SYS-CON
NOW INCORPORATING XML JOURNAL

SOA Web Services JOURNAL

April 2006 Volume 6 Issue 4

Adding Reliability to Occasionally Connected Computing in Mobile Devices

Leveraging Web Services Reliable Messaging for persistence

pg.26

Beware of Shortcuts on the Road to a Service-Oriented Architecture

How to achieve effective SOA implementation

pg.36

Using Data Services to Build Functional Services

Data management infrastructure cuts costs and accelerates development

pg.50

SOA Governance: GAINING FLEXIBILITY & Retaining Control

Avoiding chaos

pg.10

RETAILERS PLEASE DISPLAY
UNTIL JUNE 30, 2006
$6.99US $7.99CAN
04
71486 03420 9

Altova® XMLSpy® 2006 – The industry standard XML development environment.

ALTOVA® xmlspy® 2006

Bring your development plans to light

Sneak a peek at XMLSpy® 2006, and see how vital it is to master XML.

New in XMLSpy 2006:

- Schema-aware XSLT 2.0 support
- Schema-aware XQuery support
- Updated platform integration for Microsoft® Visual Studio® .NET 2005
- Updated platform integration for Eclipse 3.1

Altova® XMLSpy, the industry standard XML development environment, is indispensable for modeling, editing, debugging and transforming all XML-related technologies. Illuminate your strategy with advanced standards compliance, extended platform integration, and enlightened usability aides. Use XMLSpy to structure XML Schemas and devise XML documents, then automatically generate runtime code from schemas in multiple programming languages. Become a markup mastermind!

Download XMLSpy® 2006 today: www.altova.com

ALTOVA®

Also available in the Altova XML Suite.

MAKE YOUR SOA SECURITY MOVES WISELY...

XWALL
WEB SERVICES FIREWALL

XRAY
WEB SERVICES DIAGNOSTICS

VULCON
VULNERABILITY CONTAINMENT SERVICE

SENTRY
SOA SECURITY GATEWAY

PUTTING TOGETHER THE PIECES FOR THE WORLD'S MOST DEMANDING SOA SECURITY SYSTEMS

FORUM SYSTEMS ENTERPRISE SOA SECURITY SOLUTIONS:

- TRUSTED SOA MIDDLEWARE
- WEB SERVICES SECURITY
- XML ACCELERATION

WWW.FORUMSYSTEMS.COM

FORUMSYSTEMS

THE LEADER IN WEB SERVICES & SOA SECURITY

SOA Web Services Journal

April 2006 Volume 6 Issue 4

XML Journal Inside WSJ

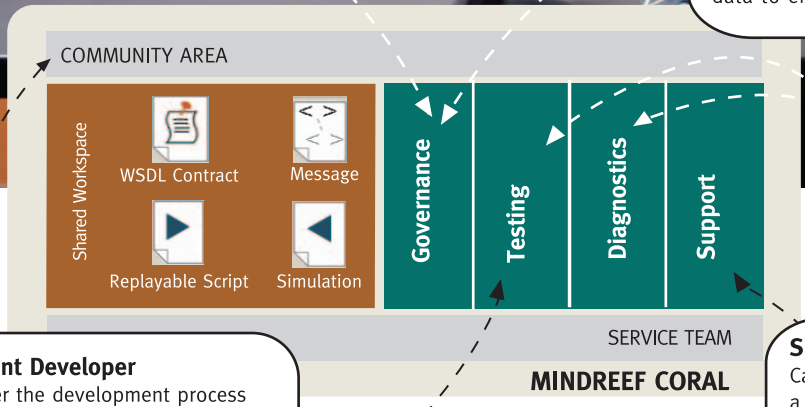
Mindreef® introduces

Mindreef Coral™

A collaborative platform for Web services/SOA teams

Corporate Architect
Codify corporate standards, best practices, and governance policy around WSDL and schema

Project Architect
Leverage rules from corporate architects while creating shared workspaces that add documentation, example scripts, and simulation data to enhance the WSDL contract



Service Developer
Use shared workspaces from Project Architects to fully understand requirements. Create unit tests automatically during development

Client Developer
Enter the development process earlier with Mindreef Coral's auto documentation and simulation features

Support Staff
Capture all associated problem artifacts into a Mindreef Shared Workspace™*, allowing any team member to reproduce a problem at the push of a button

Testing
Leverage Architect's example scripts and Developer's unit tests to quickly and easily create robust regression tests

- **Testing**
- **Diagnostics**
- **Governance**
- **Support**

The Mindreef Coral platform provides collaborative tools for the XML software layer that increase productivity and quality throughout the entire SOA lifecycle.

Our lifecycle approach—from well-adopted best practices early in the process to efficient problem resolution post-launch—drives down the cost of developing and maintaining Web services, leading to more efficient, cost-effective SOA. We designed Mindreef Coral to leverage your most valuable assets—your service team members—to achieve sustainable business results.

Pay as you grow pricing—from your first Web service project to large SOA initiatives. Mindreef Coral starts at just $999 per year for a server supporting two concurrent users. Additional licenses are just $499.

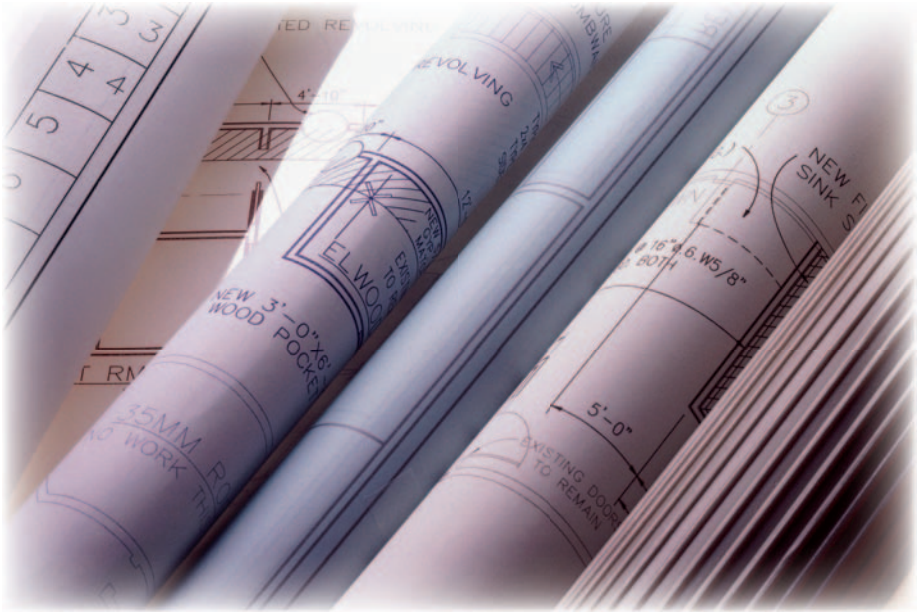
To download a FREE trial, visit www.mindreef.com, or to speak with a Mindreef representative call 603-465-2204 X510.

Mindreef®

PEOPLE MAKE SOA WORK™

* Patent pending.

BPEL is the SQL of SOA



Get started building next-generation SOA applications with the leading vendor of BPEL technologies

Download BPEL tooling & server software today

activebpel.org/soa

activeBPEL

BPEL consulting, certification and training.
BPEL design tools, servers and source code for Eclipse, Apache Tomcat, JBoss, WebSphere, WebLogic, BizTalk and Microsoft .NET.

SOA Web Services JOURNAL

INTERNATIONAL ADVISORY BOARD
Andrew Astor, David Chappell, Graham Glass, Tyson Hartman, Paul Lipton, Anne Thomas Manes, Norbert Mikula, George Paolini, James Phillips, Simon Phipps, Mark Potts, Martin Wolf

TECHNICAL ADVISORY BOARD
JP Morgenthal, Andy Roberts, Michael A. Sick, Simeon Simeonov

EDITORIAL
EDITOR-IN-CHIEF
Sean Rhody sean@sys-con.com
XML EDITOR
Hitesh Seth
INDUSTRY EDITOR
Norbert Mikula norbert@sys-con.com
PRODUCT REVIEW EDITOR
Brian Barbash bbarbash@sys-con.com
.NET EDITOR
Dave Rader davidr@fusiontech.com
SECURITY EDITOR
Michael Mosher wsjsecurity@sys-con.com
RESEARCH EDITOR
Bahadir Karuv, Ph.D Bahadir@sys-con.com
TECHNICAL EDITORS
Andrew Astor andy@enterprisedb.com
David Chappell chappell@sonicsoftware.com
Anne Thomas Manes anne@manes.net
Mike Sick msick@sys-con.com
Michael Wacey mwacey@csc.com

INTERNATIONAL TECHNICAL EDITOR
Ajit Sagar ajitsagar@sys-con.com

EXECUTIVE EDITOR
Nancy Valentine nancy@sys-con.com

ONLINE EDITOR
Roger Strukhoff roger@sys-con.com

PRODUCTION
LEAD DESIGNER
Andrea Boden andrea@sys-con.com
ART DIRECTOR
Alex Botero alex@sys-con.com
ASSOCIATE ART DIRECTORS
Abraham Addo abraham@sys-con.com
Louis F. Cuffari louis@sys-con.com
Tami Beatty tami @sys-con.com

CONTRIBUTORS TO THIS ISSUE
Abhishek Malay Chatterjee, Terance Dias, Aaron Flint, Dan Hynes, Matjaz B. Juric, Sean Kline, Geo Philips Kuravakal, Francois Lascelles, Andrew Lawlor, David Linthicum, Roy Mitchell, Srinivas Padmanabhuni, James Pasley, Varun Poddar, Sean Rhody, Vivek Singhal, Doug Todd, Ajit Sagar, Brian Wilson

EDITORIAL OFFICES
SYS-CON MEDIA
135 CHESTNUT RIDGE ROAD, MONTVALE, NJ 07645
TELEPHONE: 201 802-3000 FAX: 201 782-9637
WEB SERVICES JOURNAL (ISSN# 1535-6906)
Is published monthly (12 times a year)
By SYS-CON Publications, Inc.
Periodicals postage pending

Montvale, NJ 07645 and additional mailing offices
POSTMASTER: Send address changes to:
WEB SERVICES JOURNAL, SYS-CON Publications, Inc.
135 Chestnut Ridge Road, Montvale, NJ 07645

The GUI I Want

WRITTEN BY
SEAN RHODY

Recently I happened upon a blog site that had a discussion of one of my previous articles, one in which I proposed a need for a new "browser." The discussion was very interesting, but it was obvious to me I had not been clear about what I was asking for. Some of the folks on the board suggested that we could do what I wanted using Swing. Others realized that that required an installed or installable software distribution, and fixed on things like the Ruby language as a different approach. While all of these ideas had some merit, they didn't capture the true essence of what I think we need.

The browser, which is primarily IE and Firefox, represents widely successful and freely distributed software that render a platform-independent language (HTML) into information displayed according to the GUI elements of the underlying operating system.

What the browser is not good at is serving as the basis for a real application. HTML has only limited control over the appearance of the browser; it's a synchronous request response model and it was really geared toward presenting content, not controlling the application. I applaud the heroic measures that have been taken to fit the round peg of applications into the square hole of content, but at best it works poorly and inconsistently.

I think we need a new paradigm for application delivery. Obviously, we need to be able to send our application over the Internet. First, we need an asynchronous, bi-directional secured protocol to provide the application to the user. What's sent over that protocol should be presentation elements, minimal presentation logic (such as phone number format validations), and the data that populates the view the user sees.

Unlike HTML, the application definitions should not have to be sent repeatedly in order to update the display when data is changed. Yes, you can do some of this with HTML and AJAX right now, and that's a good start; it's just not good enough. We need the description of the application to be independent of the implementation of the application so that it isn't just Java programmers, or PERL programmers, or .NET programmers who can develop an application. Yes, that's a break from the current paradigm, but that's what we need.

There was talk a while back of Microsoft taking an approach to the Windows GUI that had the application interface sent via an XML-based language. This was a great idea, and I hope it does eventually make it into the mix. An application sent over the Internet should behave and appear no different from an application that exists natively on a user's machine.

To do this, either the OS shell or another application (aka – the new browser) should be able to receive the application definition and render the application in a manner consistent with the look and feel of the underlying computer operating system. On a Mac, it should look like a Mac. On Windows, it should look like Windows. On Linux, it should look like Linux (pick your shell flavor). X-Windows was a concept that had a lot of this built into it, but it's too bulky, has too much of a least common denominator approach, and doesn't travel well over the Internet.

Obviously we'd need tools for developing the GUI, and tying the data elements and business logic to the presentation logic. Fortunately, defining a GUI is something we have 20 years of experience in, so the tools can be adapted rather than created from scratch. We'd also probably need a translation layer that would allow existing applications to be repurposed.

Finally, this technology needs to be vendor, platform, language, and implementation independent. To be successful, it needs a standards body champion, vendor buy in, and user participation. That's a lot to ask for, but we've seen it happen in the past. To all the guys on that blog, thanks for reading and all the comments.

About the Author

Sean Rhody is the editor-in-chief of ***Web Services Journal***. He is a respected industry expert and a consultant with a leading consulting services company.

sean@sys-con.com

SOA Web Services Journal

PRESIDENT AND CEO
Fuat Kircaali fuat@sys-con.com
VP, BUSINESS DEVELOPMENT
Grisha Davida grisha@sys-con.com
GROUP PUBLISHER
Jeremy Geelan jeremy@sys-con.com

ADVERTISING
SENIOR VP, SALES & MARKETING
Carmen Gonzalez carmen@sys-con.com
VP, SALES & MARKETING
Miles Silverman miles@sys-con.com
ADVERTISING DIRECTOR
Robyn Forma robyn@sys-con.com
NATIONAL SALES & MARKETING MANAGER
Dennis Leavey dennis@sys-con.com
ADVERTISING MANAGER
Megan Mussa megan@sys-con.com
ASSOCIATE SALES MANAGERS
Kerry Mealia kerry@sys-con.com

SYS-CON EVENTS
PRESIDENT, SYS-CON EVENTS
Grisha Davida grisha@sys-con.com
NATIONAL SALES MANAGER
Jim Hanchrow jimh@sys-con.com

CUSTOMER RELATIONS/JDJ STORE
CIRCULATION SERVICE COORDINATORS
Edna Earle Russell edna@sys-con.com
Linda Lipton linda@sys-con.com

sys-con.com
CONSULTANT, INFORMATION SYSTEMS
Robert Diamond robert@sys-con.com
WEB DESIGNERS
Stephen Kilmurray stephen@sys-con.com
Wayne Uffleman wayne@sys-con.com

ACCOUNTING
FINANCIAL ANALYST
Joan LaRose joan@sys-con.com
ACCOUNTS PAYABLE
Betty White betty@sys-con.com
ACCOUNTS RECEIVABLE
Gail Naples gailn@sys-con.com

SUBSCRIPTIONS
SUBSCRIBE@SYS-CON.COM
1-201-802-3012
1-888-303-5282
For subscriptions and requests for bulk orders, please send your letters to Subscription Department
Cover Price: $6.99/issue
Domestic: $69.99/yr (12 issues)
Canada/Mexico: $89.99/yr
All other countries: $99.99/yr
(U.S. Banks or Money Orders)

Worldwide Newsstand Distribution
Curtis Circulation Company, New Milford, NJ
Newsstand Distribution Consultant:
Brian J. Gregory / Gregory Associates / W.R.D.S.
732 607-9941 - BJGAssociates@cs.com

For list rental information:
Kevin Collopy: 845 731-2684,
kevin.collopy@edithroman.com;
Frank Cipolla: 845 731-3832,
frank.cipolla@epostdirect.com

SYS-CON MEDIA

Opening SOA

WRITTEN BY
AJIT SAGAR

The feasibility of adopting new technology in the computing world is governed in large part, as it is in other paradigms, by the cost-benefit-risk equation. In the world of electronic computing, whenever a cost-benefit analysis is done, the benefit/risk of adopting open source solutions almost always comes up for discussion. Open source solutions offer very feasible cost-effective alternatives to commercial vendor products for realizing an SOA.

Consider a typical SOA stack. Although SOA is not a totally new concept, a service-oriented architecture leveraging the current technology landscape is usually comprised of a service repository/registry, a service discovery service, and a transportation protocol. Some components of the infrastructure to support these dimensions of SOA include a messaging bus for supporting reliable messaging, an Enterprise Service Bus for service orchestration and transformation, and an underlying application server, which provides the foundation for building service components and hosting the services for applications to consume. At the bottom of the stack is, of course, the operating system.

The open source community offers capabilities that address all the layers of the stack. When adopting an SOA strategy, organizations should evaluate the offerings from open source software (OSS) as well as commercial products. All the prominent vendors in the SOA arena, such as IBM, BEA, Sun, Oracle, and Microsoft, as well as newer players such as Cape Clear and Fiorano, have offerings for SOA enablement. Open source offers the advantages of low-cost, vendor-neutral, standards-based solutions.

Recently the open source community has announced several products that address the needs for implementing an SOA. At the bottom of the stack, Linux offers a commercial grade operating system. JBoss, JonAS, and the Apache server offer the core application server capabilities and have been in the industry long enough to provide mature, reliable alternatives to commercial app servers. On the messaging front, ActiveMQ, JORAM, and JBossMQ are some of the products that provide asynchronous messaging capability. Recently the open source community also released products that address the ESM space such as Mule, celtix, and open-esb. Axis, Synapse, and Sandesha are some of the other products that focus on Web service enablement in an SOA. There are several other products that are either in the process of being released or are being designed to help implement service-oriented architectures. Optaros has an interesting white paper (http://www.optaros.com/wp/wp_2_soa_oss.shtml) that provides some possible scenarios for leveraging OSS in SOA stacks.

The open source community is not completely divorced from commercial vendors. In fact, most of the open source projects are based on heavy contributions from traditional vendors such as IBM, BEA, and Oracle. IBM has invested heavily in Linux, Eclipse, Apache, etc. For example, late last year, IBM launched the Community Edition of its WebSphere Application server, which is an open source, license-free version. This project was in the works for a while and is the result of the GlueCode acquisition by IBM. WebSphere CE is the new incarnation of GlueCode's Geronimo app server. Similarly, BEA has invested heavily in the Apache Beehive projects, which leverage open source application servers.

While the products offered from the open source community for facilitating service orientation are definitely gaining ground, large organization are still wary of betting the farm on open source alternatives. However, the open source community definitely has a larger presence in service enablement than previous initiatives. This is due to several factors. The tools to support SOA are fairly new and, therefore, open source alternatives are as viable as commercial offerings. At the same time, the pressures on business to cut costs are forcing organizations to look for more cost-effective options. Meanwhile, the open source community has gained substantial trust in the industry through proven technologies and products, as well as due to the backing of vendors who have based large components of their commercial offerings on open source products.

About the Author

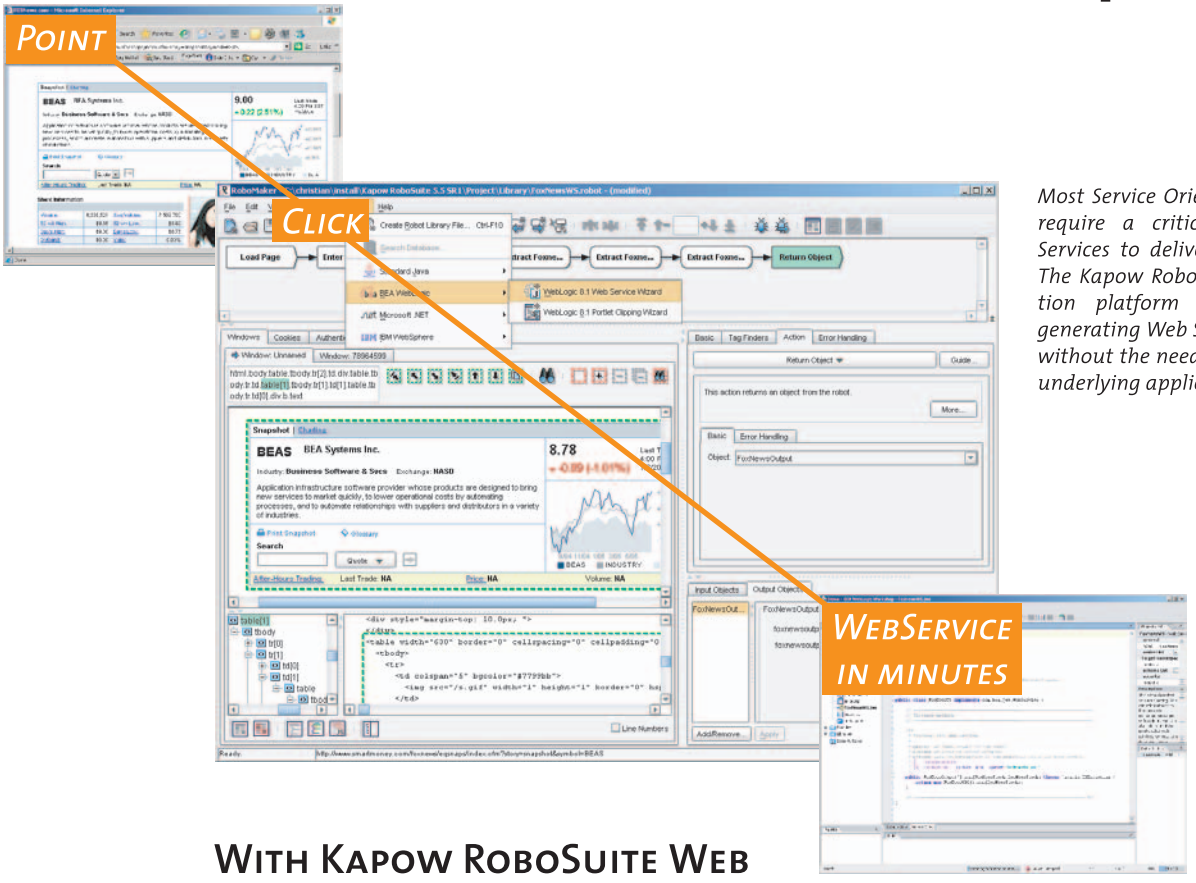
Ajit Sagar is a principal architect with Infosys Technologies, Ltd., a global consulting and IT services company. Ajit has been working with Java since 1997, and has more than 15 years experience in the IT industry. During this tenure, he has been a programmer, lead architect, director of engineering, and product manager for companies from 15 to 25,000 people in size. Ajit has served as *JDJ*'s J2EE editor, was the founding editor of *XML Journal*, and has been a frequent speaker at SYS-CON's Web Services Edge series of conferences. He has published more than 100 articles.

ajitsagar@sys-con.com

Leverage existing IT to generate Web Services

[in minutes]

For free online demonstration, Gartner "Cool Vendor" release, and software trial: www.kapowtech.com/wsjprint



Most Service Oriented Architectures require a critical mass of Web Services to deliver tangible results. The Kapow RoboSuite Web Integration platform supports this by generating Web Services in minutes without the need to re-program the underlying application.

With Kapow RoboSuite Web Integration Platform you get

- *Fast generation of Web Services of any web application – simple or complex – with visual point and click*
- *Non-intrusive solution using the browser front-end to access any web-application*
- *Wizards enable quick, low cost deployment in SOA frameworks for production-ready SOA*

kapowtech.com

kapow TECHNOLOGIES

Kapow Technologies is a leader in Web Integration – a new integration paradigm using the broadly available web front-end. The Kapow RoboSuite platform uniquely enables flexible and fast integration of content, data and applications from any source available through a browser into portals, content management systems, applications, databases or web services.

SOA Governance: GAINING FLEXIBILITY and Retaining Control

Avoiding chaos

SOA offers significant advantages, but it puts additional demands on visibility, control, and overall governance. Although enterprise SOA initiatives are typically deployed incrementally, to gain long-term value and ensure quality and consistency, you must address governance issues early in the implementation process.

WRITTEN BY
DAN HYNES & SEAN KLINE

The goal of this article is to help you understand the role of, and the requirements for, SOA governance. After reading this, you'll be better prepared to ask the right questions and define and implement an SOA governance strategy.

What Is SOA Governance?

In any discussion of SOA, the term "SOA governance" will invariably come up. Ask what it means and you'll most likely get several different answers. The definition of governance and the requirements it dictates, like SOA itself, is an evolving concept.

In essence, SOA governance may be viewed as *management architecture*: a framework that blends the flexibility of SOA with the control and predictability of a traditional IT architecture.

Why SOA Governance Matters

SOA creates an inherently dynamic and heterogeneous environment. It introduces many independent and self-contained moving parts — components that are typically widely reused across the enterprise and are a vital part of mission-critical business processes. Governance is no longer optional — it's imperative. SOA has the potential to introduce risk and, without proper governance, can disrupt business processes and create significant inefficiencies.

How can you manage changes to business services to lessen the impact on consumers? How can the consumer be sure the service is of high quality? What happens if a subcomponent of a composite service is retired? How can you be sure a new service is compliant with IT, business, and regulatory policies? How can you insure predictable uptime of a service?

These are the kinds of questions that SOA should raise in an organization. SOA brings new challenges with respect to assurances for service quality, consistency, performance, and predictability. But the greatest challenge facing SOA is engendering trust between consumers and service providers.

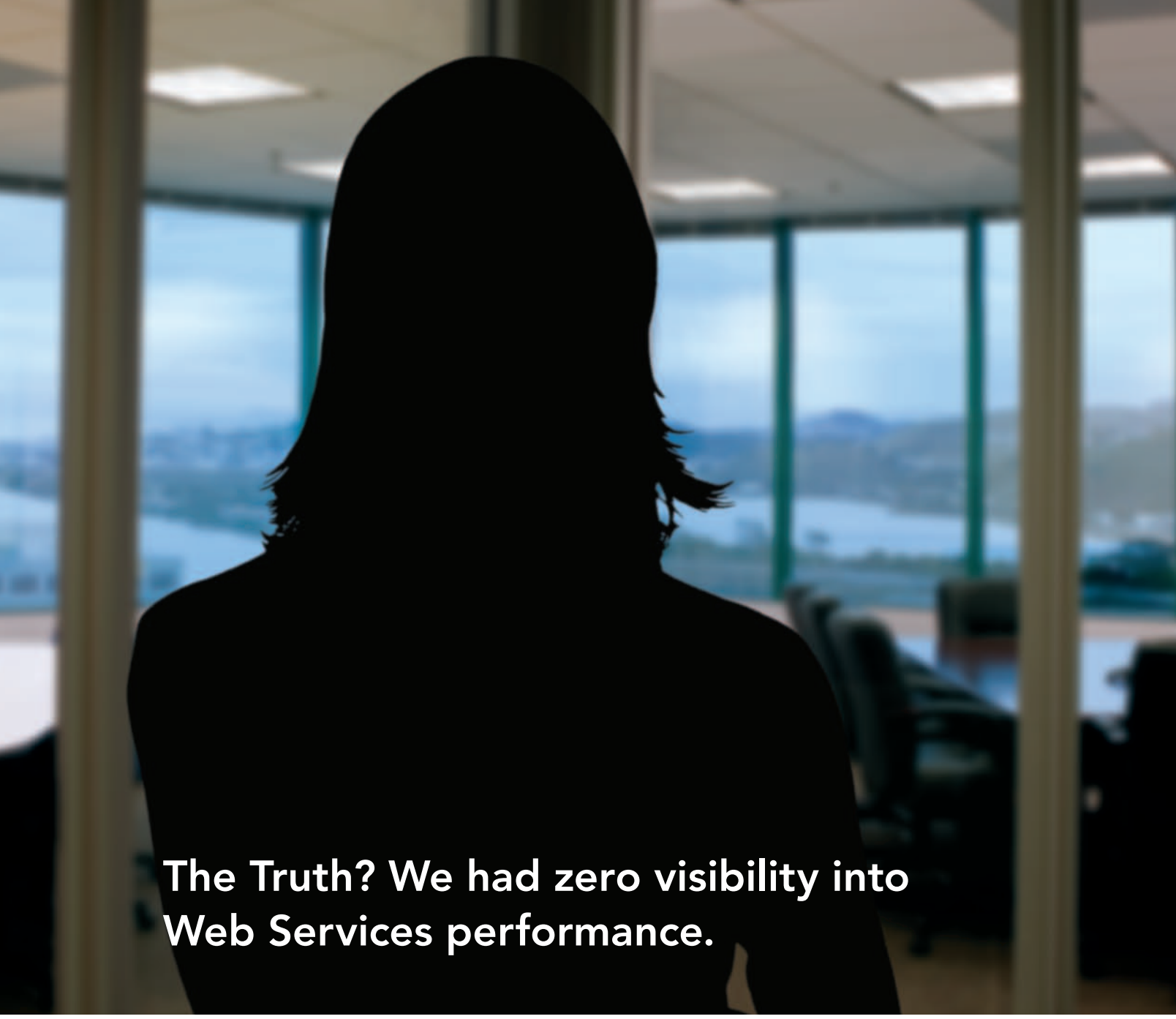
The Fundamental Importance of Trust in an SOA

Trust has become a visible issue for SOA. But what exactly do we mean by "trust"? And why is it so important?

A working SOA functions like a marketplace. And trust is a key ingredient in a functioning market.

Consider an online consumer marketplace where anonymous buyers and sellers are expected to come together and conduct business despite their total anonymity. Buyers aren't willing to do business unless they understand what's being offered, the terms and conditions of the sale, and the reputation of the seller; likewise, sellers want to be assured of the buyer's ability and willingness to pay in a timely fashion. An element of trust must exist for a transaction to take place.

In this respect SOA is no different. Without trust SOA can't succeed: Consumers simply won't reuse services if they can't be assured of the quality, predictability, and transparency of



Wily can handle the truth. We've helped hundreds of organizations around the world manage critical Web Services and proactively identify problems before customers are affected. Using a single solution, you gain control over both your application infrastructure and the customer experience. To learn more about how Wily can help you achieve customer success with Web Services, visit **truth.wilytech.com**.

Get Wily.™

Enterprise Application Management

wily™ technology

> Lack of working governance mechanisms in mid- to large-size SOA projects is the most common reason for project failure

the terms and conditions. In the same fashion, organizations can't realistically allow services to be used without solid processes for provisioning and controlling access, as well as for understanding the overall fitness of reusable services.

A significant challenge to widespread SOA adoption is that, while managing service quality is paramount, simply having quality service isn't enough. Quality is a key component in establishing consumer trust; it must be proven and demonstrated to consumers to gain their trust and create an effective shared-service environment.

Governance Is a First-Order Issue

An organization would be ill advised to start looking at governance down the road once the SOA implementation has reached a certain level of maturity. In the unique context of SOA, governance doesn't follow success; it's a prerequisite for success. It would be a mistake to discount governance as something that's optional, nice to have, or a later-phase consideration.

To be successful, you must consider an SOA governance strategy when you initially deploy an SOA. Your goal should be to establish a framework for assuring service quality and engendering trust between service providers and consumers as services progress through their lifecycles. Without governance strategies or infrastructure in place, organizations will hit roadblocks as they try to advance their SOA initiatives.

The Consequences of an Ungoverned SOA

As previously mentioned, an ungoverned SOA can become a liability for the enterprise, reversing the positive cycle, and adding costs and disrupting processes. In fact, the Gartner Group estimates that a lack of working governance mechanisms in mid- to large-size (greater than 50 services) SOA projects is the most common reason for project failure.

As with any management initiative, a key goal is to minimize risk – in this case, by defining an SOA strategy that builds governance into its core.

Potential consequences of an ungoverned SOA include:

- A lack of trust in service offerings, causing consumers not to reuse services because of unpredictable quality and performance issues;
- A disruption in processes by publishing services that don't fully conform to service-level requirements or by failing to assess the impact of change;
- Escalations in support costs through an onslaught of help desk and field service calls due to service issues and outages;
- A lack of interoperability, creating silos of business services and perpetuating the same challenges of a traditional, tightly coupled architecture;
- Non-compliance with regulations by failing to associate key policies with services;
- Security breaches by allowing arbitrary access to data and services; and
- An overall SOA failure by letting chaos reign and perpetuating a "garbage in, garbage out" environment.

The likelihood of these issues manifesting in an ungoverned SOA increases exponentially as the number of service offerings grows.

Key Components of SOA and the Role of Governance

To understand the increasing importance of governance in an SOA, let's look briefly at the road to SOA.

Initially, there were silos of monolithic applications. While silos offer the benefit of tightly controlled, application-specific functionality, a business doesn't operate in a silo. For example, customer information is often spread across multiple applications, and producing a single view of a customer's purchase, payment, and service history can be difficult. It involves creating fragile proprietary links between systems that don't handle change easily.

This was not sustainable so enterprises introduced an integration layer. Message Queue (MQ) and subsequently Enterprise Application Integration (EAI) reduced initial integration costs with adapters, but due to the tightly coupled nature of these applications, maintenance costs were enormous. Enterprises then implemented Enterprise Service Buses (ESBs) and Web Services to help address the problem. Web Services are standards-based and loosely coupled. ESBs also leverage standards and offer some loose coupling.

But the level of granularity with these technologies was too low, which led to misalignment with the business. This, in turn, led to business services.

Business services are expressly designed to align with the business needs. They may be Web Services or non-Web Services deriving from legacy systems. An example of a transformation to business services might be turning 2,000 fine-grained API-level services into a reusable set of 200 coarse-grained business services. With the advent of business services, enterprises could orchestrate these services into composite applications and implement Business Process Management and workflow.

While this new set of technologies solved the original problems of proprietary, tightly coupled, fine-grained systems, it introduced a new challenge for the enterprise: a lack of control over change. Since services were now decoupled from applications and technology, changes in these services could have a severe impact on the consumer of these services. Hence the need for governance.

The elements that help create an SOA fall

into three areas: SOA infrastructure, SOA management and security, and SOA governance.

SOA infrastructure services often include components such as:

- An ESB to integrate applications;
- A BPEL-based service orchestration engine to tie services into business processes;
- A business rules engine to capture and automate business policies; and
- A business activity monitoring solution to optimize services.

We often group SOA management and security together because they usually have overlapping functionality. That is, an SOA management and security component typically enforces policies such as authentication and authorization on services at runtime.

Finally, SOA governance usually include:

- Lifecycle management;
- Policy management;
- Contract management; and
- SOA metadata management.

Looking at the breadth of management concerns, it's apparent that there's no one single solution for SOA governance; instead, you need a suite of integrated tools. Vendors such as Oracle and Systinet are leading the way in developing such integrated tools by introducing solutions such as the Oracle SOA Suite. For example, the Oracle Service Registry, the OEM version of the UDDI v3-compliant Systinet Registry, integrates with other components in the Oracle SOA suite to provide a platform for managing several aspects of SOA management, such as lifecycle management.

Now, let's take a deeper look at the various components of SOA governance.

Lifecycle Management

As you've gathered by now, SOA's success and viability is directly related to quality and predictability, which ultimately engenders trust. System developers or architects are unlikely to start building an application against a service unless they can guarantee that the service is fully certified for quality, predictability, interoperability, and performance.

As such, managing the SOA lifecycle is an intrinsic part of SOA governance. In general, SOA lifecycle management revolves around:

- Ensuring the quality, performance, and applicability of services that are published;
- Providing a way for consumers to discover and reuse services and other artifacts;
- Managing versions, security, and the state-change of services and other artifacts; and
- Assessing and managing the impact of change across a network of consumers.

Although it's a common mistake to treat the requirements of providers and consumers similarly, their needs are quite unique. Because of the loosely coupled nature of providers and consumers in an SOA, there are actually two parallel yet distinctly different lifecycles at work in an SOA.

Service providers focus on the lifecycle of individual services as they are designed, built, and deployed. In general the provider lifecycle is centered on:

- Understanding and managing the service requirements;
- Managing service access and visibility;
- Publishing information to support the reuse of services; and
- Managing an infrastructure to deliver on quality-of-service commitments.

The consumer lifecycle is quite different. It involves:

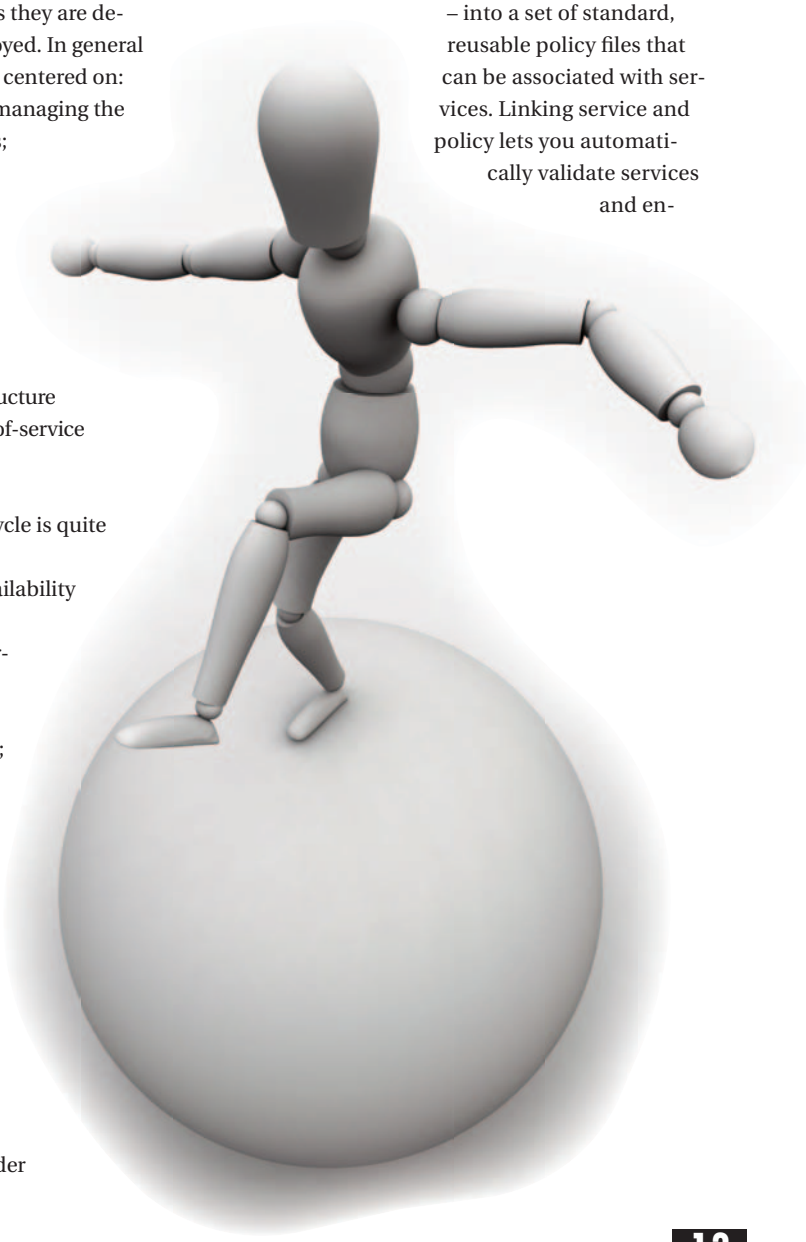
- Exploring service availability and capabilities;
- Validating the conformance of services;
- Negotiating terms of usage with providers;
- Validating and reporting on quality of service; and
- Discovering and responding to changes in services that are consumed.

Proper SOA governance depends on a strategy that addresses the needs and concerns of both provider and consumer lifecycles. Such a strategy offers the structure, control, and discipline necessary to encourage good and discourage bad behaviors.

Policy Management

An SOA policy defines configurable rules and conditions that affect services during design time and at runtime. The nature of SOA – highly distributed, heterogeneous, and very dynamic – means that it's critical for SOA artifacts to be governed by such specific business, technical, and regulatory policies.

An initial step in policy definition is to turn existing service rules – which often exist as soft-copy documents – into a set of standard, reusable policy files that can be associated with services. Linking service and policy lets you automatically validate services and en-



force specific policies.

Once they're defined, policies are used throughout the service lifecycle. For example, they're used to validate services at design-time, well before they're released to consumers, and they're used to enforce specific standards and behaviors at runtime.

The goal, of course, is to first ensure that quality issues and non-conformance are detected *before* services are put into production. Once in production, organizations should ideally implement runtime policy management capabilities for monitoring and automatically enforcing policies during service use.

The key aspects of SOA policy management include:

- ***Policy Definition:*** Transforms paper-based guidelines and rules into configurable electronic policies that provide the foundation for reuse and automation.
- ***Design-Time Enforcement:*** Automatically validates services in the design cycle to isolate policy conformance issues before the services are published.
- ***Runtime Enforcement:*** Enables Policy Enforcement Points to trigger events and actions (typically restrictive) automatically at runtime when services fail to conform to specific policies.
- ***Policy Lifecycle Management:*** Offers full control and management of policies from creation to retirement, including managing and maintaining the associations to business services.

Contract Management

Contracts are critical for communicating and enforcing policies, as well as other requirements in a heterogeneous and distributed IT environment. Just as a business contract ensures a healthy commercial relationship, a service contract ensures a healthy provider/consumer relationship.

Contracts are typically unique to a specific provider/consumer relationship and serve as the container for both formal policies, as well as agreements that are unique to the parties.

To put this in context, consider the example of renting a car. The rental agency is the provider, the renter is the consumer, and the car is the service. The contract defines information about the provider (the rental agency) and the consumer (the renter). It also specifies the service (the car), the terms and conditions (the policies), and any other provisions or agreements that are unique to the provider and consumer (for example, pre-paying for fuel). This contract is the basis for an agreement to bind the deal. A service contract is no different in complexity or purpose.

SOA Metadata Management

Effective SOA governance is ultimately the result of combining policy, process, and metadata. Metadata, or data about data, is the set of policies and descriptions of business services that let you discover and appropriately use those services. In the context of our discussion, metadata is all the information that would be centrally published to a UDDI registry.

There are three kinds of metadata generally associated with SOA:

- Business information, which includes data such as service type (for example, order entry) and line-of-business focus (for example, retail banking).
- Technical information required to consume or invoke a service, including transport type, authentication, interfaces, and implementation.
- Governance information, comprising the various policies and agreements discussed previously, along with information that identifies the relationships and dependencies between SOA elements, such as the service version or references to associated policies.

In the world of tightly coupled implementations, metadata is usually defined in the code of systems and applications. SOA requires this metadata to be externalized — that is, decoupled from the native system — so that these independent services can be classified and governed.

Conclusion

The promise of SOA is powerful and appealing. But what's apparent as organizations peel back the layers of SOA is that it radically changes traditional IT architectures. While SOA promises untold opportunities, it also introduces new challenges and risks that you must manage and mitigate.

The concept of SOA governance, while still somewhat nascent, is already a prerequisite for a successful SOA implementation. As with any sound management practice, SOA must be seen as a first-order concern with requirements that you must factor into your organization's SOA strategy at the very earliest stages of implementation because, simply put, without an effective governance strategy, SOA can lead to chaos.

Resources

- Oracle SOA resource center: www.oracle.com/soa
- Systinet SOA resources: www.systinet.com

> "While SOA promises untold opportunities, it also introduces new challenges and risks that you must manage and mitigate"

About the Authors

Dan Hynes is a principal product manager in the Java Platform Group at Oracle, focusing primarily on Web services and UDDI registry-relatedprojects.

dan.hynes@oracle.com

Sean Kline is director of product marketing for Systinet, a Mercury division. He has 15 years of product marketing and management experience in enterprise software. During this time, he has helped hundreds of companies improve software quality, security, and implement service-oriented architectures.

Altova® SemanticWorks™ 2006 – The ground-breaking visual RDF/OWL editor.

ALTOVA®
semanticworks™
2006

Give more meaning to the Web

Discover SemanticWorks™ 2006, and put the Semantic Web to work for you.

All new SemanticWorks 2006:

- Graphical composition of RDF, RDF Schema, OWL Lite, OWL DL, and OWL Full documents
- Intelligent entry helpers
- Automatic syntax and semantics checking
- Conversion of visual designs to RDF/XML or N-triples

Altova® SemanticWorks™, the brand new graphical Semantic Web development tool from the creators of XMLSpy®, lets you unearth the full potential of your Web-based and knowledge-management applications. Visually design RDF instance documents, RDF Schema vocabularies, and OWL ontologies then output them in either RDF/XML or N-triples formats. SemanticWorks makes the job easy with tabs for instances, properties, classes, etc., context-sensitive entry helpers, and automatic format checking. Wise up to the Semantic Web!

Download SemanticWorks™ 2006 today: www.altova.com

ALTOVA®

Now also included in the Altova XML Suite.

BPEL Processes and Human Workflow

Using BPEL in business processes that require human interaction

Business Process Execution Language (BPEL), one of the key technologies for Service Oriented Architecture (SOA), has become the accepted mechanism for defining and executing business processes in a common vendor-neutral way. Companies ranging from Oracle, IBM, Microsoft, SAP, and BEA to smaller organizations such as Fuego and Lombardi have committed to BPEL as a building block for SOA. BPEL, which has been designed specifically for defining business processes, supports typical interactions such as synchronous and asynchronous operation invocation, sequential and parallel flows, message correlations, fault and compensation handlers and activities triggered by events. Business processes often require human interactions as well.

WRITTEN BY
MATJAZ B. JURIC & DOUG TODD

Since the BPEL specification doesn't address them, you might think BPEL isn't suitable for business processes that involve people. But that's not the case. In this article, we'll look at different choices for human workflow support – including future possible extensions to the BPEL specification and current vendor solutions – and analyze their relevance in practical scenarios. We'll also discuss real-world scenarios in which BPEL and human workflow services are used and show how one company is using BPEL to integrate people with processes – and the benefits achieved through this approach.

User Interaction in Business Processes

BPEL business processes are defined as collections of activities that invoke services. BPEL doesn't make a distinction between services provided by applications and other interactions, such as human interactions. And that's important since real-world business processes often integrate not only systems and services but also users. User interactions in business processes can be simple, such as approving certain tasks or decisions, or complex, such as delegation, renewal, escalation, nomination, or chained execution.

Task approval is the simplest and probably the most common user interaction. In a business process for opening a new account, a user interaction might be required to decide whether the user is allowed to open the account. If the situation is more complex, a business process might require several users to make approvals, either in sequence or in parallel. In sequential scenarios, the next user often wants to see the decision made by the previous user. Sometimes, particularly in parallel user interactions, users aren't allowed to see the other users' decisions. This improves the decision potential. Sometimes one user doesn't even know which other users are involved – or whether any other users are involved at all.

A common scenario for involving more than one user is workflow with escalation. Escalation is typically used in situations where an activity doesn't fulfill a time constraint. In such a case, a notification is sent to one or more users. Escalations can be chained, going first to the first-line employees and advancing to senior staff if the activity isn't fulfilled.

Sometimes it's difficult or impossible to define in advance which user should perform an interaction. In this case, a supervisor might manually nominate the task to other employees; the nomination can also be made by a group of users or by a decision-support system.

In other scenarios, a business process may require a single user to perform several steps that can be defined in advance or during the execution of the process instance. Even more complex processes might require that one workflow is continued with another workflow.

User interactions aren't limited to approvals; they can also include data entries or process management issues, such as process initiation, suspension, and exception management. This is particularly true in long-running business processes, where, for example, user exception handling can prevent costly process termination and related compensation for those activities that have already been successfully completed.

As a best practice for human workflows, it's usually not wise to associate human interactions directly with specific users; it's better to connect tasks to roles and then associate those roles with individual users. This gives business processes greater flexibility, letting any user with a certain role interact with the process and enabling changes to users and roles to be made dynamically.

BPEL and User Interaction

So far we've seen that user interaction in business processes can get quite complex.

Although BPEL specification 1.1 (and the upcoming BPEL 2.0) doesn't specifically cover user interactions, BPEL is appropriate for human workflows. Several vendors today have created workflow services that leverage the rich BPEL support for asynchronous services. In this fashion, people and manual tasks become just another asynchronous service from the perspective of the orchestrating process and the BPEL processes stay 100% standard.

We now see the next generation of workflow specifications emerging around BPEL with the objective of standardizing the explicit inclusion of human tasks in BPEL processes. This proposal is called BPEL4People and was originally put forth by IBM and SAP in July 2005. Other companies, such as Oracle, have also indicated that they intend to participate in and support this effort.

The most important extensions introduced in BPEL4People are people activities and people links. People activity is a new BPEL activity used to define user interactions; in other words, tasks that a user has to perform. For each people activity, the BPEL server must create work items and distribute them to users eligible to execute them. People activities can have input and output variables and can specify deadlines.

To specify the implementation of people activities, BPEL4People introduced tasks. Tasks specify actions that users must perform. Tasks can have descriptions, priorities, deadlines, and other properties. To represent tasks to users, we need a client application that provides a user interface and interacts with tasks: it can query available tasks, claim and revoke them, and complete or fail them.

To associate people activities and the related tasks with users or groups of users, BPEL4People introduced people links. People links are somewhat similar to partner links; they associate users with one or more people activities. People links are usually associated with generic human roles, such as process initiator, process stakeholders, owners, and administrators.

The actual users who are associated with people activities can be determined at design time, deployment time, or runtime. BPEL4People anticipates the use of directories such as LDAP to select users; however, it doesn't define the query language used to select users.

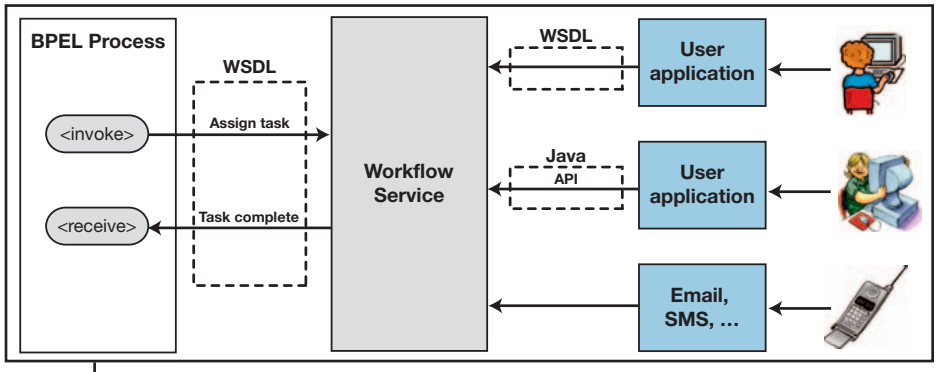


FIGURE 1 **Workflow integration with BPEL**

Rather, it foresees the use of LDAP filters, SQL, XQuery, or other methods.

BPEL4People proposes complex extensions to the BPEL specification, however so far it's still quite high level and doesn't yet specify the exact syntax of the new activities mentioned above. Until the specification becomes more concrete, we don't expect vendors to implement the proposed extensions. But while BPEL4People is early in the standardization process, it shows a great deal of promise.

Finally, as it stands today, the BPEL4People proposal raises an important question: Is it necessary to introduce such complex extensions to BPEL to cover user interactions? As described previously, some vendor solutions model user interactions as just another Web Service with well-defined interfaces for both BPEL processes and client applications. This approach doesn't require any changes to BPEL; to become portable, it would only need an industry-wide agreement on the two interfaces. And of course, both interfaces can be specified with WSDL, which gives developers great flexibility and lets them use practically any environment, language, or platform that supports Web Services. An example of such an approach is the Workflow Service in the Oracle BPEL Process Manager, which we'll describe later.

First, we should complete the discussion of standards efforts by pointing out that there are several older workflow specifications, most notably Wf-XML from the Workflow Management Coalition (WfMC). Wf-XML is an XML-based proposal for consistent data transfer between workflow engines. As far as we know, it hasn't been used in any major BPEL engine, probably because WfMC and the Business Process Management Initiative jointly released the XML Process Definition Language. XPDL focuses on the design-time interoperability of different workflow products and is therefore only of very limited relevance to the BPEL community.

Clearly, a single standard approach hasn't yet been adopted for extending BPEL to include human tasks and workflow services. However, this doesn't mean that developers can't use BPEL to develop business processes with user interactions. In the rest of this article, we're pragmatic and describe one approach that works today for integrating user interactions in standard BPEL processes.

Workflow Integration with BPEL

To interleave user interactions with service invocations in BPEL processes we can use a workflow service, which interacts with BPEL using standard WSDL interfaces. This way, the BPEL process can assign user tasks and wait for responses by invoking the workflow service using the same syntax as for any other service. The BPEL process can also perform more complex operations such as updating, completing, renewing, routing, and escalating tasks.

After the BPEL process has assigned tasks to users, users can act on the tasks by using the appropriate applications. The applications communicate with the workflow service by using WSDL interfaces or another API (such as Java) to acquire the list of tasks for selected users, render appropriate user interfaces, and return results to the workflow service, which forwards them to the BPEL process. User applications can also do other tasks such as reassign, escalate, route, suspend, resume, and withdraw. Finally, a workflow service may allow other communication channels, such as e-mail and SMS, as shown in Figure 1.

Oracle BPEL Process Manager uses such an architectural abstraction to integrate standard BPEL functionality with workflow. Loose coupling lets workflow services be deployed on any supported application server. It also allows evolving the workflow service, as specifications such as BPEL4People emerge, without having to change the existing BPEL processes. The workflow service includes a simple yet powerful set of Java APIs and WSDL interfaces for building UI workflow interfaces; these offer maximum interoperability for UI approaches, including JSF, AJAX, .NET, and Adobe Flex.

Case Study

Let's look at a case study that shows how you can incorporate human interactions with BPEL. Consider a business process for opening a new bank account; we'll call this process "New Account." The customer provides the necessary details (such as name, address, social security number, and initial deposit) to open the account. Once the process is initiated, the customer may want to track the status of the request and respond to any additional queries from the bank. This process requires a workflow to enable customer participation and process monitoring so that the customer can track the request status.

This example is based on a multi-organizational process that creates accounts for a financial service intermediary, its financial partners (insurance companies and their financial advisors), mutual fund companies, and clearinghouses. This value chain of partners collaborates to make the "New Account" business process more efficient and compliant. Each partner has compliance requirements that must be satisfied during the instantiation of the process. Regulations such as the U.S. Patriot Act, the Bank Secrecy Act, and other federal and state regulations apply and add requirements to the process for both automated system and human workflow interactions.

The example combines business process management through BPEL to orchestrate business processes in a B2B context, managing compliance in a secure environment, while greatly enhancing the efficiency of the overall process. This approach was selected because it makes the organization resilient to ongoing threats from a regulatory and competitive standpoint. These requirements were incorporated into the "New Account" business process, including the following key requirements:

- Common interfaces for maximum interoperability
- Comprehensive security, as each environment has firewall and other security requirements
- Audit trails and monitoring for compliance, security, and performance reasons

As with most business processes that require automation, human interactions play a large role in this business process. Workflow requirements for the process include:

- Electronic data routing (intra-organization as well as inter-organization)
- Data verification and approval of provided information
- Workflow routing to other departments with notification
- Escalation as a result of exceptions and/or alerts
- Approval of the entire heterogeneous transaction
- The need for workflow requirements to interact seamlessly with automated business process steps and existing enterprise information

BPEL fit these requirements perfectly and can also address security concerns due to its foundation of Web Services and integration with specifications such as WS-Security and SAML. Enterra, the company that developed this process, first evaluated how a BPEL server could function as both a process automation and workflow tool. Since workflow specifications for BPEL are evolving, as described above, the decision was based on the following criteria:

- Maturity of the BPEL product
- Support for workflow from BPEL
- Native support for relevant standards
- Ease of development, maintenance, and deployment
- Satisfying the process monitoring and audit capabilities
- Using a single security model supported by BPEL
- Taking advantage of BPEL's built-in transaction semantics

Oracle BPEL Process Manager was the selected product based on these criteria.

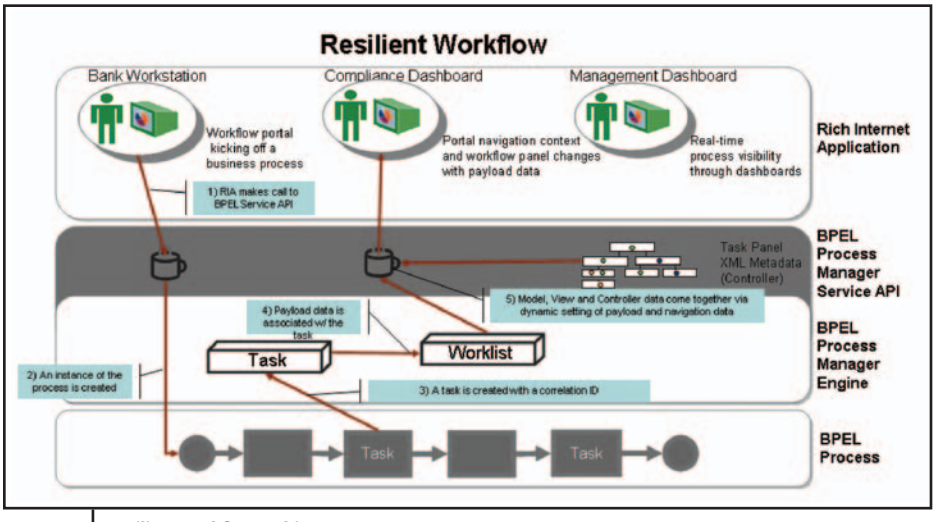


FIGURE 2 | **Resilient workflow architecture**

Workflow Interface Architecture and Benefits

Enterra used Oracle's BPEL Process Manager engine and its Java API to build an enterprise-ready resilient workflow architecture. Figure 2 shows that architecture.

The following describes the workflow:

1. The Rich Internet Interface (RIA) workflow portal instantiates the business process via the BPELService API
2. The instance ID of the process is captured to maintain state
3. When a task is initiated in the process, a correlation ID is captured
4. The payload data is sent to the task and is available via the worklist interface
5. The API pulls the worklist data and uses the

Fiorano SOA™ 2006

The Quickest Path to an SOA

Fiorano provides the industry's first framework enabling developers to rapidly create SOA applications in well-formed J2EE code.

Key Features:

- JCA-compliant Service-component Framework
- JMS based messaging
- BPEL support for composite components and applications
- Synchronous and Asynchronous Services in a single framework
- Support for multiple languages : Java, C, C++, C# and others

With Fiorano you can:

- Directly deploy Service Components and BPEL processes to any standard J2EE container
- Rapidly deliver ACID transactions across front-end (JSP) and backend processes (Databases, ERPs, etc.)
- SOA enable your existing J2EE application (EJB, JCA, JMS) without reengineering or code-generation
- Easily debug complex flows spanning multiple applications and middleware
- Perform runtime service deployment, service updates and application changes

Download your copy of Fiorano today!

http://www.fiorano.com/downloads

Fiorano®

Enabling Change at the speed of thought

correlation ID to gather the XML metadata to return to the RIA. The RIA uses the metadata to dynamically determine which panel to display next and to update the context of the visual navigation mechanisms used in the workflow portal. This data is returned to the portal and all the update events are processed.

To implement the workflow scenario shown here, Enterra developed the Rich Internet Interface and used Java APIs to interact with the Oracle BPEL Process Manager Workflow Service. The RIA user interface is the face of the application; it runs and manages the "New Account" business process. The presentation layer is built using Macromedia's Flex technology and ActionScript scripting language. It accepts the user input and passes it to the BPEL process and also displays the current status of the process. Figure 3 shows the RIA front-end for the New Account business process.

Let's review its four main components and their key capabilities:

1) ***Workflow Bar:*** This lets users view the workflow steps in the New Account process at the business level. Every step in the Workflow Bar is displayed in a detail view in the BPEL Panel.
2) ***BPEL Panel:*** The BPEL Panel offers an exploded view of every step of the New Account process. The panel dynamically loads the process model and the audit trail of a given process instance to combine the process paths available with the paths actually taken by the current instance. The panel also uses the audit trail to determine what action is pending, visually delineating where the current process is at that time. Previously executed nodes of the process can be clicked on to view their respective payload. This functionality gives operations and compliance managers real-time visibility into the past, current, and potentially future states of the current process. This panel is dynamically populated at runtime using the BPEL Process Manager API.
3) ***Workflow Forms:*** These correspond to the workflow steps in the New Account BPEL process. This view stack (developed in Macromedia Flex) communicates with the audit trail of the business process, changing the view stack form to match the context of the BPEL process. When considering different architectural approaches, we found that the RIA application would have been much more difficult to build if the data had come from two different applications (workflow and BPM). A single platform made it much easier to build a unified interface for the business user.
4) ***Zone Panels:*** Zone Panels provide alerts and feedback to help workflow and process activity users perform their respective tasks. These panels offer a superior usability experience and interact with the BPEL API. As the context of the business process changes, the Zone Panels are updated. The events that manage these changes are propagated through the Oracle BPEL Process Manager API.
5) ***BPEL Process Manager:*** The BPELService class facilitates interaction between the RIA front-end and the process activity monitoring capabilities in Oracle BPEL Process Manager.

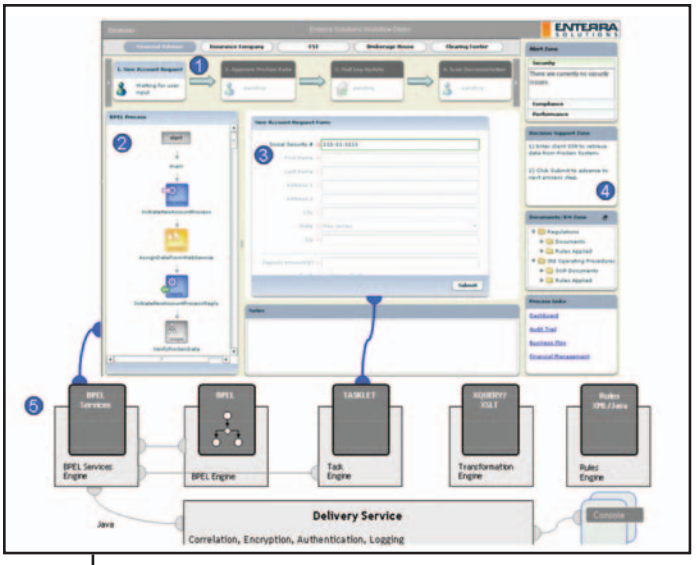


FIGURE 3 **Rich Internet Application front-end for the new account business process**

The RIA interface also provides detailed audit trails views and other business process and workflow alerts in real-time. The combination of BPEL, workflow, and RIA provides a compelling architecture for addressing compliance, security, and performance metrics.

The benefits of the New Account business process implementation are:

- Maximum interoperability without compromising security
- Real-time monitoring and auditing of the entire environment
- Robust workflow capabilities integrated cleanly in the BPEL environment
- An enterprise-ready infrastructure that supports transactions, availability, and advanced exception handling

Conclusion

When automating business processes, it's critical to include the element of human interaction in the design. Standards play a key role in business process automation, and BPEL is clearly the de facto standard for process execution, but human workflow doesn't have such a single widely adopted standard. In this article, we've highlighted the options related to human workflow support in BPEL, discussed the emergence of BPEL workflow extensions, shown how to incorporate workflow into BPEL servers today, and presented a real-world BPEL process that includes human workflow.

Acknowledgments

We would like to thank Harish Gaur and Dave Shaffer for their helpful comments.

About the Authors

Dr Matjaz B. Juric is a professor at the University of Maribor. He is the author of the book Business Process Execution Language for Web Services, 2nd Edition, published by Packt Publishing (www.packtpub.com), and several other books, articles, and conference presentations. He is also the author of courses offered by BPEL-mentor.com - a training, mentoring, and consulting company.
matjaz.juric@uni-mb.si

Doug Todd is CTO of Enterra Solutions in Yardley, Pa. He has more than 20 years of experience in systems architecture, applications architecture, systems integration, and applications integration with major corporations. Todd is responsible for Enterra's overall IT strategy and tactical implementation, enterprise information architecture, and technology product offerings.
dtodd@enterrasolutions.com

Ah, remember when EDI was young and full of promise?

The future of EDI is B2B over IP

EDI sure had promise in the sixties – but its complexity, inflexibility, and the bottleneck of the VAN make it very costly today.

Simple to integrate, easy to manage, and blazingly fast, **terminalONE** is *the* B2B over IP platform. It intelligently transports, transforms, and routes all your data transactions.

We're about ebusiness adaptability: your business world changes fast – wouldn't it be nice if your data interchange adapted quickly and painlessly along with it?

terminalONE™

Secure, intelligent ebusiness transactions

*high-**velocity***
*high-**volume***
*high-**availability***

Take **terminalONE** for a test drive – ***today***
www.xenos.com/VAN

1 888 242 0695
1 905 763 4468
terminalONE@xenos.com

xenos

WS Security Performance

Secure Conversation versus the X509 Profile

The WS Secure Conversation specification describes a mechanism letting multiple parties establish a context (using the WS Trust Request Security Token standard) and secure subsequent SOAP exchanges. Each WS Secure Conversation session has an associated shared secret. Instead of using this shared secret directly to sign and encrypt the conversation's messages, symmetric keys are derived from the secret itself. Deriving new keys for each message and different keys for signature and encryption limits the amount of data that an attacker can analyze in attempting to compromise the context.

WRITTEN BY

FRANCOIS LASCELLES

AARON FLINT

Derived Key Tokens are tokens in a SOAP Security Header that refer to the derived keys. Using the context's shared secret and hints provided by the Derived Key Token element, the message's recipient derives the key used by the requestor either to verify a signature or decrypt parts of the message.

How Derived Key Tokens are used is best understood by looking at Listing #1, which illustrates a *SOAP* message signed and encrypted as detailed by the WS Secure Conversation specification. Notice how the element *Header/Security/SecurityContextToken* refers to the pre-established WS Secure Conversation context. Both parties participating in this message know the shared secret associated with the context. Two Derived Key Tokens are declared in the Security header. Both of those Derived Key Tokens refer to the same *Security Context Token* but the associated derived keys are different as per the derivation *Nonces* provided. The element *Header/Security/Signature/KeyInfo* refers to one of the derived keys and the *Body/EncryptedData/KeyInfo* refers to the other.

Derived Key Tokens Beyond Secure Conversation Contexts

The Derived Key Token mechanism described in WS Secure Conversation relies on a shared secret. This shared secret doesn't have to be in the form of a WS Secure Conversation context key. It can be as simple as a password (think *UsernameToken*) or a Kerberos ticket (think Kerberos *BinarySecurityTokens*). Any form of shared secret that can be mapped to a security token can effectively serve as the basis for deriving keys (although for an implementation to interoperate easily one should stick to the derivations defined by standards). For example, the Web Services Security UsernameToken Profile 1.1 specification describes a mechanism where the password associated with a username is used to derive a secret key to protect the integrity or confidentiality of the message content. This has the advantage of not requiring that the context be pre-established.

Another interesting approach to key derivation that avoids the offline establishment of a context is to derive keys on an *EncryptedKeyToken*. In this case, the requestor makes up a secret, encrypts it, and sends it to the recipient. This generated secret is shared between the requestor and the recipient and only the recipient can decrypt it. Of course, this shared secret alone can't be used for authentication purposes but derived keys based on such a shared secret can still be useful for encrypting a message and signing it for ensuring integrity. The WS Security 1.1 spec also allows subsequent messages to refer to an encrypted key defined in a previous message. Deriving keys based on this previous secret has the advantage of avoiding the expensive operation associated with deciphering a new encrypted key for each message. Of course, any use of EncryptedKeyTokens requires the initiator to know the X509 cert of the recipient to encrypt the initial key.

WS Security Performance

Messages secured on a pre-established WS Secure Conversation are processed by both parties using symmetrical cryptography only. This contrasts with other mechanisms such as the ones described in the *X509 Token Profile* specification where XML digital signatures are based on an X509 *BinarySecurityToken* and where encryption is based on a key that is itself encrypted using the recipient's public key. In that case, both signature and encryption operations require using asymmetrical cryptography.

Your CPU will tell you that cryptography is generally expensive and that asymmetrical cryptography is extremely expensive. So it's reasonable to expect WS Secure Conversation-based WS Security to be processed at faster rate than X509-based WS Security. The question is how significant this performance advantage is in a real-world deployment

burdened by other overheads such as XML processing.

Secure Conversation vs. X509 Profile Benchmark

As illustrated in Figure 1, an XML gateway is introduced between a number of WS requestors and a WS server. This XML gateway gets security-decorated SOAP requests coming in from clients, deciphers them, and verifies the digital signature. The XML gateway then forwards the request to the back-end Web Service that returns a SOAP response. This response is then secured by the XML gateway (XML encryption and signature) before it's forwarded back to the original requestor. The response security is achieved using the same mechanism used to secure the request.

The WS Security method used for securing these SOAP messages is dictated by a WS Policy document published by the XML gateway. By altering this policy document we can switch between messages secured using Derived Key Tokens associated with a WS Secure Conversation session versus messages secured using an X509 token profile mechanism. The key derivation algorithm used by the XML gateway is the standard PSHA-1 described in the WS Secure Conversation specification.

In this scenario the number of messages per second the gateway was able to process for each of these WS Security mechanisms was measured. Listings 1 and 2 illustrate sample messages processed by the XML gateway for Derived Key Tokens and X509 respectively. Also measured was the number of requests per time unit processed by this same gateway in a case where messages didn't involved WS Security at all and were exchanged through SSL as well as a benchmark measurement taken with no security policy present al all.

On the requestor side, five systems running Apache benchmark were simultaneously sending pre-formatted SOAP requests to the XML gateway inside an isolated network. The gateway was deployed as a single node. On the back-end, an Apache server returned static unsecured SOAP responses. In these tests, all of the WS Security processing was delegated to the gateway, both the requestors and the back-end service were sending hard-coded SOAP messages; this ensures that we focus the bottleneck and isolate the real throughput of the XML gateway with regards to WS Security processing as much as possible.

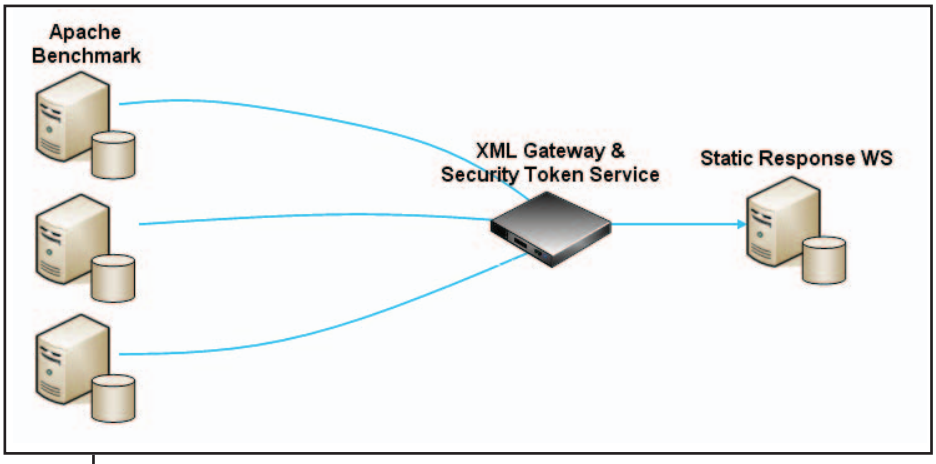


FIGURE 1 **Benchmark testing configuration**

Security Mechanism	Messages per second	CPU load	Throughput (kB/s)
X509 XML Signature & Encryption	352	99	2,403
WS Secure Conversation XML Signature & Encryption	798	98	5,679
SSL with HTTP Basic	2,918	95	3,181
None (message routing only)	5,088	96	5,419

TABLE 1 **Benchmark results for 25 concurrent requestors**

Benchmark Results

The numbers shown (see Table 1) are for messages processed per second by the single node XML gateway (note that each request and response is processed as separate messages). As you can see, when processing messages secured using WS Secure Conversation, the XML gateway was able to handle as many as 798 messages per second as compared to 352 messages per second for X509-based signatures and encryption. The anticipated performance gain predicted is very significant; the throughput more than doubles for the single XML gateway node.

To provide context the number of messages the same XML gateway processed when security was based purely on transport mechanisms (in this case SSL) was also measured. In that case, the single node XML gateway processed 2,918 messages a second.

Summary

Using a purely symmetric crypto approach to WS Security as is possible with Derived Key Tokens produces a processing performance advantage over WS Security achieved through the X509 profile that relies on public key crypto. This performance gain has the potential to translate into significant throughput gains in a production environment where WS Security processing is involved in a bottleneck. Derived Key Tokens are also a practical approach to WS Security; they can be used in conjunction with a number of different mechanisms such as Kerberos, passwords, and WS Secure Conversations, and they don't need a public key infrastructure.

However, the mechanisms described in the X509 token profile should by no means be regarded as inferior. The public key aspects of the X509 token profile provide functional advantages over WS Security relying exclusively on Derived Key Tokens.

Indeed, the performance advantage

> WS Secure Conversation means more performance, while the X509 profile allows for non-repudiation

provided by signing and encrypting messages using exclusively symmetrical crypto comes at a price. Because the messages are signed with something based on a shared secret, those signatures can't form the basis of non-repudiation. Both parties knowing the shared secret can produce such signatures. Conversely, when message signatures are based on an X509 token, they prove the possession of a private key to which the recipient doesn't have access; the signing party can't claim that the other party forged his or her signature. Obviously, asymmetrical crypto is just one piece of the complicated non-repudiation puzzle, but an essential one nevertheless.

Another advantage of using X509 mechanisms over session-based security is that digital certificates and their associated private keys typically have a longer lifecycle than security contexts such as WS Secure Conversation sessions or Kerberos tickets. The ephemeral nature of security contexts restricts (if not eliminates) the ability to audit a message offline long after it's been processed. Once a session has expired, and the associated shared secret is forgotten, encryption can no longer be undone and signatures become meaningless. On the other hand, messages including signatures and encrypted elements that refer to X509 certificates can be saved for later auditing; they can be decrypted later, their signatures can be verified.

About the Authors

Long before terms like Web Services and SOA were coined, Francois Lascelles was developing applications using SOAP and other XML standards. He joined Layer 7 Technologies in its early days and helped shape the vision of its SecureSpan product line. Today, as a member of Layer 7's' engineering team, Francois assists corporations in taking full advantage of Web Service Security technologies.

flascelles@layer7tech.com

Over the past 10 years, Aaron Flint has worked, in increasingly senior positions, to ensure quality of enterprise-level server applications. He joined Layer 7 Technologies to lead the QA department early on and has been helping to build and release a quality SecureSpan product line since then.

aflint@layer7tech.com

Listing 1: WS Security achieved through DerivedKeyTokens

```
(some namespaces have been truncated to fit this page)

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/
soap/envelope/"
  xmlns:wsse="http://docs.oasis-open.org/...wssecurity-
secext-1.0.xsd"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
  xmlns:wsu="http://docs.oasis-open.org/...wssecurity-
utility-1.0.xsd">
  <soap:Header>
    <wsse:Security xmlns:wssc="http://schemas.xmlsoap.org/
ws/2005/02/sc">
      <wssc:SecurityContextToken wsu:Id="scContext">
        <wssc:Identifier>uuid:5e78f6fc-bf05</wssc:
Identifier>
      </wssc:SecurityContextToken>
      <wssc:DerivedKeyToken wsu:Id="signingkey">
        <wsse:SecurityTokenReference>
          <wsse:Reference URI="#scContext"/>
        </wsse:SecurityTokenReference>
        <wssc:Length>16</wssc:Length>
        <wssc:Label>WS-SecureConversation</wssc:Label>
        <wssc:Nonce>5zIc5sVkvSzZgDcB98G9qg==</wssc:Nonce>
      </wssc:DerivedKeyToken>
      <wssc:DerivedKeyToken wsu:Id="encryptingKey">
        <wsse:SecurityTokenReference>
          <wsse:Reference URI="#scContext"/>
        </wsse:SecurityTokenReference>
        <wssc:Length>16</wssc:Length>
        <wssc:Label>WS-SecureConversation</wssc:Label>
        <wssc:Nonce>mh9fNSRNMT6QTlyW7ovZnQ==</wssc:Nonce>
      </wssc:DerivedKeyToken>
      <xenc:ReferenceList>
        <xenc:DataReference URI="#encryptedContent"/>
      </xenc:ReferenceList>
      <Signature xmlns="http://www.w3.org/2000/09/xmld-
sig#">
        <SignedInfo>
          <!-- ... -->
        </SignedInfo>
        <SignatureValue>R1bt...tB9RI.M=</SignatureValue>
        <KeyInfo>
          <wsse:SecurityTokenReference>
            <wsse:Reference URI="#signingkey"
              ValueType="http://schemas.xmlsoap.org/
ws/2005/02/sc/dk"/>
          </wsse:SecurityTokenReference>
        </KeyInfo>
      </Signature>
    </wsse:Security>
  </soap:Header>
  <soap:Body>
    <xenc:EncryptedData Id="encryptedContent"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/
xmlenc#aes128-cbc"/>
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <wsse:SecurityTokenReference>
          <wsse:Reference URI="#encryptingKey"
            ValueType="http://schemas.xmlsoap.org/
ws/2005/02/sc/dk"/>
        </wsse:SecurityTokenReference>
      </KeyInfo>
      <xenc:CipherData>
        <xenc:CipherValue>Hqjdc...+LEV51oMqtg=</xenc:
CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </soap:Body>
</soap:Envelope>
```

Listing 2: WS Security achieved through X509 Certificate Profile

```
(some namespaces have been truncated to fit this page)
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/
soap/envelope/"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
  xmlns:wsu="http://docs.oasis-open.org/...utility-
1.0.xsd">
  <soap:Header>
    <wsse:Security
      xmlns:wsse="http://docs.oasis-open.org/...secext-
1.0.xsd"
      soap:mustUnderstand="1">
      <wsse:BinarySecurityToken
        EncodingType="..."
        ValueType="http://...x509-token-profile-
1.0#X509v3"
        wsu:Id="x509bst">
          MIICLj...=
      </wsse:BinarySecurityToken>
      <xenc:EncryptedKey>
        <xenc:EncryptionMethod
          Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-1_5"/>
        <dsig:KeyInfo xmlns:dsig="http://www.
w3.org/2000/09/xmldsig#">
          <wsse:SecurityTokenReference>
            <wsse:KeyIdentifier
              EncodingType="..."
              ValueType="http://
...#X509SubjectKeyIdentifier">
                tVuqKSO89...Mw=
            </wsse:KeyIdentifier>
          </wsse:SecurityTokenReference>
        </dsig:KeyInfo>
        <xenc:CipherData>
          <xenc:CipherValue>OjGpFrTg...4MQ=</xenc:
CipherValue>
        </xenc:CipherData>
        <xenc:ReferenceList>
          <xenc:DataReference URI="#bodyencdata"/>
        </xenc:ReferenceList>
      </xenc:EncryptedKey>
      <ds:Signature xmlns:ds="http://www.
w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <!-- ... -->
        </ds:SignedInfo>
        <ds:SignatureValue>tbDUa...kqio=</ds:
SignatureValue>
        <ds:KeyInfo>
          <wsse:SecurityTokenReference>
            <wsse:Reference URI="#x509bst"
              ValueType="http://...-x509-token-pro-
file-1.0#X509v3"/>
          </wsse:SecurityTokenReference>
        </ds:KeyInfo>
      </ds:Signature>
    </wsse:Security>
  </soap:Header>
  <soap:Body>
    <xenc:EncryptedData Id="bodyencdata"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/
xmlenc#aes128-cbc"/>
      <xenc:CipherData>
        <xenc:CipherValue>mrsIN...WQ==</xenc:
CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </soap:Body>
</soap:Envelope>
```

Adding Reliability to Occasionally Connected Computing in Mobile Devices

Leveraging Web Services Reliable Messaging for persistence

Mobile devices use wireless networks that have limited range. Therefore, they may not always be connected to a network. This kind of intermittent connectivity in mobile devices has been one of the factors inhibiting enterprise-level adoption of pervasive mobile applications. Occasionally Connected Computing (OCC) deals with this problem. The challenge that mobile application developers face today is to develop applications that will let users interact uniformly with the application regardless of the connection status.

In the recent past, Web Services have appeared as a key technology for mobility with support from leading mobile platforms. However, leveraging Web Services for occasionally connected mobility applications falls flat because of a lack of reliability. In this article we present a framework that enables OCC using Web Services Reliable Messaging (WSRM), which can be a catalyst for enabling occasionally connected mainstream mobility applications. For our framework, WSRM provides the reliability required for Web Service communication in occasionally connected environments.

WRITTEN BY


ABHISHEK MALAY CHATTERJEE


GEO PHILIPS KURAVAKAL


TERANCE DIAS


VARUN PODDAR


SRINIVAS PADMANABHUNI

This article is divided into three parts. The first part surveys the relevant background material on Web Services support in mobile platforms, WSRM, and occasionally connected computing. The second part explains a typical use case for occasionally connected mobile computing and explains our framework and implementation details. The last section provides conclusions on this work and identifies further scope for future work.

Overview of Mobile Platforms and Web Service Frameworks

Mobile devices today not only include the common mobile phones, but also smartphones, PDAs, small form factor tablet PCs, and so on. These devices vary significantly on hardware and software configurations. Here is a brief overview of the most common mobile platforms. Most modern rich mobile devices use one of the following operating systems:

1. Symbian OS is currently the most popular Smartphone operating system. It is an independent Open Source project. The OS isn't found as such on devices but custom-

Welcome to the Future of Video on the Web!

REGISTER NOW!
www.iTVcon.com

CALL FOR PAPERS NOW OPEN!

LIVE SIMULCAST!
AROUND THE WORLD ON SYS-CON.TV

iTVCON.COM
INTERNET TV CONFERENCE & EXPO 2006

Coming in 2006 to New York City!

"Internet TV is wide open, it is global, and in true 'Web 2.0' spirit it is a direct-to-consumer opportunity!"

For More Information, Call 201-802-3023
or Email itvcon@sys-con.com

Welcome to the Future!

Did you already purchase your ".tv" domain name?
You can't afford not to add Internet TV to your Website in 2006!

2005 was the year of streaming video and the birth of **Internet TV**, the long-awaited convergence of television and the Internet. Now that broadband is available to more than 100 million households worldwide, every corporate Website and every media company must now provide video content to remain competitive, not to mention live and interactive video Webinars and on-demand Webcasts.

20 years ago the advent of desktop publishing tools opened the doors for the creation of some of today's well-known traditional print media companies as well as revolutionized corporate print communications. Today, with maturing digital video production, the advent of fully featured PVRs, and significant advances in streaming video technologies, **Internet TV** is here to stay and grow and will be a critical part of every Website and every business in the years to come.

It will also very rapidly become a huge challenge to network and cable television stations: **Internet TV** is about to change forever the $300BN television industry, too.

The Internet killed most of print media (even though many publishers don't realize it yet), Google killed traditional advertising models, and **Internet TV** will revolutionize television the way we watch it today. You need to be part of this change!

Jeremy Geelan
Conference Chair, iTVCon.com
jeremy@sys-con.com

PRODUCED BY SYS-CON EVENTS

List of Topics:

- Advertising Models for Video-on-demand (VOD)
- Internet TV Commercials
- Mastering Adobe Flash Video
- How to Harness Open Media Formats (DVB, etc)
- Multicasting
- Extending Internet TV to Windows CE-based Devices
- Live Polling During Webcasts
- Video Press Releases
- Pay-Per-View
- Screencasting
- Video Search & Search Optimization
- Syndication of Video Assets
- V-Blogs & Videoblogging
- Choosing Your PVR
- Product Placement in Video Content
- UK Perspective: BBC's "Dirac Project"
- Case Study: SuperSun, Hong Kong

Track 1	Corporate marketing, advertising, product and brand managers
Track 2	Software programmers, developers, Website owners and operators
Track 3	Advertising agencies, advertisers and video content producers
Track 4	Print and online content providers, representatives from traditional media companies, print and online magazine and newspaper publishers, network and cable television business managers

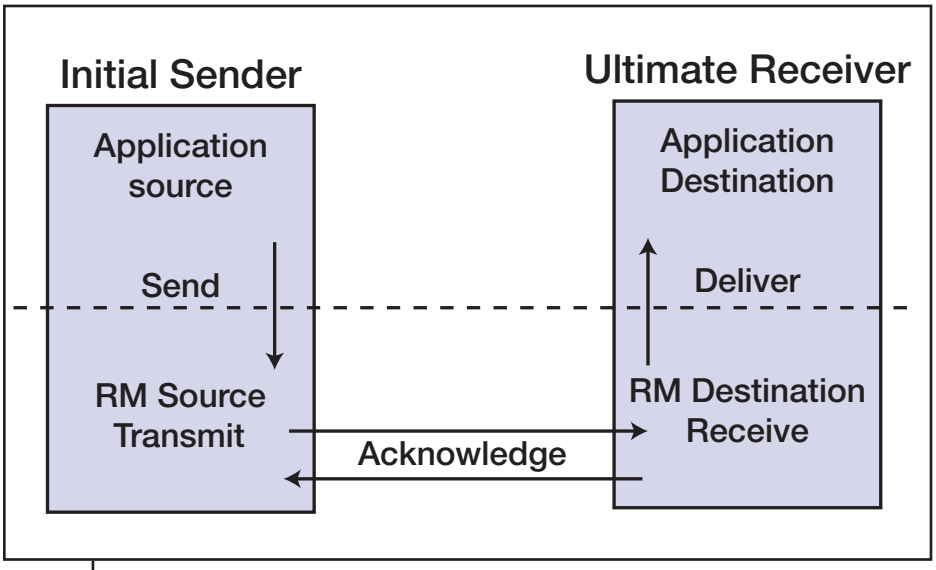


FIGURE 1 **Reliable Messaging model**
Source: http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/wsrm-ws_reliability-1.1-spec-os.pdf

ized according to a manufacturer's needs. Nokia phones use a series of Symbian-based interfaces known as the S series (such as S60, S90) while Sony Ericsson devices have UIQ platform interfaces. Symbian runs mainly J2ME applications and has a dedicated Java execution layer. It's not found in PDAs without phone functionality.

2. Windows CE runs on a variety of mobile devices such as PDAs, Smartphones, and hybrids. The various versions of CE are given different names such as Windows Mobile 2002 and 2003 along with different editions of the same such as PocketPC, Smartphone, and PocketPC Phone. The latest version is Windows Mobile 5.0. It uses the .NET CF platform for development.
3. PalmOS can also be found in many mobile PDAs and phones, such as the popular Treo line of devices. Being the first dedicated mobile OS, it has a huge library of applications. Coding is done primarily in C/C++.
4. Other OSes such as Linux and its embedded versions can be found in specialized devices. BlackBerries use a proprietary OS. However, they represent a small niche of the market.

At the language level, most of these devices natively support either the Java or .NET platform.

1. Java is implemented on mobile devices using the Java 2 Micro Edition (J2ME) platform. Programs written for J2ME are similar to normal Java programs, except they have smaller libraries and are written for a more compact virtual machine. Due to the large Java developer base and robust and open community, J2ME is by far the most popular mobile language platform.
2. Microsoft's .NET Compact Framework (.NET CF) for mobile devices lets developers program for devices that run Windows CE versions such as PocketPCs and Windows Mobile Smartphones. Integrated into the popular Visual Studio toolset, it has an intuitive and easy-to-use GUI to build and test mobile applications as well as excellent native XML support.
3. Other languages such as C/C++ are also supported on some OSes such as Symbian, Palm, and BlackBerry.

The diversity in application platforms as detailed above poses serious interoperability problems and is a cause of the high cost involved in developing enterprise mobility applications.

A recent trend in these mobile platforms is the inclusion of support for open standards like Web Services. This can be a crucial enabler of interoperability across diverse mobile platforms and of enterprise mobility applications that can be treated as extensions of conventional enterprise applications exposed via services to mobile devices. We explore the support for Web Services in different platforms below:

1. .NET CF offers native support for calling and handling Web Services, as well as complete XML processing capabilities, such as support for XML schemas, Xpath, and XML serialization. Due to the integrated nature of the framework and full support for SOAP and other protocols, the Web Service support in the CF is well-rounded and simulates functionality available on larger devices
2. The J2ME platform also supports Web Services, but not at the advanced level found in .NET CF. Basic Web Service clients can be made using the Java SOAP toolkit but until now such functionality wasn't natively available in J2ME itself. Sun has come out with JSR-172 to address these issues. JSR-172 aims to provide access to remote SOAP- and XML-based Web Services and to provide XML parsing capability in J2ME. With wider adoption of these new features among developers and frameworks like kSOAP and MIC already available, native Web Service handling in J2ME should come up to the level of the .NET CF implementation.

Web Services Reliable Messaging

WSRM is a specification by Microsoft, IBM, Tibco Software and BEA Systems for the reliable delivery of SOAP messages. By using this specification the sender can make sure that the Web service message actually reaches the Web service receiver and is not lost in transit. To implement this, the sender may store the message before sending it across to the receiver and delete it only when it gets an acknowledgment from the receiver. If the receiver fails to acknowledge the message, it's assumed to be lost and is re-sent. This is the most basic form of WSRM and is called a *'once and only once'* delivery assurance profile. There is also a *'at most once'* profile where one tries to reach the Web Service just once. If it fails, the message is lost. Although this defeats the purpose of having WSRM, it can be quite useful as we show in our use case.

In Figure 1 the initial sender sends a message for reliable delivery. The source accepts the message and transmits it one or more times. After getting the message the destination acknowledges it. Finally, the destination delivers the message to the ultimate receiver.

Current Work in Occasionally Connected Computing

Before proceeding further, it's imperative that we understand the current work and research done in this area. While not many initiatives or products have taken off, the existing attempts can give us valuable pointers as to the advantages and drawbacks of such a scheme.

To handle occasionally connected scenarios, the MiddleWare Company extended the SOA Blueprints Reference to come up with the Occasionally Connected Profile (OCCP) v0.1 in 2004. This reference architecture was later handed over to Oasis for ratification and standardization through a community process. The profile covers an architectural framework that states that the user should be allowed to work offline and cache data locally when a connection isn't available and when the connection is restored, the cached data should be sent to the server. However, this profile has some limitations like its requirement for a mobile database for persistence, its scope of coverage in terms of the range of mobile devices covered, etc. Notwithstanding certain limitations in the draft OCCP specification, it's a farsighted attempt to specify a framework for mobile applications considering that when it was drafted, Web Services and XML weren't as prevalent as they are now. Most mobile platforms didn't support them natively either so the OCCP draft's shortcomings have to be evaluated in this context. It's a genuine forbearer of architectures such as the one we're suggesting.

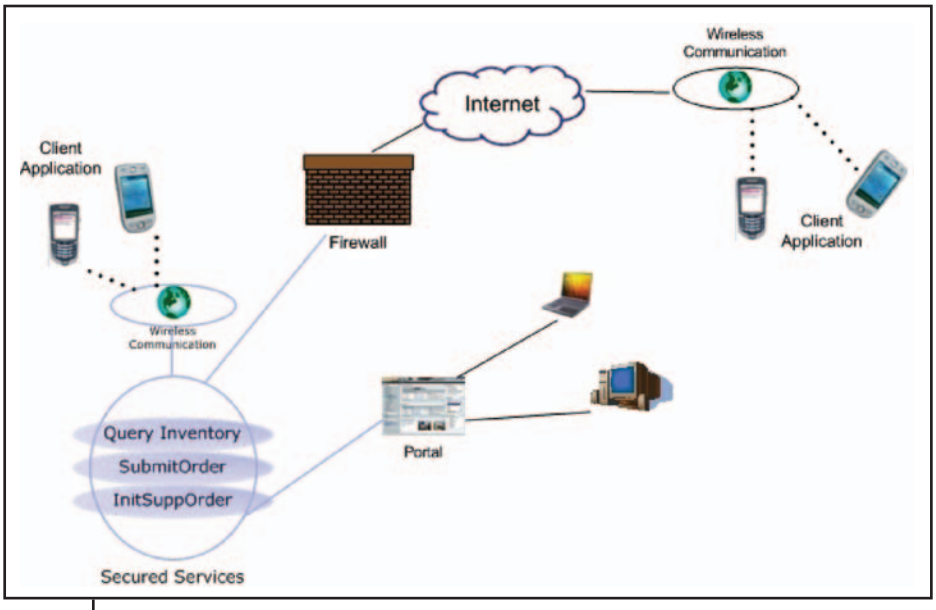


FIGURE 2 | **Application setup**

Use Case

To drive home the requirements of a realistic architectural framework leveraging OCC, we propose a comprehensive use case. We assume an "Inventory Management System" in the context of a mobile sales force application to depict the ever-increasing need for connectivity in an enterprise for its mobile work force. This system provides services like submitting a customer order, querying the current inventory status, and initiating the order process with a supplier to replenish inventory.

These services have to cater to the needs of three kinds of users, all of whom can place a customer's order and view the required inventory status: a field salesperson who is on the move and has to access these services through a Symbian smartphone, a manager moving around with a PocketPC device, and an in-house salesperson/manager/employee who has a laptop or PC and can place a customer's order or order from the inventory supplier through the company portal. To make the services available to the different users, they are available as Web Services. Access to the portal requires that the devices always be online. But with mobile devices having intermittent connectivity, this isn't always possible. So these devices have a client application installed through which they can access these Web Services. This setup is shown in Figure 2.

Access to these Web Services is role-based. The salesperson can see the current inventory status and submit customer orders whereas the manager has the extra privilege of initiating the order process with the supplier for refilling the stock.

We have assumed that the enterprise uses an optimistic approach for its sale process. Here the salesperson will have a fixed quota out of the total stock in inventory. This will enable the salesperson to create and submit customer orders when he's not connected. In this case he will create orders only within the allotted quota. The client application lets the user submit orders as if he were online and when he's connected again, the submitted orders will be sent to the central repository and the order processing may start immediately. However, if he's connected he can view the real-time inventory status and take orders beyond his quota. The decrease in inventory may result in triggering a re-order alert to replenish the inventory. The trigger will make

> "Despite the inherent occasionally connected nature of mobile devices architectures can be devised to enable enterprise-level mobile applications"

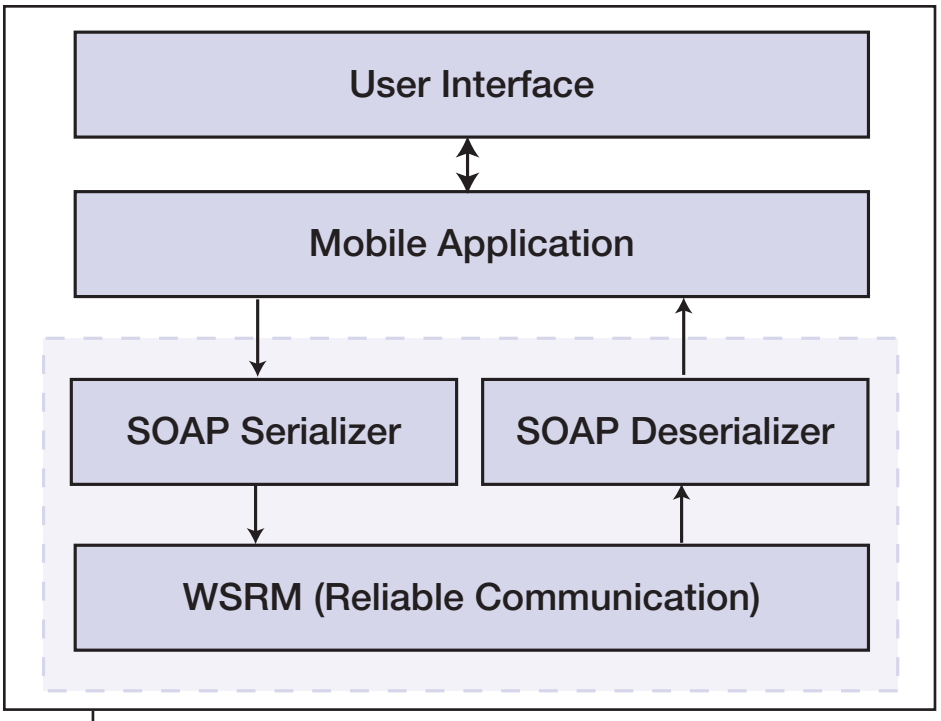


FIGURE 3 **The framework**

the manager aware of the current inventory status so that he can initiate the re-order with the supplier. Hence a perfect balance can be maintained between the current stock and the customer orders. The manager can also initiate the re-order at his discretion and market speculation. The application will make sure that the order to the supplier is initiated even if his mobile device isn't connected by using WSRM to send the request upon receiving connectivity transparently.

Framework

Occasionally connected computing requires the presence of a client application that lets the user work and store his data offline. Figure 3 shows the high-level components needed to develop such an application.

The user interface lets the user enter inputs into the system. The mobile client application will handle the events generated by the user's interaction. It will also make sure that the application works normally even if the device isn't connected (offline mode). This can require keeping the application state information and caching some content in the mobile device storage (depending on the scenario).

The application will call the Web Service residing in a remote server using the SOAP serializer and WSRM in Figure 3. The SOAP serializer will create a SOAP request message as required by the service and send the message to the WSRM component for reliable communication. The response is passed from the WSRM to the SOAP deserializer to convert the SOAP response to the application-specific format.

WSRM ensures that the reliability level (exactly once, at least once, etc.) of the communication during the Web Service invocation is attained. Depending on the requirements of the application the reliability level for the Web Service invocation can be set. In the case of a reliability level being set at 'at most once,' the application can switch over to offline mode if a response isn't available. It's also possible that the application may not have to have an offline execution mode. In that case the user is kept oblivious of the connection status of the device and the WSRM component will ensure that the Web service is called whenever the connection is available.

Figure 4 shows the flow of the request process for placing an order.

The first flowchart depicts the flow of placing an order with the server. Here two Web Services are used. The first one is the QueryInventory Web Service for checking the status of the inventory. This Web Service uses a "at most once" delivery assurance profile. So if the WSRM component tries to send it once and it fails, it informs the application. Depending on the whether the response has arrived from this Web Service, the application decides whether to use the inventory value or the quota. The second Web Service is for placing the order, i.e., the SubmitOrder Web Service. This Web Service uses the 'exactly once' delivery assurance profile. Once the application has created the request, the WSRM component tries to send it; if it fails, it tries twice more and then tries again after some specified period of time.

Implementation Details

In our current implementation we're using Microsoft's .NET Framework 2.0 running on IIS 5.1 to implement Web Services. To work with WSRM messages we're using Web Service Enhancements (WSE) 3.0. On the client side we're using a Symbian smartphone running J2ME and PocketPC running .NET Compact Framework. To persist the data on the client side, we're using files.

The WSRM component of the client application installed on these mobile devices creates the WSRM-compliant messages and stores them in files. The application resends messages in case of failure. The WSRM component takes care of the *'once and only once'* and '*at most once*' delivery assurance depending on the configuration file associated with each Web Service.

Conclusions and Future Work

We've shown that despite the inherent occasionally connected nature of mobile devices architectures can be devised to enable enterprise-level mobile applications. Two key contributing technologies to do so include at the core a Web Services-based framework that is predicated on the universal support of Web Services in most mobile platforms. Further, for reliability, we proposed a framework based on WSRM to handle the occasionally connected problem in mobile Web Services. We've used WSRM, which is not yet an Oasis standard. Alternatively, WS-Reliability can be used, which is an Oasis standard. We decided to go with WSRM because it's consistent with other WS-* standards.

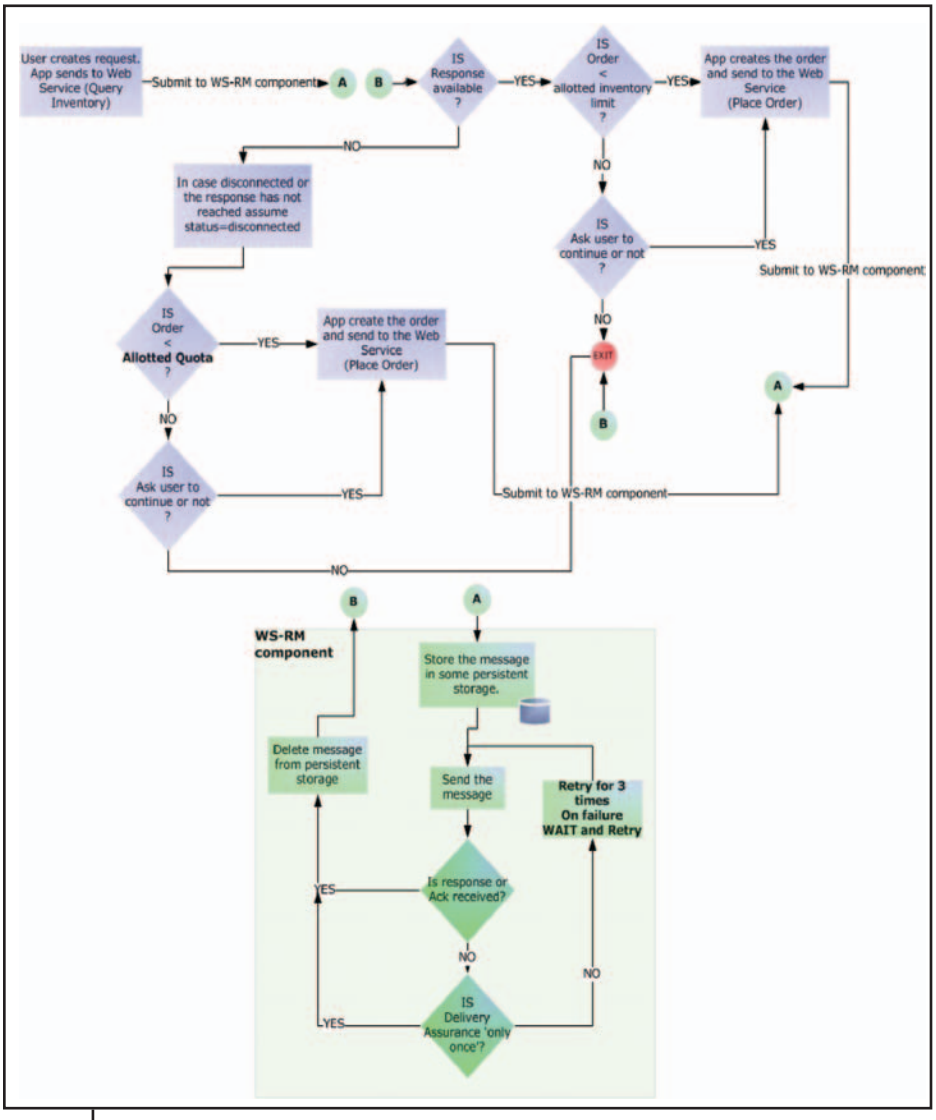


FIGURE 4 | **Request flow**

Our framework stores data in files, which is not optimal in enterprise applications. Future work can include more optimal ways of storing data on the client side.

References

- Symbian technology page with overviews of its mobile operating systems: http://www.symbian.com/technology/technology.html
- Home page for Windows Mobile range of mobile operating systems: http://www.microsoft.com/windowsmobile/default.mspx
- Home page for the Palm operating system: http://www.palmsource.com/palmos/
- Blackberry OS developer page with a brief description of the OS framework: http://www.blackberry.net/developers/started/blackberry.shtml
- Sun's resource page for J2ME: http://java.sun.com/j2me/
- Microsoft's developer documentation and resources page for .NET CF: http://msdn.microsoft.com/netframework/programming/netcf/default.aspx
- The Java Community page for JSR-172: http://www.jcp.org/en/jsr/detail?id=172
- Ksoap, the Open Source J2ME SOAP API: http://ksoap.objectweb.org/
- Product page for MIC: http://www.agile-delta.com/MIC.htm
- IBM's WSRM page: http://www-128.ibm.com/developerworks/library/specification/ws-rm/
- Occasionally Connected Profile v0.1, an extension of the SOA Blueprints Reference Example to deal with occasionally connected rich client interfaces: http://xml.coverpages.org/SOA-BlueprintsOccConnPv01.pdf
- Information on WSE 3.0: http://msdn.microsoft.com/webservices/webservices/building/wse/default.aspx
- Web Services Reliable Messaging TC WS-Reliability 1.1: http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/wsrm-ws_reliability-1.1-spec-os.pdf

About the Authors

Abhishek Chatterjee currently works as a part of the Web Services/SOA center of excellence at SETLabs, the R&D arm of Infosys Technologies Ltd., and has substantial experience in publishing papers, presenting papers at conferences, and defining standards for SOA and Web Services.

abhishek_chatterjee@infosys.com

Geo Philips Kuravakal currently works with the Web Services/SOA Center of Excellence at SETLabs. He is currently involved in the development of an enterprise Web Service management framework. He is also interested in the Semantic Web and its applicability to Web Services, new programming methodologies such as AJAX, and mobile Web Services.

geo_kuravakal@infosys.com

Terance Dias is a member of the Web Services/SOA Center of Excellence at SETLabs. He has substantial experience in publishing papers, presenting papers at conferences, and defining standards for SOA and Web Services. His fields of interest include SOA-enabling technologies like ESB, Web Services interoperability, and grid computing.

terance_dias@infosys.com

Varun Poddar is a member of the Web Services/SOA Center of Excellence in SETLabs. He is currently working on XML technologies using Java. His interests are theoretical computer science, the Semantic Web, mobile applications, AJAX, and SOA.

varun_poddar@infosys.com

Dr. Srinivas Padmanabhuni leads the Web Services/SOA Center of Excellence at SETLabs, the R&D arm of Infosys Technologies Ltd. His research interests include service-oriented enterprise architecture, enterprise security, semantic information integration and intelligent agents. He has been published in numerous international journals and conferences.

srinivas_p@infosys.com

Blending Discovery, Governance, Security, and Management in SOA

Left unchecked, SOA's flexibility will result in chaos

With power comes responsibility. The promise of Service Oriented Architecture (SOA) offers significant opportunity for service reuse and the realization of a fully integrated enterprise. But left unchecked, the flexibility enabled by an SOA will result in a Wild Wild West of enterprise IT. To properly harness the power of SOA while delivering value for the enterprise, certain controls are essential. Incorporating service discovery, service security, service management, and policy governance in a ubiquitous and transparent framework is essential to the success of any enterprise SOA deployment.

WRITTEN BY
ANDREW LAWLOR

With power comes responsibility. The SOA computing model offers tremendous promise to integrate the enterprise and enable reuse. But with this power comes the responsibility for its proper use. If an organization doesn't exercise sufficient control over the flexibility enabled by SOA principles, chaos will ensue.

To effectively realize the benefits of service orientation, some notions must be applied to harness its inherent flexibility for the benefit of the organization. Effective SOAs employ service discovery, service governance, service security, and service management in proper proportions. Each of these critical notions has to be understood independently and as they relate to each other. When they're combined into one seamless architecture the whole is more than the sum of its parts.

Publish-Find-Bind Notion

At the heart of a Service Oriented Architecture is the Publish-Find-Bind notion. This paradigm of loosely coupled computing is depicted in Figure 1.

Under this computing paradigm a service provider exposes its services (as service end-points) and waits for consumers to invoke those services. (Step 1 – Publish.) To find those consumers who would benefit the most from using its services, the provider "advertises" its services by publishing its offerings in a registry. This registry (usually either UDDI or ebXML standards-compliant) maintains a database of all services published in the enterprise along with details about that service to aid the consumer's discovery process. In practice, this information usually includes a WSDL file, which contains its invocation signature, as well as the URI of the end-point(s) where the service offer resides. The registry may also contain a full repository to hold additional details about each service's functions and characteristics.

Once a service is published in the enterprise registry an interested consumer may query the registry looking for services that meet his needs. (Step 2 – Find.) Developers then construct their systems using the services that they discover in the registry. The final step in a loosely coupled interaction (Step 3 – Bind) occurs at runtime. At execution time the consumer binds to the provider using the end-point URI supplied by the registry during the Find operation.

Under the Publish-Find-Bind loosely coupled paradigm, providers and consumers need not be familiar with each other prior to service consumption. The power of this notion introduces significant opportunity for chaos. For example, how to ensure that only authorized consumers can access a given service? How to ensure that all service providers secure their services properly from unauthorized consumers? How to ensure that organizational standards are followed throughout the enterprise SOA? To manage the power inherent in the loosely coupled model, additional layers of control must be introduced to ensure the benefits of SOA are realized while maintaining order.

Publish-Time Governance

Without control over an enterprise SOA, an unbounded number of internal service providers will expose and publish all manner of services for use by any consumer. A layer of governance must be introduced into this process at publishing time to ensure only high-quality services are published and that those services conform to enterprise standards. This is known as change time governance.

With change time governance, the registry validates the published service against a set of rules established by the enterprise and rejects any service provider that attempts to publish a service that doesn't conform to all established policies. Rejected services can't be consumed by an enterprise-level consumer. These policies can perform any number and manner of checks against the published service before registering it as an enterprise service. For instance, the governance tool can mandate that all enterprise service offerings follow the WS-I Basic profile and that they follow the doc/literal SOAP encoding (critical for successful WSDL first-design

methodology). Many organizations adopt a specialized SOAP header for all enterprise-level Web Service invocations. These organizations frequently employ change time governance to validate compliance with those headers. Service offerings that lack this standard enterprise SOAP header are rejected and denied publication in the registry.

Figure 2 shows where the change time governance function occurs in the loosely coupled lifecycle (step 1a – Validate). This governance tool (sometimes part of the registry product) manages all the corporate SOA policies.

Runtime Intermediaries

To inject control over an enterprise SOA, an intermediary must be injected between all consumer-provider interactions. This intermediary, usually referred to as a Policy Enforcement Point (PEP), can then perform a number of functions necessary to bring order to an SOA in real-time:

1. Runtime governance enforcement
2. Security-related functions
3. Management activities

These functions are done transparently to both the consumer and the provider (save for some impact on invocation performance). Figure 3 illustrates the process by which this intermediary is introduced into the publish-find-bind paradigm.

After change time publishing and governance validation (Steps 1 and 1a), the registry notifies the PEP that it will be acting as the intermediary for the given service (Step 1b above). The PEP then responds to the registry with the new end-point at which the enterprise can access the original service. Subsequently, all inquiries to the registry for this service will return the new end-point established on the PEP (Step 2). This process is known as end-point re-writing. *All* enterprise level interactions with this service will go through this intermediary.

All consumers attempting to bind to the given service (Step 3) will in fact be binding to a PEP acting as the provider. It's at this point that the intermediary can perform its critical functions. Once the PEP has completed its work (assuming all went well) the consumer is bound to the provider (Step 4). From that moment onward the PEP's work is done and it removes itself from the interaction (i.e., it effectively hard-wires the consumer to the provider).

Runtime Governance

With an intermediary in place the architecture can be used to ensure that all service invocations throughout the enterprise conform to established organizational standards. The PEP is used to enforce this runtime governance. For instance, service consumers that attempt to consume a service without the proper enterprise SOAP header can be rejected. Keep in mind that service providers would be unaware that a consumer attempted to consume their service in a way that breaks organizational policies.

Figure 4 shows the addition of runtime governance to our loosely coupled lifecycle. Step 3a shows the PEP validating the consumer against the organization's runtime governance policies before letting the consumer bind to the provider's service offer.

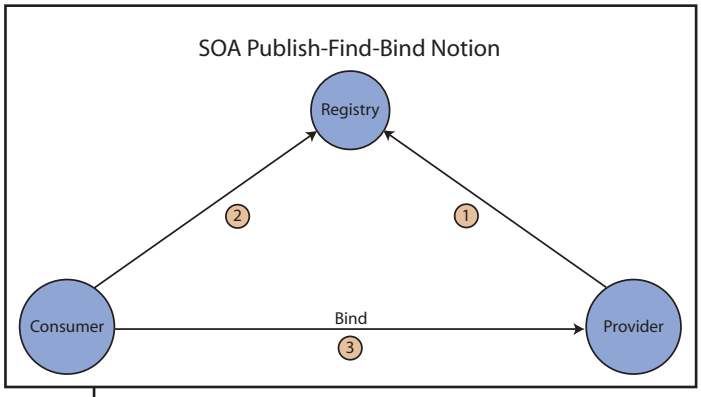


FIGURE 1 **Loosely coupled service interaction**

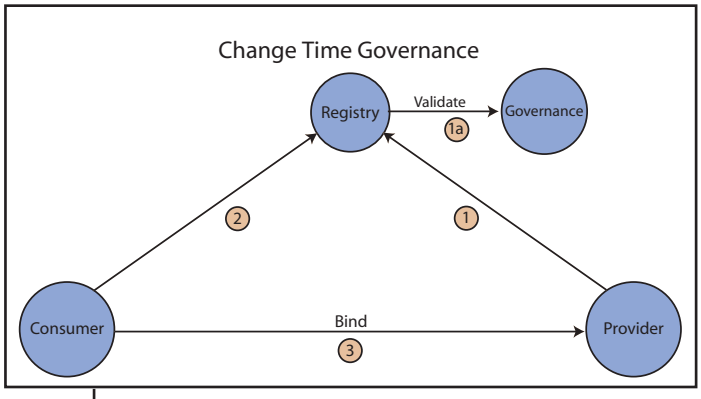


FIGURE 2 **Change time governance**

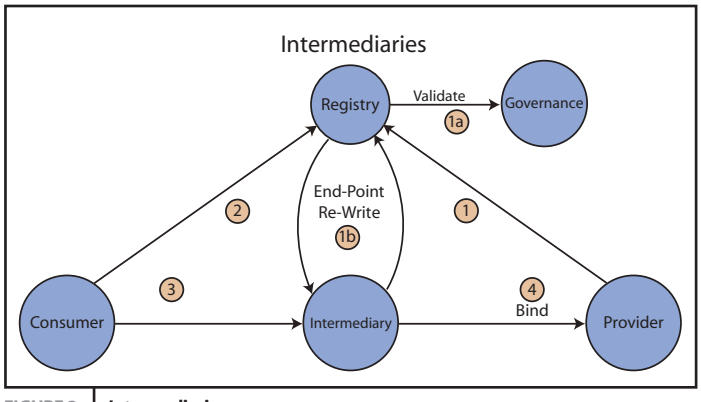


FIGURE 3 **Intermediaries**

Runtime Security

A PEP is frequently used to handle security-related functions such as authorization. Moving much of the security-related functionality to a PEP can significantly reduce the security overhead burden otherwise the responsibility of the service providers. This strategy lets the service providers concentrate on their core functionality. Additionally, these security-related functions can be consolidated in one place (the PEP), fostering re-use and administration benefits. These are significant benefits of the SOA paradigm and one of the driving forces behind the adoption of SOA.

Step 3b in Figure 5 shows the PEP invoking a Policy Resolution Service (PRS) to authorize the consumer's request to consume a particular service. To determine a consumer's eligibility to invoke a given service, the PEP forwards the

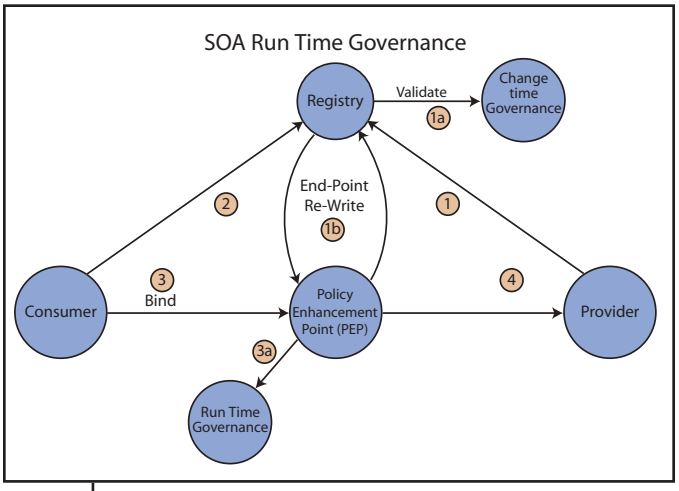


FIGURE 4 **Runtime governance**

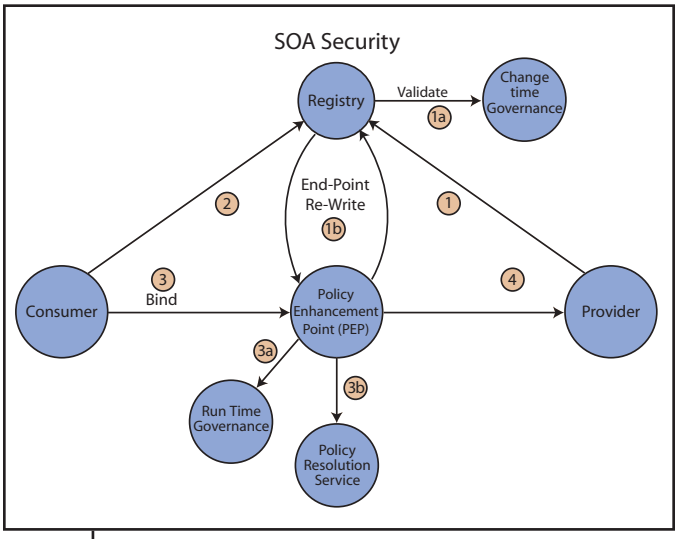


FIGURE 5 **Authorization**

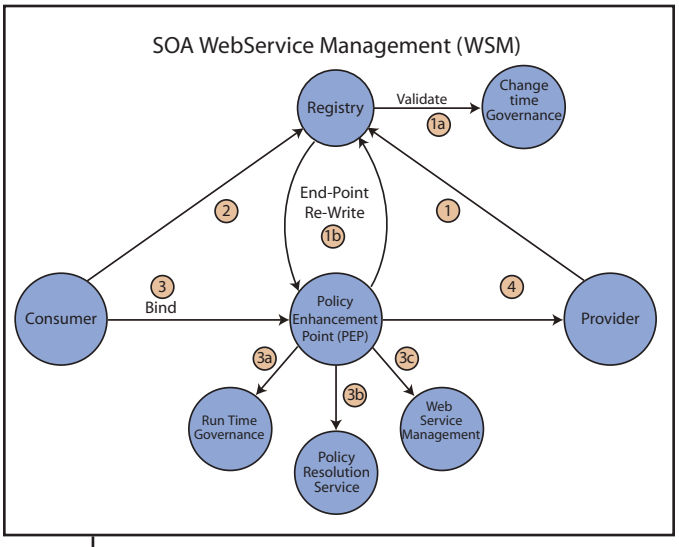


FIGURE 6 **Web Service management**

consumer's request to the PRS. The PRS extracts identity information from the consumer's SOAP header (which usually conforms to the WS-Security standards and contains a SAML assertion, which are beyond the scope of this article) and authorizes the consumer using the policies in its policy store.

If a consumer isn't authorized to consume the service that he attempts to access, he is rejected by the PEP (a WS-Security SOAP error is returned) without the knowledge of the service provider.

Web Service Management (WSM)

A PEP can also be used to do transparent Web Service management. This capability lets an organization monitor the "health" of its enterprise SOA. Metrics can be gathered for each and every service invocation. Capturing this data lets an organization:

1. identify when services become unavailable and take proactive action
2. identify the most popular services in the enterprise
3. identify the consumers that consume the most services in the enterprise
4. capture response times of all enterprise Web Services
5. compile availability figures of all enterprise Web Services

The addition of Web Service management is shown in step 3c of Figure 6.

WSM can take action when a service becomes unavailable. WSM can proactively, and without the knowledge of the consumer, redirect the Web Service call to an alternative end-point. WSM can also notify the registry that a given end-point is unavailable and suggest an alternative end-point where the service can be found.

In more advanced installations, WSM periodically provides automatic feedback to the registry with metrics on service availability, response time, or other relevant characteristics. This data can then be used by consumers to identify the best (most highly available, fastest responding, etc.) service for their needs.

Performance Considerations

The PEP has a host of responsibilities, each of which may consume significant computing resources and introduce latencies. To alleviate this bottleneck PEPs can be implemented using specialized hardware. These XML/PKI-aware devices can significantly reduce the impact (processing delay) when introducing an intermediary between each Web Service consumer and provider. Numerous vendors offer such devices today, sometimes called Application Oriented Networking (AON) gear.

Putting It All Together

A real-world SOA blends service discovery, governance, security, and management. The most effective SOA deployments will factor in all of these controls. Without these controls harnessing an enterprise SOA, the enterprise IT equivalent of the Wild Wild West will surely result.

About the Author

Andrew Lawlor is president and chief Architect of S.O. Solutions Inc. (www.sosolutionsinc.com), a Washington, DC, area consultancy concentrating on the application of service-oriented architectures. He has specialized in the integration, Web services and SOA technologies for over five years while assisting a numerous Fortune 500 and government clients. Most recently Andrew assisted the DoD's NetCentric Enterprise Services (NCES) project develop its cutting-edge SOA infrastructure.

adlawlor@sosolutionsinc.com

JAVA J2EE-Ready Managed Dedicated Hosting Plans:

Xeon III

SAMEDAY SETUP

Dual 3.2 GHz Xeons
4GB RAM
Dual 73GB SCSI
1U Server
Firewall
Linux
Monitoring
NGASI Manager

$399 *monthly*

Pentium 4 I

SAMEDAY SETUP
FREE SETUP

2.4 GHz P4
2GB RAM
Dual 80GB ATA
1U Server
Firewall
Linux
Monitoring
NGASI Manager

$199 *monthly*
2nd month FREE

4Balance I

1 Database Server and 2 Application Servers connected to 1 dedicated load balancing device.
Dual Xeons.
High-Availability.

$1724 *monthly*

NEW! Geronimo 1.0 Hosting . Virtual Private Servers(VPS) starting at $69/month

At **WebAppCabaret** we specialize in **JAVA J2EE Hosting**, featuring **managed dedicated servers** preloaded with most open source JAVA technologies.
PRELOADED WITH:
JDK1.4 . JDK1.5 . Tomcat . Geronimo . JBoss . Struts . ANT . Spring . Hibernate
Apache . MySQL . PostgreSQL . Portals . CRM . CMS . Blogs . Frameworks
All easily manage via a web based control panel.
Details:
-All Servers installed with the latest Enterprise Linux
-Firewall Protection
-Up to 60 GB daily on site backup included at no extra charge per server.
-Database on site backup every 2 hours
-Daily off site database backup
-A spare server is always available in case one of the server goes down
-Intrusion detection.
-24x7 Server and application monitoring with automatic self healing
-The Latest Bug fixes and Security updates.
-Tier 1 Data Center. 100% Network Uptime Guarantee
-Guaranteed Reliability backed by industry-leading Service Level Agreements

Log on now at **http://www.webappcabaret.com/jdj.jsp** or call today at **1.866.256.7973**

NGASI POWERED

WebAppCabaret sm

JAVA J2EE Hosting

Beware of Shortcuts on the Road to a Service-Oriented Architecture

How to achieve effective SOA implementation

The concept of a Service Oriented Architecture (SOA) existed long before the current set of Web Services standards. However, it's the widespread adoption of these standards that has enabled the idea of SOA to enter the mainstream and to start delivering the level of connectivity and savings it has promised for so long. Now that SOA has hit the mainstream, some are attempting to show how SOA can be successfully implemented using pre-Web Services technologies. This article will show why these approaches fail to fulfil all aspects of SOA and become exercises in rediscovering why SOA depends on Web Services technology.

WRITTEN BY
JAMES PASLEY

Before looking at some of the shortcuts that people take, let's start with a quick recap of the core principals of SOA. SOA is an approach to software development that is most commonly used where a number of distributed systems have to be connected. SOA emphasizes the use of standards to achieve interoperability and the reuse of existing assets. One aspect of SOA that is often overlooked is the use of a standardized protocol that provides a message envelope to enable the processing of messages by intermediaries. Of course, at the heart of SOA is the concept of the service, which should perform a specific task and be self-contained. It should be invoked remotely, have a well-defined interface, and be loosely coupled.

The As-You-Were Approach

The adoption of a new architecture requires some due diligence, part of which must be a comparison of it with existing practices and architectures. At this point, finding elements of SOA already in use is good news for everyone. However, this doesn't mean that no change is required and that the existing system just needs to be described in SOA terminology.

Consider what might be uncovered in an organization during an analysis prior to introducing SOA. Most large organizations have an existing queue-based transport infrastructure (if not several of them) that is typically used to support a variety of point-to-point integrations between applications. The good news is that these systems exhibit many of the characteristics of loosely coupled services: they perform specific tasks, exchange messages asynchronously, and can be accessed remotely via message queues. Most message formats are text-based. However, they've been documented as part of the design process.

It might appear that there's little left to do to roll out a SOA in an organization — perhaps all that's needed is to standardize a few design practices, such as getting people to think about the kinds of tasks these systems perform. Sections of the design documents can be published in a new document repository so that the message formats can be made public. Finally, organizations can standardize on the use of a single middleware vendor to provide the queue-based transport. They may have cut a few corners in using Web Services standards,

but most of the SOA principles are covered.

The problem with this approach is that while the owners of the existing systems feel good about their architecture, the tasks required in enabling a new application to act as a client of such a service haven't changed. The client developers must start with reference documents and write code that constructs the appropriate message formats. This code might already exist in the service, but sharing it exposes the service implementers to all the issues associated with software distribution. It also assumes that the clients are implemented in the same programming language. The client applications have to make use of the messaging transport from the same vendor as chosen by the service implementers. Even if it were logical to standardize on a single vendor across the entire organization, this kind of vendor lock-in isn't necessarily a good thing.

The Too-Clever-by-Half Approach

Another approach is the demo that seeks to show how a SOA can be constructed using nothing more than the basic features provided by a programming language. For example, a Java application is developed that discovers all its components using JNDI. Each component implements an identical interface that provides methods that pass a java.util.HashMap as a parameter. This "loosely coupled" interface enables new data to be passed across APIs without recompiling.

Again, a number of SOA principals have been applied in this approach. However, it demonstrates the development of a new application in a single environment. These may be useful programming practices, but they're not solving integration problems, and they're limited to the microcosm of a single application.

Doing It Right

Each of these approaches set out to follow SOA principles, but fell short. The result is an architecture that is limited to a particular environment. In some cases, these limitations can be justified by specifying the limited set of circumstances in which the applications will be reused. However, if there's one thing that can be said without doubt, it's that over time, applications will be reused in ways that the original designer never imagined. Fundamentally, these approaches fail because they simply improve on, rather than replace, existing practices when it comes to solving integration. The result is a failure to deliver on projected cost reductions.

Where two distinct systems have to be connected, work must be done to solve the integration issues. A key issue is the question of where this work is done. The first example represents an exercise in tidying up an existing system and exposing it as is to clients. This pushes the integration work onto the client side. As a result, cost is encountered every time the system is to be reused. In an environment in which services are to be reused, any cost expended on the server side will be incurred only once. Doing it right requires the use of Web Services standards. The cost of solving integration issues is done on the server side when exposing the existing system as a well-designed Web Service. This will be recouped every time a new client reuses the service.

The most important Web Services standards are shown in Figure 1. There are a few ways in which these standards change integration practices and deliver on cost reduction.

The messaging-related standards are shown in blue. While Web Services messaging can operate on top of existing messaging infrastructure, it also represents a compelling alternative to proprietary messaging solutions at key points in an organization. These messaging protocols have been designed to work over the Internet to avoid having to choose one messaging solution for inside an organization and another outside. The work of the WS-I ensures that implementations of these standards are interoperable. Using these standards doesn't tie consumers of these services down to a single vendor. In fact, in large organizations there's growing acceptance that selecting a single vendor across the entire organization isn't possible. Even if it were, acquisitions would make such a practice an ongoing cost in terms of migrating the systems of the acquired companies.

The interfaces to services are described using WSDL and XML Schema. This provides an unambiguous description of the message formats. Tools support is provided for creating, validating, and parsing such messages, eliminating the need for developers of client applications to write code to construct the messages. It also eliminates the temptation for developers of the service to distribute sample client code, thus enabling them to avoid the support costs associated with software distribution.

WS-Policy provides a framework in which the Quality of Service (QoS) aspects of a service can be expressed declaratively. In particular, reliable messaging and security can

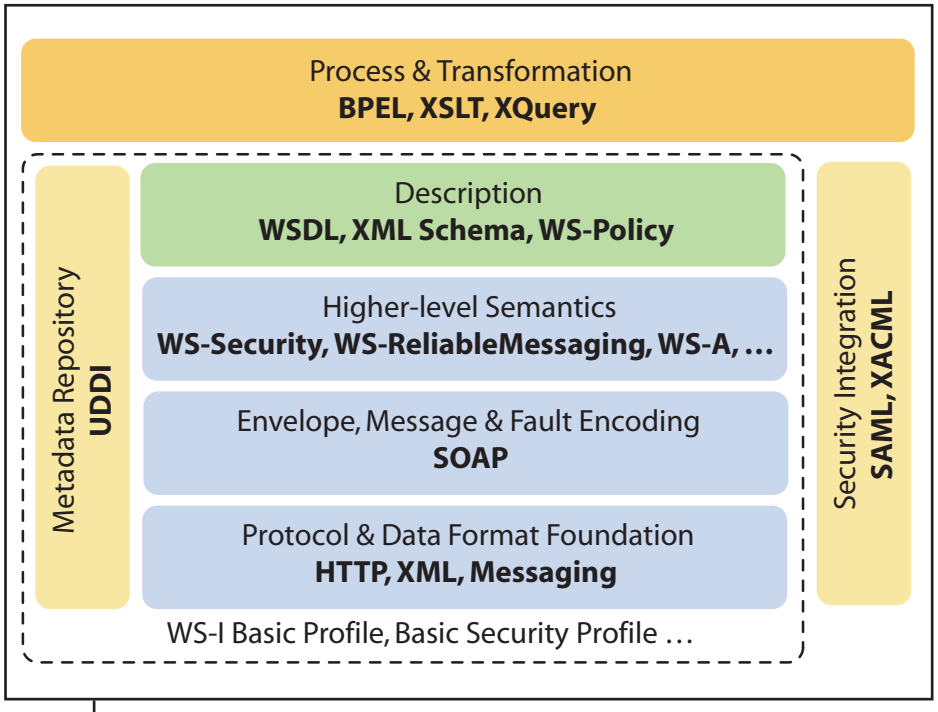


FIGURE 1 | The core Web Services standards

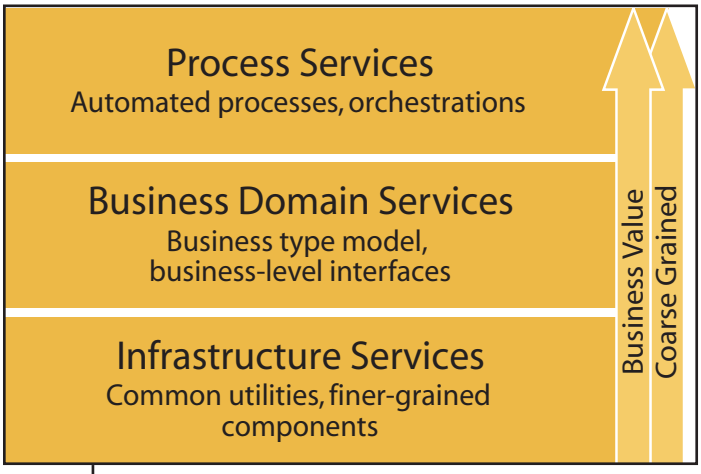


FIGURE 2 | **Service layers in an organization**

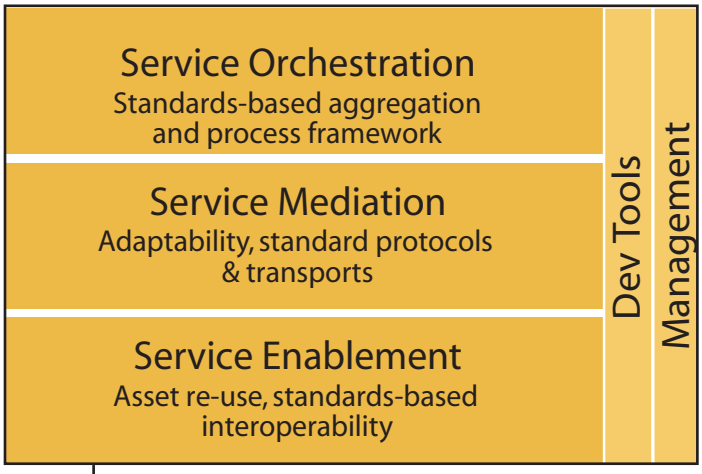


FIGURE 3 | **Stages in the creation of a SOA**

be expressed this way. This turns what may have previously been development tasks into configuration tasks.

The Benefits of a True SOA

Using these standards ensures that integration issues are addressed once by the creators of the service, thus radically reducing the cost of reusing the services. It also makes the fewest assumptions about how the services will be reused. This enables the use of technologies such as BPEL to model business processes through the reuse of services. This is where SOA can deliver the greatest flexibility.

SOA advocates the reuse of existing assets to create new services. Figure 2 illustrates how to build on the benefits of reuse. The first tier of Web Services will solve many of the integration issues and provide a basis for adding value. However, they may still reflect some of the structure or function of the IT infrastructure rather than the business need. From these infrastructure services, new business services can be created that reflect the services performed by the business. It's these services that should be exposed for reuse by an organization. The interfaces and function of these services remain stable over time, because they reflect the core business functionality. For example, a bank has customers and provides them with accounts. In the history of banking the concept of accounts and the movement of money between them has changed very little, while the underlying infrastructure that supports it has changed radically.

Once created, this appropriate set of business services can be combined with each other and with new services to model business processes. This is where agility is of the greatest value. In the banking example the business service that provides a customer with a loan can be used in the bank by tellers or from a call center to provide telephone banking. Through orchestration, it can also now be used within a greater process that uses information about the purpose of the loan to sell car or holiday insurance to customers.

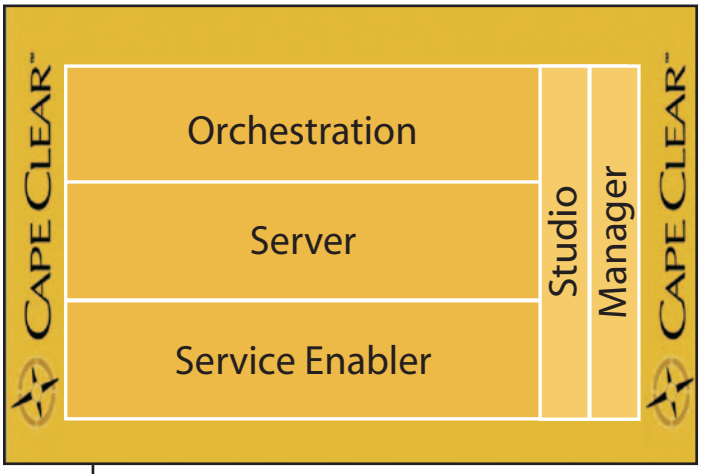


FIGURE 4 | **The Cape Clear Enterprise Service Bus**

So Why Do They Do It?

There are a number of factors that motivate people to create something that is less that a true SOA. The as-you-were approach isolates the owners of the existing systems from exposure to new technology. It may also appear to cause them less upfront work during the integration phase of a project, as more work is pushed onto the client side.

However, in many cases the motivation is the perceived difficulty of supporting all the principles of SOA. This is the result of a failure to select the appropriate products to support the creation of a Service Oriented Architecture. The Enterprise Service Bus provides the necessary tools and servers to support the creation of a SOA based on Web Services standards. This will not only ensure that the principles of SOA are followed, but that the goals of reuse and cost reduction are achieved.

About the Author

As chief technology officer, James Pasley is Cape Clear Software's lead technologist responsible for ensuring that the company's enterprise service bus (ESB) technology supports the evolving needs of its global customer base. Pasley joined Cape Clear in 2001 as a lead developer for Cape Studio and is now a technology visionary boasting several years of experience implementing messaging and middleware standards. In 2003, he was appointed chief architect and oversaw the development of Cape Clear's ESB. Prior to joining the company, Pasley worked for Siemens Nixdorf developing secure X.400 messaging and public key infrastructure (PKI) solutions for a range of military and civilian products. James also served as a primary developer of the CORBA conformance test suite for the Object Management Group (OMG). He has a computer science degree from Trinity College, Dublin.

james.pasley@capeclear.com

Don't Miss the 2006 JavaOne℠ Conference

More than 300 technical sessions and Birds-of-a-Feather sessions will be offered at the 11th Annual JavaOne℠ Conference. See and hear from the Industry leaders and technology experts over four content-rich days. Included are:*

PLATINUM COSPONSORS

GOLD COSPONSORS

SILVER COSPONSORS

Introduction to AJAX

Ben Galbraith, *Consultant* | Dion Almaer, *Adigio, Inc*

This session provides an introduction to AJAX and an orientation to the state of the AJAXian universe. It demonstrates the basic AJAXian techniques through live coding and demonstrates and deconstructs more-advanced examples of AJAX.

Effective Java™ Reloaded

Joshua Bloch, *Google, Inc.*

It has been five years since *Effective Java*™ was released. The Java platform has evolved, and we've learned more about how to use it to best effect. Therefore, a second edition of *Effective Java*™ is being released to coincide with the 2006 JavaOne conference. This presentation covers new material that has been added to the second edition, material that should be useful to every working Java technology programmer.

Spring Framework Update

Rod Johnson, *Interface21*

Rod Johnson, the father of Spring, brings attendees up to date on some of the many significant new features in the Spring 1.3 and 1.4 releases. He discusses what's new and cool in the Spring world and examines the implications of these new features for best practice in developing applications with the Spring Framework. Johnson shows code examples throughout the presentation, leaving attendees ready to try these features out for themselves.

To see more information on the conference offerings visit java.sun.com/javaone/sf

*Content subject to change.

Register by April 14, 2006 and SAVE $100.

java.sun.com/javaone/sf

JavaOne™ Conference | May 16-19, 2006

JavaOne℠ Pavilion: May 16–18, 2006, Moscone Center, San Francisco, CA

JavaOne
Sun's 2006 Worldwide Java Developer Conference

Corporate Applications on a Mobile Device Near You

Web services as the *lingua franca* of the mobile enterprise

In the enterprise building mobile applications is as much about integration and the corresponding challenges as it is about pure application development.

WRITTEN BY
ROY MITCHELL

Recent industry reports reveal that more than 70% of mission-critical data and most of the pivotal business logic that runs worldwide commerce still resides on existing host systems. Based on this dependency, as well as their speed and power, host systems are unquestionably here to stay for most large organizations and will continue to be a foundation for business success as organizations design and implement new business initiatives.

However, many of those same organizations have also invested in packaged applications (SAP, Siebel, Oracle Applications, PeopleSoft, etc.) to manage their businesses. When building mobile solutions that leverage these systems, enterprise developers want to use the existing business logic of these applications (to ensure data integrity, security, etc.) and not have to recreate this logic in new systems.

For many organizations a mobile enterprise solution will need to expose business functions from numerous core business systems in a single, seamless, easy-to-use interface, delivering real business value.

That core business functionality is housed on multiple platforms, including structured data such as databases and unstructured data such as host, ERP, CRM applications and middleware. Without a comprehensive mobile platform with built-in connectivity to these core systems, delivering mobile solutions becomes very difficult.

Therefore, in the enterprise, mobile applications are an extension of the company's existing systems and processes and are very rarely standalone applications in their own right. They are about allowing a mobile workforce to have real-time and continuous access to corporate business processes and information.

Many mobile enterprise applications require real-time access to multiple existing business systems. For example, to provide the sales team with a complete view of the customer, the mobile application might need to interact with the corporate SAP system for sales information, the Siebel system for customer care issues and the mainframe for customer records. To be practical, (limited real estate, data bandwidth, etc.) the mobile enterprise solution should only provide the specific data required for the task and not try to be general-purpose.

As such, mobile integration follows the same principles and tenets as any other integration issue in the corporation. Today's guiding principle for integration is Service Oriented Architecture (SOA) and its major delivery mechanisms XML and Web Services. SOA promises to decouple the end devices and their operating environments from the integration of mobile services with corporate applications. The adoption of XML and Web Services will help this come about.

Today, the consensus supports the adoption of mobile middleware. It's clear that off-the-shelf integrated application adapters combined with the capability to define and coordinate transactions across multiple back-end corporate applications (with process management) is the recipe for delivering corporate business processes and applications to mobile devices.

Mobile Computing

The spike in laptop, notebook, and tablet-type devices over the last few years underscores the demand for information that travels with people as they move around. These devices have become essential tools for many different types of mobile workers, from sales people to service technicians, who need to access and enter data electronically.

In recent years, the market for small handheld computers such as Personal Digital Assistants (PDAs) has grown rapidly and trends are pointing toward exploiting mobile technology even more. Research shows that while laptops are still the mobile device of companies, smaller handheld devices are being used to deliver mobile e-mail to field personnel. The RIM Blackberry has been at the forefront of this new age of mobile communication, but other vendors are able to deliver mobile e-mail on other handheld devices such as those using

> "Companies can get a competitive advantage using mobile technology to deliver crucial corporate information to employees, partners, and customers wherever they're located"

LinuxWorld® Conference & Expo

Conferences: April 3 – 6, 2006
Expo: April 4 – 6, 2006

Introducing: OpenSolutions World™

Boston Convention & Exposition Center

BOSTON'06

2 Conferences
10 Tracks
100+ Sessions
Over 200 Exhibitors

LinuxWorld Conference tracks

- *Mobile & Embedded Linux*
- *Network Management*
- *Kernel and Driver Development*
- *Linux on the Desktop*

OpenSolutions World Conference tracks

- *Business Case*
- *Security*
- *Open Source Applications*
- *Enterprise Application Development*
- *Dynamic IT: Cluster/Grid/Virtualization*
- *Emerging Trends*

Open source means business.

Find out what it can mean for your business at LinuxWorld Conference & Expo—in ten tracks and more than 100 in-depth sessions, tutorials, and workshops led by the open source movement's top minds.

Go beyond Linux.

Expand your vision at OpenSolutions World, a new conference at LinuxWorld covering the full spectrum of open source solutions and strategies for the enterprise.

Get the full picture.

Explore a comprehensive exhibition of open source products, technologies, and services from more than 200 key Linux and open source vendors—all under one roof.

Put it to work.

Come to LinuxWorld Conference & Expo April 3 – 6, 2006, and harness the power of the entire open source community for your business. The future of enterprise technology is wide open.

Register by 3/3 and save
Enter priority code D0302 and save up to $500
www.linuxworldexpo.com/boston

IDG International Data Group
IDG World Expo

D0302

> The enabling technology for the new mobile economy will be Web Services under the umbrella of SOA

the Microsoft Windows Mobile and Symbian platforms. In addition, many employees have purchased handheld devices and are looking to leverage these devices in their business lives. Many companies now face the challenge of coming up with a strategy that turns PDAs into useful productivity and communications devices. These companies face an equal challenge due to the rapid evolution of PDA technology – much faster than conventional PCs and laptops.

Nowadays most of these devices come loaded with more than one wireless option for data communications including Bluetooth, Wi-Fi, General Packet Radio Service (GPRS), and other 3G network technologies such as EDGE. All of these devices come with browser support, which in most cases, goes beyond support for Wireless Application Protocol (WAP) to support standard HTML-based Web sites. Most of these modern devices also support development frameworks, such as J2ME and the Microsoft .NET Compact Framework, which creates new opportunities to develop solutions beyond mobile e-mail. With this, the availability of literally thousands of Wi-Fi hotspots has meant that a combination of 3G GPRS and Wi-Fi can realistically be used to deliver the mobile infrastructure needed by the enterprise.

The real opportunity for companies is to move beyond mobile e-mail and begin to leverage the millions of mobile devices already in the hands of salespeople, service representatives, and customers, delivering corporate information directly to the point where it's needed.

The Mobile Internet – Reality versus Hype

In the past the media – with plenty of eager help from overzealous marketing campaigns – has generated a massive amount of hyperbole about the potential of the mobile Internet. By confusing "Internet-capable" with "can do everything that the wired Internet can do," the mobile Internet has been over-sold. Many companies' inflated expectations have been dashed by the realities of yesterday's first-generation implementation of the mobile Internet.

If we look back to the year 2000, it was estimated that by 2004 the number of Internet-capable cell phones being shipped and used would outstrip – by a large margin – those that didn't have a data communications capability. That milestone was reached in half the time.

According to Forrester Research in 2005 mobile networks covered 80% of the world's population, which means that more than five billion people are within range of a cellular network. And a quarter of the world's population – some one-and-a-half billion people – use mobile services. In addition, 78% of those users are connected to GSM networks, with the greatest penetration of mobile use in Europe followed by the United States.

IDC has estimated that more than 55 million mobile devices were sold in 2005, a boost of 165% over 2004. In other surveys IDC has said that approximately 40% of these devices will possibly require middleware in an enterprise environment. It's estimated that there are tens of millions of phones capable of running J2ME applications – all of which can integrate with the enterprise through their HTTP and Web capabilities.

Gartner, the research company, estimates that mobile terminal sales is likely to reach 848 million by 2008. Gartner also estimate that in the third quarter of 2005 shipments of Smartphones exceeded 12 million units. World Smartphone shipments surpassed those of PDAs for the first time in 3Q04 by 1.2 million units, and in 3Q05 this gap widened to nine million units.

Lessons Learned

Mobile Internet pioneers have learned some important lessons:

1. The mobile Internet is not suitable for all of the same applications you find on the wired Internet. The key is choosing applications that fit in the screen space and speed constraints of today's mobile devices. Applications that provide on-the-spot order entry, sales force automation, customer lookups, and equipment service are all excellent candidates for mobile Internet use.
2. Mobile Internet technology, at all levels, is moving very fast – handsets, network speeds, interoperability standards, and protocols are all evolving rapidly. Mobile phone manufacturers, network infrastructure players, and mobile network providers are investing massive amounts of resources in developing and deploying the next-generation networks. Today's devices have bigger full-color screens as well as faster networks that enable even more types of applications and services to go mobile.
3. Mobile network technology is not a North American technology, or European, or is it GSM-specific or tied to any transport network or technology. Mobile applications mask all the local differences and enable worldwide deployment of mobile applications using virtually any handset and wireless network.

Many corporations have started to deploy mobile applications to small projects and a substantial number in experimental or pilot stages. The types of mobile applications being deployed range from packaged mobile applications, used to deliver solutions for specific groups (such a field service), to custom mobile applications where they need access to multiple corporate systems in a singe solution.

When it comes to the benefits of mobile technology in the enterprise, most companies cite better employee productivity as the major gain, with the advantages of real-time access to corporate information a close second. The more information people have at their disposal, the better decisions they will make.

Forrester defined mobile enterprise technology as a set of technologies – includ-

ing networks, infrastructure, and portable devices – that enable employees and systems to use applications in a mobile environment. Key deliverables from mobile-enabling corporate applications include a boost in customer service, an increase in productivity, and a reduction in costs.

The concept of multi-channel and multi-modal applications means that corporate applications should inherently be able to support access from any device. Of course, in reality this is not the case and won't be for quite some while. In the meantime, mobile application middleware will deliver solutions that allow both legacy custom-developed applications and contemporary packaged applications to be accessed from handheld devices from a multiplicity of vendors. Obviously, any mobile middleware solution should have built-in support for as many enterprise applications, old and new, as possible.

Back in 2004 Gartner said that companies should plan for a widening range of mobile application platforms to become available. It advised companies to plan on mobile applications becoming a strategic part of their IT portfolio and not just tactical solutions.

Gartner estimated that while only 5% of a company's IT budget was spent on mobile in 2003, that percentage would double by 2007 with most of the spending focused on middleware with multi-channel access capabilities. That is a significant investment in moving organizations to the mobile enterprise. Giga, now part of Forrester, in assessing mobile application development in 2004 cited choosing middleware with versatility and flexibility as being uppermost in the selection criteria (with a focus on Web Services and strong connectivity to existing applications and a good development framework being key).

Where Is WAP?

A mobile phone used to be considered "Internet-capable" when it had a specialized piece of software called a *WAP browser* on it. Even today, most cellular providers still offer an Internet service that uses the capabilities of the WAP browser (often called a *micro-browser*), even though the capabilities of the phone and its form factor have changed considerably.

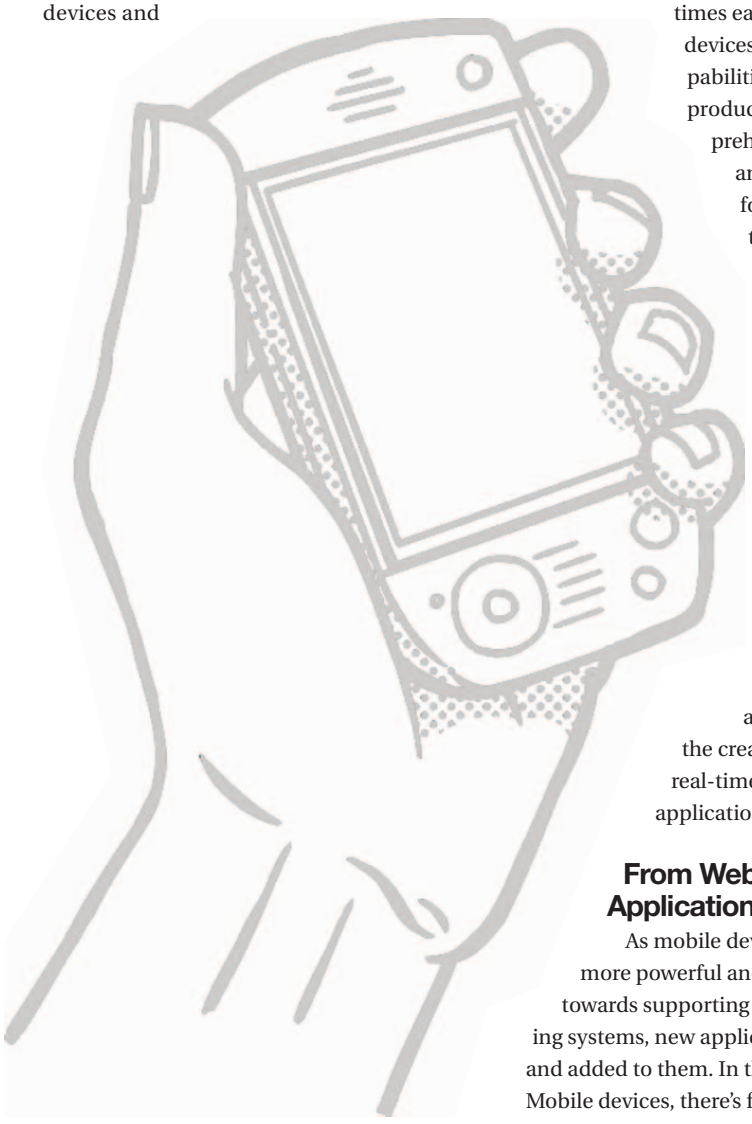
Although it's called a *browser* and although it can be useful to think of a WAP-based site as a "minimalist Web page," the screen and keypad of a mobile phone present very different challenges for the application designer than do the rich output and input capabilities of a modern Web browser. Existing Web sites – with their frames and graphical navigation aids – can't be translated easily into the menu-oriented approach of the WAP browser either.

From WAP to Web

The latest round of mobile devices offers much more capability than the simple cell phone with a micro-browser. Whether the

device is based on Palm, Microsoft Windows Mobile, Symbian, or Java technology, the display capabilities have improved dramatically and the means of interacting with the device have also been extended – from touch screens to thumb-operated keypads. The human/machine interface is significantly better than the original simple monochrome WAP microbrowser, which had a very small text-only screen. In addition to user interface improvements, these devices now have fully capable Web browsing capabilities and support many of the standards one expects to find on a desktop or laptop browser, such as Internet Explorer, Mozilla Firefox, or Opera.

As with WAP technology, there are two elements to Web-based support on mobile devices and



dards here are the standards of the Internet. The underlying communications technology is IP and the underlying network is the Internet, in much the same way that it is for desktops and laptops. The end-user application is a Web browser capable of rendering text and graphical pages from data delivered in HTML and XML or their mobile equivalents. As such, very complex interactive application presentations are possible, bringing the mobile corporate enterprise one step closer to reality through the mobile Web. However, these mobile browsers still don't provide the full functionality of a desktop Web browser, presenting some limitations on the full Web experience.

Therefore, mobile-enabling corporate information is many times easier with smart mobile devices with full browser capabilities. More importantly, products that provide a comprehensive development and deployment platform, while supporting the creation of new corporate applications and transactions that can be Web-enabled for PDAs and Smartphones, are critical to making this a reality. It's also possible to have a product like this extract and *re-present* transactions from a corporate application, enabling the creation and delivery of real-time mobile enterprise applications.

From Web to Mobile Application

As mobile devices have become more powerful and capable and move towards supporting programmable operating systems, new applications can be built and added to them. In the case of Windows Mobile devices, there's full support for the Microsoft programming environment. The Compact .NET Framework is implemented on PDAs, Phone Edition PDAs, and Smartphones. This has helped Microsoft gain a leading 48%t market share (and the greatest growth rate) in PDA shipments as its dominance of the enterprise market for PDAs continues. In Western Europe, that market share is a record 70%. In the case of Palm devices and many phones based on the Symbian operating system (such as Nokia and Sony-Ericsson), there's support for J2ME, which lets Java programs be written for and run on even generic cell phones. Delivery of these applications to the mobile device can be done automatically through a capability called Over-The-Air provisioning (OTA), where delivery and installation is done wirelessly through the cell network.

This programmability for mobile devices opens up opportunities for corporate use and integration with corporate applications. If the corporate programming environment extends down to the mobile device as with the Microsoft .NET Compact Framework (.NET CF), it's possible to develop applications to run on the mobile device that have the look-and-feel of a mobile application but are interacting seamlessly with corporate assets over GPRS or Wi-Fi networks. For example, a GPRS wireless PDA application can be built that interacts with a mobile middleware solution, which in turn interacts with the mainframe legacy system or contemporary systems such as SAP.

An example of this type of application would be in customer relationship management where a mobile application on a wireless PDA lets the salesperson on the road access customer contact information. An application that looks and behaves like a PDA application running locally on the PDA would let the salesperson tap the PDA's touch-sensitive screen to input customer or company names to retrieve contact details. In the background, the application running on the PDA uses the GPRS network to communicate with a mobile middleware server, which communicates with an IBM iSeries system that runs the customer database. Information from the iSeries system is passed back completely transparently by the middleware server to the PDA application. To the salesperson it looks as though

the whole sequence has run locally on the PDA. The salesperson has simply tapped the screen and gotten the information he or she was looking for. The key standard delivering this transparency for future mobile applications will be Web Services.

From Mobile Web to Mobile Web Services

The enabling technology for the new mobile economy will be Web Services under the umbrella of SOA. The principles of SOA aren't new. Existing client/server systems, such as DCE, CORBA, and DCOM, have delivered distributed applications for quite some time – some more successfully than others. The key difference with SOA is that the intercommunication technology is based on standards manifested through Web Services, which has the advantage that information providers can be developed completely independently of information consumers.

In a corporate environment the information providers will be the existing applications that drive the business. These will be legacy IBM mainframe apps or more modern contemporary applications like SAP or, more likely, a mixture of many applications across many platforms. For these applications to provide information in a standardized manner, an integration layer must exist between the "old" applications and the "new" Internet-enabled mobile economy. So the challenge is to Web-enable or mobile-enable existing corporate assets so that transactions within them (or entire business processes that span more than one application) can be presented as services to be consumed by other applications and systems.

The information consumers in the mobile economy are the mobile devices. Far from just presenting static information like WAP card decks or Web pages, these mobile devices are fully programmable interactive systems with applications built for them by conventional development environments like Microsoft Visual Studio. In today's world applications for mobile devices can consume Web Services and work easily with any corporate systems.

Mobile Middleware Platforms

When building enterprise mobile solutions, it's critical that any mobile middleware platform selected be capable of providing real-time access to back-end business systems as well as having the tools necessary to encapsulate specific business functions from those systems as Web Services to be leveraged in mobile solutions.

With the limited real estate available on many devices, mobile solutions only have to expose the functions of back-end business systems that are critical for the applications to use; there's little value in duplicating an existing application UI on a mobile device.

To realize the value from mobile solutions, the mobile middleware platform must provide a wide array of business systems connectors out-of –the-box and should include connectivity to legacy systems since this is a core business platform for most large organizations.

That platform must enable mobile applications developers to leverage the existing business logic of these applications (to ensure data integrity, security, etc) and not have to re-create this in new systems. However, many of the current mobile middleware solutions available today have roots in database synchronization with infrastructures based around this technology. However, unstructured data (most business information) by its nature doesn't lend itself to a database-centric solution, so a new approach is needed to deliver true real-time enterprise mobility.

Summary

Companies have an excellent opportunity to improve their competitive advantage using mobile technology to deliver crucial corporate information to employees, partners, and customers wherever they're located. The "go anywhere" coverage of mobile networks plus the ever-growing deployment of millions of Internet-enabled mobile phones and PDAs makes the mobile enterprise a viable and cost-effective proposition.

Web Services are the *lingua franca* of the mobile enterprise. They are also the common denominator between business systems, the Java world, and the .NET world. The more a company's application assets are Web-enabled and its business processes presented as Web Services (or perhaps more accurately as *business services*), the more those company assets can be used to deliver high-value mobile solutions easily.

About the Author

Roy Mitchell is a senior product manager at NetManage (www.netmanage.com), a software company that provides solutions for accessing, Web enabling, and integrating enterprise information systems. Mitchell has more than 10 years of product management experience in enterprise software solutions.

BY NOW THERE ISN'T A SOFTWARE DEVELOPER ON EARTH WHO ISN'T AWARE OF THE COLLECTION OF PROGRAMMING TECHNOLOGIES KNOWN AS AJAX!

REAL-WORLD AJAX SEMINAR

DoubleTree Hotel — **April 24, 2006** San Jose, CA

The Roosevelt Hotel — **June 5-6, 2006** New York, NY

AjaxWorld Conference & Expo — Rich Internet Applications — **Oct 9-10, 2006** San Francisco, CA

How, in concrete terms, can you take advantage in your own projects of this newly popular way of delivering online content to users without reloading an entire page?

How soon can you be monetizing AJAX?

This "Real-World AJAX" one-day seminar aims to answer these exact questions...

HURRY! LIMITED SEATING THIS SEMINAR WILL SELL-OUT CALL 201-802-3022 TO REGISTER!

Led by "The Father of AJAX" himself, the charismatic Jesse James Garrett of Adaptive Path, "Real-World AJAX" has one overriding organizing principle: its aim is to make sure that delegates fortunate enough to secure themselves a place – before all seats are sold out – will leave the seminar with a sufficient grasp of Asynchronous JavaScript and XML to enable them to build their first AJAX application on their own when they get back to their offices.

Jeremy Geelan
Conference Chair, Real-World AJAX
jeremy@sys-con.com

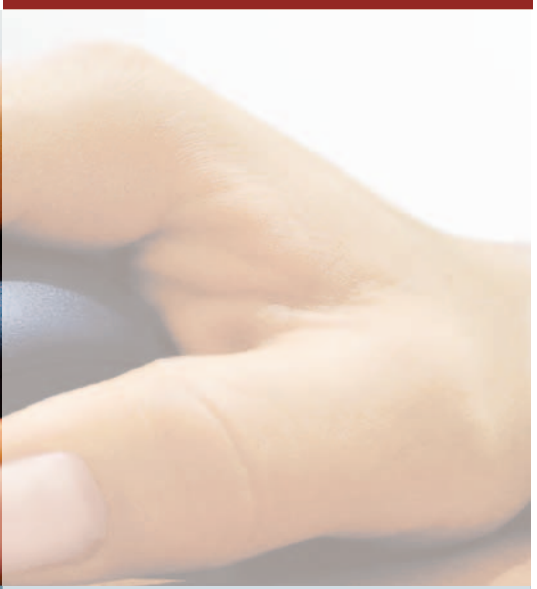
Register at...
www.AJAXSeminar.com

Before April 24, 2006 (San Jose, CA)	**$995**
(On-site Price: $1295) Call for group discounts	
Early Bird Price (New York, NY*)	**$1,295**
(Before March 31, 2006) Call for group discounts	
Discounted Price (New York, NY*)	**$1,495**
(After March 31, 2006) Call for group discounts	
On-Site Price (New York, NY*)	**$1,695**
(On June 5-6, 2006) Call for group discounts	

*Includes SOA Web Services Edge plus Enterprise Open Source Conference & Expo

LIVE SIMULCAST!
AROUND THE WORLD ON SYS-CON.TV

SPONSORED BY Adobe · telerik · BACKBASE · TIBCO / Perficient · Microsoft



What you'll walk away with...

A New understading of AJAX and how to make it work for you, plus:

A) **Conference Proceedings Binder**
Full-conference proceedings in a self-contained commemorative binder.

B) ***Ajax in Action* Book**
A copy of the Best-Selling AJAX book by Dave Crane.

C) ***Real-World AJAX "Secrets of the Masters"* Book + DVD**
New AJAX book edited by Dion Hinchcliffe (Release Date: Summer 2006)

D) **Notebook**
Executive style keeps you looking sharp for note-taking wherever your business might take you!

E) **Computer Back-Pack**
A hip new way to conceal your laptop! The Urban Wonder Compu-Pack is an ingenious creation - a 600-denier polycanvas and nylon knapsack!

F) **T-Shirt**
100% preshrunk 6.1 oz. heavyweight cotton tagless shirt features shoulder-to-shoulder tape, and double-needle sleeves and bottom.

G) **Porcelain Mug**
Lightweight and durable, you can use it as a teacup, coffee mug or keep it as a pencil holder.

H) **Satin Silver Contemporary Pen**
Satin-silver plastic barrel with the look and feel of brass. Rubber comfort grip improves writing control and comfort.

I) **Real-World AJAX Seminar on DVD**
Watch complete seminar video and slide presentations (Release Date: Summer 2006)

Featuring... Real-World AJAX Rock Stars!

Jesse James Garrett (San Jose April 24)
Father of "AJAX" Who Coined the Term in 2005

Jesse James Garrett is the Director of User Experience Strategy and a founding partner of Adaptive Path, the world's premier user experience consulting company. He is author of The Elements of User Experience (New Riders), and is recognized as a pioneer in the field of information architecture. Jesse's clients include AT&T, Intel, Crayola, Hewlett-Packard, Motorola, and National Public Radio. Since starting in the Internet industry in 1995, Jesse has had a hands-on role in almost every aspect of Web development, from interface design and programming to content development and high-level strategy. Today, information architects around the world depend on the tools and concepts he has developed, including the widely acclaimed "Elements of User Experience" model. He is co-founder of the Information Architecture Institute, the only professional organization dedicated to information architecture. He is also a frequent speaker and writer whose work has appeared in numerous publications, including New Architect, Digital Web, and Boxes and Arrows.

Adam Bosworth (San Jose April 24)
Vice President of Engineering, Google
One of the Fathers of XML & the Creator of MS Access

Adam Bosworth is Vice President of Engineering, Google. He joined Google in 2005 from BEA Systems, where he was Chief Architect & Senior VP of Advanced Development. Prior to joining BEA, Bosworth co-founded Crossgain, a software development firm acquired by BEA. Known as one of the pioneers of XML, he previously held various senior management positions at Microsoft, including General Manager of the WebData group, a team focused on defining and driving XML strategy. While at Microsoft he was also responsible for designing and delivering the Microsoft Access PC Database product and assembling and driving the team that developed the HTML engine of Internet Explorer 4.0.

Dion Hinchcliffe (San Jose April 24)
Cofounder & CTO, Sphere of Influence Inc.
Editor-in-Chief, Web 2.0 Journal

Dion Hinchcliffe, newly appointed editor-in-chief of SYS-CON's pioneering Web 2.0 Journal, is cofounder and chief technology officer for the enterprise architecture firm Sphere of Influence Inc., in McLean, Virginia. A veteran of software development, Dion works with leading-edge technologies to accelerate project schedules and raise the bar for software quality. He is highly experienced with enterprise technologies and he designs, consults, and writes prolifically. Dion actively consults with enterprise IT clients in the federal government and Fortune 1000. He is a frequent speaker on AJAX, Web 2.0 and SOA and is currently the top-read SYS-CON.com blogger.

Christophe Coenraets (San Jose April 24)
Senior Technical Evangelist, Adobe
AJAX/Flex Integration Guru

Christophe Coenraets currently works as a Senior Technical Evangelist at Adobe. Before joining Adobe, Christophe was an evangelist at Macromedia, focusing on Rich Internet Applications and Enterprise integration. Prior to Macromedia, Christophe was the head of Java and J2EE Technical Evangelism at Sybase, where he started working on Java Enterprise projects in 1996. Before joining Sybase in the US, Christophe held different positions at Powersoft in Belgium, including Principal Consultant for PowerBuilder, and Manager of the Professional Services organization. Before joining Powersoft, Christophe worked as a developer and architect on several retail and BPM projects. Christophe has been a regular speaker at conferences worldwide for the last 10 years.

Paul Rademacher (San Jose April 24)
Google, Creator of HousingMaps.com

Paul Rademacher is the creator of HousingMaps.com, which combined Craigslist and Google Maps for the first web mashup. Paul holds a Ph.D. in Computer Science from UNC-Chapel Hill, and worked as an R&D Engineer at Dreamworks Animation on such movies as Shrek 2 and Madagascar. Since creating HousingMaps, Paul is now at Google.

Jouk Pleiter (San Jose April 24)
Co-Founder & CEO of Backbase

Jouk Pleiter is the CEO of Backbase, a leader in the field of Rich Internet Applications and AJAX development software. Backbase's clients include ING, ABN AMRO, TNT, KPN, Comsys and Heineken. Backbase operates globally with offices in San Mateo (North America) and Amsterdam (Europe). Since 1995, Jouk has been an entrepreneur: he founded three successful software companies. Prior to Backbase, Jouk was part of the founding team at the web content management company Tridion, where he led the product management operations, and was driving the company's efforts to become a leader in the European WCM software market. Jouk previously was part of the founding team at the Interactive Agency Twinspark where he grew the company to a leading market position in Europe and was instrumental in the sale of Twinspark to Agency.com. He has an MBA from the University of Groningen.

Kevin Hakman (San Jose April 24)
Director of Product Marketing for TIBCO General Interface TIBCO Software

Kevin Hakman is the director of product marketing for TIBCO General Interface, the award winning AJAX and Rich Internet Application framework and toolkit. Kevin Hakman pioneered AJAX in the enterprise co-founding General Interface in 2001. Since that time General Interface (aka 'GI') has been powering Web applications that look, feel and perform like desktop applications, but run in the browser at Fortune 500 and U.S. Government organizations. General Interface was also the first to use its own toolkit to provide full visual tooling for AJAX when it released it's 2.0 Version in 2003. TIBCO acquired General Interface in 2004 to extend its vision for service oriented applications to the end user. Kevin is a contributor to the SOA Web Services Journal and the AJAX Developer's Journal.

Shanku Niyogi (San Jose April 24)
Product Unit Manager of the UI Framework and Services Team Microsoft Corporation

Shanku is Product Unit Manager of the UI Framework and Services (UiFX) team, which is responsible for delivering high-productivity UI framework technologies for the .NET platform, including ASP.NET, Atlas, Windows Forms, and frameworks for smart clients. Prior to his current role, Shanku was Group Program Manager of the Web Platform and Tools team on the Whidbey release of ASP.NET and Visual Web Developer. Shanku joined Microsoft in 1998 as a developer, having spent several years shipping products in the Windows ISV industry. Shanku holds a Bachelor of Mathematics degree in Computer Science from the University of Waterloo.

Coach Wei (New York June 5-6)
Chairman, Founder and CTO, Nexaweb
The Creator of First Commercial AJAX Applications

Coach Wei combines in-depth IT industry expertise with extensive education and research experience at MIT to drive technology innovation and business direction for Nexaweb. He founded Nexaweb in 2000 and served as CEO until summer 2003. Before founding Nexaweb, Coach architected and designed enterprise software for managing storage networks at EMC Corporation. As a graduate researcher at MIT, Coach developed software and hardware systems for non-destructive evaluation as well as signal/image processing algorithms. Coach was a finalist in the 1999 MIT $50K entrepreneurship competition and holds several U.S. patents. An accomplished writer and speaker, Coach has published numerous articles on topics including: AJAX, J2EE and .NET, RIA development, XML, signal/image processing, composite materials and ultrasonic imaging. He has spoken at top industry events, such as JavaOne and Web Services Edge. Coach holds an MS in information technology from MIT.

Ajit Jaokar (New York June 5-6)
CEO, futuretext
Author, "Mobile Web 2.0"

Ajit Jaokar, based in London (England), is the CEO of a publishing company, futuretext (www.futuretext.com). He is currently writing a book about Mobile Web 2.0 (Mobile Web 2.0: The Innovator's Guide to Developing and Marketing Next Generation Wireless / Mobile Applications). Ajit also chairs Oxford University's Next-Generation Mobile Applications Panel and, since January 2006, has been a member of the Web 2.0 Workgroup. In his "Real-World AJAX" conference session, Ajit will discuss the "AJAX Use in Mobile Applications" as part of the wider impact of Web 2.0, sometimes referred to as the "Global SOA."

Jonas Jakobi (New York June 5-6)
AJAX Evangelist and Co-Author, "Ajax and JSF: Friend or Foe?"
Jonas will autograph a copy of his book for all delegates!

Jonas Jacobi is a principal product manager and evangelist for Oracle's Java/J2EE tool offering, JDeveloper, and over the past three years has been responsible for JavaServer Faces, Oracle ADF Faces, and Oracle ADF Faces Rich Client development features within Oracle JDeveloper. Jonas has been in the software business for 15 years. Prior to joining Oracle, he worked at several software companies in Europe, covering many roles including support, consulting, development, and project team leadership. Jonas' new book "Ajax and JSF: Friend or Foe?" released by Apress on February 25, 2006.

John Fallows (New York June 5-6)
AJAX Evangelist and Co-Author, "Ajax and JSF: Friend or Foe?"
Jonas will autograph a copy of his book for all delegates!

John Fallows, former lead developer for Oracle ADF Faces Rich Client, has been working in distributed systems for over a decade. After five years spent focused on designing, developing the JavaServer Faces standard to provide AJAX functionality, playing a leading role in the Oracle ADF Faces team, he recently joined an AJAX start-up. Originally from Northern Ireland, John graduated from Cambridge University in the United Kingdom and has worked in the software industry for more than ten years. Prior to joining Oracle, he worked as a research scientist for British Telecommunications Plc.

Steve Benfield (New York June 5-6)
Well-known AJAX Evangelist and CTO of Agentis Software
Steve's first talk on "Aspect-Oriented Programming & AJAX"

Steve Benfield is CTO of Agentis Software and one of the pioneers of AJAX technology, a gifted writer and a technical visionary. A technology marketeer and strategist with 20 years of software entrepreneurism experience, a combination of qualities that made him the perfect choice of editor-in-chief for SYS-CON Media's inaugural publication 12 years ago. Steve's proven ability to determine marketing and technology strategies that align with market needs led to successful stints at SilverStream, where he started as technology evangelist and ended as CTO, and at ClearNova, an open source AJAX company, where he was CTO and AJAX evangelist.

Jeremy Geelan
Conference Chair
Group Publisher & Editorial Director, SYS-CON

Jeremy Geelan is group publisher and editorial director of SYS-CON Media, and is responsible for all print titles and online i-technology portals for the firm. He regularly hosts SYS-CON.TV, is executive producer of the "Power Panels with Jeremy Geelan" iTV series, and represents SYS-CON at conferences and trade shows, speaking to technology audiences both in North America and overseas. His i-Technology Blog is at jeremy.linuxworld.com and he is conference chair of the upcoming iTVcon - "Internet TV Conference & Expo 2006".

For more information...

Call 201-802-3022 or email events@sys-con.com

SYS-CON EVENTS **For more great events visit www.EVENTS.SYS-CON.com**

NOTE: SPEAKER LINE-UP SUBJECT TO CHANGE WITHOUT NOTICE
VISIT WWW.AJAXSEMINAR.COM FOR THE MOST COMPLETE UP-TO-DATE INFORMATION

Why Services Are Like Craigslist

And why the big software vendors will be singing the blues

I caught a review in Fast Company of an interview that Craig Newmark of Craigslist had with ABC's Nightline News. I didn't see the interview myself, but Fast Company did a good job highlighting the more important points, including the fact that Craigslist, which offer free classified ads, is killing the local newspapers.

WRITTEN BY
DAVID LINTHICUM

Get this:

"…There was the complaint that the site is hurting the newspaper business, stealing away those who would buy classified ads. According to Nightline, this shift has created an annual loss of $50 million in San Francisco alone."

Fast Company further says:

"Why must Newmark and Craigslist answer for the papers' failings? Craigslist has become one of the top Web sites in the world (three billion page views per month) as new features and additional cities have been added to the line-up. Users preach the virtues of the free classified service. I think Craigslist shows the value of presenting people a free service with wide capabilities (Google also comes to mind). If anything, traditional media should learn from such online innovators and adapt more quickly to the new landscape, rather than complain about it."

Exactly, and spot on! He who creates the best innovative and cost-effective service wins the game. This includes Google, eBay, Amazon, and, of course, Craigslist. Moreover, this is also very much like the battle brewing between the traditional software powerhouses and the up-and-coming Web Services marketplace, including guys such as StrikeIron, Salesforce.com's AppExchange, and NetSuite's NetFlex.

While most SaaS providers and service marketplaces don't give away business applications and services like Craigslist gives away classified ads, the number of free or cheap business services available on the Web is expanding rapidly. For instance, I'm not sure you can buy better or more cost-effective mapping software for a business than Google Maps or create an auction site better than eBay, and those services are free or inexpensive, and best of all they're on-demand.

> "The larger software houses are going to have to adapt to survive SaaS"

Moving up, you have guys like Saleforce.com and NetSuite who are providing world-class CRM applications as a service, using a subscription model that's only a fraction of the cost of traditional enterprise applications. Moreover, you have the emerging Web Services marketplaces that are providing application services that you can mix and match into composites using similar subscription models, again, at only a fraction of the price of buying complete packages or building those services yourself.

In addition, the notion of mash-ups is going to push this along even faster as we learn how to mix and match both content and services to create new abstractions for use in hybrid Web-delivered applications and in the enterprise through emerging SOAs. We could get to a point were most application services actually exist outside the enterprise and thus enjoy the resulting reduction in costs and the ability to leverage best-of-breed applications. This is the final destination of Web 2.0.

So, in the same way that the large newspapers are complaining about new free on-demand services such as Craigslist, it's clear that the larger software vendors are going to be singing the same blues and will have to adapt to survive as things move quickly to a SaaS model. In fact you may find that many of the major SaaS players are sucked up by the larger software guys to hobble this trend or to buy into this emerging market. Sometimes I wonder if they see this coming.

What's key here is that organizations can get better IT services through this trend and at a reduced cost of development and deployment. Moreover, as businesses change, the need to maintain both an architecture and a new paradigm is going to be critical, including developing their SOA as a platform for business processes and leveraging outside services (SaaS) to complete these processes.

About the Author

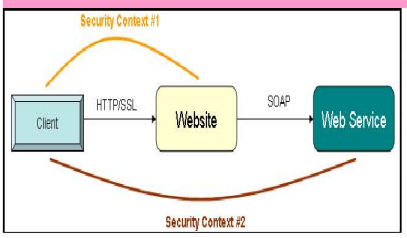
David S. Linthicum (www.davidlinthicum.com) is the author of three books on application integration and SOA, a frequent speaker at industry conferences, and the host of the "Service-Oriented Architecture Expert Podcast" (www.soaexpertpodcast.com).
linthicum@att.net

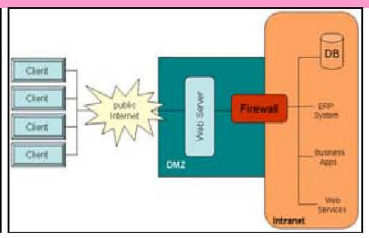
The 2006 Web Services Security Conference & Exhibition

May 25-26 Marriott Conference Center, University of Maryland, College Park, Maryland

THE INN AND CONFERENCE CENTER
UNIVERSITY OF MARYLAND UNIVERSITY COLLEGE
Marriott Conference Centers

Get the Details and Register at: http://www.unatek.com/conference.htm or Call 301-583-4629





XML Gateway
High performance for XML traffic

RSA SECURED · Ready for IBM Tivoli e-business software · nCIPHER READY · Entrust Ready

Keynotes and Power Panels' Topics Include: State-of-the-Art in Intrusion Prevention, XML Cutting-Edge Products, SOA Web Services, Single Sign On Authentication, Identity, Data Theft, Risk Management and Legal Issues, ***And Many More!!!***

ABOUT THE CONFERENCE

The 2006 Web Services Security Conference (WSSC) is the forum for senior executives, industry and government leaders, and legal and academic experts to present the state-of-the-art of online transactions security practice. With the mix of strong developer, infrastructure and messaging content, WSSC 2006 is the principal educational conference for all those who engage in online transactions and a technical training conference for a wide range of professionals.

In addition to the in-depth breakout sessions, attend a selection of Hands-on demos, Chalk-&-Talks, Panel Discussions and more!

WHO SHOULD ATTEND?

- Business and Corporate Executives, CIOs and CTOs, Financial and Banking Professionals, Government Representatives
- Technical Decision Makers, IT Managers, Network and System Administrators, Developers, Software Architects, Web Development Managers, Security Specialists, and anyone looking to stay current with the latest in Web services and online transactions security.

FUN AND INFORMATIVE NETWORKING EVENTS

- Birds-of-Feather Gatherings, Roundtables and Panels
- Expo Floor Opening Party & the Conference Party

POWER SPEAKERS

Melise R. Blakeslee
Partner, McDermott Will & Emery LLP

Dr. Anton Chuvakin
Chief Strategist, netForensics, Inc.

John Starke
VP, *TransGlobal Business Systems*

Kerry G. (Kwasi) Holman
President / CEO, PGCEDC

Dr. Charles Iheagwara
CTO, Unatek, Inc

Ralph Bazilio
President, TCBA

Mamoon Yunus
Founder and CTO, Forum Systems, Inc.

Andre Yee
President/CEO, NFR

Martin Roesch
Founder and CTO, Sourcefire, Inc.

Dr. Ephraim Feig
CTO, Kintera, Inc.

Donald Deboit
Director, Computer Associates

Register Sponsor Panels Speakers Agenda Exhibit Conference Call for paper Hotel Info

Sponsors: **Unatek** Forum Systems **netForensics** Transglobal SOA**WebServices**Journal + **More!**

Using Data Services to Build Functional Services

Data management infrastructure cuts costs and accelerates development

In almost every significant SOA deployment, a few services have advanced requirements that force those services to intelligently manage the data that they use. A fault-tolerant service might be deployed on a cluster of machines, which means that the instances of the service must share data across several machines.

WRITTEN BY
VIVEK SINGHAL

A high-performance service might use in-memory caching to achieve fast response time. An aggregation service might retrieve or update data from multiple sources, transform that data into a standard representation, and integrate the resulting data in real-time. For these kinds of services, the data management infrastructure can significantly reduce development effort and deployment risks.

In a SOA, services act as the building blocks for implementing business processes. Each service, hereafter called a functional service, offers a set of operations. The implementations of these operations usually involve querying and updating data from one or more data sources. Data services are the next layer in the SOA architecture. Data services support functional services by acting as a high-level abstraction for data: rather than directly exposing functional services to the complexity of data replication, data transformation, and data federation, data services hide those details and present a simple view of enterprise data. Data services expose high-level data manipulation operations, whereas functional services expose business domain-oriented operations. Numerous commercial data management infrastructure products are valuable for writing data services. These infrastructure products reduce the cost, improve the reliability, and accelerate the development of data services.

This article describes several categories of data management infrastructure that are available today that can be used to build data services. A case study from a major hotel chain illustrates how data services were used in a real-world SOA. We'll discuss the next generation of data management infrastructure, which blends independent data management capabilities to provide a cohesive platform to deliver consistent, reliable, and pervasive access to data.

Data Access Infrastructure

Enterprise data can originate from a variety of data sources. Relational databases are a common source. But data might also originate from non-relational databases (such as IMS or object-oriented databases), files (such as XML files), applications (such as other functional services), or mainframe resources. The challenge with heterogeneous data sources is that each type of data source offers a distinct interface for retrieving and updating data.

A data access infrastructure can simplify the use of heterogeneous data sources by providing a view of the data that's independent of the underlying data source type. For example, data might originate from relational databases or mainframe resources, but data services could present that data to functional services as Java objects. As data services use new types of data sources, there's little or no impact on the functional services. The functional service simply gets richer data from the data services.

Several vendors offer infrastructure for accessing data sources. For example, there are products that provide a JDBC interface to mainframe applications, so that data services can use a familiar interface to query or update the mainframe. Similarly, XQuery products make it easier to manipulate data stored in XML files. Data services can exploit these data access products to connect to a broad range of data source types.

Data Replication Infrastructure

While data sources serve as the definitive origin and repository for enterprise data, often a data service needs a local copy of the data. For example, if a data service needs to rapidly query data, it might use an in-memory cache to store a local copy of frequently used data. To recovery quickly from process failures, the data service might write a copy of the cache to disk, so that it can quickly reconstruct the cache without having to query the original data sources. If a data service needs to operate reliably even when a data source is unavailable, the data service could maintain a local copy of the data so that processing can continue even without the primary data source.

Making a local copy of data is easy. Ensuring that the copy stays up-to-date is more challenging. A data replication infrastructure can automate the replication of data so that both the initialization and subsequent updating of the local copy occurs automatically. A data replication infrastructure can offer a spectrum of "qualities of service" (QoS) that meets the varied requirements of data services. For example, a local copy might simply correspond to a snapshot of a data source that's periodically refreshed to reflect recent changes. Or the local copy might be continually updated via distributed transactions so that the copy is guaranteed to be identical to the original data source. Another QoS is whether the local copy is writeable or read only. Yet another QoS is whether the local copy is recoverable after process failure (because the local copy is backed up to disk) or not (because the local copy is stored in volatile memory).

Not only does replication infrastructure unburden a data service from the drudgery of synchronizing data, it also provides the data service with a high-level abstraction for managing data. The data service merely declares its QoS requirements and the replication infrastructure hides the complexity of synchronization in accordance with those requirements. These powerful abstractions accelerate the development of data services and they ensure that data services have reliable access to replicated data.

> "Future data management platforms have to integrate multiple technologies"

Data warehouse technology is commonly used for disk-based data replication. Traditional data warehouses use a batch-oriented approach to initialize and update the local copy. Real-time data warehouses support continuous incremental updates to the local copy, which means that the local copy is nearly synchronized with the original data source. Both traditional and real-time data warehouses produce a read-only local copy of data, which means that updates are disallowed because they aren't propagated back to the original data sources.

Distributed in-memory caching, which automatically synchronizes data across a group of high-speed caches, is another example of replication infrastructure. Each cache is typically deployed directly within an application process, which provides the application instant access to the cache's data. This infrastructure accelerates the performance of data services by provisioning data directly into the application address space, but with the limitation that the local copy isn't fault-tolerant.

Another data replication technology can provision data for disconnected mobile applications. This represents a powerful abstraction because a data service simply relies on the replication technology to manage the complexity of synchronizing data whenever a mobile computer is connected to the network. The data service is mostly unaware of whether the machine on which it's deployed is connected to the network, shifting the burden of data management to the replication infrastructure.

Data Integration Infrastructure

Occasionally, a data source stores data in exactly the format required by the data service that uses it. More often, the storage format doesn't match the requirements of the data service, so some data transformation is required. For example, suppose a data service that's implemented in an object-oriented programming language needs to use data from a relational database. Object-relational mapping is a standard infrastructure for performing this transformation. Object-relational mapping automates the translation of data from the rows and columns representation of a relational database to the instances and attributes representation of an object-oriented programming language. It also eliminates hand-coded database queries, optimizes the use of database connections, and offers a high-level abstraction for data manipulation.

But consider a more complex example. Suppose a data service needs to aggregate data from multiple relational databases. And suppose that the relational databases use different schema. In this example, not only must relational data be converted to an object representation, but the schema differences have to be reconciled too. Data originating from the different data sources has to be integrated into a common representation that's easy for the data service to use.

Infrastructure for data integration and data federation can address these challenges. For example, enterprise information integration products provide a data service with a tailored view of data, where the data originates from distributed data sources with different schema and data types. This infrastructure is valuable for building data services because it abstracts the underlying data sources and presents a unified data representation that's independent of the format and location of data.

BONUS TRACK ADDED! (FREE $1,295 VALUE!)
"Real-World AJAX" 2-Day Seminar

SOA WebServices & Enterprise OpenSource Conferences

The #1 i-Technology Educational and

Register at... www.SOAEOSConference.sys-con.com

The **10th International SOA Web Services Edge Conference**, colocated with the First Annual **Enterprise Open Source Conference**, will take place on **June 5-6, 2006** at the historical **Roosevelt Hotel in New York City**. SOA Web Services Edge and Enterprise Open Source Conference together deliver the #1 i-technology educational and networking opportunity of the year.

This year, more than 100 distinguished conference faculty members will present **96 cutting-edge SOA, Web Services and Enterprise Open Source topics** in educational classes, panels, and keynotes in six simultaneous tracks for two information-packed days, plus a bonus track, a **"Real-World AJAX"** two-day seminar, a $1,295 value!

The conference program does not merely present a comprehensive view of all the development and management aspects of integrating a **SOA strategy** and an **Open Source philosophy** into your enterprise, its organizing principle is that delegates will go away from the intense two-day program replete with why-to and how-to knowledge delivered first-hand by industry experts.

According to analyst firm Gartner Group, **by 2008 more than 60 percent of enterprises will use SOA** as the guiding principle when creating mission-critical applications and processes. **"Businesses that ignore the potential of SOA will find themselves outpaced by rivals who improve their agility and transform themselves into new kinds of enterprises,"** says Gartner analyst Yafim Natis.

Meanwhile, the **Open Source revolution** has spread into traditional areas of enterprise IT, with all major vendors and a global group of **Open Source entrepreneurs** developing the technology and delivering the solutions that are changing the way the world does business and the way governments serve their citizens.

Receive FREE WebCast Archives of all Entire Conference!

The best news for this year's conference delegates is that your "Golden Pass" registration now gives you full access to all 96 conference sessions. We will mail you the complete content from all the conference sessions in seven convenient DVDs after the live event takes place.

This on-demand archives set is sold separately for $995

Receive 8 Days Worth of Education F

Networking Opportunity of the Year!

SOA 10th International WebServices Edge conference

2006 ENTERPRISE> OPENSOURCE CONFERENCE

Early Bird Price (Before March 31, 2006)
Incl. Golden Pass to Both Conferences* $1,295

Discounted Price (After March 31, 2006)
Incl. Golden Pass to Both Conferences* $1,495

Conference Price (On June 5-6, 2006)
Incl. Golden Pass to Both Conferences* $1,695

*BONUS: "Real-World AJAX" Two-Day Seminar!

Attention Sponsors:
A Forum will be available to display leading Web services, OpenSource, and AJAX products, services, and solutions.

TOPICS INCLUDE...

SOA Web Services Edge:
- > Transitioning Successfully to SOA
- > ebXML
- > Orchestration
- > The Business Case for SOA
- > Interop & Standards
- > Web Services Management
- > Messaging Buses and SOA
- > SOBAs (Service-Oriented Business Apps)
- > Delivering ROI with SOA
- > Java & XML Web Services
- > Security
- > Systems Integration
- > Sarbanes-Oxley
- > Business Process Management

Enterprise Open Source:
- > Open Source Licenses
- > Open Source & E-Mail
- > Databases
- > ROI Case Studies
- > Open Source ERP & CRM
- > Open-Source SIP
- > Testing
- > Grid Computing
- > Open Source Middleware
- > LAMP Technologies
- > Professional Open Source
- > Open Source on the Desktop
- > Open Source & Sarbanes-Oxley
- > IP Management

AJAX User Experience:
- > Rich Internet Applications
- > Flash-AJAX Integration
- > Developing Interfaces
- > AJAX on Rails
- > Best Practices
- > Enterprise AJAX
- > Technology Demos
- > Lessons from the Frontline
- > Bringing Desktop Apps to the Web
- > AJAX Power Panel
- > and more...

SYS-CON EVENTS www.EVENTS.sys-con.com

Packed into 2 Information-Filled Days!

> New data management capabilities will further enable flexible and ubiquitous access to data

Case Study

A major hotel chain embarked on a large-scale effort to rebuild their reservations systems as an SOA. As part of that effort, they identified two critical functional services that required data services: Availability and Permissibility. The Availability service must rapidly respond to inquiries about room vacancies on particular dates at specific hotel properties. The Permissibility service evaluates whether reservations conform to rate plan rules. Both services are written in an object-oriented programming language, retrieve data from a relational database, support very high transaction volumes, and promise highly available operations.

To build the data services that supported the Availability and Permissibility functional services, they needed data infrastructure that solved several problems. There was a mismatch between the relational representation used to store data and the object-oriented index data structures used for computation. Data had to be synchronized across a cluster of processes that implemented each functional service. And the functional services required instant access to data to deliver fast responses to requests.

To build its data services, the hotel chain selected Progress ObjectStore Enterprise from Progress Software. ObjectStore acts as a high-performance distributed durable cache for object-oriented data. It offers transparent storage of object-oriented data structures; it delivers data automatically to distributed in-memory caches; it guarantees strong transactional consistency semantics; and it provides instant access to data.

One key challenge overcome by the data services was to transform data from a representation optimized for storage into a different representation optimized for computation. The relational representation of data was normalized and semantically complete, but very inconvenient to manipulate. To answer a typical query about room availability, an expensive multi-table join operation was required. In response, they developed an optimized index structure that contained the same semantic information as the relational database, but in a different format that could efficiently answer queries.

The initial construction of the index structure required a full scan of the relational database, a process that took several hours. Once the index structure was built, updates to the relational database required corresponding incremental updates to the index structure. It was mandatory to ensure that the index structure was backed up to disk so that a temporary failure of the data service wouldn't result in a lengthy outage to reconstruct the index.

The use of a data management infrastructure enabled the hotel chain to build data services that supported the operational requirements of the Availability and Permissibility functional services. While they contemplated building an in-house implementation of data management infrastructure, they quickly determined that it was quicker, more cost-effective, and less risky to buy a commercial infrastructure product instead.

Next-Generation Data Management Infrastructure

Today's data management infrastructure products tend to focus on solving individual data challenges. As these products mature, they must evolve into data management platforms that integrate multiple technologies. Such platforms will be capable of meeting the demands of tomorrow's sophisticated data services.

For example, in-memory caching, object-relational mapping, and data integration are all implemented by independent products. However, if a data service needs a fast object-oriented view of data that originates from multiple data sources, the data service is forced to manually orchestrate three data management infrastructure products. The complexity is worse if the materialized view must be backed up to disk for fault recovery, or if some data sources are non-relational, or if the data service wants to update the view of data.

The QoS model for data management represents a simple paradigm for exposing data management capabilities in a simple declarative manner. Rather than requiring a data services developer to separately procure and carefully orchestrate independent data management technologies, the developer merely declares the data services' QoS requirements to a data management platform. The platform is responsible for activating the appropriate technologies and ensuring that they're together.

New data management capabilities will further enable flexible and ubiquitous access to data. Innovative data replication technology will automatically coordinate multiple data sources so that data services can update any data source and the corresponding updates are replicated automatically with full transaction consistency. Emerging data integration technology will perform bi-directional transformations between object-oriented, relational, and XML representations, so that a data service can read or update data from any source, yet manipulate it in a computationally convenient format. Future data management platforms will deliver pervasive, consistent, and reliable access to data by synthesizing these new technologies with contemporary data management infrastructure.

About the Author

Vivek Singhal is vice-president of technology at the Progress Real Time Division. He leads a team responsible for defining the architecture for the Real Time Data Services products. Previously, Vivek was vice-president of engineering at Persistence Software, responsible for product development and product management. During his 10 years at Persistence (which was acquired by Progress in November 2004), he designed and patented several of the core technologies of Persistence's enterprise data management products. He defined and evangelized the product strategy, orchestrated the activities of the engineering team, and led the delivery of numerous product releases.

XML JOURNAL SUPPLEMENT

This Month

Leveraging SOA to Web-Enable Back-End Applications

Brian Wilson

Most organizations today have internal applications that span across different systems, networks, and technologies. These applications would provide more value to customers if only the data that they manage were exposed. In the past, this was often difficult due to the inability of systems to communicate with one other. However with Web service standards in place for several years now, services are finally gaining acceptance throughout the industry. Consequently, service-oriented architectures (SOAs) have finally come to fruition as companies start to rethink how they build enterprise applications.

XML-Based Interop, Close up

In addition to the strategy side of Web services, there is also the protocol-oriented side of things, the XML side. Embracing not only XML itself but also the full range of mainstream XML-based technologies like XPath, XSLT, XML Schema, and SOAP. *XML Journal* has been delivering insightful articles to the world of developers and development managers since the year 2000.

It is our privilege to bring XML-Journal directly to readers of Web Services Journal, and vice versa. Anyone already familiar with the Web services world of SOAP, UDDI, and WSDL will find here articles and features each month that will interest them – about the cutting-edge technologies and latest products that are changing not only our industry, but the way the world exchanges information. To make it easy for you to find your way around, we have four distinct sections:

Content Management:
Organization, dissemination, and presentation of information

Data Management:
Storage, transformation, representation, and general use of structured and unstructured data

Enterprise Solutions:
Systems and applications that manage mission-critical functions in the enterprise

XML Labs:
Product reviews, book reviews, tutorials, and standards analysis

Leveraging SOA to Web-Enable Back-End Applications

An in-depth look at a real-world services implementation

WRITTEN BY
Brian Wilson

Most organizations today have internal applications that span across different systems, networks, and technologies. These applications would provide more value to customers if only the data that they manage were exposed. In the past, this was often difficult due to the inability of systems to communicate with one other. However with Web service standards in place for several years now, services are finally gaining acceptance throughout the industry. Consequently, service-oriented architectures (SOAs) have finally come to fruition as companies start to rethink how they build enterprise applications.

In the past, organizations typically approached large-scale development efforts as one monolithic project. This was often due to constraints of the underlying technology as well as the difficulty of making systems communicate. If applications did communicate, they were invariably tightly coupled.

> "As companies look to building future enterprise applications, strong consideration should be given to a service-oriented architecture approach"

The concept of SOAs has changed the model of large-scale application design. Web service standards now provide the ability for companies to build core services that provide both business agility and functional reuse; as a result, applications become thinner by nature and rely on integrated services to do the heavy lifting.

Web service standards also allow organizations to build service wrappers around existing functionality, thereby getting more value out of existing systems. Finally, services are decoupled from consuming applications, allowing them to be built independently of one another.

The remainder of this article will explain how the aforementioned concepts apply to a real-world SOA implementation recently built by Analog Devices.

The Need for a Service-Oriented Architecture

Analog Devices, Inc. (ADI) designs, manufactures, and markets high-performance analog, mixed-signal, and digital signal processing (DSP) integrated circuits (ICs) used in signal processing applications. ADI's target audience comprises design engineers who evaluate semiconductor products for many different applications. In an effort to support these customers, the company constantly builds Web-based design tools to allow for product evaluation prior to purchasing. These design tools simulate the electrical behavior of an integrated circuit and allow the engineer to study the electrical response based on certain input signals without even touching the soldering iron.

ADI faced several ongoing challenges that impeded their online tool development. First, tools were often tightly coupled around a particular product or small group of products.

These tools provided value to the customer for a particular part but did not scale to support multiple parts within a given product line. It would often take many months before a new tool could be created for new or emerging products.

Another problem the company faced is that the information required building design tools for customers resided within different existing internal applications and systems. As with most large organizations, ADI suffered from the dissemination of data throughout their organization. The company needed a way to encapsulate the common functionality required by all design tools and make it available for reuse.

Finally, ADI was looking to leverage a third-party application to build simulation engines that would provide the core value for their online design tools. The product of choice was National Instrument's LabVIEW, a powerful third-party application that provides a sophisticated tool suite to build test and simulation applications. However, LabVIEW was not originally designed for distributed Web-based applications and exposing application functionality from this product via the Web was not seamless.

A new strategy was needed that would solve these business and technical issues.

Approach

ADI, with the help of Molecular, Inc., decided to build a service-oriented design tool framework that would serve as the foundation for future online design tool development. In this architecture there are two types of services: core business services and simulation services.

Core business services encapsulate functionality that could be reused by multiple design tools. These services integrate with the appropriate internal systems necessary to provide the desired functionality as well as adhere to common Web service specifications. These reusable services speed up the development of future design tool applications.

The second set of services allowed ADI design engineers to expose simulation engines built in LabVIEW as services. In this architecture simulation services are tightly coupled to a particular design tool application.

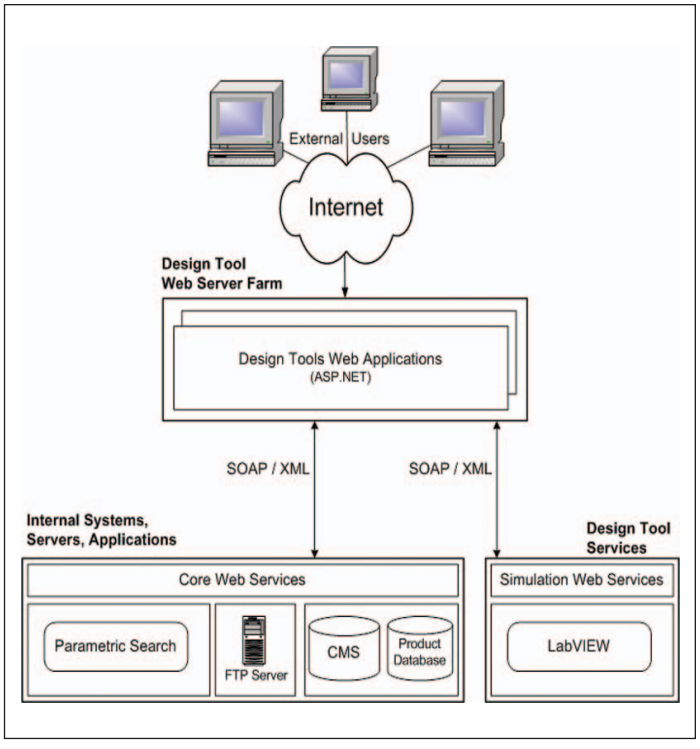


Figure 1 • A high-level illustration of the Design Tool architecture

> "Service-oriented architectures have finally come to fruition as companies start to rethink how they build enterprise applications"

Architecture

Figure 1 gives a high-level illustration of the Design Tool architecture. Each online design tool is developed as an ASP.NET application and integrates with the core set of services as needed. The interface to the specific Design Tool simulation engine (developed in LabVIEW) is wrapped in a standardized Web service SOAP header and made available to the client design tool application.

Service Development

The first step was to identify the core set of services that could be reused by multiple design tools. Common functionality was identified from several internal sources, including ADI's product database, content management system, FTP server, and parametric search applications. For each of these core services, a standardized SOAP interface was created and made available to calling applications.

Next, the team defined the service interface for the design tool simulation engine. This type of service was different from the core services because it was not intended to be used by future design tools. As mentioned earlier, each online design tool consists of a simulation application developed in LabVIEW.

The development team consisted of two groups: design engineers and Web application developers. Design engineers have the ability to build sophisticated simulation engines using the LabVIEW product, but they are not experts in Web application technology. As a result, Web developers built the online Web application and provided the necessary service plumbing to expose the simulation logic. A service-oriented architecture made sense for ADI because each service defined a clear delineation of work between team members.

Rather than create a detailed service interface, the team chose to expose only one parameter in the WSDL, a serialized XML Stream of data that adhered to a valid W3C XML Schema. By simplifying the Web service interface the team was able to focus exclusively on designing the XML data structures that would be passed back and forth, thus abstracting the Web service technologies during initial design.

Once the XML data contracts were agreed upon, LabVIEW application developers were able to go off and design the simulation engine independent of the consuming design tools application. Each simulation engine was compiled as a shared library.dll, imported into a .NET Web

"Web service standards now provide the ability for companies to build core services that provide both business agility and functional reuse; as a result, applications become thinner by nature and rely on integrated services to do the heavy lifting"

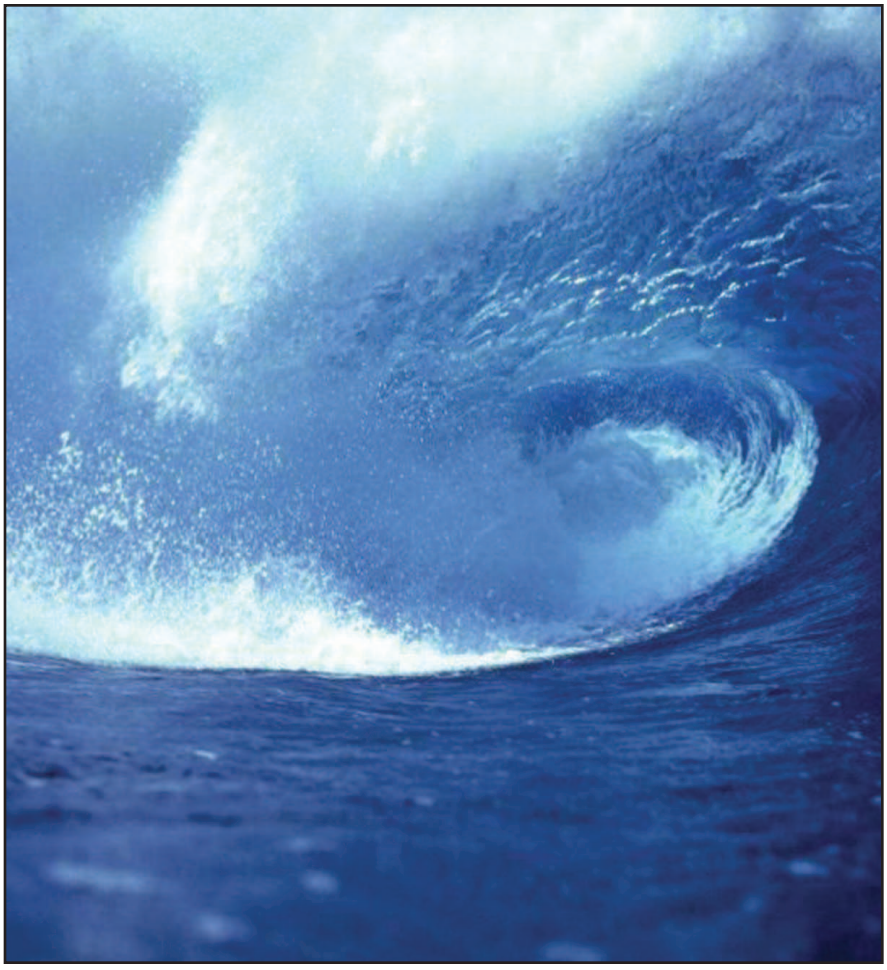
Service assembly via DLLImport, and then exposed as an internal Web service.

The team also took advantage of the XSD Sample Code Generator (XsdObjectGen) tool provided by Microsoft. Once the data contract was defined as a valid W3C XML Schema, the XsdObjectGen tool generated a set of managed .NET classes (either VB.NET or C#) that was then used to interface with the service. Application developers interacted with the service via method calls rather than by directly manipulating XML.

Benefits

By leveraging an SOA, ADI was able to achieve the following benefits for their Design Tools Framework:

- The ability to build reusable components that were not dependent on a particular application. By building a set of core services to extract data from existing internal systems and databases, future design tools can now easily incorporate existing functionality with minimal effort.
- Web service standards allowed the company to build service wrappers around existing internal systems and applications that provide a mechanism to expose this knowledge externally to customers.
- The ability to define service interfaces up front allowed the company to better align its core competencies to its areas of expertise. Simulation services were decoupled from the client applications, allowing each to be built independently.
- By streamlining the Web service interfaces, ADI is able to quickly define XML data contracts passed between simulation services and consuming applications.



Conclusion

As companies look to building future enterprise applications, strong consideration should be given to a service-oriented architecture approach. Services provide companies with a mechanism to unleash knowledge that in the past was trapped within internal applications.

By identifying existing knowledge that could potentially be bundled as business services, companies can reap the benefits of functional reuse, loose coupling, and technical platform independence.

This article has also shown a real-world example how Analog Devices implemented a service-oriented architecture to build out a new online design tools framework, as well the benefits the company received from using this approach.

Author Bio

Brian Wilson currently has over 12 years of consulting, architecture, and implementation experience within the industry. He is a technical consultant at Molecular, an Internet consulting firm in Watertown, MA.

bwils06@gmail.com

IBM WEBSPHERE PRESENTS

YOU VS REWRITE, REVISE RE-EVERYTHING

INFLEXIBILITY: MEET SOA

PLUS: TANGIBLE BUSINESS BENEFITS ★ BEST PRACTICES FOR BEST RESULTS

OVER 10 YEARS OF WORLD-CLASS INTEGRATION EXPERTISE

FEATURING:

A FASTER, EASIER WAY TO IMPLEMENT TRUE SOA | SUPPORT FOR OVER 80 O.S. CONFIGURATIONS | UNPARALLELED INDUSTRY KNOWLEDGE & PROCESS SKILL

IBM MIDDLEWARE. POWERFUL. PROVEN.

FIGHT BACK AT WWW.IBM.COM/MIDDLEWARE/SOA. THIS IS A RIP-AND-REPLACE-FREE EVENT.

Ensure Interoperability.

Validate functionality.

Eliminate security vulnerabilities.

Test performance & scalability.

Confirm compliance.

Collaborate and reuse.

Ensure Secure, Reliable Compliant Web Services

PARASOFT®

SOAtest™

As enterprises adopt Service Oriented Architectures (SOA) to deliver business critical data, ensuring the functionality and performance of Web services becomes crucial. Complex web services implementations require the means to thoroughly validate and test them to assure they are truly production ready.

Parasoft SOAtest is a comprehensive, automated tool suite for testing web services and complex Service Oriented Architecture (SOA) solutions to ensure they meet the reliability, security and performance demands of your business. SOAtest provides a total and holistic testing strategy for your SOA implementations including automated unit testing, graphical scenario testing, scriptless performance/load testing, security penetration testing, standards validation, message authentication, and more.

If you are building serious web services, you need SOAtest. For more information regarding Parasoft SOAtest, call 888-305-0041 (x-3501).

Download a copy of SOAtest for a free evaluation today at www.parasoft.com/WSJmagazine

Parasoft SOAtest clients include: Yahoo!, Sabre Holdings, Lexis Nexis, IBM, Cisco & more.

PARASOFT®

We make software work.™

Automated Error Prevention™