

Web Services: Beyond the Hype

Steven J. Vaughan-Nichols

For the past six months, the technology media has been full of articles about Web services (WS), focusing largely on Microsoft's .NET initiative. Proponents call it "the next big thing" in computing, and although WS vendors have not delivered their systems yet, there is already a *Web Services Journal*.

WS comprises a set of platform-neutral technologies designed to ease the delivery of network services over intranets and the Internet.

Cross-platform capabilities are one of WS's key attractions because interoperability has been a dream of the distributed-computing community for years.

Barry Morris, CEO of Iona Technologies, an object-oriented distributed-computing company, added, "The most important aspect of Web services is that it's a standards-based,



service-oriented architecture that is supported by every major software and hardware company in the world."

Indeed, Microsoft isn't the only company promising WS. Major vendors BEA Systems, Hewlett-Packard, IBM, Oracle, and Sun Microsystems are working on competing Java-based Web services. There are even two open source WS projects: Mono and DotGnu. The "Web Services Web Sites" sidebar lists URLs.

Philip DesAutels, Microsoft's product manager of XML Web services strategy, said WS will prove useful in many ways.

On the other hand, analyst David Smith with market-research firm Gartner Inc. said Web services are "exciting, but they're no big deal," because they represent just another way to deliver network services.

Added CEO Avery Lyford of Linux-Care, which develops Linux-based applications, "Show me a customer need for these new services, and I'd be more excited."

Currently, therefore, it isn't at all certain whether WS will become an important new computing approach or just a niche technology.

WHAT ARE WEB SERVICES?

WS would, in essence, integrate PCs, other devices, databases, and networks into one virtual computing fabric that users could work with via browsers.

The services themselves would run on Web-based servers, not PCs, thereby moving functions from the desktop to the Internet. Users could work with the services over any WS-enabled machine with Internet access, including handheld devices. Web services would thus change the Internet into a computing platform, rather than a medium in which users primarily just view and download content.

This would also move data and applications from the desktop to a WS provider's servers, a potential source of user concern about security, privacy, and accessibility.

Application servers will be a critical part of Web services because they typically handle the complex, transaction-based-application operations between users and an organization's back-end business programs or databases.

Some industry observers say WS is not really a new concept and reflects much of the network-computing concept that was popular several years ago.

Gartner's Smith said that WS is basically a loosely coupled remote procedure call that would replace today's

Web Services Web Sites

- **BEA WebLogic Server 6.1:** <http://www.bea.com/products/weblogic/server/index.shtml>
- **DotGnu project:** <http://www.dotgnu.org/>
- **HP Web Services Platform:** http://www.bluestone.com/products/hp_web_services/default.htm
- **IBM Web Services Toolkit:** <http://www.alphaworks.ibm.com/tech/webservicestoolkit>
- **Microsoft .NET:** <http://www.microsoft.com/net/>
- **Mono project:** <http://www.go-mono.net>
- **Oracle9iAS Web services application server:** <http://otn.oracle.com/tech/webservices/content.html>
- **Sun ONE:** <http://www.sun.com/sunone/>
- **Web Services Journal:** <http://www.sys-con.com/webservices/>

tightly coupled RPCs, which require application- and protocol-specific application programming interface (API) connections. WS uses XML, rather than C or C++, to call procedures.

Still other experts say WS is just a type of middleware-based API, with XML providing the front end to Java 2 Platform, Enterprise Edition (J2EE) or .NET application servers. Like middleware, WS would link the application server and client programs.

STANDARDS: THE HEART OF WS

WS enables interoperability via a set of open standards, which distinguishes it from previous network services such as Corba's Internet Inter-ORB Protocol (IIOP).

XML is the most important Web-services standard and is the basis for the other WS standards. As a metalanguage, XML lets a set of users define its own markup tags. The tags provide information about the data in a document to users on most platforms. This permits cross-platform communications and also lets organizations integrate different data types within their systems.

Systems would use SOAP (simple object access protocol) to run WS applications. SOAP lets a program working in one operating system communicate with a program working in another by using HTTP and XML as information-exchange mechanisms. SOAP specifies how to encode an HTTP header and an XML file to achieve this interoperability. Thus, an operating system or browser will need only SOAP compatibility to work with any Web service.

The XML-based Web Services Description Language describes the online services a business offers. WSDL also helps users access a Web service by providing information such as the nature of its interface.

Finally, businesses use WSDL to list their Web services on the Internet in an XML-based registry based on the UDDI (universal description, discovery, and integration) protocol. UDDI lets companies find publicly available

Web services on the Internet or corporate networks, as Figure 1 shows.

In essence, WS provides developers with a widely applicable API. Client-server systems use hard-coded interfaces and protocols between applications, noted Rick Caccia, director of product management for KnowNow, a WS and system-integration software company. This requires users to work with proprietary standards for each client-server software package.

Caccia said technologies based on Corba's object request brokers (ORBs) are a bit more flexible because they move the application coupling to a higher level. ORBs let programs treat blocks of code as objects without worrying about interior details.

However, explained Dan Gisolfi, solutions architect for IBM's jStart emerging-technologies team, "Vendors compete on ORB implementations and thus no [business] motivation exists ... to achieve [full] interoperability."

WS makes the process more abstract than ORBs by delivering an entire external service without users having to worry about moving between internal code blocks. The overall WS process would depend on several key elements, as Figure 2 shows.

WS FUNCTIONALITY

According to Annraí O'Toole, chair of WS-vendor Cape Clear Software, WS would permit the increased integration of online businesses. Businesses could thus use WS in many ways.

For example, software vendors could use WS to sell their applications over the Internet on a per-use or sub-

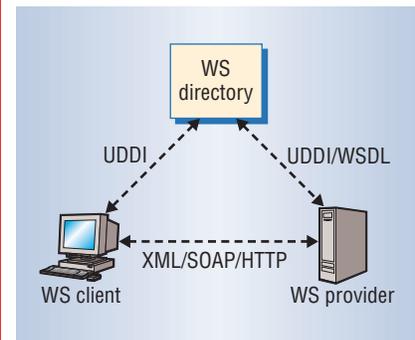


Figure 1. A Web services provider uses the universal description, discovery, and integration protocol to update the UDDI-based WS directory about the availability of its services. The provider sends the information encoded in WSDL. A client then sends a request for a service to the directory. The directory tells the client about the service's availability, and the client and provider interact via HTTP, SOAP, and XML.

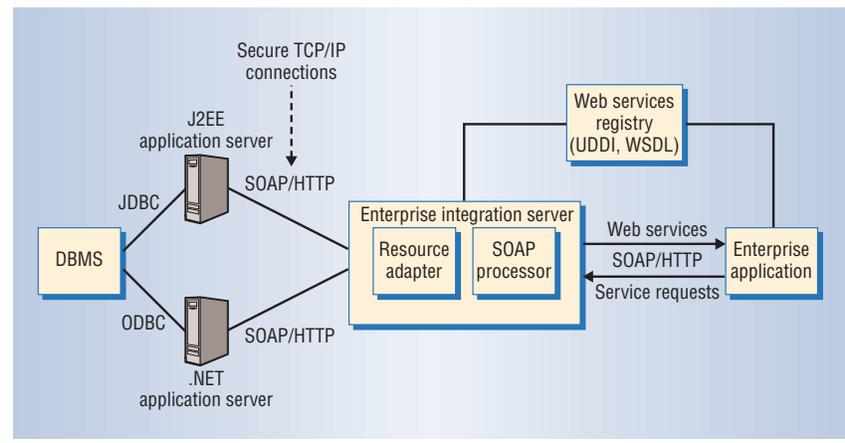


Figure 2. In a model Web-services (WS) process, an enterprise application requests a specific service, after determining its availability via the Web services registry, from a WS application server based on either Microsoft's .NET or J2EE. The resource adapter makes the connection between WS requests and the application server and also translates requests into specific network system calls. The processor translates SOAP into the appropriate requests for legacy servers and DBMSs that don't understand the protocol.

scription basis, rather than as a one-time purchase. This could change the face of the software industry.

According to KnowNow's Caccia, WS could also provide data updates for programs and permit data exchange among applications. He said many desktop applications have Web interfaces and use HTTP. "Someone is going to want to wire some of those together to exchange data. Web services can play a role by making that happen easily." He explained that WS's interface would permit this while hiding the complexities of each service's APIs and RPCs.

Meanwhile, companies could use WS to access software for use as components to build their own applications and services, such as a customer-service program. Component software could provide functionality that is general, such as storage, or industry specific, such as production scheduling.

Individuals could also access Web-based personal services—such as address books, appointment schedules, or travel applications. "If access to these services moves to desktop applications, as Microsoft proposes with some of its ... offerings," said KnowNow Chief Architect Steve Dossick, "it is more likely that consumers will use Web services directly."

Proponents say companies could also use WS as a platform on which to integrate their existing applications. According to Caccia, "It is more necessary than ever to integrate internal and external systems in a simple fashion, and Web services may be very helpful in doing so."

Developers would be able to connect network applications, like databases and end-user programs, by using WS's near-universal interface rather than writing DBMS- and application-specific connections.

According to senior analyst David Schatsky with Jupiter Media Metrix, an Internet research firm, a recent WS survey found that 60 percent of responding CEOs plan to use Web services internally, for application integration and data exchange, this year.

MICROSOFT'S .NET

Microsoft's .NET is the most high profile and well developed of the WS initiatives. Anders Hejilberg, Microsoft distinguished engineer, said the company started work on .NET when it began the Windows Distributed Internet Architecture project in 1997.

Companies could use WS to access software for use as components to build their own applications and services, such as a customer-service program.

.NET's use of XML-based standards makes it platform independent. In addition, .NET consists of numerous key elements that make it language independent.

For example, the Common Language Specification is a set of rules intended to promote language interoperability. And the Common Language Runtime multilanguage environment uses a just-in-time compiler to let code written in any of a number of languages, such as Cobol and C#, deliver WS via XML.

".NET is an attempt to integrate all popular programming languages in one runtime and development system," said UserLand Software CEO Dave Winer, one of SOAP's creators. The initiative's key Web-development tool will be Visual Studio .NET, currently in beta and planned for release in the near future.

.NET My Services (formerly called HailStorm) represents a set of XML-based services that users can call on to standardize and simplify e-commerce and other Web activities. For example, the .NET Profile service includes information such as user name and address, and .NET Wallet includes preferred payment instruments. Microsoft has also added to .NET additional services such as the Passport user-authentication system.

JAVA-BASED WS INITIATIVES

Like .NET, the Java-oriented WS initiatives are based on XML, SOAP, WSDL, and UDDI. However, they use J2EE, rather than .NET services, for their core application servers. J2EE is a platform-independent, Java-centric environment for developing and deploying Web-based enterprise applications online. Programs are developed in Java and delivered by J2EE application servers. This could encourage Java programmers to deliver network services as Web services.

To help with this effort, Sun and its partners are building full support for the WS standards into J2EE's next version, due by early next year. Sun has also started the Liberty Alliance to create an alternative to Microsoft's Passport user-authentication system.

A key difference between the J2EE initiatives is that Sun ONE (Open Net Environment) attempts to provide developers with an almost-universal WS development environment. BEA Systems, HP, IBM, and Oracle, on the other hand, are developing WS infrastructures that work best with their own products or those of partners.

Sun ONE

Given Sun's role as developer and caretaker of Java, it is not surprising that Sun ONE hopes to be the leader among the Java-based WS initiatives. In essence, Sun ONE adds XML to Java-based network services. Sun has attempted to make the approach universal by designing it to work with virtually any J2EE implementation or database.

Sun, which only began its WS efforts in February 2001, is lagging behind Microsoft's .NET, said Alan Zeichick, principal analyst for Camden Associates, a media-technology-research firm. Sun expects to release its full Sun ONE package by the middle of this year.

Sun is trying to catch up with its JAX (Java API for XML) technologies. Peter Kacande, a Sun senior product manager, said JAX "is an all-in-one download of Java technologies for XML."

JAX has several elements, including

- JAX-RPC, which lets developers build Web applications and services that incorporate XML-based RPCs via SOAP; and
- JAXP (Java API for XML processing), which would provide a standard way to integrate any XML-compliant parser with a Java-based application, thereby letting systems read, manipulate, and generate XML documents via Java APIs.

Sun ONE will work with the company's iPlanet application server and Forte development environment. Sun ONE will also provide Java-based software-development tools.

Other Java-based WS initiatives

Other companies' Java-based WS efforts, scheduled for release this year, are very similar. The key difference is that each uses its sponsoring company's own J2EE implementation and works best with its own DBMSs.

BEA. Already a leading application-server vendor, BEA is first out of the J2EE-enabled WS gate with its BEA WebLogic Server 6.1.

HP. The HP Web Services Platform initiative began with the open source e-Speak project, which originally was designed to deliver Web services using HP-created technologies. However, the rise of XML for delivering WS made e-Speak less important. Today, HP is focusing on delivering WS using XML-based specifications and the recently acquired Bluestone J2EE engine as the core application server.

IBM. IBM is adding WS to its WebSphere application-server suite via the Web Services Toolkit, a software development kit that includes a runtime environment, architectural blueprint, tools, components, a demo, and examples to help in designing and executing WS applications.

Oracle. Oracle says the company has its own WS offerings, though not under a bannered initiative. Oracle

says that building WS capabilities is more of an evolutionary process in software. Oracle can use its market-leading database technology in conjunction with its WS initiatives.

To help provide WS, the company is developing its Oracle9iAS Web Services application server. The company is also working on Jdeveloper, a development kit that can be seen as a Java-based counterpart to Microsoft's Visual Studio.NET.

WS could help users save money by making it easier for them to develop and integrate their network applications.

CONCERNS

Industry observers, such as Steve Vinoski, Iona's chief architect and vice president of platform technologies, are worried about WS performance. One main reason is that XML, unlike binary-based IIOP, is text-based and thus entails more data for systems to process. XML therefore runs more slowly over HTTP. Adding a security protocol like Secure Socket Layer (SSL) would slow performance even more. This could make WS impractical for activities over low-bandwidth connections such as dial-up modems.

Because Web services have no built-in security model, they must rely on SSL, virtual private networks, or other external measures. In general, said Scott Dietzen, BEA Systems' chief technology officer, public-key infrastructure and SSL will provide sufficient security.

Nonetheless, vendors such as Netegrity, Oblix, and OpenNetwork Technologies are working on products to manage WS security via, for example, authentication and encryption.

In addition, the Organization for the Advancement of Structured Information Standards is developing the Security Assertion Markup Language, a vendor-neutral format for WS-transaction authentication. However, OASIS

doesn't plan to approve SAML as a draft standard until the second half of this year.

WS: BUST OR BOOM?

With the exception of current J2EE application servers, no vendor is shipping WS tools. At most, the tools are in late beta.

While there are some publicly available WS applications (for a current list, see <http://www.xmethods.net>), most are relatively trivial, such as zip code and MP3 finders.

That may change quickly, though. Gartner predicts that 75 percent of companies with more than \$100 million in annual revenue will use WS by the middle of this year and that the technology will reach mainstream users by 2004.

Jupiter's Schatsky said WS could succeed because it will help users save money by making it easier for them to develop and integrate their network applications. In the process, .NET and J2EE vendors will hope to make money from sales of and support for their application servers, as well as application-integration services.

As for the marketplace battle, KnowNow's Dossick said, "Just as [Java] and [Microsoft] camps coexist today, they will continue to do so. While Microsoft has articulated an excellent packaged vision for Web-services creation and hosting that includes development tools and server support, it is likely that the J2EE camps will do so as well."

Nonetheless, KnowNow's Caccia was cautious about WS's future. "Right now we are in the easy, simple, pie-in-the-sky phase," he said. "The next year or two will bring out many hard obstacles." ■

Steven J. Vaughan-Nichols is a freelance technology writer based in Arden, North Carolina. Contact him at sjvn@vna1.com.

Editor: Lee Garber, Computer, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org