percent of respondents were familiar with the ASP model⁷. Yet health care organizations are beginning to actively pursue, evaluate, and acquire ASP service offerings⁸.

Recent data on health care organization's plans to contract with ASPs (as presented in *Modern Healthcare*, April 2000) indicate that depending on the application, between 8% and 14% have signed or intend to sign a contract with an ASP, while 19% to 25% have not made a decision but are considering using an ASP.

Group practices, mid-market health plans, and small hospitals were among the first to adopt ASPs in the health care market. Increasingly, it is expected that larger health plans and health delivery organizations will begin using ASPs for niche and ancillary applications.

The ASP market is young and can expect substantial growth over the next several years. Like many emerging markets experiencing high growth, it will likely go through consolidation. GartnerGroup predicts that "by the end of 2001, 60 percent of today's ASPs will fail, due to poorly thought-out business models, poor choice of partners, inability to execute, and consolidation in the ASP market."⁹ Likely winners will be large established vendors that have extensive experience in hosting applications.

In part, the success of ASPs will be dependant on the level of service they provide and the extent to which health care organizations accept standard, non-customized functionality. One thing is certain, ASPs will influence the way many health care organizations think about maintaining their clinical and administrative applications.

What to Ask an ASP Vendor

Selecting an ASP deserves the same attention and due diligence as a traditional vendor selection or outsourcing contract. (See Table 5 for list of questions a health care organization should ask prior to contracting with an ASP.)

One of the first decisions to be made is whether to contract with a software vendor that offers an ASP version of its products or an independent ASP. The vendor is likely to have greater experience and knowledge of its product, but if an enterprise wants a large selection of applications from a single source, the vendor-neutral ASP might be the better choice.

Since the ASP market is young and will likely go through substantial consolidation, when evaluating an ASP, it is important to not only evaluate the application and service level offerings, but the vendor's overall viability. Choosing an ASP that will be a long-term player is not easy. When selecting a vendor, choosing one with a large client base and a positive revenue stream is less risky compared to one that is just starting to build a customer base and is primarily financed through venture capital. Additionally, a vendor with a commitment to health care is a lower risk alternative than one that might abandon the health care market for other opportunities.

Selecting an established health care legacy vendor that offers ASP services and has extensive experience hosting applications can also mitigate risk. If its ASP offering fails or you decide that traditional hosting or licensing the product is a better approach, the vendor should be in a position to accommodate those needs.

Most important, if an ASP fails or is purchased by a competitor, the contract should stipulate rights of succession. Health care organizations need to be confident that their business relationships will not be jeopardized in the event of an ownership change.

Application functionality, up-time, cost, and quality are the next factors to be evaluated. Ultimately, it is the health care organization that will be held responsible for an application's functionality and up-time.

The same point applies to security and standards. The health care organization is ultimately responsible for meeting security and standard requirements, so it should be careful to select a vendor that is committed to meeting stated expectations. This is especially important in light of upcoming HIPAA mandates.

In addition to the comparative cost and pricing estimates that should be conducted, it is important to keep in mind the number of service contracts the health care organization must manage. The more contracts it has, the greater its challenge in coordinating and managing those contracts. Depending on what the ASP includes as part of its offering, an enterprise might also have to find and manage services needed to run the applications (e.g., telecommunications). Minimizing complexity is desirable. ASPs that offer comprehensive service should be more attractive than those that limit the scope of their service offerings. Also in negotiating contracts, service level agreements should be used to commit vendors and ASPs to both quality and performance levels.

Data ownership and access also require attention. Since data resides at the ASP site, access to and the ownership of the data must be specified. Health care organizations have historically had difficulty accessing data for in-depth data analysis in traditional outsourced relationships. Health care organizations should be able drill down into data using sophisticated business intelligence tools to support their decision making. In order to accomplish detailed reporting, health care organizations must not only have access to the data but understand what form that data is in. Tools that facilitate access and interpretation, such as data warehousing and data dictionaries, should be provided as part of the ASP contract.

Since a health care organization's competitor may also contract with a particular ASP, owners of the data and what rights they have to it must be clearly established upfront. Ideally, in the event the data at the ASP cannot be accessed, a copy of the client's data would reside at the client site to allow them to conduct analysis to support critical business decisions.

Since the typical ASP model is a one-to-many relationship, when upgrades are made to an application, they are made by the ASP on behalf of all clients. Organizations therefore need to understand the ASP's change management guidelines for making modifications and know what influence they will have in approving those changes. In the event a client is able to make customized changes to an application to meet a unique business need, the details should clearly indicate who must pay to re-tailor the application when customization is lost during the upgrade. Therefore, in addition to software change guidelines, how and who customizes an application after an upgrade, along with who provides integration to new and legacy enterprise systems, should be considered carefully.

Since ASPs include access to an application, the use of a data center, maintenance, and various other components; clients are able to forgo some of the investment involved with purchasing hardware, supporting IT staff, and paying for a software license. As a result, total cost of ownership for a client organization can be significantly reduced. However, clients are still responsible for purchasing and maintaining the computers needed to access the application locally, as well as the networks required for connections. Ultimately, health care organizations should conduct a value assessment for both an ASP offering and a traditional licensing agreement and compare the two. The trade-offs around issues such as infrastructure cost, administrative complexity, data center operations, and staffing needs must be recognized. This exercise of building a business case forces the enterprise to make the license-versus-rent decision only after clearly identifying and weighing the benefits and limitations of fielding an application under both alternatives.

Table 3: Questions an Organization Should Ask When Evaluating an ASP

1. What experience does the ASP have in the health care market? – ASP age, size, customer base and references, help desk and technical support experience, average ASP staff tenure, company long term vision?	11. Does the ASP understand the privacy, security, and EDI requirements to comply with the Health Insurance Portability and Accountability Act of 1996 and how does it plan to comply?
2. Does the ASP offer a single vendor solution or can it offer a number of different applications to choose from?	12. Is the application stored on a dedicated or shared server?
3. What level of service does the ASP offer? Are different levels available?	13. What type of connections are available – Internet, secure line, VPN?
4. Who are the ASP partners? What is their experience? Who provides the first line of support?	14. What types of hardware and software are required at the client site?
5. What is its pricing model? – Are upgrades included in the cost?	15. How will ASP software interact with currently installed applications?
6. What is the contract duration? – Can the application be brought in-house or transferred to another ASP?	16. Will the application scale? – How many users can the ASP handle given organizational growth?
7. What type of response time can be expected? – Does the ASP guarantee network performance or only system and application performance?	17. How does the ASP handle security – Firewalls, encryption, tunneling or physical locks, personnel selection, training, and audits? What are the controls to separate data from other ASP client's data?
8. How are problems resolved? – What are the levels of escalation?	18. Who is hosting the data and how will it be backed-up? – ASP or subcontractor?
9. How, what, and when will performance reporting be handled? How will performance penalties be levied if service level agreement targets are not met?	Number and location of data centers? How is data transferred in the event the application is brought back in-house or transferred to another ASP? Will client data be resold? How often is its disaster recovery plan practiced?
10. How are software upgrades preformed? Who verifies the configuration?	

Resources

RECENT ARTICLES

“Application Service Providers, A Market Overview,” by Norin Leong, Senior Analyst Internet Research Group, 2000.

<http://www.irgintl.com/publications.shtml>

“ASPs Are for Real...But What’s Right For You?,” An IDC White Paper by Clare Gillian and Meredith McCarthy, International Data Corporation. <http://www.idc.com>

“Making Sense of the ASP Market,” *Mainspring*, July 21, 2000.
www.mainspring.com

“Is There An ASP In Your Future?,” *Internet Healthcare Magazine*, July – August 2000.
www.internethealthcaremag.com

“Evaluating ASP Pricing Models and Contract Terms,” *Healthcare Financial Management*, August, 2000.
www.hfma.org

“ASPs Can Be Cure for Cash Crisis,” *Modern Healthcare*, May 29, 2000.
www.modernhealthcare.com

ASSOCIATIONS/WEB SITES

ASP Industry Consortium: (www.aspindustry.org) an international advocacy group of well over 500 companies in 21 countries, formed to promote the industry by sponsoring research, promoting best practices, and articulating the measurable benefits of this evolving delivery model.

ASP News.com: (www.aspnews.com) a Web site of news and analysis for and about the Application Service Provider industry.

ASP Island (www.aspisland.com): a Web-based community dedicated to the Application Service Provider industry.

Glossary

Application Service Provider (ASP) A vendor that deploys, hosts, and manages access to a packaged application to multiple parties from a centrally managed facility on a subscription basis. The applications are delivered over networks or via the Internet.

Architecture: This structure term refers to a system's form and how its pieces communicate and work together. Also see client/server and tiered architecture.

Bandwidth: A measurement describing how much information can be transmitted at once through a communications medium such as analog transmission, radio frequency, or digital transmission. When the Internet experiences a "traffic jam," it's usually caused by too many people trying to access or send data at once – more data than the bandwidth can handle.

Browser: A software program that interprets documents written in HTML, the main programming language of the World Wide Web. A browser such as Netscape or Microsoft Explorer is required to experience the photos, video, and sound elements on a Web page and assists in quick, easy travel around the Web.

C+/C++: C is an established programming language found in many operating systems, including UNIX. As object oriented technology gains popularity, C++, a daughter program based on objects, is quickly becoming a favored programming language. Also see Java.

Client: Computer hardware that retrieves information over a network from a server.

Client/Server: A network system where a dedicated computer (server) handles some of the processing tasks while multiple smaller computers (clients) complete other processes by tapping into the server's shared files and programs. Also see distributed computing.

Customer Relationship Management (CRM): Information systems and software that enable an organization to manage customer relationships in an organized way with the objective of building better customer relationships.

Data Dictionary: A list that describes the specifications and locations of all data contained in a system.

Data Entry: The transcription of information from the original source into a machine-readable form. Although keyboard entry is the most familiar, other fast-growing methods include scanners, speech recognition, and automatic device-to-system technology.

Data Mining: The comparison and study of large databases in order to discover new data relationships. Mining a clinical database may produce new insights on outcomes, alternate treatments, or effects of treatment on different races and genders.

Data Repository: A database acting as an information storage facility. Although often used synonymously with data warehouse, a repository does not have the analysis or querying functionality of a warehouse.

Data Warehouse: This vast database stores information like a data repository but goes a step further, allowing users to access data to perform research-oriented analysis.

Data: Pieces of information or commands.

Database Server: A computer that stores data centrally for network users. It often uses client/server software to distribute the processing of data among itself and other workstations on the network.

Database: An aggregation of records or other data that is updateable. Databases are used to manage and archive large amounts of information. Also see relational database.

Distributed Computing: A system where tasks are divided among several computers rather than having all processes originating from one central computer. Client/server systems are one type of distributed computing.

Electronic Data Interchange (EDI): A standard transmission format for business information sent from one computer to another using strings of data. EDI also can accommodate encryption.

Encryption: Coding attached to data with the intent to keep the information secure from anyone but the addressee. Encryption can include a password, public and private keys, or a complex combination of all.

Enterprise Resource Planning (ERP): An information system that integrates business applications for an organization allowing them to manage its resources. Included can be applications for manufacturing, order entry, accounts receivable and payable, general ledger, purchasing, warehousing, sales, and human resources.

Enterprise-wide Network: A system where all computers in a health care system's various buildings are connected to exchange information.

Extranet: A loosely defined term generally describing a network that uses Internet technologies and protocols to securely exchange information between trading partners. For example, physicians that have a working relationship with a hospital may access its internal intranet through use of an extranet.

Fat Client: A computer that handles the majority of the application logic and user interface.

File Server: A PC or computer dedicated to managing the flow of information among networked computers and used as a storage location for programs and files shared by network users.

Firewall: A security device situated between a private network and outside networks. The firewall screens user names and all information that attempts to enter or leave the private network, allowing or denying access or exchange based on pre-set access rules. Also see encryption.

Graphical User Interface (GUI): An interface that allows a person to operate a software program through visual images (called icons), drop-down menu choices, and button or tool bars instead of complex keystrokes. The most common manipulating device is a mouse, and the GUI is what makes "point and click" capabilities possible.

HL7 (Health Level 7): 1. A standard interface for exchanging and translating data between computer systems. 2. A non-profit organization accredited by the American National Standards Institutes (ANSI) that develops standards for data transfer.

Home Page: The first or "main" page of a Web site. This usually acts as a table of contents for the layers of pages and additional hypertext links available within the site.

Host: A computer that acts as a source of information or capabilities for multiple terminals, peripherals, and/or users.

HTML: Hypertext Markup Language. The basic programming language for sites on the World Wide Web. This “skeleton” of codes surrounds blocks of text and/or images and contains all the display commands. A browser program is needed to interpret HTML and turn it into a graphical display on a computer screen.

http: Hypertext Transfer Protocol. A language protocol used when sites communicate. When http appears as part of a site address (called a URL), it indicates to Web browsers, “HTML spoken here.”

Interface: The electronic connection where two parts of a system are joined, such as where a software program meets a hardware component, or where hardware meets an input device. Also used to describe software that joins two different information systems. Also see graphical user interface.

Internet Service Provider (ISP): A company that provides modem or network users with access to the Internet and the World Wide Web. Although some ISPs charge by the hour, most offer monthly or yearly flat rates. Recently, telephone companies have begun to address the notion of combining Internet access rates with local telephone service.

Internet: An international network of computers that operates on a backbone system without a true central host computer. Today’s Internet links thousands of universities, government institutions, and companies, but when it was created in the 1960s the Internet linked just four computers. Technically, the Internet and the World Wide Web are not interchangeable terms; the Web is an integral child of the Internet whose ease of use has made it much more popular than its less graphical parent.

Intranet: A member-only network that looks and acts like the World Wide Web. Intranets allow companies to take advantage of Web-based technology and create a private means of exchanging images and text among their networked users.

Java: A platform-independent, object-oriented programming language developed by Sun Microsystems and modeled on C++. Java applets, miniature applications designed to run within another program, now are popular features of Web sites.

Leased Line: A telecommunications line dedicated to a particular customer.

Legacy System: An older computer system, often centered around a mainframe, that has been in place for a long time. Since rather old technology is difficult to upgrade, owners of legacy systems often are faced with weighing the cost of replacing a system that technically “still works” with a faster, less bulky, fully integrated system.

Local Area Network (LAN): A network of computer and peripherals in close proximity, usually in the same building. A LAN can facilitate high speed exchange of text, audio, and video data among hundreds of terminals.

Master Patient Index: A software database program that collects a patient’s various hospital identification numbers, perhaps from the blood lab, radiology, admission and so on, and keeps them under a single, enterprise-wide identification number.

Mission Critical: Data relating to essential business operations.

Mission-Critical Applications: A software application vital to the running of a business that includes applications such as appointment scheduling, billing, and claims processing.

Network Computer: See thin client.

Network: A general term for terminals, processors, and devices linked either by cable or wireless technology. Peripherals, programs, and applications can be shared by the network users.

Outsourcing: The transfer of an organization's IT shop, or large segments of an organization's internal IT infrastructure, staff, processes or applications, to an external company (outsourcer) that then runs those segments on behalf of the organization for a fee.

Outsourcing: Electronic processing capabilities that are not inherent in the system and so arranged through or given over to a third-party administrator.

Packaged Software Application: A non-customized computer program developed for sale to multiple customers as is. While some tailoring of the program may be possible, it is not intended to be designed for each user or organization.

Performance: A term used to describe how well a computer system is functioning and relates to the overall productivity of a system, availability, and response time.

Push Technology: The automatic delivery of Web news and other information without a request from the user. Unlike Web pages and most e-mail where a user downloads or "pulls" information each time in order to view it, push technology requires a user to be recognized only once and sends the news as it becomes available, without further prompting. PointCast and Microsoft's Internet Explorer 4.0 both include health care push news services.

Relational Database: A database where all information is arranged in tables containing predefined fields. Using structured query language, reports and comparisons can be generated by selecting fields of interest from the original database and creating new tables. Changing a field in one record automatically changes the same record in all related databases, allowing for easy global updating.

Remote Access: The connection of two or more computers, via a network, for the purpose of accessing applications and information located in another, remote location.

Remote Computing: Using a network to contact someone else's computer system, usually through telephone lines, in order to access software and other information. Through remote computing agreements, health care organizations can use their own networks to access resources.

Scalability: The ability to add users and increase the capabilities of an application without having to making significant changes to the systems or application software.

Server: A computer on a network that manages a specific set of network resources. A server may manage network traffic, printer use, store files, or run remote applications.

Service Level Agreement (SLA): A contract between a service provider and a user that specifies the level of service that is expected during a contract term. SLAs determine how performance will be measured and, in the event of under performance, how the penalties will be calculated and paid.

Structured Query Language (SQL): A standard command language used to interact with a database.

T1,T3,T4: Types of transmission lines in the T-carrier telecommunications system. T1 lines can transmit about 1.5M bps of data. A T3 line contains 28 T1 lines together and can transmit about 45 times the data of a single T1, enough for full motion video. Six T3 lines make one T4 line, capable of transmitting about 274 Mbps.

Thin Client: A computer designed with little processing power whose primary function is to access applications and and/or data from a central server where heavy computing is handled.

Tiered Architecture: A client/server network whose structure is divided into several levels. The system's application responsibilities are apportioned to different parts of the network, relieving the main server of some of its duties.

Total Cost of Ownership (TCO): Takes a long-term view of all costs associated with a specific technology investment. These costs include the cost of acquiring, installing, using, maintaining, changing, and disposing of a technology during its useful life.

URL: Short for Uniform Resource Locator. Much like a postal system, every page on the World Wide Web has a unique address, or URL. To visit a Web site, the user types the address into the browser program. The URL also reveals whether the site originates from an educational (.edu), corporate (.com), governmental (.gov), or other source. Most international sites add a two character country code.

User Interface: The part of an application that allows the user to access the application and manipulate its functionality. It includes menus, forms, and command buttons.

Virtual Private Network (VPN): A network that is designed using a secure, encrypted private Internet connection.

Web Ring: A group of Web sites on related topics linked with each other in a loop, allowing a user to take a circular "tour" within specific subject area. As of August 1997, there were 66 rings on health subjects, including an international medical library ring of 20 sites.

Web Server: A networked computer that stores and transmits documents written in hypertext markup language, the formatting language for Web-based documents, or other languages, and other data to browsers via the hypertext transport protocol, an Internet-based data transfer protocol.

Web site: A group of related files, including text, graphics and hypertext links, on the World Wide Web. Accessed by typing its unique address, a site usually includes layers of supporting pages as well as home page. Also see URL and browser.

Web-enabled: Software programs that can be used directly through the World Wide Web, including financial and human resources applications. Web enabling extends the client/server concept to the largest server in the world-the Internet.

Webmaster: The person in charge of a particular page or collection of pages.

World Wide Web: An international group of databases within the Internet that uses hypertext technology to access text, pictures and other multimedia with a click of a mouse. Sites on the Web usually are created in HTML, Java or both. A browser program is needed to access multimedia aspects. Also see URL.

Citations

- ¹ The ASP Consortium, Frequently Asked Questions (www.aspindustry.org)
- ² Terdiman, R., “ASP Trends: The ASP Model Moves Closer to ‘Prime Time’,” GartnerGroup Research Note, January 11, 2000.
- ³ Porter and Associates, ASP Market Needs and Opportunities, June, 2000, p.50.
- ⁴ Williams, Arthur, “Measuring the ASP Market Today and Tomorrow,” *Gigia IdeaByte*, March 31, 2000.
- ⁵ Williams, Arthur, “Measuring the ASP Market Today and Tomorrow,” *Gigia IdeaByte*, March 31, 2000.
- ⁶ Williams, Arthur, “Measuring the ASP Market Today and Tomorrow,” *Gigia IdeaByte*, March 31, 2000.
- ⁷ Gartner Dataquest, Application Service Provider Opportunities in the Health Care Industry: An Emerging Model, August 14, 2000, p.3.
- ⁸ Morrissey, John, “ASPs Can be Cure for Cash Crisis,” *Modern Healthcare*, May 29, 2000, p.21.
- ⁹ Apfel, A., ASPs: Myths and Realities, GartnerGroup Research Note, January 14, 2000.