

Implementing a Web Strategy

Enterprise Information Portal Deployment Considerations

In this time of rapid organizational change, increased integration and interdependency among organizations, and information overload, companies are searching for ways to provide more and better service to their customers, business partners, and employees. The enterprise information portal concept (referred to in this paper simply as “portal”) has emerged as a strategy for facilitating the dissemination of information, providing self-service capabilities, and improving communications and interaction with and between partners. This paper examines the portal concept in more detail, reviews the components of a portal, and identifies architectural issues that must be considered in any implementation effort.

What Is an Enterprise Information Portal?

As is often the case with computer industry terminology, the term “enterprise information portal” has different meanings to different people. Generally, portals are thought of as a means to aggregate content and services to make them more easily accessible by users; in real life, the “content and services” provided via portals span a very broad range, and any given portal can implement many different services concurrently.

A significant number of portals are used for some form of information distribution or knowledge management, whether that be Human Resources (HR) documentation and policy handbooks, product literature, legal briefs, or syndicated industry news. By logically grouping material and providing robust search and metadata capabilities, portals can make it much easier for users to find, access and repurpose information. The prototypical example of this is the corporate intranet, used by many organizations to disseminate information (such as the corporate address book) to employees and to allow employees to easily request corporate services. Other information

distribution examples include customer self-service and support (providing product information or descriptions of the latest bug reports and resolutions), partner extranets (providing sales, marketing and training information to corporate channel partners), and customizable streaming media and news. Along with the information itself, portals may also provide a content management workflow that allows administrators to review information before it is made available and to govern which information is presented, how, when and for how long.

Another typical usage of portals is to streamline business processes. Many companies are either evaluating or implementing portals for “interaction-intensive” business processes such as those found in HR departments. With an HR portal, employees can find answers to questions about benefits or insurance coverage, submit requests for changes to personal information, benefits coverage or insurance beneficiaries, maintain a corporate skills database or review the performance of their 401(k). HR personnel can benefit

IT Solutions. Fixed Price. Fixed Time.™

www.eforceglobal.com

from having a single point from which they can initiate processes such as hiring and termination, recruiting, EEOC and OSHA interactions, and others, while taking advantage of employee self-service and updating. The resultant benefits of re-purposing HR personnel toward higher value activities while decreasing processing time and manual errors can be significant. Each of these functions typically involve multiple steps - gathering information, submitting it to multiple groups, checking on progress, and forwarding results, for example – and most of these steps can be improved through the use of a portal.

Other portals focus on facilitating collaboration amongst the users of the portal and between users and the portal's corporate host. In its simplest form this collaboration can take the form of chat, news, e-mail, and other communications tools; more advanced applications require facilities such as shared workspaces or collaborative design tools.

In some situations, companies find it helpful to facilitate off-site or off-hours work by providing access to a “virtual desktop” containing links to applications used by internal users (employees), external users (partners, suppliers, etc.) or both. These links can be created by developing Web interfaces to commonly used applications, by invoking 3270/5250 terminal emulation programs, by using desktop access tools such as PCAnywhere or Hummingbird's XWindows emulator, or by creating a secure network pipe through which TCP/IP based applications such as “telnet” can be used.

In addition to providing access to information, applications and collaborative tools, portals can also allow users to personalize their workspace to meet their particular needs. Personalization capabilities allow users to see only that information that is relevant to them, and to arrange their view of the portal in a way that facilitates their own personal work style.

ENTERPRISE INFORMATION PORTAL USERS

User Community	Portal Content and Services
Employees	“Classic” intranet, including company news, corporate directory, human resources functions, etc.
Customers	Customer “self-service” portal, including product information, online technical support, product training, account/contact information updates, electronic bill presentment and payment, etc.
Partners and Resellers	Partner information portal, including product literature, sales and marketing information, product training, internal corporate contact information, etc.
Suppliers	Business process functions, including order placement and status, bill presentment, online lists of product needs with bidding facilities, etc.

The key to a successful portal is to identify a community of users with common interests or needs, then implement targeted content and services to benefit that community.

Enterprise Information Portal Users

When considering a portal implementation, the key to success is to identify a community of users with a common interest or need, and then bring together content and services that are targeted specifically to that community.

User Community - Portal Content and Services

The community served by the portal can be composed of any individuals that interact with the enterprise; in fact, many portals are designed to serve communities that span multiple different types of users. Both customers and partners of an enterprise can, for example, benefit from access to product and technical support information, training materials, development tools, and links to related technologies.

Components of an Enterprise Information Portal

Depending on its purpose, a portal can be composed of many different components. Some of these components are specific to the portal, while others are key enterprise infrastructure components or shared services that are leveraged across multiple applications.

Gartner Group forecast that some 60% of Fortune 500 corporations will be deploying an enterprise portal by 2003.

Hardware

Selecting the best hardware platform for a portal requires the same thought process as is used for any other enterprise application. The decision should be based on criteria such as corporate preferences and standards, the number of concurrent users and the overall size of the user community, disk space requirements, system availability and fault tolerance considerations, user training requirements, software development skills and other factors.

While it is often not necessary to run a portal on a dedicated hardware platform, there are some situations in which it may be highly desirable to do so. For example:

- Some or all of the portal's hardware may be placed in a network "demilitarized zone", where it can be more easily accessed by outside users without compromising enterprise security.
- The size of the portal user community may warrant a dedicated system that can be scaled without impacting other applications.
- The choice of software to support the portal may mandate a certain type of hardware and/or operating system, which is not currently available within the enterprise.
- The administration of the portal is the responsibility of a group that should not have administrative access to other enterprise systems.

Portal Foundation Software

Successful, functional portals can be built from a wide range of different software. The requirements for a portal's foundation software are dependent on the functionality to be provided, the number and type of users to be served, security considerations, and other factors.

The classes of software that can be used are discussed in more detail below.

Application Servers

Application servers such as BEA's market-leading WebLogic Server form an excellent framework for highly scalable portal systems. These systems provide support for J2EE components such as Java Server Pages (JSPs) and Enterprise Java Beans (EJBs), performance features such as database connection pooling, excellent support for LDAP-based security systems, and strong integration with external applications and packaged software through message-oriented middleware and other facilities. Thanks to the widespread adoption of the J2EE framework and the Java programming language, it is fairly easy to find contract or permanent personnel with the skills required to implement portal systems and perform application integration using an application server platform.

In addition to the core application server, BEA, Sun Microsystems, ATG, IBM and other application/commerce server vendors offer many complementary packages that are appropriate for portal implementation. For example, BEA's Personalization Server offers portal implementers two key additional

capabilities: portal construction tools that ease the development of customizable user interfaces, and facilities that allow customization of a user's interaction with the portal or the content delivered based on user profile characteristics, past interactions, or even a user's clickstream during the current session.

Application servers such as WebLogic or WebSphere, for example, used alone or in cooperation with other personalization or commerce products, are good candidates for portal implementation in almost any situation. However, keep in mind that it may take more work to implement a portal if the user interface needs are substantial (heavy user customization, for example) or when there is no existing code that can be leveraged to build the required functionality. In situations such as these, it may be appropriate to consider complementing the application server with a dedicated portal server.

Dedicated Portal Servers

Dedicated portal servers, such as Epicentric's Foundation Server, have recently become strong contenders in the portal market. These products ease portal development by including commonly used, customizable functions such as support for document folder access, streaming media and news, e-mail, and pre-integrated connectors to popular applications such as Microsoft Outlook, the Documentum 4i document management solution or the Siebel customer relationship management system. Portal servers also provide tools and application interfaces to support role-based security and ease system administration, and can support performance features such as caching, concurrent requests and database connection pooling.

Portal sever architectures vary widely. Some portal servers, such as the Epicentric Foundation Server, leverage the substantial functionality and extreme scalability of J2EE application servers as their foundation. Others provide their own proprietary foundation and application development framework, and yet others provide no application development facilities at all, preferring to let users leverage CGI, ASP and other technologies for application development.

The choice of a dedicated portal server depends on cost, on the amount of "out of the box" functionality that can be leveraged, the availability of application servers, the development skill set available, and the scalability needs of the portal.

Content and Document Management Systems

Content management systems such as Vignette and Interwoven and document management systems such as Documentum's 4i suite can be excellent choices for building systems that focus on distributing dynamic content such as news, streaming media, product information, documents, and content-oriented services to large numbers of users. These systems generally provide very strong development and management/workflow

capabilities that allow for editorial review, automated content publishing and expiration, document storage, search and retrieval, and integration with XML-based data sources. In conjunction with built-in or third-party personalization tools, these systems can optimize use of the portal by customizing it for each individual user based on their profile, interactions during the current work session, or similarities to other users.

Because they specialize on content management, however, these systems are often not the best choice in situations where tight integration with other applications is required. Other products (dedicated portal servers, for example) may provide an appropriate level of content management capability and provide much stronger integration facilities. Consider building a portal based on a content or document management system when the focus is on information dissemination, particularly if the content is highly dynamic. If very strong content management capabilities are required along with application integration (and you can justify the cost), consider the use of both a dedicated portal server and a document management solution.

“Freeware” Portal

Depending on the nature of the portal, the functionality required and the number of concurrent users to be supported, it is often quite possible to build a simple portal with nothing more than a Web server, a set of Perl scripts, and a database. More sophisticated portals can be built for the cost of labor, using free tools such as the Apache product suite (which includes a web server, JSP engine and a many other tools), Microsoft’s Active Server Pages (ASP), and the MySQL database. If cost is a major factor, the functionality is straightforward, the user community is of moderate size, and the development expertise is available, then this is a very viable solution.

Security

Security is often a major concern for enterprises considering the creation of a portal, particularly if that portal will serve users outside the organization, such as partners or suppliers. Because portals often integrate applications across diverse operating systems – mainframes, midrange systems, Unix and Windows NT – building and implementing a security system that supports all of these users can be a large undertaking.

There are several different aspects to security that must be considered:

- Firewalls: How do I prevent unauthorized outsiders from accessing the portal?
- Choice of security repository: Is an LDAP-compliant directory service needed, or can an existing security system such as Windows NT Domain security be used?
- Authentication vs. authorization: How can I explicitly identify portal users and control their access to the system?
- Delegated administration: How do I allow authorized group administrators to administer their portion of the portal?
- Roles and permissions: What are the roles that will use this portal, and what do they need access to? How do I handle users with multiple roles (employees that are also managers, for example)?
- Integration with external packages: How do I integrate with third-party authentication systems such as RADIUS? Can I synchronize my local security repository with other corporate security systems, such as IBM’s RACF?
- Single sign-on: How do I manage security and sign-on across multiple diverse applications?
- Session-level encryption: Do I need to use SSL to protect sensitive data while in transit? If so, what do I encrypt?
- Entity-level encryption: Do I need to encrypt sensitive content using a PKI solution?

Several systems handle some of these needs, but no system handles them all. The solution may include a combination of products: LDAP directories such as the Sun Microsystems Directory Server to handle roles and permissions, firewalls such as Check Point Software’s Firewall-1 or SonicWall’s SonicWall GX for network access control, or products such as Netegrity’s SiteMinder for single sign-on. The decision on how much security is enough (or too much) depends on the sensitivity of the data and applications being shared, the composition of the user community, and any current security mechanisms in place.

User Interfaces

The choice of a user interface (or a combination of user interfaces) is, in some ways, linked to the choice of portal foundation software. The use of web browsers as the primary portal user interface is almost a given, and all packages provide support for distribution, branding and display of content in HTML and/or XML format; however, once alternative interfaces such as those for wireless devices and voice-driven interfaces are introduced, the product field thins considerably. The key difficulty with support for alternative interfaces is in translating the robust, highly graphical content into a form that smaller, less powerful devices with lower bandwidth can support. Support for wireless devices such as cell phones, personal digital assistants (PDAs) and other hand-held devices is growing as the underlying technology matures and as increases in available telecommunications capacity become available.

Similarly, some portal applications benefit from the ability to personalize the interface to meet a particular user’s content and usage needs. Content management systems and dedicated portal servers can provide some personalization capabilities (such as the ability to rearrange modules on a page, change colors and fonts, etc.), but to provide real-time personalization of delivered content based on a user’s preferences it is often necessary to use the capabilities of a third-party personalization server such as BEA’s WebLogic Personalization Server.

Databases

Use of a database management system is almost a given in all but the most basic portals. Relational databases are the type most often used, and can store content and user preferences, support portal-based applications, and, using stored procedures and triggers, even do data-level application integration. The two leaders in the relational database market, Oracle and SQL Server, both provide robust features and can scale to support thousands of users.

Workflow

In some portals, workflow is an indispensable part of the overall system. A well-designed workflow system can streamline business processes, reducing labor costs, improving process accuracy and repeatability, and reducing the number of times work is delayed. Workflow systems can also help in tracking the status of requests, letting users know who is currently handling requests and when a given request might be completed.

Depending on the complexity of the business processes to be handled, a simple workflow system can be constructed using facilities provided by the portal foundation software. Some dedicated portal servers have very simple workflow systems (typically used for content management) which can be adapted to basic tasks, and the workflow engines provided with some content and document management systems may be suitable for some tasks. For

more complex processes, however, a dedicated workflow engine is preferable.

Content and Applications

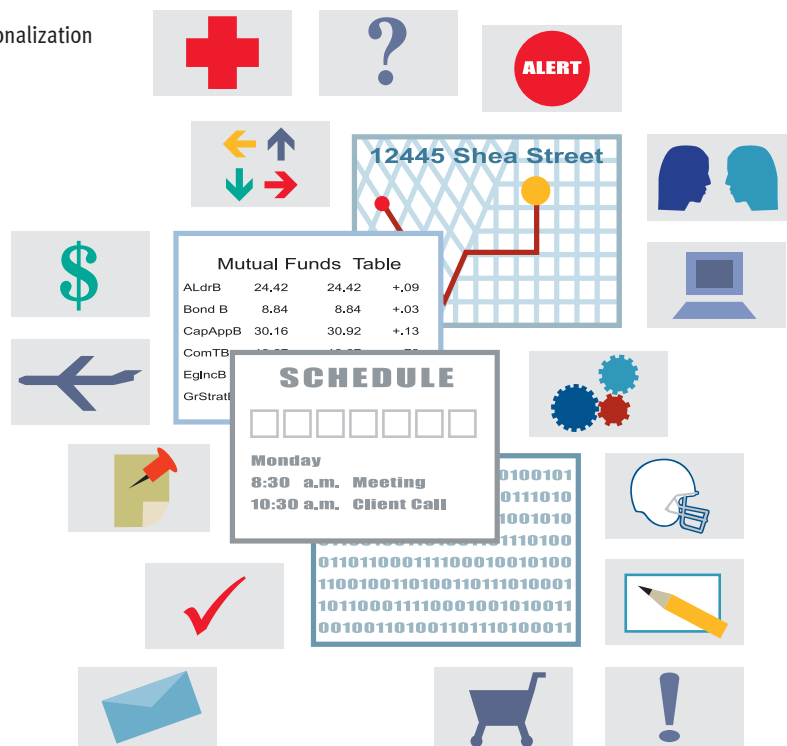
No portal is complete without content and applications. Many of these features are provided “out of the box” by dedicated portal servers such as the Epicentric Foundation Server. Epicentric and other portal server vendors not only package tools such as those listed above (called “modules”, “portlets”, “gadgets”, or some other name, depending on the vendor) with their systems, but also provide on-line networks where users can exchange, download or purchase additional modules. Vendors of packaged software products such as Microsoft, SAP, PeopleSoft and Siebel have also worked with these vendors to develop adapters that allow their products to be accessed through dedicated portal server software.

Of course, all of the portal framework systems can also support development of portal applications. These applications fall into three classes:

- “Native” portal applications, in which the complete functionality required is implemented within the portal;
- “Front-end” applications, where the portal application serves as a front end to applications running on other systems;
- “Pass-through” applications, which either provide a simple link to another application or pass the user interface of the application through to the user without modification.

There is an amazing diversity of portal features that can be provided to users, including:

- File/document directories, with upload and download
- Syndicated content (news, etc.), including filtering and personalization
- Streaming media
- Search and indexing
- E-mail
- Bulletin boards
- Bookmark managers
- Calculators
- Calendars
- HTML and XML page display (from other sites)
- User directories
- Discussion boards
- Event planning tools
- Messaging interfaces
- Notepads
- Suggestion box
- Task list
- Web directory
- Web search
- Polls
- “Voice over IP” telephony
- Newsgroups and mailing lists
- “Shared workspace” or “collaborative design” tools



Native Portal Applications

Typically, native portal applications are composed of up to four components:

- A “main view” which appears on the main portal page and provides an entry point into the application. Because it must co-exist on the portal user’s main page with many other applications, this view is generally only a very small window into the application.
- A view that allows properties of the application to be customized for a particular user. While this view is not required, it is highly desirable in order to enhance the usability of the system.
- A view that allows system administrators to configure the application for all users. Again, this view is not required, but is recommended to provide maximum flexibility.
- One or more Web pages that comprise the application itself. These pages can be created either by the portal application or by a remote application (such as another Web site).

Because they exist entirely within the portal environment, the creation of native portal applications is fairly straightforward. Different portal frameworks provide different applications interfaces and different infrastructure capabilities, but almost all allow application development using Java and some or all components of the J2EE framework.

Front-end Portal Applications

In situations where users require access to an enterprise application that does not expose an HTML or XML user interface, it may be appropriate to implement that interface inside of the portal itself. As with native portal applications, the implementation of the user interface is fairly straightforward; however, difficulties may arise when integrating that interface into the application. Depending on the capabilities of the enterprise application, it may be necessary to perform development in both the portal and the application in order to allow the user interface to invoke back-end functionality using message-oriented middleware, CORBA, COM+, “screen-scraping”, or some other mechanism.

The portal and the application must also agree on the identity of the user and on that user’s permission to invoke application functionality. Because enterprise applications usually have an investment in some sort of security (based on LDAP, another security package such as RACF, or a proprietary security mechanism), using a portal to implement the user interface may require the integration of the application’s security with that of the portal itself. If both security systems support a standard such as LDAP, then this integration may be easy; however, if the systems are dissimilar it may be necessary to provide a batch synchronization tool in order to propagate user information between the two systems.

Pass-through Portal Applications

When the remote application already has an HTML or XML user interface, then simply exposing that interface to the user through the portal may be the most straightforward approach. Even when the application does not support HTML or XML, it may be possible to provide access via a third-party terminal emulation program, a remote desktop access tool such as PCAnywhere, or even via a Unix or Windows NT command line.

With pass-through applications, it is still necessary to provide a small application view that can be displayed on a user’s main portal page.

Implementing an Enterprise Information Portal

Making the business decision to implement an enterprise information portal is often the easiest part of the implementation process. Moving from concept to implementation, on the other hand, can be a challenging and frustrating experience unless the right amount of prior planning is done up front.

One Portal or Many?

Once it is decided that portals can benefit the organization, the next step is to decide *how many* portals to implement. Often, a lack of clear understanding of the purpose of the portal and how the user community will use it leads organizations to design a single, massive portal with all possible functionality included. While it is possible for such a strategy to succeed (albeit with heroic efforts on the part of all involved), a single portal is often not the best approach, for a variety of reasons:

- An organization may need to segregate portals along business lines in order to allow better allocation of development and maintenance costs, group similar applications, or simply eliminate any possibility that a portal security breach may compromise sensitive applications.
- It can be much easier to develop portal ROI estimates, find senior management “champions” and get stakeholder involvement if the portal development effort is broken down by line of business or user community.
- Development effort can often be reduced and performance improved if the portal is co-located with the data or applications it will access.
- A distributed portal architecture can reduce the load on a given machine or cluster and improve overall scalability.
- An unrecoverable error (or a system maintenance window) in a single large portal may cause a loss of availability of the entire portal. Organizations that operate twenty-four hours a day, 365 days a year may find these outages unacceptable.
- Different portal implementation considerations, such as the desire to use a particular technology because of benefits it can provide in application integration, may lead to excessive development effort when implementing other portal related functionality.

By clearly defining the users that need access to each portal-related application, a set of discrete portals can be identified, each serving a distinct user community. Access to all portals can be provided through a corporate “launchpad”, and yet each portal can be developed, maintained and used independent of the others.

Gathering Portal Requirements

Gathering requirements for the portal, especially if it serves a diverse user community, can be the most challenging part of the implementation effort. It is critical at this stage to understand the overall goal of the portal, the nature of the user community, and the business priorities and relative benefit of each proposed piece of functionality.

If the portal being considered is the organization's first portal effort, then it is particularly important to start with a modestly-scaled effort that implements high-benefit, "low-hanging fruit" functionality. Keeping the initial effort small gives the development team an opportunity to become familiar with the portal framework software and development process, reduces the time to benefit for the organization, allows for changing business and functional priorities, and improves the likelihood of having stakeholders remain motivated and highly involved in the process. Requirements for subsequent portal development iterations should also be scaled to ensure that stakeholders see benefit in a relatively short amount of time.

Some questions that should be asked during the requirements gathering process include:

- Who will use the portal? What is their level of familiarity with computers, and what level of additional training will be necessary to allow them to use the portal productively?
- By stakeholder and user standards, how will the portal be judged to be successful?
- Approximately how many users will register to use the portal, and how many will be actually using the portal at any given time?
- What are the portal availability and performance/scalability needs?
- Should the portal support internationalization and localization? If so, what languages character encodings, date-time formats, etc. should be supported?
- Will the portal be used only from within the organization's offices, or will it be accessed from outside the organization? How fast are the connections used to access the portal from outside the organization?
- Is there a need to access some or all of the portal from a wireless device?
- Should the portal provide a consistent look-and-feel to all applications that are accessed through it?
- What are the portal's eCommerce needs?

Choosing the Portal Framework

Once the business requirements are gathered, it should be a fairly straightforward exercise to choose the best technology to meet those requirements. Of course, it's not always possible to purchase software solely for use in building a portal, especially with the current cost of portal foundation software; however, it's still a good idea to review the options so that the cost of purchasing new software (both initial cost and ongoing maintenance fees) can be weighed against any increase in development effort and cost, opportunity costs, and so on.

Issues to consider when selecting portal framework software include:

- Performance and scalability, including support for clustering and distributed architectures (including the ability to distribute portions of the portal across firewalls and/or "demilitarized zones")
- The flexibility of the security model
- Fault tolerance and disaster recovery capabilities, including support for system monitoring packages such as CA-Unicenter, BMC Patrol and HP OpenView.
- Extensibility, including support for third-party packages
- Personalization capabilities, including both look-and-feel and personalization based on user preferences and behavior
- The availability of pre-built portal components that provide desired functionality
- System administration facilities, including the ability to delegate administration of portions of the portal to others.
- Support for various client-side technologies, particularly HTML, XML and WAP
- Whether or not the portal supports the downloading and execution of applets, web-based terminal emulators, etc.
- Internationalization and localization support
- The available developer skill set
- Remote application integration facilities
- Support for industry standards

It's always a good idea to talk with customers of vendors being considered, and to request benchmark information (along with explicit descriptions of how the benchmarks were conducted).

Development and Testing

By the time the portal reaches development many of the difficult decisions have been made, and success rests on the development team's ability to execute. As with any software development project, successful portal projects depend on well-defined requirements, strong project management, control of project scope, risk assessment and mitigation, effective user interface design, good technical design and development skills, and thorough testing.

A key point to remember is that the portal architecture should handle upgrades to its foundation software as gracefully as possible. Generally, this is only a major problem if the development team uses undocumented interfaces, modifies the foundation software in some way, does not follow good development practices, or if the vendor removes support for some aspect of the software that is being used by the portal. Use of an abstraction layer such as the J2EE framework can help insulate the portal applications from changes in the underlying software.

Another, often-overlooked aspect of development is the creation of user- and system-level documentation. While the creation of extensive documentation can be considered a low-priority task by some organizations, implementation of a comprehensive help system can improve user adoption of the portal and reduce support costs.

Portal Deployment

The level of effort required to deploy a portal depends heavily on the nature of the portal itself and its relationship to other corporate systems. Self-contained portals can be fairly straightforward to deploy, while portals that are dependent on concurrent updates to other enterprise systems may require careful coordination and post-deployment monitoring.

After a period of use, it is important to follow up with a survey of users to ensure that the portal is meeting its business objectives.

ROI

The return on investment (ROI) derived from use of a portal can be realized across a number of different metrics. Those metrics are determined by the business goals of the organization, and the way in which it uses the portal. For example, human resource departments have attained measurable cost savings from deployment of Web-based employee self service portals that encourage personnel across the enterprise to take responsibility for updating personal benefits information. As a result, HR personnel can be redeployed to more value-added activities, while manual data entry errors are minimized. Recurring questions about 401K, medical and dental benefits can be centralized in a FAQ feature within the portal, further relieving the support burden on HR personnel, while shortening the time it takes employees to solve their problems. An independent research audit by the Delphi Group showed up to a 1,800% return on portal investment based on a reduction in ‘searching’ time per employee, combined with a dramatic reduction in paper and printing costs.

J.D. Power and Associates correctly identified that for companies to stay competitive in the U.S. Web-based multichannel market, companies need to offer consistent service levels across the customer care infrastructure. Customer Portals result in strong customer satisfaction metrics, which translate into incremental sales and revenue, through benefits such as ease of use, content suitability, availability, billing and speed. Datamonitor, for example, determined that poor customer service cost retailers \$6.1 B in lost sales in 1999.

ROI resulting from portal deployment can be attained, and should be expected, whether the portal is deployed for customers, trading partners or employees.

What Happens Next?

Even before the current portal development effort is completed, work should start on defining the requirements for the next iteration of development. Overlapping work in this way reduces the time between iterations, accelerates time to benefit, and keeps the development team, stakeholders and executive sponsors engaged and excited.

Summary

Portal implementation, while not substantially more difficult than any other enterprise software development initiative, does require significant planning and a clear understanding of the requirements in order to be successful. There are many choices to be made – which users are to be served, which portal foundation technology is the best fit, and how to integrate with existing applications, among others – but with a thoughtful ROI analysis in hand and a realistic approach to project execution, the benefits in improved productivity and accelerated business processes can be well worth the effort.

About eFORCE

eFORCE specializes in the Fixed Time, Fixed Price implementation of solutions encompassing all areas of the Enterprise Value Chain - Enterprise Portals, CRM, EAI, Business Intelligence, Enterprise Content Management and Enterprise Infrastructure. Combining expertise in business architecture, technical architecture, design, deployment and maintenance, eFORCE delivers production-scale enterprise solutions that result in measurable ROI for Global 1000 customers. eFORCE delivers solutions based on best-in-class enabling technologies such as ATG, BEA Systems, BroadVision, Documentum, E.piphany, HP, IBM, Interwoven, Netegrity, Plumtree, Sun Microsystems, MatrixOne, Microsoft, Oracle, SeeBeyond, Siebel Systems, Stellent, TIBCO and webMethods. eFORCE (www.eforceglobal.com) is headquartered in Silicon Valley, has Development Centers in North America, Europe and India, and through its Global Delivery Methodology provides both onshore and offshore full lifecycle deployment – design, development, deployment, maintenance, support.

Contact eFORCE

800.295.1914
 510.265.5800
sales@eforceglobal.com

eFORCE Clients

Alcatel	CapitalOne	Fidelity Investments	The Hartford	Kingfisher B&Q	Nortel Networks
American Express	Charles Schwab	France Telecom	Hoover's	Lucent Technologies	Pearson Group
Avaya	Compaq	GE Capital	Intel	MasterCard	Reuters
Bank of America	Credit Suisse	GTE Genuity	Janus	Merrill Lynch	Visa USA
British Telecom	FedEx	H&R Block	Johnson Controls	Mitsubishi	Wells Fargo