

An Introduction to
Web Services

TABLE OF CONTENTS

What Are Web Services?	3
Definitive Characteristics	4
What Do Web Services Do?	5
Integration	5
Service Re-Use	5
Business Flexibility	6
Building for the Future	7
Web Services ROI	7
Keys to Web Services Success	8
Contact	9

WHAT ARE WEB SERVICES?

A Web Service is, in essence, programmable application logic accessible using standard Internet protocols. Web Services combine the best aspects of component-based development and the Web. Like components, Web Services represent functionality that can be easily reused without knowing how the service is implemented. Unlike current component technologies that are accessed via proprietary protocols, Web Services are accessed via ubiquitous Web protocols (ex: HTTP) using universally accepted data formats (ex: XML).

In practical business terms, Web Services have emerged as a powerful mechanism for integrating disparate IT systems and assets. They work using widely accepted technologies and are governed by commonly adopted standards. Web Services can be adopted incrementally with little risk and at low cost. Today, enterprises use Web Services for point-to-point application integration, to reuse existing IT assets, and to securely connect to business partners or customers. Independent Software Vendors (ISVs) embed Web Services functionality in their software products so they are easier to deploy.

From a historical perspective, Web Services represent the convergence between the service-oriented architecture (SOA) and the Web. SOAs have evolved over the last 10 years to support high performance, scalability, reliability, and availability. To achieve the best performance, applications are designed as services that run on a cluster of centralized application servers. A service is an application that can be accessed through a programmable interface. In the past, clients accessed these services using a tightly coupled, distributed computing protocol, such as DCOM, CORBA, or RMI. While these protocols are very effective for building a specific application, they limit the flexibility of the system. The tight coupling used in this architecture limits the reusability of individual services. Each of the protocols is constrained by dependencies on vendor implementations, platforms, languages, or data encoding schemes that severely limit interoperability. And none of these protocols operates effectively over the Web.

The Web Services architecture takes all the best features of the service-oriented architecture and combines it with the Web. The Web supports universal communication using loosely coupled connections. Web protocols are completely vendor-, platform-, and language-independent. The resulting effect is an architecture that eliminates the usual constraints of DCOM, CORBA, or RMI. Web Services support Web-based access, easy integration, and service reusability.

As stated earlier, a Web Service is an application or information resource that can be accessed using standard Web protocols. Any type of application can be offered as a Web service. Web Services are applicable to any type of Web environment: Internet, intranet, or extranet. Web Services can support business-to-consumer, business-to-business, department-to-department, or peer-to-peer interactions. A Web Service consumer can be a human user accessing the service through a desktop or wireless browser; it can be an application program; or it can be another Web Service. Web Services support existing security frameworks.

Today, Web Services products like will operate anywhere, in almost any existing IT setting.

DEFINITIVE CHARACTERISTICS

Web Service exhibits the following definitive characteristics:

- A Web Service is accessible over the Web. Web Services communicate using platform-independent and language-neutral Web protocols. These Web protocols ensure easy integration of heterogeneous environments.
- A Web Service provides an interface that can be called from another program. This application-to-application programming interface can be invoked from any type of application client or service. The Web Service interface acts as a liaison between the Web and the actual application logic that implements the Service.
- A Web Service is registered and can be located through a Web Service Registry. The registry enables service consumers to find services that match their needs.
- Web Services support loosely coupled connections between systems. Web Services communicate by passing messages to each other. The Web Service interface adds a layer of abstraction to the environment that makes the connections flexible and adaptable.

WHAT DO WEB SERVICES DO?

Web Services are really just applications, so fundamentally they do what your applications do today. However, the way they do things is different. Consider three things that Web Services are especially good at:

INTEGRATION

In most large organizations, the data and logic of one application are basically useless to other applications. When an application and its data are isolated from other applications, we often say that they are in “silos.” In some of the most technology-savvy companies today it is not unusual to find a billing application that cannot ask a shipping application whether a delivery has been made. After many years of ballooning applications within an enterprise, enabling cross-silo communications was the emerging business challenge in the 1990’s and it continues today. The first generation solution to the silo problem was met with what we already regard as traditional Enterprise Application Integration approaches. In these approaches, an application communicates with another application through a bus or a hub that packages and translates information in one application according to specific rules provided by the receiving application. All of the packaging instructions for the receiving application are tied closely to the sending application, making each application ‘integration’ a one-to-one relationship, that is, a single application communicating with a single application. These relationships are described as ‘tightly coupled’.

Although cumbersome, these new approaches were a big step forward, allowing companies to leverage their investment in their most critical business applications, rather than rewriting the application or migrating it to a new technology. However, the approach presented a new set of problems revolving around the integration software costs, complexity of the effort, and inflexibility of the solutions. These three factors – cost, complexity, and solution inflexibility - have driven companies to choose the Enterprise Application Integration approach for only their most critical business requirements.

Today, application integration continues to be one of information technology’s most important challenges. Businesses spend millions of dollars on it every year. Web Services-based application integration offers the next big step in application-to-application communication. It attacks the three problems identified above – cost, complexity, and solution inflexibility – with a new model that is lower cost, easier to learn and deploy, and more adaptable to changing business needs. The result is a faster, easier application-to-application communication, allowing companies to connect many more applications within their company, and opening the door for better portals and Business-to-Business (B2B) solutions. Now many, rather than only the most critical, applications can talk to each other, presenting enormous opportunities for improved business performance.

SERVICE RE-USE

For years it has been known that reusing code, or parts of applications, rather than rewriting them for a new application increases productivity. One of the most powerful features of Web Services is the ability to re-use a service (business function application) many times rather than creating a new service for the same function over and over to meet the requirements of the receiving application. This is possible by isolating the packaging or delivery requirements of the service from the business function. Putting only the ‘business logic’ in each service and maintaining the packaging or delivery information outside of the service, allows the service to be used again and again, without change, whenever a receiving application requires that business logic. Service re-use can increase your organizations’ productivity, allowing your people to develop new services around new business functions rather than rewriting services to meet the requirements of new client applications.

BUSINESS FLEXIBILITY

The third area where Web Services shine is in providing increased business flexibility. Historically, business applications are rather rigid. Updating them with new functionality to meet changing business requirements has traditionally been expensive, complex, and time consuming. However, Web Services provides a substantial step forward, allowing you to build new Web Services based integration solutions quickly, at a reasonable cost, so you can adapt to changing or time-sensitive business requirements more easily. And, since it is easier to reuse services for multiple clients in the Web Service environment, you achieve additional business flexibility. As an example, you might want to build a dashboard to view orders, shipping, and billing information for your customers. Using a Web Services based integration, you could rather easily integrate information from your order-entry, shipping, and billing systems to build the new view. You could quickly build a 'Customer Dashboard' that provides a complete history of all the orders, shipments, and billing activities for a customer over time.

Corporate dashboards are becoming increasingly important. Often legal requirements drive the need for certain dashboards, particularly in company financials. Using Web Services can reduce the development time substantially. It is fundamental to the promise of Web Services that you can connect applications for internal use, portals, or B2B opportunities and efficiencies – quickly, at reasonable cost – that allow you to adapt to changing business needs.

Although Web Services are still quite new, the technology is being adopted at an amazingly quick pace. Once companies have some experience with the easier development cycle and the lower cost, they quickly move on, using Web Services for more complex projects. Within just a couple years, there has been a progression from internal Web Services integrations to portals (internal and external), to robust B2B implementations. As Web Services use expands, so does the need for additional standards. After all, these standards are the foundation for Web Services today – and tomorrow. Three dominant standards boards, OASIS, W3C, and WSI, lead this critical effort, insuring that as your usage expands, the standards will be in place to support your next set of Web Services initiatives.

Smart managers are cautious about any rapidly evolving technology. While there is a lot of hype about Web Services, there is also real benefit. They are being used by some of the world's largest companies to rapidly solve real problems with lower than expected costs. Every major business technology platform provider has adopted Web Services, including Microsoft, IBM, Sun, Oracle, BEA, etc. Their promise for the future makes them a smart investment in the present.

WEB SERVICES ROI

Web Services provide several essential business benefits that translate into a rapid ROI.

1. Open standards. Because Web Services use open standards, the cost point is substantially reduced. You do not have to use the more traditional and expensive EAI product suites to build successful solutions.
2. Ease of use. Open standards are easier to learn and to use than traditional methods. The faster learning curve translates into faster deployment, increasing your productivity and producing a rapid ROI.
3. Reuse capability. The services reuse capability reduces the amount of new services that need to be developed, again increasing your overall productivity.
4. Flexibility. Web Services based integration solutions are more flexible, decreasing the time and effort required to meet changing business needs. Because the packaging and delivery information is isolated and maintained independently from either the provider or consumer application, business service developers need only focus on the business logic and functionality. Services focus on the business function. Within your organization, the right skills can be focused on the right issues - one on business function, one on the communication and delivery function. This specialization ability makes it much easier to change and/or add services as your needs change.

KEYS TO WEB SERVICES SUCCESS

Web Services is not a technology that requires you dive in headfirst- your enterprise can adopt Web Services incrementally. For example, you can start with a departmental project, and gradually open your Web Services to other business units, and eventually business partners. With each step forward, you can validate the key Web Services technologies, and demonstrate the business value of the Web Services architecture. Over time, you will have developed a successful Web Services operation.

We suggest keeping these key principals in mind as you begin to build your Web Services Strategy:

- ***Web Services are enterprise applications.***

Web Services are applications that your business will depend on. Build and manage them with the same rigor you apply to any enterprise technology. Insist on security, quality assurance, and management of Web Services throughout their lifecycle.

- ***Web Services can be adopted incrementally.***

Unlike some technologies, you can usually start small with Web Services. As a first step, build a new Web Service, or expose an existing application as a Web Service. You can take on more ambitious Web Services initiatives as you gain experience.

- ***Building a Web Service is an architectural decision.***

Building a Web Service involves making lot of decisions about how your business will operate. Don't let your Web Services architecture evolve in an ad-hoc manner. Establish your long-term objectives and business principles for your Web Services at the start.

- ***Web Services are a "how," not a "why."***

You must have a business purpose for using Web Services. Before building your Web Services architecture, consider your business objectives, and plan for regular review to see whether your Web Services are bringing you closer to your business objectives.

- ***Web Services technologies are based on open standards.***

What makes a Web Service special is its XML-based components. Fortunately, the designs of these components are specified by independent standards bodies. Stick to the standards, and you will avoid strategic dead-ends.

- ***Web Services promote services reuse.***

Services reuse increases your organization's productivity, allowing you to focus on new business functions rather than re-hashing existing ones. Keeping this in mind as you move through the process is important since it will yield the most cost effective, and operationally efficient models.

- ***Web Services will change the way your people work.***

Your initial projects may not have a strong impact on the roles and responsibilities of your employees. However when you start using Web Services to work more closely with your customers and partners, your business will run differently. Prepare your people for the change.

- ***Inform your organization about Web Services at every level.***

Your technologists need to stay abreast of the major developments in Web Services standards and technologies. At the same time, your business managers should closely follow the changing role of Web Services in your company and your industry.

CONTACT

Whether you're looking for significant online revenue, increased traffic and brand recognition, business process streamlining, or all of the above, we can help. Our open source technology and award-winning designers and engineers will give your eBusiness the power of content, commerce, and control, with a best-of-breed solution that will give your customers a reason to come in- and come back.

Give us a call today. We'd like to make you our next success story.

San Diego Media
9820 Willow Creek Road, Suite 450
San Diego, California 92131

Toll Free: 800.808.8696
Local: 858.536.5505
Fax: 858.536.5504

Email: sales@sandiegomedia.com

