September 2003 Volume 3 Issue 9

WebServices JOURNAL

.NET J2EE XML

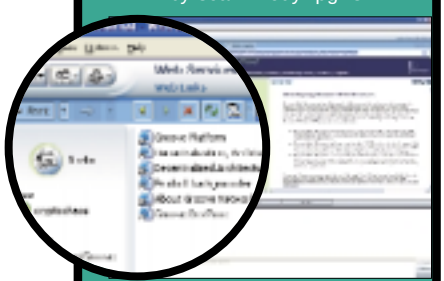
WSJ2.COM

Full Conference Program Inside!

Page 41

FOCUS ON SUCCESS STORIES

From the Editor
Making the Case
by Sean Rhody pg. 3



RETAILERS PLEASE DISPLAY UNTIL OCTOBER 31, 2003

$6.99US $7.99CAN

09>

0 71486 03420 9

SYS-CON MEDIA

WebServices JOURNAL
.NET J2EE XML

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

SYS-CON MEDIA

Making the Case

Written by
Sean Rhody

Author Bio:
Sean Rhody is the editor-in-chief of Web Services Journal and managing editor of WebLogic Developer's Journal. He is a respected industry expert and a consultant with a leading consulting services company. SEAN@SYS-CON.COM

This past August, I celebrated a major milestone – 10 years in the publishing business. Over the last decade, I've had the opportunity to participate in, and in some cases, even anticipate major technology changes. First with client/server, then with the Internet and Java, and now with Web services.

What I've noticed is that technology runs in stages, and the publishing surrounding technology runs that way as well. Gartner has a chart they call the "Hype Curve," which describes the adoption of technology over time. There's a correlating cycle for publishing, which for lack of a better name I'll call the Rhody Publishing Life Cycle (hey, I noticed it, why shouldn't I name it?).

The cycle starts with the release of some new technology. At this point in time, little is known about the technology, there are only a few experts at it, and those folks are busy up to their eyeballs doing projects, training people, and generally reaping the benefits of being first on board. This is the "Trial Stage," where new technologies get tested. Most magazine publishers won't touch a magazine at this stage, for fear of losing all the startup and branding costs if the technology never makes it. So typically the technology becomes a guest section of another magazine, while the wait for adoption occurs.

The next stage is what I call the "Stealth Stage." Companies begin to adopt the technology, at least on a trial basis. But no one says anything about it. Publishers sacrifice small animals in an attempt to lure a company into discussing it enough for a case study. Most companies demur, either from fear of affecting the stock price by announcing they're using something that isn't older than dirt, or because they view the technology as a competitive advantage and want to keep that edge as long as possible. I hate this stage.

The next stage I call the "Widespread Panic Stage." Everybody's doing it, everybody wants to talk about it, little startups are everywhere, and a publisher has to take the phone off the hook, disconnect the computers, and try to refrain from shooting friends who say "I've got this friend who just started such and such – maybe you could do an article on it." This is the glory time for a magazine. You're done running "Hello World" articles and can discuss topics in depth, with detailed information for the hard-core programmer, and high-level concepts for the enterprise architect or CIO. If you're lucky, this stage lasts a long time.

But like everything in this world, there is a time, and a place. After that time, we get to the final stage, the long, agonizing descent into oblivion that I like to call the "Dustbin Stage." Some technologies fade quickly, others linger in the dustbin stage for decades – too vital to replace (or too costly), but nothing new ever happens. All right, perhaps Objective Cobol interests someone, but you get the idea.

For Web services, we're somewhere between Stealth and Widespread Panic (sounds like a double bill at some local gin joint). Case studies are rare, but we are starting to see them; in fact there are two in this month's issue. So sit back, relax, and get ready for full on Widespread Panic. Coming soon to a computer near you.

Mindreef

www.mindreef.com

Future-Proofing Solutions with Coarse-Grained Service-Oriented Architectures

Written by John Medicke & Thomas Pack

Web services and service-oriented architectures are transforming application construction. The ubiquity of Web services support by all leading platform venders brings the promise of a flexible application environment with simplified interface techniques, location transparency, and platform-neutral interoperability. This dynamic infrastructure brings about a new implementation approach, the service-oriented architecture.

However, to date most Web services projects have really only created simplified communication mechanisms for the invocation of those same old complicated legacy interfaces that we have always had. To truly realize the creation of service-based components, a new design approach is needed, one that produces simple, straightforward coarse-grained service interfaces that conceal the ugliness of the legacy low-level interfaces. Designing coarse-grained interfaces is not as easy as it sounds. This article discusses how to recognize good coarse-grain interfaces and how to design coarse-grain interfaces for maximum flexibility and longevity.

When you use a service, be it a bank, a restaurant, or a store, you expect to easily interact with that service. How would the restaurant-goers experience be if there was a different process for ordering each part of the meal, or when withdrawing cash at an ATM, there were a hundred different menu options. Service interfaces are expected to be simple and intuitive. This is what makes a service successful. New services should be easily discoverable and the consumption of those interfaces undemanding. ATM machines are ubiquitous because everyone knows or can easily learn how to use one.

The proper design of a service should take an "outside in" approach to constructing the interface by focusing on the client application's perspective and how the component plays within the larger business process context. Sadly, many interface designs take an "inside out" approach by basing the interface design on the dirty details of the existing implementation rather than the specific requirements for client utilization. The interface should be on a "need-to-know" basis, only exposing interface details that are meaningful to semantics of the larger interaction with client applications.

Ongoing software maintenance is a constant burden for business applications. Today's development is tomorrow's legacy. Loosely coupled service-oriented architectures create an opportunity to

AUTHOR BIOS:

John Medicke has been a senior consulting IT architect for the last six years, responsible for the design of leading-edge solutions across multiple industries, including retail, banking, insurance, finance, government, automotive, and electronics. He is the coauthor of Integrated Solutions with DB2, *as well as several articles in* DB2 Magazine.
MEDICKE@US.IBM.COM

Thomas Pack is a solution architect for IBM's On Demand Solution Center (ODSC). He has 11 years of experience with IBM. Thomas's areas of expertise are enterprise transformation and integration, distributed computing, J2EE, and Web services.
PACKWT@US.IBM.COM

reverse this escalating legacy cycle. Proper design of the interfaces can minimize changes as requirements evolution occurs, making these services future-proof.

The Problem

Most poor interfaces are the result of allowing a convoluted implementation to bleed through into the interface. Even a good implementation that is transparently presented as a Web service becomes a thorny service interface. A component often contains dozens of classes and hundreds of methods. Exposing this detail as a service is analogous to an ATM machine with hundreds of menu options. Complicated interface designs impose unnecessary responsibilities on the client application, for example:

- They result in excessive chattiness over the Web services interface.
- They overburden the client application with the maintenance of the service component's context.
- They force the client application to become co-dependent with the service component requiring dual development for the life of the solution.

This poor interface design will sign a death sentence for the component and the service-oriented implementation by tightly coupling the client application with the component, requiring future maintenance of intricate interaction semantics and proprietary interaction context.

Interaction Semantics

Part of the problem is that the mechanism for interacting with a component is different from – and often in some very bizarre ways – any other component. The client application needs costly custom implementation for invocation of the interface of each component. This is reminiscent of Indiana Jones navigating through hidden passageways to find the lost treasure with secret incantations required for entrance to each secret chamber. In their book *BPM: the Third Wave* (Meghan-Kiffer Press, 2001), Howard Smith and Peter Fingar expanded this thought: "Imagine a world where people speak a language that brilliantly describes the molecular structure of a large object but can't tell you what that object is – or that it's about to fall on you." The reason for failure of component interoperability from this perspective is self-evident.

We remember a recent engagement for the integration of physician practice applications with a hospital IS system that had outlandish interface semantics. Ignoring reasonable practices and embracing the quirks of their implementation, the system required that the patient's first and middle name fields be left blank and the last name field contain the patient's full name. This is just one of the many urban legends of strange system interfaces that litter the IT landscape.

Burden of Context

Another problem of low-level proprietary interfaces is that the client application is pulled into the internal context of the called component. With multiple method invocations for any coarse-grained operation, there is a need to maintain implementation context in the client application in order to provide that context on subsequent method calls. One example of a system from a recent project would require special codes indicating proprietary state and numeric status codes (e.g., code="F", Status="989"), which had no relevance to the client application, to drive the invocation of each operation. Implementation-specific state information was forced onto the client application by the component. The delineation between the relevant context of the interaction (that which is meaningful to both the client and the component) and irrelevant context (the exclusive state information of the component) is often not considered during interface design.

Context in terms of resource references, processing state, and method parameter values are unnecessarily forced into the interface. As the interface grows so does the interaction context. The client application has its own context to maintain and does not need to be burdened with the component's implementation context.

Example: Customer Management System Interface

To illustrate the difficulty presented by exposed low-level interfaces, look at a customer-service application that has a component for customer. With objects

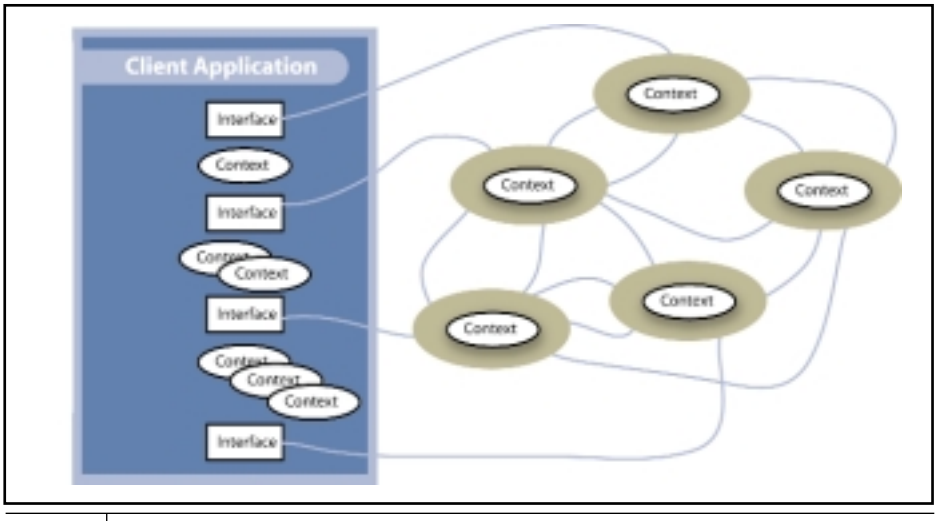


FIGURE 1 | **Client application**

for atomic elements of the customer information results in the client application, implementation might look like the following:

```
customer =
CustomerManage.findCustomer("123456789")
;
 customerID = customer.getCustomerID();
 addressVector =
addressList.findAddresses(customerID)
 homeAddress =
addressVector.findAddress("Home")
 homePhone = homeAddress.getPhone();
 shipTo =
addressVector.findAddress("Ship To");
 shipToZip = shipTo.getZip();
```

This demonstrates how the client application is pulled into an unnecessary interaction context that it has no concern with.

Designing for Serviceability

Creating a coarse-grained interface that truly embodies a service must be a conscious part of the interface design process. Serviceability comes from an interface that is easy to exploit, straightforward, and has a life span beyond the first version. To discuss these characteristics of coarse-grained service design, we'll borrow the ACID acronym from the transactional processing domain:

- ***Atomic:*** Any one business operation can be completed through one service interface. The coarse-grained interface is close to a 1:1 ratio of business operations to service interfaces. In document-oriented Web service interfaces the interface is simplified to the semantics of document exchange.
- ***Consistent:*** There is a consistent behavior to the interface within a domain that makes new services in that domain easily recognizable and understood. Consistency extends beyond the interfaces of one component to a common interface format for all components within a domain. Locating services instances, establishing context, retrieving data, performing updates, and executing business operations all have consistent interaction. One of the best ways to do this is with a FCRUD semantic where service resource objects are operated with straightforward semantics of the Find resource, Create resource, Retrieve resource information, Update resource information, and Delete resource. Consistent interaction semantics empower the client application developer to easily utilize any of the service components once he or she has experience developing an interface to one.
- ***Isolated:*** Any one interface can be invoked independent of other service interfaces. Component implementation context and detailed invocation sequencing are not forced onto the client application. Loose coupling exists rather than tight semantic coupling. Predecessor and successor invocation requirements are not part of the interaction beyond the degree that they are part of the shared process between the client and the component. Isolation of interface design enables the client application to invoke any service interface with a minimum of preconditional interaction, perhaps only a Find resource call.
- ***Durable:*** This interface has been designed with a vision of the future and has longevity built into it. The interface envi-

sions the broad range of usage scenarios and future implementation enhancements and has been designed with an eye towards ease of migration. A durable interface is future-proof, not that the interface will never change, but it has the capability to easily incorporate future enhancements. Document-centric Web service interfaces and other loosely coupled interface techniques minimize the impact of extensions to the interface.

How do you design ACID characteristics into your service component interfaces to best ensure successful exploitation in a service-oriented architecture? The best way is by proper modeling of the component interface. Modeling often looks inward. It is easy for programmers to immediately concern themselves with the implementation while the interface and its usage become a background activity. Taking an "outside in" approach to the modeling ensures that consideration is properly given to the continuum of utilization.

Correct modeling takes a top-down approach beginning with the business processes and requirements for the solutions that the component service envisions being exploited in. Correct modeling should begin at the domain level by identifying the business context of the component usage and what business processes would exploit the component. For each business process, identify the use cases or scenarios that demonstrate the various ways the process could be executed. These process use cases lead to specific use cases of component interaction.

This top-down approach ensures that the context of the process is firmly established before design looks inward at the interface and implementation of the component. Although it may seem like overkill, establishing use cases at the domain level will ensure that correct specification of the service is created and future-proofs that component by anticipating all possible usage scenarios.

Proper Service Interface – Customer Management System Interface

Returning to our customer component example, you can see how a proper coarse-grain interface results in a simplified client implementation without the burden of component context or proprietary interaction semantics:

```
customerdocument =
CustomerManageService.getCustomerDocumen
t("123456789");
homePhone =
customerdocument.getHomePhone();
shipToZip =
customerdocument.getShiptToZip();
```

This document-centric interface requires only one service invocation, which returns a document containing all the necessary information the client application requires without the burden of implementation details.

Creating Coarse Grain Implementations

Typically a Web services project focuses primarily on migrating existing application functionality to a Web services interface. The preexisting application can be anything from legacy systems to J2EE Enterprise JavaBeans (EJBs). How do you map this to a coarse-grain service interface? Very seldom will traditional low-level interfaces naturally

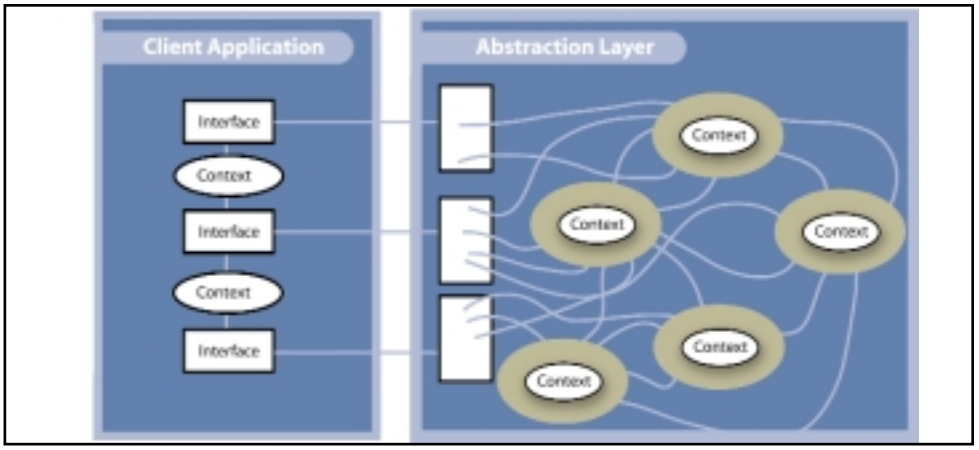


FIGURE 2 | **Abstraction layer hiding implementation details**

map to a proper coarse-grained structure. An abstraction layer is required to hide the details of the implementation from the user behind a facade. This abstraction layer encapsulates:

- Multiple low-level interfaces that comprise the business operation
- Multiple data sources that need to be aggregated for the service
- Legacy system interaction
- The sequencing of low-level calls
- Maintaining context for the low-level implementations
- Transactional coordination of updates to multiple low-level interfaces

The abstraction layer can be constructed using either of two approaches:

1. ***Build to integrate:*** Using application development techniques, a facade or mediator is implemented to provide the interface that aggregates the lower-level interfaces. Coarse-grained components are created that broker the interactions with multiple classes. Traditional application development tools and techniques can be employed for development of this facade. Component development environments like J2EE or Microsoft's .NET provide application environments to host both the facade component and the implementation.
2. ***Enterprise application integration (EAI)/Business process integration (BPI):*** EAI tools exist for the purpose of integrating applications together. They provide rich tools and functionality for rapidly integrating all types of applications, including legacy, Web, and packaged software applications. BPI tools extend this capability to provide choreography of the business process and application flow outside the application. Web services–based integration is now a component of almost all EAI/BPI tools. This means that these tools can expose their integration flows through coarse-grain Web services interfaces.

Which is a better choice depends on a number of factors. Some things to consider are if the implementations are of similar technologies, what interfaces these lower-level interfaces currently support, and whether legacy systems are part of the equation. Often it comes down to whether the primary focus of the project is application development or business integration.

Conclusion

The success of service-oriented architectures depends on a rich universe of available services that are easily located, understood, and utilized by a diverse community of users. These interfaces must have a life span beyond the first implementation, which can only be achieved by proper design of coarse-grained interfaces that are truly coarse-grain in nature and not just a weak veneer on top of an existing tortuously complicated interfaces. By taking an "outside-in" approach to modeling the service component interface, it is possible to identify the full spectrum of usage of the component.

As you design services, remember the ACID acronym and ask yourself if the interface models a full atomic business operation; is there consistency to the interface across the family of components; can any one interface be invoked in reasonable independence from other interfaces; and has the interface been designed with a view towards future usage scenarios. This perspective will lead to components that can truly become ubiquitous services in the Web services world.

Web Services Journal
.NET J2EE XML

PRESIDENT AND CEO
Fuat Kircaali fuat@sys-con.com

VP, BUSINESS DEVELOPMENT
Grisha Davida grisha@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
DIRECTOR, SALES & MARKETING
Megan Ring-Mussa megan@sys-con.com
ADVERTISING SALES MANAGER
Alisa Catalano alisa@sys-con.com
ASSOCIATE SALES MANAGERS
Carrie Gebert carrie@sys-con.com
Kristin Kuhnle kristin@sys-con.com

SYS-CON EVENTS
PRESIDENT, SYS-CON EVENTS
Grisha Davida grisha@sys-con.com
CONFERENCE MANAGER
Michael Lynch mike@sys-con.com
REGIONAL SALES MANAGER
James Donovan james@sys-con.com

CUSTOMER RELATIONS/JDJ STORE
CIRCULATION SERVICE COORDINATORS
Niki Panagopoulos niki@sys-con.com
Shelia Dickerson shelia@sys-con.com
Edna Earle Russell edna@sys-con.com
JDJ STORE MANAGER
Rachel McGouran rachel@sys-con.com

SYS-CON.COM
VP, INFORMATION SYSTEMS
Robert Diamond robert@sys-con.com
WEB DESIGNERS
Stephen Kilmurray stephen@sys-con.com
Christopher Croce chris@sys-con.com
ONLINE EDITOR
Lin Goetz lin@sys-con.com

ACCOUNTING
FINANCIAL ANALYST
Joan LaRose joan@sys-con.com
ACCOUNTS RECEIVABLE
Kerri Von Achen kerri@sys-con.com
ACCOUNTS PAYABLE
Betty White betty@sys-con.com

SUBSCRIPTIONS
SUBSCRIBE@SYS-CON.COM
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

SYS-CON MEDIA

Written by Rickland Hollar

Moving Toward the Zero Latency Enterprise

Speed is key to results

The Internet makes it possible to deliver information almost instantaneously – anytime, anywhere – and is redefining the traditional boundaries around organizations and their IT systems. The Internet has turned buyers into sellers, sellers into buyers, and set new expectations for how services should be delivered. These expectations raise the bar for applications in terms of their need for interconnectivity and responsiveness. For businesses to remain competitive in this environment – or in the case of government agencies, responsive – they must embrace the idea that speed not only matters, but that it is now a key discriminator. Enterprises able to leverage the Internet's real-time nature and its technologies create competitive advantages that let them reap the benefits of greater efficiency, responsiveness, market share, and profitability. This promise has led businesses to look to interconnect their enterprise resource management (ERM), supply chain management (SCM), and customer relationship management (CRM) systems, both internally and externally; and government agencies to look for better ways to connect their systems with the public, their suppliers, and each other.

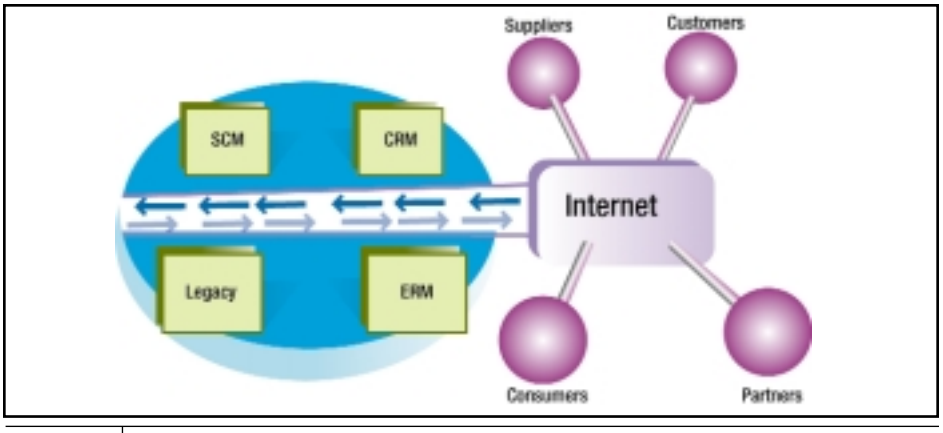


FIGURE 1 | **The zero latency enterprise**

The Gartner Group coined the term "zero latency enterprise (ZLE)" to describe organizations that can exchange information with employees, trading partners and customers in near real time). The original focus was on internal systems, but much of what Gartner said applies equally to eliminating latency between internal and external systems. Figure 1 illustrates many of the concepts underpinning ZLE. In a ZLE organization, business events trigger system events that post actions and send responses throughout the enterprise. Bill Gates calls this level of interconnectivity a "digital nervous system." Like the human nervous system, the applications in a ZLE organization interconnect in such a way that they eliminate latency, which is the time gap between when the system receives information at one point and uses it, wherever needed, at others.

Becoming a ZLE

Transforming your organization into a ZLE is a three-phase process. You must understand where latency exists within your current processes and systems and your options for reducing or eliminating it. You must then create an architecture that focuses on minimizing latency along the critical path of as many business processes as possible. Finally, you must translate the architecture into an implementation plan that provides the roadmap for yours becoming a ZLE organization.

The first step in becoming a ZLE organization is identifying the major business processes within your organization that the ZLE architecture must support. One goal at this stage is to establish the architectural boundaries of the effort; will it deal with internal systems, external systems, or both? Another goal is to understand the dynamics of each business process: its tempo, meter, natural pauses and breaks. Understanding these dynamics is critical to recognizing latency

Author Bio

Rickland Hollar is a senior applications architect with the Central Intelligence Agency and has over 30 years of experience in the industry. Prior to joining the CIA, he was president of a Virginia-based software development firm. RICK_HOLLAR@YAHOO.COM.

and bounding the parameters for fixing it. It is important to remember that what constitutes latency in one process may be completely acceptable in another, even for the same application.

The next step is to decompose each business process into its applications and identify any latency points that exist. You should ask: What applications make up this process? Is latency a problem in either the overall process or one or more of its supporting applications? If so, how much of a problem? How much does its timing need to change? The result of asking and answering these questions is a list of business processes and applications that have latency problems.

Next you need to learn as much as possible about each latency point so that you can later devise techniques for removing, or at least minimizing, the latency. For internal systems, latency stems from several root causes (see Figure 2). Legacy systems are often stovepiped applications that were developed independently, over time, using different technologies. These applications create islands of information and functionality that are by their very nature difficult to integrate and share.

The same data belonging to different applications may be in different formats, follow different data validation and business rules, or be updated through completely different business processes. Interfaces within these older applications tend to be synchronous, tightly coupled, and driven more by the underlying technologies than the business needs they serve. Proprietary drivers, proprietary APIs, and proprietary formats represent only the tip of the iceberg when it comes to tying these systems together. One question you should ask is: What are each application's processing characteristics: batch, on-demand, or continuously running? Some may be batch oriented where you need them to be real time, others may have availability and reliability problems in cases where you need them to be 24x7. These issues frequently reflect age and technology differences that increase the difficulties in creating a coherent architecture.

Latency's causes multiply when you look at connecting internal and external systems. Each external system potentially represents a different set of technology, security, reliability, and manageability characteristics that your architecture must address.

The Architecture

At the end of the first phase, you should have a good understanding of your organization's internal and external business processes and the latency points you need to address within each. You're now ready to lay out the major business processes and applications and begin developing an overall ZLE architecture. It's important that your architecture address four key elements: business process management, data communications and routing, data transformation and formatting, and applications connectivity.

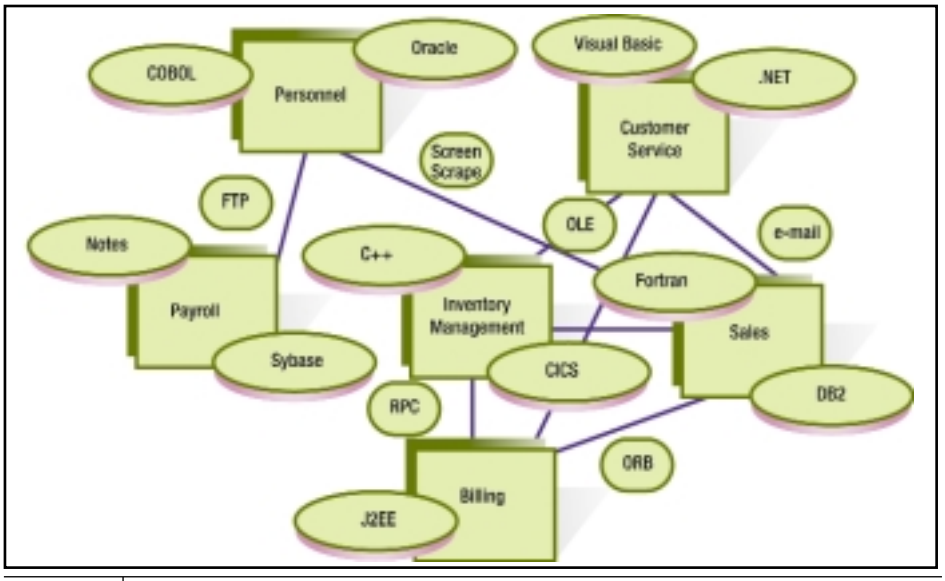


FIGURE 2 | **The legacy nightmare**

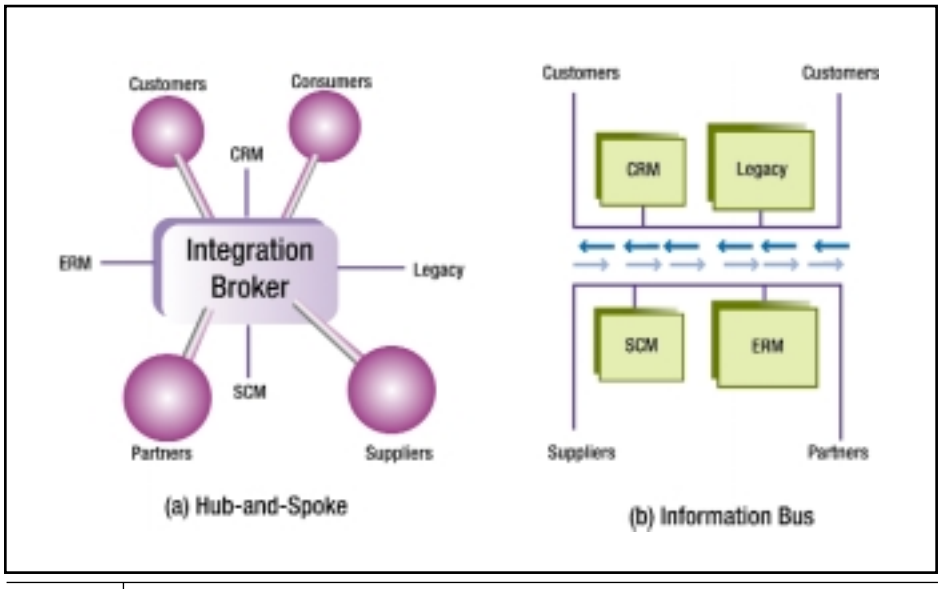


FIGURE 3 | **Data communications and routing architectures**

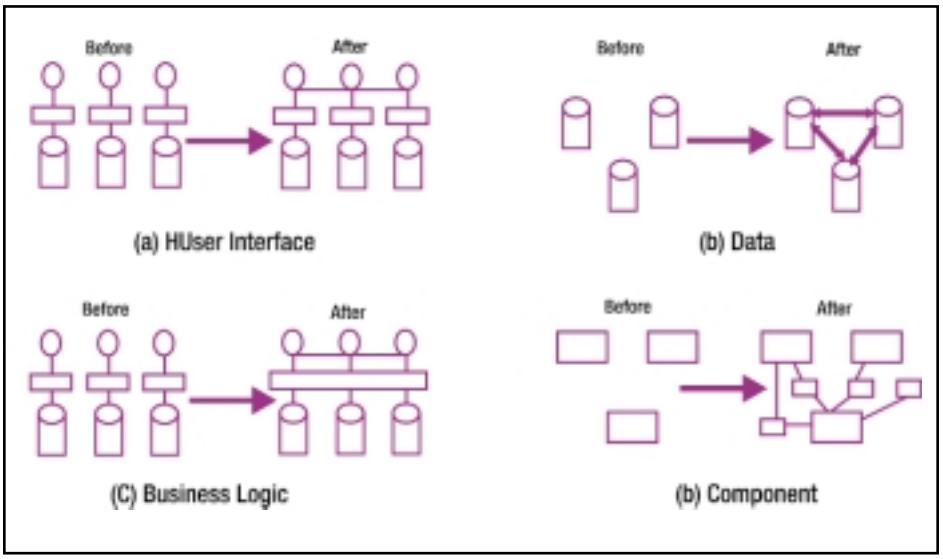


FIGURE 4 | **Application integration strategies**

Business process management is, in my opinion, the most important part of the architecture; it's the glue that ties applications together. It should reflect the enterprise's business processes: assembling, sequencing, and orchestrating applications to align them with the business's natural processes and work flows. A workflow manager, a rules engine, and collaborative tools can be critical components at this level. Employees, business partners, and customers should find easy-to-use, intuitive interfaces supporting your core business processes.

Data communications and routing in conjunction with business process management create the central nervous system for the ZLE architecture. Two fundamental architectures, shown in Figure 3, have evolved in this area: hub-and-spoke; and data, or information, bus. The hub-and-spoke architecture uses a central integration engine and message queuing products, such as MQSeries and MSMQ, to integrate across applications. In this architecture, applications deal with one another through the central hub; this is responsible for extracting, transforming, and routing data and coordinating activities throughout the overall system.

The information bus architecture takes a decentralized approach. This architecture implements a common messaging framework, frequently using a publish and subscribe model, for intercommunication. Applications connect to this bus through application adapters and pass messages to one another by placing them onto the bus. The information bus may use either a messaging or workflow manager to assist in routing messages. You can use either architecture internally; the information bus is clearly superior when connecting between internal and external systems.

XML has become the lingua franca for solving the data transformation and formatting problem. It provides a flexible, extensible syntax for expressing both information and its structure in a meaningful format. Legacy applications can apply Extensible Stylesheet Language Transformations (XSLTs) to XML documents to convert information within those documents into whatever format they need. Data transfer and replication tools are also available for extracting, transforming, cleansing, and loading data for those wanting to make minimum modifications to existing applications.

Application integration can occur at many different levels (see Figure 4). A key question is whether there is overlap in the data the applications process or the business rules they enforce. User interface integration integrates applications at the presentation layer. This level of integration is valuable for connecting independent applications into common business processes. Data integration integrates applications at the database level by copying, transferring, or replicating information from one data source to another. This is a good strategy when transfers are timely and business rules are sufficiently compatible. Business logic integration integrates applications' middle tiers, allowing each application to retain its original business rules and logic. This level

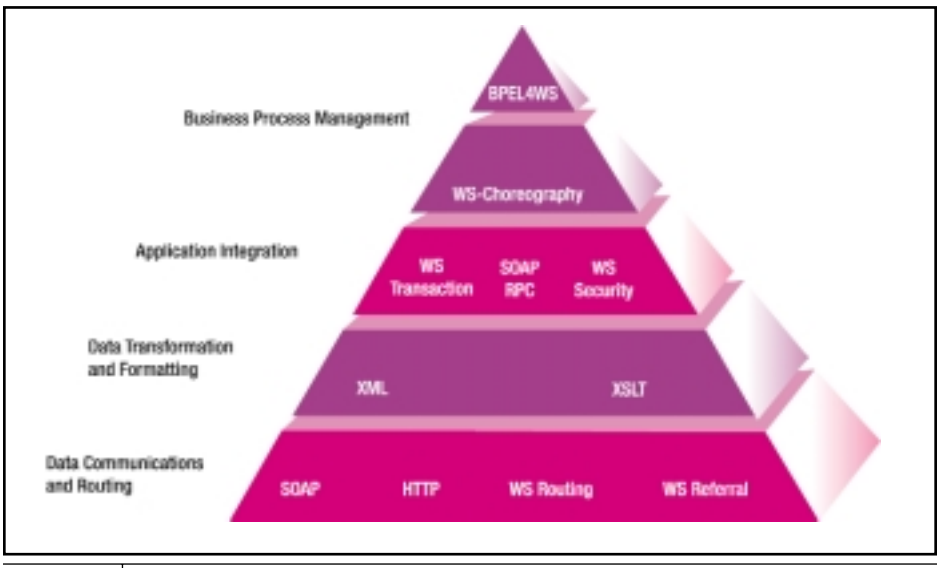


FIGURE 5 | **Web services technologies vs ZLE architectural elements**

of integration works best for tying existing, interdependent applications together into more streamlined processes. Component integration integrates applications through their application programming interfaces (APIs), common components, or function calls. Integration at this level may require you to write proxy interfaces for some components; change call interfaces from direct to RPC for others; or adopt a distributed object model such as DCOM, CORBA, or Web services. This integration form is most useful for creating components several applications or processes can share.

A critical part of the application integration analysis is looking closely at each latency point to determine both the level of integration and corrections needed. The first step is to identify the appropriate integration level for each application: presentation, business logic, or data. Simply changing the application's invocation characteristics may be enough to also change its latency characteristics for some applications.

In situations where that is not the case, the next step is to drill down into the application and its interfaces with an eye towards improving the application's performance characteristics. The first, and simplest, corrective measure is to identify and remove any inefficiencies or chokepoints within the application. A second option is to look at overlapping the application's processing with that of others by making it an asynchronous process. Making an application asynchronous is straight-forward; you simply need to add a queue and alerting and rendezvous mechanisms. This can also be a good approach for dealing with reliability and availability problems caused by older systems. A third, and sometimes only, option is to redesign and rewrite the application.

You may need to make several passes through each of the four architectural elements to finalize the ZLE architecture. That isn't unusual. It's important that you come away with an overall architectural strategy, a list of integration points, and an idea of the integration strategies you'll need to address as part of the implementation process, which is the next step. Before proceeding to the implementation phase, it's a good idea to create a set of guiding principles to help in making architectural tradeoffs and selecting products. Questions you should answer include: Are the number or types of products you use of concern? How about the amount of code you write? Is it important to use the same solution for solving the latency problem between both internal and external applications? Is it important to use the same integration solution for integration points at the same level? Do you have large investments in ERM, SCM, or CRM solutions that will drive the implementation? With the answers to these questions in hand, you're now ready to look at options for implementing the architecture.

Implementation

Web services provides a lightweight, standards-based solution for implementing a ZLE architecture. Web services offers an integration model that brings applications together as loosely coupled components within a larger architectural framework. This standards-based framework closely aligns to the four elements in the ZLE architecture (see Figure 5). Business Process Execution Language for Web Services (BPEL4WS) and WS-Choreography are standards proposals for modeling, defining, orchestrating, and implementing business processes. WS-Transaction and WS-Security supply protocols for implementing atomic and business transactions, and security features such as authentication and encryption that are necessary for tying applications together into new business processes. The Simple Object Access Protocol (SOAP), HTTP, and TCP/IP create the backbone for data communications. WS-Routing, and WS-Referral address the data routing problem.

XML, which is the heart of Web services, provides a standard for data representation. XSLT adds a language for data transformation and formatting. SOAP-RPC contributes a lightweight, standards-based, platform-independent component model for implementing distributed components. In short, Web services provides all the elements necessary to implement whatever ZLE architecture you ultimately develop. With several of the standards still evolving, the issue is that products lag behind standards; that means you have to write more code.

If that is a concern, off-the-shelf enterprise application integration (EAI) products offer a good foundation for moving towards a ZLE organization. EAI products provide message broker and adapter technologies that quickly integrate applications to exchange and share information at the data, business logic, or presentation layers. Most EAI solutions implement either a message broker or bus concept corresponding to the hub-and-spoke and information bus architectures. If you decide this is the best approach for you, choose a product that fits into your overall integration strategy by providing the greatest number of integration adapters corresponding to the integration levels, points, and products you identified as part of your analysis.

EAI and Web services are extremely powerful together – EAI for fine-grained interfaces, Web services for coarse-grained interfaces. Many EAI vendors, such as SeeBeyond, TIBCO, webMethods, and IBM, recognize this synergy and offer products that are in fact a marriage between traditional EAI technologies and Web services. These products give you a best of both worlds option. Ultimately, the question boils down to which strategy works best with your architecture within your organization.

Summary

The costs of not becoming a ZLE organization are high; they translate to frustrated customers, disappointed partners, and missed opportunities. The challenges are in understanding critical business processes and developing an architecture that removes the problems creating latency both in the enterprise's internal systems and in their connections to systems belonging to trading partners and customers. Web services standards, which EAI products are rapidly adapting, lay out the framework you need for implementing this architecture. As more companies adopt them, low cost, standards-based solutions for implementing ZLE applications may finally become a reality.

Integrating CICS Applications as Web Services

written by Russ Teubner

Web services promise to lower the costs of integration and help legacy applications retain their value. This article explains how you can use them to integrate mainframe CICS applications with other enterprise applications.

Web services are platform-independent interfaces that allow communication with other applications using standards-based Internet technologies, such as HTTP and XML. They provide an opportunity for organizations to reduce the costs and complexities of application integration inside the firewall and create new possibilities for legacy applications to participate in e-business. One problem with traditional integration techniques is the proliferation of point-to-point communication and data conversions that must change as new applications are integrated or data formats change. The problem quickly gets complicated when you add business partners, subsidiaries, mergers, or acquisitions to the integration mix. Web services simplify integration by reducing the number of APIs to one (SOAP) and the number of data formats to one (XML). While there is often variation in the syntax of the XML payload, such as the FinXML, FIXML, OFX, and IFX standards used within financial services, the fundamental knowledge and skills needed to work with each of these standards is the same.

Web services are standards based and platform independent. Independent standards bodies control the direction of the core components, rather than individual corporations pushing their own technologies. By using standards-based technologies and widely available skill sets, Web services allow companies to develop flexible integration solutions that can evolve as needs change.

Web services sound promising for new applications that support built-in Internet technologies, but what about legacy applications, like CICS? How can they benefit from Web services integration and what does it take to integrate legacy applications as Web services?

AUTHOR BIO:

During his 23-year career, Russ Teubner has focused much of his creative energy on solving the difficult problems associated with the integration of legacy systems with technologies that have emerged more recently, such as XML. Russ currently is CEO of HostBridge Technology.
RUSS@HOSTBRIDGE.COM

CICS Applications and Web Services

IBM's CICS (Customer Information Control System) is a family of application servers that provides online transaction management and connectivity for legacy applications. Some simple facts about CICS demonstrate the importance of integrating CICS with business-critical initiatives that involve Web services.

- 30 years and $1 trillion invested in CICS applications (IDC)
- 20,000+ CICS/ mainframe licenses worldwide
- 14,000+ CICS customers worldwide
- Used by 490+ of IBM's top 500 customers

- 30 million end users of CICS applications
- 150,000+ concurrent users/system
- 5,000 CICS software packages from 2,000 ISVs
- 950,000 programmers earn their living from CICS
- CICS handles more than 30 billion transactions/day valued at over $1 trillion/week

Figure 1 shows a high-level taxonomy of CICS applications.

CICS represents the broadest category of mainframe applications. CICS transactions fall into two subcategories: "visual" and "nonvisual." A "visual" transaction is one that expresses a presentation interface to an end user at a terminal. You could also refer to a "visual" transaction as a "terminal-oriented" transaction. In contrast, "nonvisual" transactions don't interact with an end user. Instead, another program invokes these transactions. (This type of transaction is also referred to as a "COMMAREA transaction" because the input/output parameters are passed to/from the transaction using an area of storage referred to as the "communication area," or COMMAREA.)

The nature of CICS applications makes them complementary to Web services technologies. Nonvisual COMMAREA applications take one request and return data to the requesting application in a single step. This maps well with the Web services model because a single SOAP request would yield the required host data. Meanwhile, the majority of visual applications use a component of CICS called Basic Mapping Support (BMS). BMS essentially handles the presentation logic of the transaction and relieves the application developer of having to encode and decode 3270 terminal data streams. BMS expresses fields and data as name/value pairs, which can be converted to XML for consumption by Web services.

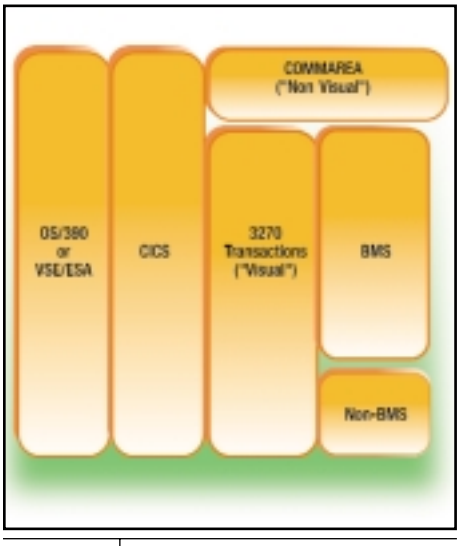


FIGURE 1 | **CICS application taxonomy**

While mainframes continue to be the most reliable and scalable platforms for handling large amounts of data and large numbers of transactions, many organizations want to move to newer technology and stop relying on the dwindling numbers of COBOL developers and system administrators with mainframe skills. However, in most cases these same organizations realize that their legacy applications "work", their business processes rely on them, and that making changes to decades of business logic is both time-consuming and risky. Moreover, integrating applications is usually cheaper than rewriting them to operate on other platforms. Now by enabling your legacy applications as Web services you can deliver integration projects faster and cheaper because your legacy application groups and Internet groups use their existing knowledge to integrate the applications.

Integration Models

There are two basic models for integrating CICS applications as Web services, both of which include the use of adapters. The differences between these models depend upon where the Web services exist, how they operate under the covers, and the types of applications you want to integrate. In this article, we will refer to these models as connectors and gateways.

- ***Connectors*** run on the mainframe and can use native interfaces that permit seamless integration with the target application.
- ***Gateways*** run off the mainframe on middle-tier servers and often use traditional methods, such as screen-scraping (that is, capturing legacy data based on row/column coordinates of the application screen).

In the purest sense, adapters might seem antithetical to the Web services model because the target application is not itself acting as a Web service. Nevertheless, when you face the daunting task of rewriting trillions of lines of code and millions of legacy COBOL/Assembler applications, the need for adapters becomes apparent. The question then becomes "what type of adapter do you want to use?"

The choice between using connectors or using gateways often depends upon the types of applications you need to integrate. Integration vendors have traditionally provided facilities that allow you to use gateways to access CICS. These gateways commonly use screen-scraping techniques. However, with the release of CICS Transaction Server, IBM began providing facilities that allow the use of connectors to access legacy applications, so you can choose between connector and gateway models based on your needs.

The recent availability of connectors that support both visual (terminal-oriented) and nonvisual (COMMAREA) CICS application types allows remote applications to invoke almost any CICS application as a web service. Because most shops have a mix of CICS application types, companies should seek out this kind of connector to avoid multiple software licenses and additional training on how to integrate the different application types within your organization.

In the connector and gateway examples below, we focus on using Web services with visual (terminal-oriented) applications because they are typically more difficult to integrate than nonvisual (COMMAREA) applications.

Connector Model for Web Services

Connectors allow you to transform your legacy applications into Web services without requiring the use of additional hardware, without changes to the legacy application, and without falling back upon brittle techniques like screen scraping. Compared to gateways, connectors yield better performance by running on the host, and more reliable operation due to the elimination of the many layers data must pass through due to screen-scraping (Figure 3 details these layers). Connectors also provide enhanced access to application information such as state and error codes. This information is lost when data is sent to a terminal emulation technology such as a 3270 emulation client.

Figure 2 shows the basic model for accessing legacy applications as Web services through connector technologies. It illustrates a Web services architecture: a Provider that supplies the Web service, a Requester that uses a Web service, and a

Broker that finds Providers for Requesters. In this model, the legacy application is the Provider. The following steps represent how to find and use a Web service connector (steps 1-3 are optional).

1. The ***Provider*** uploads a WSDL specification to publish its Web service with a Broker.
2. The ***Requester*** (usually a Java or .NET application) queries the Broker for a Web service by name or category.
3. The ***Broker*** selects a Provider and returns the Provider information to the Requester.
4. The ***Requester*** uses the information from the Broker to format and send a SOAP message to the service provider.
5. The ***Provider*** returns a SOAP/XML response to the Requester with the legacy application data enclosed.

Gateway Model for Web Services

Unlike connectors, gateways typically run on a physical or logical middle tier. Where the gateway runs is important because there are so few options for accessing the host from the middle-tier servers, which means gateways usually involve some form of screen-scraping. The solution is tightly coupled in that the integration is between the gateway and a specific application. Any changes to the application will break the integration.

When gateways communicate with terminal-oriented legacy applications they open a terminal session with the legacy application, send a request to the application, receive the terminal datastream, use HLLAPI to capture the screen data, process the screen data, convert the contents to XML, and ship the XML document to the requester. (A variation of the gateway model is to use FEPI on the mainframe instead of a middle-tier terminal emulation client. This simply moves the middle tier onto the mainframe.) The most common components of the gateway model appear in Figure 3.

Gateways allow you to get an application into the Web services mix, but screen scraping creates performance bottlenecks and multiple points of failure between the legacy application and the Web service. For this reason, gateways are best for short-term projects, either as a transition to using connectors or as a stopgap measure during application reengineering or platform migration.

Figure 4 shows the basic model for accessing legacy applications indirectly using a gateway technology such as a screen scraper.

Again, the diagram in Figure 4 illustrates a Web services architecture: a Provider that supplies the Web service, a Requester that uses a Web service, and a Broker that finds Providers for Requesters. The legacy application is not itself a Web service, but is accessed by an off-host Provider. The following steps represent how to find and use a Web service gateway (steps 1–3 are optional).

1. The ***Provider*** uploads a WSDL specification to publish its Web service with a Broker.
2. The ***Requester*** (usually a Java or .NET application) queries the Broker for a Web service by name or category.
3. The ***Broker*** selects a Provider and returns the Provider information to the Requester.
4. The ***Requester*** uses the information from the Broker to format and send a SOAP message to the Provider.
5. The ***Provider*** starts an emulation session and conducts a series of transactions with the legacy application to collect the requested data.

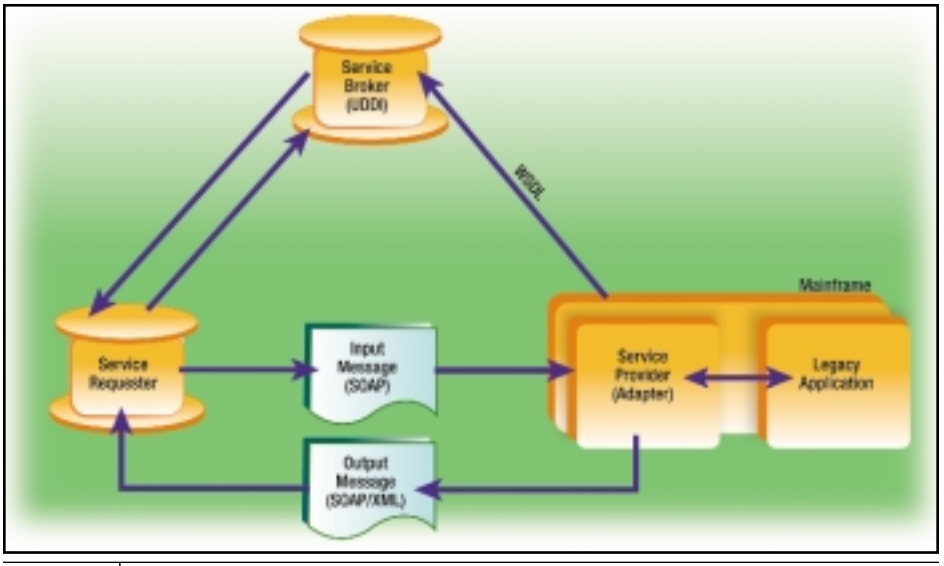


FIGURE 2 | **Connector model**

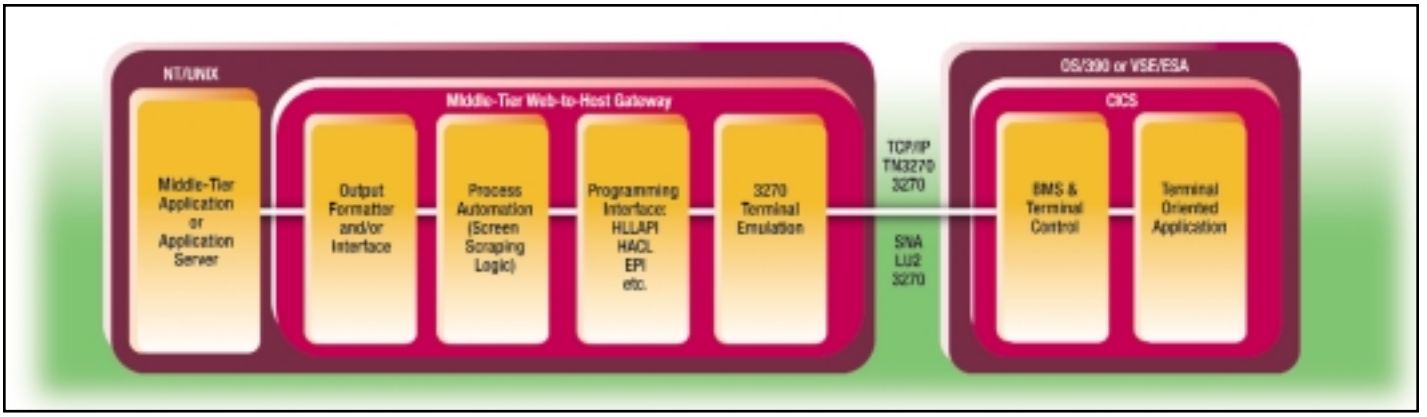


FIGURE 3 | **Typical middle-tier gateway architecture**

WowGao

www.wowgao.com

6. The ***Provider*** converts the legacy data stream into an appropriate SOAP/XML response and returns the response to the Requester with the legacy application data enclosed.

Dynamic Connectors vs Code Generators

As we have seen, connectors provide several advantages over gateways when it comes to Web services integration. How those connectors operate can be just as important. Products that implement the connector model for Web services usually fall into two camps: those that use code generation and those that are dynamic. Code generators have been around for some time and require developers to adhere to a fairly structured process. In the case of CICS applications, something like the following takes place:

1. Download program source code, copybooks, or screen maps to a workstation running a proprietary tool.
2. Use a proprietary tool to generate Web services "wrapper" code.
3. Upload generated code to the mainframe.
4. Compile, install, and test the generated code.
5. Repeat this process for each program and, again, whenever a program changes.

While generated connectors can jumpstart the initial development process, they create several maintenance problems. First, the integration developer is generating code off the host and uploading the code to the mainframe. In most cases, this developer will know a lot about Web services and XML, but will know little about the mainframe. Thus, any changes to the mainframe application require coordination between the Web developer, the application developer, and the CICS administrator. Second, generated connectors create "net new code" that must be managed. Changes to either the legacy application or to the Web service will require repetition of the code generation process to keep the generated code in sync with the integrated application. Without proper management, you are likely to drown in a sea of generated code.

These issues led to the development of dynamic connectors. Dynamic connectors operate with little or no configuration and changes to legacy applications are automatically incorporated into the SOAP/XML output. In many cases there is no configuration required, while some cases may require a single-step process to specify Web service information for each application. As a result, there is no generated code and there is a clear division of labor: the CICS administrator installs the connector and the Web developer simply invokes the connector as a Web service.

> ***"Web services are not a trend, but an industry-wide movement that can provide a long-term solution for companies that want to integrate legacy applications and data with new e-business processes"***

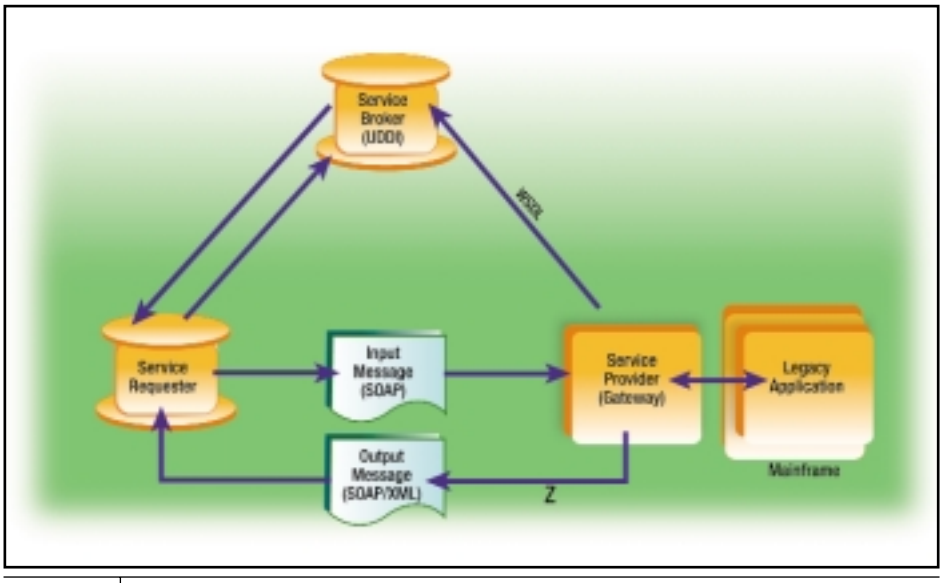


FIGURE 4 | **Gateway model**

Conclusion

The subsystem under which your legacy application runs determines your top-level integration choices. IBM's CICS Transaction Server includes facilities that allow third-party vendors to create connectors that can immediately enable legacy applications as Web services. These facilities provide additional benefits over gateways, such as improved performance and increased stability compared to their screen-scraping counterparts. By using the same industry-standard technologies as Web services, some connectors make it possible for applications to transparently invoke CICS transactions within a Web services architecture and receive the resulting data as well-formed XML. For organizations that want to retain the value of their CICS applications, the combination of XML-enabling connectors and Web services offers a practical and powerful integration solution.

Web services are not a trend, but an industry-wide movement that can provide a long-term solution for companies that want to integrate legacy applications and data with new e-business processes. In the end, companies need to assess the value of the data contained in their CICS applications. Most companies have already determined that such data is highly valuable and they are looking for ways to preserve their investments. Given that recent surveys show the top strategic priorities of CIOs and CTOs are integrating systems and processes, the use of Web services for legacy integration will grow rapidly.

Reviewed by Paul Maurer

Author Bio:
Paul Maurer is a principal in the Financial Services practice of a leading consulting services company.
PAUL@PAULMAURER.NET

Groove Workspace 2.5 from Groove Networks

Getting into the Groove!

In my spare time I participate in a nonprofit entity along with a number of other busy colleagues. Most of the hard work is done in our spare time and we occasionally meet to polish the work.

We started out collaborating in the endeavor in the typical ad hoc fashion, using e-mail for discussion, activity coordination, and file sharing. This mechanism quickly became awkward and unworkable. Long e-mail conversations lack context, unless of course you include the previous message. But then, after a few exchanges, distinguishing the response from the context becomes awkward.

It quickly became clear that some sort of collaboration tool was in order.

Enter Groove

Then a colleague recommended Groove Workspace, a peer-to-peer collaboration product from Groove Networks. The Groove Workspace (see Figure 1), or just "Groove," allows users to create a workspace that can contain a number of tools such as Calendar, Contact Manager, Discussion Manager, File Sharing, and more. Once the workspace is created, the owner can share the workspace with others by sending them an invitation. When a user accepts a Groove Workspace invitation, the workspace is replicated to their local Groove desktop. Users can then work in the environment and Groove manages the synchronization among all users of the workspace.

Features

Here are a few of the features you'll find in Groove 2.5 Workspace:

- ***Autonomy:*** Groove is a desktop application that requires no server setup. Users are in full control of their shared spaces!
- ***Distributed transparency:*** Seamlessly connects users across enterprise boundaries without imposing special setup requirements.
- ***Disconnected productivity:*** Allows the user to continue to work while disconnected. Work is stored for later update when the user is back online.
- ***Context:*** All information shared by the team is captured in a single space. Members can see the entire history and progress of work in the space.
- ***Security:*** All content is encrypted on disk and over the wire.
- ***Web services:*** Best of all, features can be accessed as a Web service.

Enterprise Integration Server

Groove Networks has created an integration server that allows companies to deliver enterprise information through the Groove Workspace environment, allowing automated discovery and connection to users on the desktop. This feature is delivered through agents (Bots in Groove terminology) that plug into this server.

Agents can be developed to manage communications between Groove shared spaces and a variety of external systems such as ERP, CRM, and knowledge management systems.

The Web Service Groove

You may ask, "What does this have to do with Web services?" Well, as Groove Workspace has moved into the enterprise workplace, it has become clear that, although the integration server works well at bringing enterprise resources into the workspace, the ability to expose Groove's capability to other applications on the desktop and in the enterprise is still needed.

Enter Web services. Groove has decided to expose its core functionality as Web services for access by other parties through SOAP. This makes it possible to access Groove data from a separate process using any number of SOAP-enabled toolsets.

Local Versus Remote

The current version of Groove Work-

COMPANY INFORMATION
Groove Networks, Inc.
100 Cummings Center
Suite 535Q
Beverly, MA 01915
Phone: (978) 720-2000
Fax: (978) 720-2001
Web: www.groove.net
E-mail: info@groove.net

DOWNLOAD INFORMATION
Groove Workspace 2.5: www.groove.net/downloads/groove
Web Services GDK: www.groove.net/devzone/default.cfm?pagename=GWS_GDK

LICENSING INFORMATION
Developer Licenses:
Groove Developer Kits Free
Developer Partner (Premium Support) $1495
Groove Networks recommends that developers purchase the Professional Workspace Edition (see below)
Runtime License:
Groove Professional Workspace Edition $149
Groove Workspace Professional Edition Upgrade $105
Groove Workspace Standard Edition $69

TESTING ENVIRONMENT
OS: Windows-XP Professional (Service Pack 1)
Hardware: Intel Pentium III: 996MHz – 512MB RAM

FIGURE 1 | **Groove Workspace**

space allows Web service access from the same machine as the running workspace – called Local Groove Web Services. This release also includes a preview version of Remote Groove Web Services, which allow access from other network endpoints. The Remote version is considered a preview because Groove Networks has not fully worked out the security model and should therefore not be used in production environments. I'll focus on the local Web services.

Getting Started

In order to start developing a local client you'll need to install a copy of the Groove Workspace application as well as the free Groove Web Services GDK (Groove Developers Kit). The GDK comes with ample documentation, sample SOAP clients, WSDL and XSD files, and utilities. Support for developers is provided through the Groove "devzone" site.

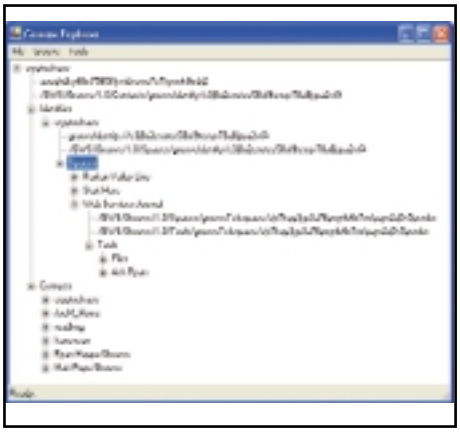


FIGURE 2 | **Groove Explorer**

The samples provided with the GDK are written in C# and Perl. I was able to compile and run the C# Groove Explorer client quickly and without a hitch. Keep in mind that you'll need to have the Groove Workspace running in order for the clients to work. Otherwise, they'll just time out with a connection error. If the workspace is running but you aren't logged in, the authentication panel will pop up when a SOAP request is sent to the GrooveAccounts Service. This may seem odd initially but makes sense since only local clients in the same Windows account can send requests to the GrooveAccounts Service and are likely to have come from an integrated desktop application.

The Groove Explorer shown in Figure 2 gives you a good feeling for the types of information you can request by showing a hierarchical view of the Groove Environment.

Security

The Groove Environment enforces Local Web Service access restrictions by requiring a special Groove Header in the SOAP request that must contain a "WebServices-Nonce" value. This value is generated and stored in the Windows registry each time the Groove Workspace starts. SOAP clients must retrieve this value from the registry and pass it in the Groove Header along with each SOAP request. The Groove Environment checks this value for each request and rejects those that don't match. The mechanics of the access to Windows registry data ensure that the request has come from a client running under the same Windows account. As a developer who favors the Java programming environment this led to some consternation. Java doesn't provide for Windows registry access (it's not portable) so I would need to write a small native method and call it through JNI (Java Native Interface, Yuk!). Luckily the Groove Developers Zone provides sample JNI code to assist with this requirement.

SOAP Endpoints

Since the Groove Workspace is a very fluid environment, workspaces, tools, and other features can appear or disappear from the environment due to changes by its members. This means that SOAP endpoints also become moving targets and must be generated on the fly. Most Groove services return URIs that can be used to generate POST URLs for SOAP requests. For example, The GrooveAccounts Service returns URIs for all GrooveSpaces for an account. Each URI is appended to "http://localhost:port-number" and used as the SOAP endpoint for a GrooveSpaces SOAP request. This method propogates as each space will return the list of Tools (URIs) contained by it for use as endpoints for SOAP requests to the GrooveTools service.

Conclusion

The Groove Workspace product makes clever use of the peer-to-peer model to facilitate on-the-fly collaboration while making setup and administration almost nonexistent. The enterprise features allow corporations to develop Bots that allow Groove users to access corporate resources from within its collaboration environment. Finally, SOAP access to Groove services allows developers to incorporate and control Groove's features from core business applications.

Costco Electronic Hardware Services

Taking risks that pay off

In today's business universe, the boardroom mantra ringing out across the globe is "control costs at all cost." Executives that have survived in a difficult business environment know that if you want to compete successfully, you must have increasing transparency into the costs of doing business.

AUTHOR BIO:

Bringing a unique blend of business, sales and computer experience, Jennifer Peers has a distinctive approach to interviewing a client and encapsulating the business value and benefits of their particular situation.

As a direct result of the benefits offered by the Microsoft .NET platform, sweeping new improvements in information systems are helping managers to understand and drive down costs in managing both the goods and the service business. By leveraging the power of Teamplate for .NET with MBS Great Plains and an array of Microsoft .NET applications, system consultant and integrator Tectura has helped Costco Wholesale dramatically reduce the costs associated with managing its computer and peripherals business.

Costco Electronic Hardware Services: A Case Study

Costco Wholesale Corporation operates an international chain of membership warehouses, mainly under the "Costco Wholesale" name, that carry quality, brand-name merchandise at very competitive prices. Its Electronic Hardware Services (EHS) division was created in 1999 to recover cost and reduce losses on computers and peripherals returned by Costco members. Costco's EHS central refurbishment site handles returned merchandise from 346 Costco locations in North America and Puerto Rico. During the process, computers and associated peripherals are moved through the refurbishment cycle – repaired, tested, cleaned up, repackaged, and sold as refurbs. The process also includes filing and tracking warranty claims with manufacturers.

The mission of Costco EHS is to provide the highest quality, lowest cost refurbishing service whether through its in-house facilities or through third-party refurbishers. In order to gauge the effectiveness of its in-house processes, Costco EHS compares its own results to the equivalent refurbishment services offered by its partners.

Tracking and Accessibility Taken to Another Level

The key to determining the cost-effectiveness of in-house refurbishment is the collection of statistics and application of cost-accounting principles at various points in the refurb process. Data collection had to satisfy the following requirements:

- Unique tracking of inventory costs at discrete steps within the refurbishment process and integration with financial systems
- Track peripherals that were linked to serialized and reserialized CPUs and also linked to the receiving PO at the originating warehouse
- Create a single database that 60 people at two locations could access as a single system
- Automate processes within the refurbishment cycle
- Create, submit, and track warranty claims
- Gather support metrics to provide a clear indication of business profitability
- Operate in a Microsoft-standard environment

Workflow and customization of the accounting software were essential to the solution. The real business pain that EHS was trying to alleviate was its inability to accurately track inventory during the refurbishment process and the collection of decision support metrics. In addition, over 60 EHS employees were at two different locations – corporate headquarters in Issaquah and Sumner, Washington – including skill-specific technicians, warranty claim administrative staff, shippers, receivers, inventory management, parts clerks, sales representatives, and operations management. The tracking and measurement system had to provide the capability to have all employees act as one team and replace multiple site-specific databases with a single, secure, high-performance database.

One of the main challenges was the sheer volume of returns processing. In one week, moving refurbished items from "receiving" to "saleable" totaled 1,000 items received to 600 items in saleable condition. Each item moves through inventory transaction sites up to eight times during the refurbishment process – over 7,000 inventory transactions per week would be required to clearly understand the finer details of the business. Prior to the solution implementation, the refurbishment site was tracking only two inventory transaction sites – Received or For Sale. The Teamplate for .NET solution now provides the sales staff with real-time status of the refurbished items prior to their availability for resale.

Shay Reed, EHS Manager, along with teams from TECTURA and Teamplate, leveraged the strengths of Microsoft Business Solutions and the Teamplate for .NET workflow automation tool to develop a solution to meet Costco EHS's unique requirements. This solution provides

the ability to track equipment at the serial number level, and also enables EHS to track the components and peripherals related to a specific serial number as distinct inventory items through the entire refurbishment process. The system also sustains the relationship of "item" to "parent serial number" through to the originating warehouse. The item cost is also captured to compare it to the market resale value at the resale step. The ability to track 7,000 inventory site movements per week was made possible using the Template for .NET workflow automation capabilities and creates the ability to determine profit or loss for a particular item.

As a computer moves through the refurbishment process, warranty claims to the manufacturers for parts and labor are initiated at the click of a button. Teamplate for .NET launches a new process enabling the Warranty Claim Administrator to enter the warranty claim at the manufacturer's site and tracks the specific warranty claim until it is reconciled in the financial system. Previously, the majority of warranty claims were on the larger items, such as CPUs and monitors. Other peripheral items, like keyboards and mice, were not as frequently claimed due to the difficulty in tracking the peripheral and linking it to a specific CPU serial number.

The automated process also helps to ensure the appropriate skill level and cost labor is being utilized to maximize resources available.

The Teamplate for .NET interface automates inventory transactions and updates the Great Plains database, replacing the site-specific databases and providing the exact status of inventory position. Now able to break down the available information to the component level, EHS can effectively manage product flow and business activity to ensure the highest cost recovery position. For example, EHS may identify a number of inventory items to move through the refurbishment process more quickly to meet a sales order delivery date and create a higher profit.

Teamplate for .NET is the user interface at the Sumner, Washington location. It provides management of the automated workflow affecting the appropriate use of business rules of the refurbishment life cycle. At the Issaquah corporate location Great Plains and Teamplate for .NET operate in tandem as the user interface, providing financial reporting and analysis capability.

After reviewing nine different possible solutions, Costco selected MBS Great Plains integrated with Teamplate for .NET. The selection was based upon flexibility, expandability, cost, and the ease with which the average user could understand and apply the program to their daily functions. Technically, the solution had to be compatible with the Microsoft product family currently in use at EHS in order to provide data integrity and seamless integration. Additional criteria, including financial stability of the vendors and ongoing support were considered prior implementation.

As a result of the success realized with the refurbishment tracking and the analysis system, other business areas have been earmarked for automation with Teamplate for .NET workflow. They are:

- Interfacing technical calls with the refurbishment process
- Integrating the warehouse receiving
- Incorporating parts inventory into Great Plains
- Automating the sales process

> ***"…there were no limitations or extended development efforts encountered in working in the .NET world"***

- Determining additional cost recovery through innovative packaging processes

The Power Behind Automation

Managing the business from a workflow perspective is an innovation – knowing the status of processes in real time provides the division with the visibility into "soft costs" that were previously unavailable. Now, Costco EHS can determine the real costs of their repair business, which items provide the highest cost recovery to repair, or which inventory to use for particular end products. All of this provides management with information that enables a streamlined, successful operation.

The unique requirement of tracking inventory through the refurbishment process has created an innovative way to apply standard financial accounting software. Perpetually tracking the inventory item, and its cost at a specific process step, has created the "Specific Life Cycle Cost Tracking" accounting approach. This required unique customization of the Great Plains system and the Teamplate for .NET workflow. In most cases, companies track standard costs by aggregating product costs. Costco EHS wanted to break every item received into its many component pieces, cost it, and then tie it back to the originating Costco location at any given time.

By providing low-cost, high-quality refurbishment services, Costco EHS reduces the cost associated with returned merchandise. Ultimately, by reducing the cost of returned computers, EHS provides Costco with a means to achieve their business objectives – leveraging creative ideas to exceed business expectations and providing its members with cost-effective products. In the computer market, this translates to consistently pricing equipment more competitively and minimizing inventory write-downs.

The innovative techniques of automating the refurbishment process have created another unique position in the marketplace. Now that EHS controls refurbishment by serial number, Teamplate for .NET instantly flags if a serial number has been through the process before and captures how it arrived at the site a second time. EHS can now sufficiently track the item to maintain program standards originally set by the manufacturer, ultimately providing a higher quality product for resale and, therefore, improved cost recovery. Before the solution was implemented, it was impossible to provide proof that every component in a unit had been through the refurbishment process and met those standards. In turn, this provides confidence to the warehouse and legal department that all items sold have met Costco's high standards, and eliminates the risk of legal action.

Positive Impacts

Several business problems were solved by Teamplate for NET. The most impactful were:

- Unique tracking of inventory costs at discrete steps within the refurbish process integrated with financial systems
  - *Business Impact:* Information captured during the refurbishment process is automatically integrated with financials, enabling management to make business decisions with real data
- Track peripheral items to serialized and re-serialized CPUs while linking the receiving PO to the originating warehouse
  - *Business Impact*: Clearly identifying costs associated with a single item to establish a true chain of costs associated with the refurb process
- Create a single database that 60 people in two locations could access as one
  - *Business Impact:* Confidence in data integrity
- Automate the refurbishment process
  - *Business Impact*: Better resource planning and utilization, reduced training on how refurbishment process works and reduction in administrative tasks for technical staff
- Create, submit, and track warranty claims
  - *Business Impact:* Increased capability to submit all eligible warranty claims and increase cost recovery percentages
- Gather support metrics to provide a clear indication of business profitability
  - *Business Impact:* A true cost recovery view is now available, triggering plans for increasing cost recovery and reduction of administration efforts on two third-party programs
- Operate in a Microsoft-standard environment
  - *Business Impact:* Successful implementation of an integrated solution and rapid development of changes or enhancements as the business dictates

One of the most dramatic results from implementing the Teamplate for .NET solution was the ability to identify repeat trouble patterns before the manufacturer did. In one example, this observation resulted in 100% cost recovery from the vendor and Costco was proactive in removing the item from sale, thereby avoiding further returns. Prior to EHS, these defective, returned computers would have been sold to a standard salvager at 22 cents on the dollar. This 100% cost recovery was valuted at more than $300,000 compared to $88,000 from salvage – a 354% increased cost recovery in this instance.

By controlling refurbishment in-house, Costco EHS is saving costs associated with hiring third-party refurbishment companies. These costs include flat refurbish rates of $55 to $90 per unit and a sales commission, plus the cost of replacing necessary parts. The average cost of third-party refurbishing is $150 per unit.

The Technology

The integrated solution incorporates the following technologies:

- Great Plains, Version 6.0, 10-user license: Implemented modules include:
  - General Ledger
  - Inventory Control
  - Sales Order Processing
  - Purchase Order Processing
  - Receivables Management
  - Payables Management
  - Landed Cost
  - Refund Checks
  - Customer/Vendor Consolidation
  - FRx Desktop
- Teamplate for .NET
- Microsoft SQL Server 2000
- Microsoft Windows 2000
- Microsoft Exchange and Office
- Microsoft Access
- - Microsoft Internet Explorer

> *"knowing the status of processes in real time provides the division with the visibility into 'soft costs' that were previously unavailable"*

Costco EHS provides Costco with a flat rate of $50 per unit without any additional cost for the part or sales commissions. This is a saving of $100 per unit – a 66% reduction in refurbishment costs compared to industry standards.

Next year's business plans consider bringing all units currently sent to third-party refurbishers in-house. Based on projected volume, estimated cost recovery for that year will be over 3 million dollars – an increase of 66%.

The Next Frontier

The entrepreneurial spirit that Costco is recognized for is founded on taking risk, changing, and adapting to change to facilitate business.

Costco EHS was looking for a powerful solution secure enough to handle their needs but with unlimited flexibility to add new functionality to the solution, as their business needs changed. This directly corresponds to the decision to go with a .NET platform. As an early adopter of this technology, EHS decided to take a risk on the newest technology on the horizon. Weighing the implementation costs and the life expectancy of the solution with the stability of the .NET environment and its future capabilities, EHS was able to make their product decision. They also had a very good comfort level that TECTURA and Teamplate would create the right solution and support any issues encountered.

The decision to go with .NET directly translated into quick, low-cost development of application enhancements to meet the ever-changing demands of Costco EHS. Most changes are expected on the fly. Creating solutions to difficult issues and working way outside of the box, TECTURA found there were no limitations or extended development efforts encountered in working in the .NET world. Flexibility within the solution was key and the speed of development was evident in creating the new process. After requirements were determined, the solution development and implementation happened in less than 14 days.

An important EHS philosophy on successful technology implementations is to minimize the number of platforms, products, and vendors applied to a particular solution. There is also a requirement for using core Microsoft products with a SQL foundation. In this way, data integrity, which is of primary importance, can be more easily maintained.

EHS is looking forward to a continued partnership with TECTURA, automating additional processes that can benefit from Teamplate for .NET as well as integrating with Microsoft Business Solutions.

Written by Sean Maxwell

Direct Sales Notification Service

Simplicity that can meet – and then exceed – the clients' needs

Marketing executives across the country love to know how well their business is doing, especially how much money it's making. This need has produced a number of solutions across the market in all ranges of complexity.

This article outlines one way executives and marketing teams can be easily updated on the revenue figures driving important business decisions.

As a system grows in volume and the business team looks for more real-time data, IT professionals may be tempted to use direct database queries to check on sales results. More technically savvy individuals may simply open a Query Analyzer session and issue dynamic queries to get the information they are looking for. Certainly, this type of behavior raises many potential problems, including security concerns, performance and locking issues in the database, and the possibility for users to issue poorly performing dynamic queries against the production transactional system.

By leveraging the power and ease of Web services for distributed data access, development teams can easily provide interesting bells and whistles, along with the core functionality of the application. If you choose, as in this example, to use a richer desktop application, you can still have centralized data access, database security, and performance control. It isn't necessary to support dozens of users doing direct database access. Database security can still be maintained easily, and firewall rules don't have to be changed as long as the contract (the WSDL) isn't altered during production environment migration to a .NET Framework. The client does not need to be updated and will work transparently with the new version.

A Richer Application

The solution we came up with involves a desktop/taskbar application that makes calls to a Web service residing on an application server servicing multiple production e-commerce Web sites. Each Web site uses an ASP front end that is driven with Visual Basic 6.0 components connected to a SQL Server 2000 database. The sites reside on a series of load-balanced Web servers running Microsoft Windows 2000 Server.

The server side of the application started out as a simple ASP page receiving XML requests from the client and made a single database call to retrieve the sales figures in an XML format. Once we had SOAP installed in production, it was obvious that we needed to leverage the technology to create a more flexible Web service. The current version now uses the SOAP 3.0 toolkit with an ASP broker and VB 6.0.

One of the requirements of the server application is that it not put unnecessary stress on the server itself. We wanted to make sure that we don't make database calls every time a request is made for the data. This was accomplished by programming every SOAP call to check when the last call was made. To keep this information in scope the last request date and last response date are placed into ASP application variables. If the current request is within a preset time limit, the old response is returned; otherwise, the service goes to the database for updated information.

Since keeping the data in scope is tricky, a few methods were examined, including keeping the data in a separate table within the database or writing it to a file. Since the original version did all of the work in an ASP page, we decided to use the application object that is created with every IIS application. Once the server component started using SOAP, we realized we still wanted to use this object even though all the functionality was moved to a VB6 component. To keep using the application object, a reference was made to the COM+ services–type library. Using this library, and the Microsoft Active Server Pages object library, we can access the current application object that is in context. This is where the last request time and response are held in memory. The first thing that must be

> *"An e-commerce site can use a simple device such as this to give a marketing team or business executive exactly what they need"*

Author Bio

Sean Maxwell is a senior consultant with Fusion Technologies, Inc. He has four years of experience in IT development, specializes in Microsoft technologies, and is an expert in C#, VB.NET, VB, SQL, and scripting languages. MAXWELLS@FUSIONTECH.COM

done is to set the MTSTransactionMode property of our class to NoTransactions and then create a new COM+ server application, in the Component Services Manager, that includes this component. The next four lines are all the code needed to reference the current instantiation of the application object:

```
Dim oContext As COMSVCSLib.ObjectContext
Dim oApplication As
ASPTypeLibrary.Application
Set oContext = GetObjectContext
Set oApplication = oContext("Application")
```

One of the difficulties in using a Web service is dealing with complex data types. A common way to keep data in an easy-to-use format is to simply return ADODB recordsets to calling functions. If you need to send or return a complex type with a Web service, extra steps are needed to create a complex type node in the WSDL. To keep the data in an easy-to-use XML format we use the "For XML Explicit" option in our SQL statement. This way we can specify the format of the data, and SOAP can treat it as a string. This makes the SOAP implementation much easier than if we tried to return the recordset.

The client application is a Windows application written in C# and consists of a few main pieces. The first is a small Windows form that contains a tab control with two tabs. The first tab shows the sales of each Web site returned from the Web service. The second contains all the configuration options available to the user. The tab control was chosen because we want to keep the form small, and allows the flexibility required for any future enhancements. The second part of the client is a simple notify icon control that will sit in the task bar. This icon will show the current total sales for all sites. It will have the same functionality as most taskbar icons, with a right click bringing up a context menu. The only way to close the application is an option on this menu.

The focus of the client is the sales alert functionality. This popup-like form shows the current sales at user-defined intervals. When the client first starts, it will call the Web service to get the current sales. We use the Timer control to get the sales at regular intervals. When the Tick event of the Timer occurs, the Web service is called and returns the current sales. If the new sales amount is greater than the amount the user has chosen, we show the user the alert. The popup form then rises with the new total in its center (the form appears for only a few seconds and then disappears without stealing focus from the user's current workflow).

This application shows just how insignificant the technical issues dealing with interoperability are when using a Web service. We have a VB 6.0 Web service that calls a stored procedure that does all the work. If we called this function without the Web service, we would have to worry about the cost of interoperability. However, since we are using the Web service this is a non-issue.

Using .NET in our client makes calling the Web service simple. The first thing is to create a Web reference. If we right-click on the References node in the Solution Explorer of the .NET IDE we can choose to add a Web reference. In the window that pops up, we can enter the URL and click the Go button. The IDE will then search the URL for the specified WSDL file and give us a list of functions available that we can use. Clicking the Add Reference button at this point will finish the process.

Now that we have a Web reference, all we have to do is call the function. Just by treating the Web reference as an object reference, we can call our function. Since the data of our service is returned as a string of XML, we can use a few lines of code to create an XmlDocument that can be used through the entire application (see Listing 1).

Once all the real functionality is finished, we need to polish the user interface a little. We have done this by adding a few bells and whistles – literally. One of the options we've given the user is to play a sound when the popup occurs. If they choose to play the sound, we decided to use a cash register "ka-ching" sound every time we see the popup.

After researching our options, we decided to use the Windows API to play the sound. We create the reference to the PlaySound function in WinMM.dll with the syntax:

```
[DllImport("Winmm.dll")]
public static extern bool
PlaySound(byte[] data, IntPtr hMod, UInt32
dwFlags);
```

Our sound has been added to the project as an embedded resource, which can be played on command using the function in Listing 2.

Conclusion

This application is just a simple, yet robust tool that can be used before the sales volumes can truly support a dedicated real-time reporting infrastructure. An e-commerce site can use a simple device such as this to give a marketing team or business executive exactly what they need. This application is just a simple mechanism that meets the clients expectations, then exceeds them.

Listing 1

```
private System.Xml.XmlDocument
GetSalesFromService()
{
 System.Xml.XmlDocument oSalesXml =
new System.Xml.XmlDocument();
 MyWebReference.DirectSales oDirSales
= new MyWebReference.DirectSales();

oSalesXml.LoadXml(oDirSales.GetSales(
));
 return oSalesXml;
}
```

Listing 2

```
public const UInt32 SND_ASYNC = 1;
public const UInt32 SND_MEMORY = 4;
public static void
PlayWavResource(string wav)
{
 // get the namespace
 string strNameSpace=

System.Reflection.Assembly.GetExecuti
ngAssembly().GetName().Name.ToString(
);
 // get the resource into a stream
 Stream str =
System.Reflection.Assembly.GetExecuti
ngAssembly().GetManifestResourceStrea
m(strNameSpace +"."+ wav);
  if ( str == null )
   return;
  // bring stream into a byte array
  byte[] bStr = new Byte[str.Length];
  str.Read(bStr, 0, (int)str.Length);
  // play the resource
  PlaySound(bStr, IntPtr.Zero,
SND_ASYNC | SND_MEMORY);
```

Download the code at
sys-con.com/webservices

Written by Alastair Green

Transacting Business with Web Services PART I

The coming fusion of business transaction management and business process management

Business transaction management (BTM) is a promising new development in general-purpose enterprise software. Most large companies are devoting significant resources to the problem of reliable, consistent integration of application services.

BTM offers previously inaccessible levels of application coordination and process synchronization, radically simplifying the design and implementation of transactional business processes. Business process management (BPM) needs to be enriched by BTM for users to see the potential value of BPM realized in practice.

XML is already widely deployed as a useful *lingua franca* enabling the creation of canonical data standards for particular industries, trading communities, and information exchanges. The extended family of Web services standards (clustered around the leading duo of SOAP and WSDL) is gaining growing acceptance as an important way of providing interoperable connectivity between heterogeneous systems. Many organizations are also examining the use of BPM technologies, exemplified by the current OASIS initiative, Web Services Business Process Execution Language (WS BPEL). Increasingly, attention is turning to the special problems associated with building transactional business processes and reliable, composable services. This is where BTM technology comes into its own.

In this article – which follows the excellent introductions by Mark Little and Jim Webber to WS-Coordination, WS-Transaction, OASIS Business Transaction Protocol, and BPEL (*Web Services Journal*, Vol. 3, issues 5–8) – I'm going to look at the rationale for and current status of BTM, and how vendors and users are thinking about the integration or fusion of BTM with BPM, particularly in the OASIS BPEL standardization effort. BPEL, as a special-purpose programming language designed to make processes portable across different vendors' execution engines, can become a very useful standard programming interface for business transactions in the Web services world.

> ***"The extended family of Web services standards... is gaining growing acceptance as an important way of providing interoperable connectivity between heterogeneous systems"***

Why Is BTM Needed?

Companies would like to minimize the costly, slow, and risky business of "reconcile and repair." Automating or improving the internal flow of transaction processing, spanning multiple disparate applications, is a major issue for most corporations: the finance industry's focus on straight-through processing is an emblematic case. The business model of the Internet (which can cynically be summarized as "make your customers your clerks") continues to drive efforts to offer automated transactional platforms or portals directly to consumers and to business clients. This push towards "self-care" makes reliable, consistent, and recoverable composition of back-end services more important, as inconsistencies and failures become quickly visible outside the call center or the operations room. Equally, customers and suppliers who transact business electronically need well-formed protocols which model the interactions of their business relationship: they want near-instant certainty and "assured common knowledge" about the outcomes (good and bad) of those interactions.

Automated business transactions are a new category, wider than historical data-centric local or distributed transactions. This "third generation" of transaction management builds out from core transactional technology, particularly the concept of a

Author Bio

Alastair Green is CEO/CTO and cofounder of Choreology Ltd, the business transaction management (BTM) company. A coauthor of the OASIS Business Transaction Protocol standard and an active member of the OASIS WS BPEL committee, he has spent over 20 years helping produce distributed transaction products for software vendors, and deploying them in end-user business processes, particularly in banks.

ALASTAIR.GREEN@CHOREOLOGY.COM

www.LinuxWorld.com

The Leading Magazine for Enterprise and IT Management

LinuxWorld Magazine

There is no escaping the penetration of Linux into the corporate world. Traditional models are being turned on their head as the open-for-everyone Linux bandwagon rolls forward.

Linux is an operating system that is traditionally held in the highest esteem by the hardcore or geek developers of the world. With its roots firmly seeded in the open-source model, Linux is very much born from the "if it's broke, then fix it yourself" attitude.

Major corporations including IBM, Oracle, Sun, and Dell have all committed significant resources and money to ensure their strategy for the future involves Linux. Linux has arrived at the boardroom.

Yet until now, no title has existed that explicitly addresses this new hunger for information from the corporate arena. *LinuxWorld Magazine* is aimed squarely at providing this group with the knowledge and background necessary to make decisions to utilize the Linux operating system.

Look for all the strategic information required to better inform the community on how powerful an alternative Linux can be. *LinuxWorld Magazine* does not feature low-level code snippets but focuses instead on the higher logistical level, providing advice on hardware, to software, through to the recruiting of trained personnel required to successfully deploy a Linux-based solution. Each month presents a different focus, allowing a detailed analysis of all the components that make up the greater Linux landscape.

Regular features include:

- *Advice on Linux Infrastructure*
- *Detailed Software Reviews*
- *Migration Advice*
- *Hardware Advice*
- *CEO Guest Editorials*
- *Recruiting/Certification Advice*
- *Latest News That Matters*
- *Case Studies*

SAVE 30% OFF!

REGULAR ANNUAL COVER PRICE $71.76

YOU PAY ONLY

$49.99

12 ISSUES/YR

**OFFER SUBJECT TO CHANGE WITHOUT NOTICE*

SUBSCRIBE TODAY!

WWW.SYS-CON.COM

OR CALL

1-888-303-5282

FOR ADVERTISING INFORMATION:

CALL 201 802.3020 OR

VISIT WWW.SYS-CON.COM

> *"Companies would like to minimize the costly, slow, and risky business of 'reconcile and repair'"*

multi-phase distributed outcome ("two-phase commit" in conventional database/messaging transactions). Business transactions may be of split-second duration, but may also be long-running, i.e., may endure for periods that exceed the anticipated service cycles of participant systems. Business transactions retain the driving ambition of consistency. However, they are more flexible and sophisticated in their means – and often more modest in their goals, reflecting the imperfect determinism of real business processes (which are necessarily designed to cope with innumerable variations and with incremental and partial successes).

- Business rules determine the set of viable outcomes, which may involve noncritical partial failures, or selection among contending service offerings, rather than the strict "all-or-nothing" assumption of conventional ACID transactions.
- Business rules govern the duration and character of participation in a transaction. Provisional results are often revealed deliberately, to allow probabilistic inventory management. Tentative service or product offerings may be withdrawn spontaneously or after a declared interval, even if this disrupts the final outcome.

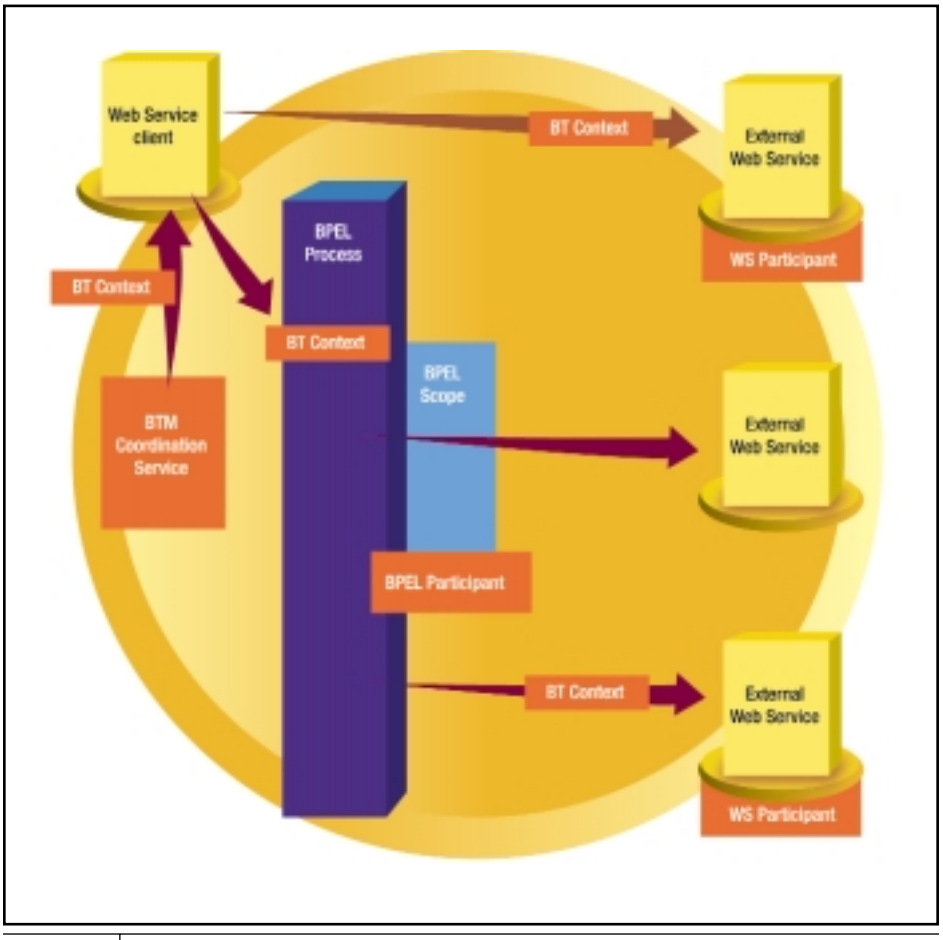


FIGURE 1 | **Example of a BTM coordination service**

Both of these features are relevant to the work of the OASIS WS BPEL Technical Committee, whose charter foresees work over a minimum period of nine months (from May this year to January next) to achieve a recognized standard. In the remainder of this article I'll concentrate on the integration of a transactional coordination protocol with BPEL.

Exception and Recovery Handling in BPEL Today

The current WS BPEL 1.1 draft specification (jointly submitted by IBM, Microsoft, BEA, SAP, and Siebel) incorporates the notion of processes that can contain nested sub-processes or scopes, to any depth. Processes are expected to carry out "real work" (work which affects persistent records and external systems) by invoking WSDL-defined operations. While BPEL processes can define and manipulate variables, these variables are designed for use in tracking process state and controlling conditional flows. If processes wish to communicate with other processes then they can do so using "partner links," which essentially create sets of WSDL-defined services as access points for bilateral message sequences between defined roles in a collaborative exchange.

Processes are initiated by receiving inbound Web service invocations. BPEL, therefore, expects that external stimuli will arrive and substantive processing work will be invoked as Web service documents, typically delivered using SOAP messages.

Processes and scopes can declare fault handlers and compensation handlers, in addition to the normal sequence of "happy path" activities. A fault handler is designed to deal with exceptions during normal processing. A compensation han-

BTM: Protocols and Standards

Full-scale BTM software needs to implement interoperable protocols that define three phases of any transactional interaction (whether integrating internal systems, or automating external trades and reconciliations).

- ***Phase One: Collaborative Assembly***. The business-specific interplays of messages that assemble a deal or other synchronized state shift in the relationship of two or more services. A useful general term for such an assemblage of ordered messages is collaboration protocol. Examples include RosettaNet PIPs, UN/CEFACT trade transactions, and the FIX trading protocol. In the future, BPEL abstract processes should help greatly in defining such protocols. Reliable messaging has an important role in this assembly phase, but as a subordinate part of a new, extended concept of GDP (guaranteed delivery and processing).

- ***Phase Two: Coordinated Outcome***. The coordination of an outcome that ensures that the intended state changes occur in all participant systems, consistent with the business rules or contracts which govern the overall transaction. Examples of relevant coordination protocols are WS-Transaction (Atomic Transaction and Business Activity, supplemented by WS-Coordination) and BTP (the OASIS Business Transaction Protocol) and the recently released WS-TXM (Transaction Management, part of the WS-Composite Application Framework [WS-CAF]).

  A coordination protocol requires three related sub-protocols: a *control protocol*, which creates and terminates a coordination or transaction (present in BTP); *a propagation protocol*, which allows a unique transaction identity to be used to bind participating services to a coordination service (this sub-protocol is mostly defined by WS-Coordination); and an *outcome protocol*, which allows a coordination service to reliably transmit the instructions of a controlling application to the participants, even in the event of temporary process, processor or network failures. WS-T, BTP, and WS-TXM, contain very similar outcome protocols.

  A coordination protocol is typically useful only if it is inherently reliable, utilizing persistent checkpoints and message retries to survive planned or unplanned interruptions in service.

- ***Phase Three: Assured Notification***. Notification of the result of the transaction to the parties involved, ensuring that they're all confident of their final relationship to their counterparties. Ideally, this requires a *reliable notification protocol*, which allows the different legal entities or organizational units to receive or check the final outcome, including partial or complete failures. There are no current examples of standardized reliable notification or termination protocols that I am aware of.

Current BTM products and standards focus on Phase Two, Coordinated Outcome. WS BPEL may prove a useful vehicle for creating a new, wider view of the scope of business tranaction management.

dler is designed to reverse the effect of a completed scope, and can only be invoked when the normal work of the scope is finished (this gives the compensation handler a clean starting point for its reversal activity). Compensation handlers can be invoked by the fault handler of an enclosing scope – which can decide which compensations to process, and in which order – in order to roll back partial work in the event of failure.

Integrating BTM with BPEL BPM

Vendor and user companies in the WS BPEL committee, including Choreology Ltd, are discussing how BPEL will use BTM. (In this context BTM tends to be equated with WS-Transaction Business Activity [WS-T BA]. As we shall see, some features of OASIS BTP must also be taken into account, and the use of WS-Coordination must be made more precise.)

To set the scene, it's useful to look first at the relationship of BPEL processes to Web services. Figure 1 shows a BPEL process, a scope (sub-process) external Web services, and a WS client. Note that a BPEL process may receive messages from a Web service client. It can also call out to Web services, including other BPEL processes that offer Web service interfaces. BPEL processes can also execute nested scopes. (Scopes are effectively processes that happen to be initiated by a colocated process, and have direct access to control variables declared in enclosing scopes/processes.)

BTM features are superimposed on this basic structure. Figure 1 shows a BTM coordination service. (This might be "physically" located within a BPEL execution engine, or operate as a free-standing Web service.) It also shows the use of a business transaction context to connect or relate participant services and processes to a particular business transaction.

In looking at the integration of BTM into WS BPEL the technical committee is likely to consider four main areas:

- How to propagate business transactions between Web services and business processes
- How to propagate business transactions between business processes and nested scopes
- How to initiate and terminate new business transactions within a business process
- How to define the reaction of processes and sub-processes as participants in the progress of a business transaction

In another article, I'll discuss the specific changes proposed for BPEL in order to address these questions.

Get Smart

Connecting services to desktop applications

Smarter client technology is bringing a new class of Web service–powered applications to the desktop. Case in point: PhotoWorks Inc. is leveraging a common Web services infrastructure to power multiple sales channels.

AUTHOR BIO:

Kevin Hakman is the cofounder of Versalent Inc., a leading provider of enterprise client technology. Prior to Versalent, he founded a series of successful emerging Internet technology and ecommerce ventures. Kevin has also written for various business publications.

KHAKMAN@VERSALENT.COM

To date, standardizing and streamlining systems interoperability has been the central objective of Web services. And while the benefits of system-to-system interoperability are key to driving Web services adoption, a complementary trend in the advancement of client technology is enabling enterprises to bring a new class of applications to their customers, partners, and employees.

One of the central reasons enterprise IT shifted to the Web was the significant cost advantages of delivering applications through a browser versus the high costs of developing, distributing, and maintaining incumbent client/server software. For enterprises, it was easy to understand that by reducing the costs of delivering an application to almost zero, the Web offered IT a way to get more done in less time and with smaller budgets. However, many enterprises did not realize that as they were gaining the benefits of low-cost distribution, they were taking a hit on productivity. Forrester Research's December 2002 survey of Fortune 1000 IT executives revealed that "73% prefer to deliver applications in a browser." Yet at the same time, "the majority cited poor productivity and usability" as a key pain that resulted from the shift to browser-based software. Users were frustrated by the limited functionality of each page and the "click-wait-refresh" pacing. The intuitive ease-of-use of the desktop applications to which they were accustomed was jettisoned in the rush to the cost savings promised by switching to Web applications. While IT productivity improved, it was at the expense of user productivity. According to AMR, call centers that shifted to browser-based solutions with the intent of lowering their total cost of ownership (TCO) saw call handle times significantly increase, which had a greater impact on the bottom line. It's easy to see that when compared to the rich interface, speed, and performance of client/server applications, delivering performance through HTML pages just doesn't come close. The key problem has been that enterprises have had to make a tough choice: go with the high productivity, high performance, and high cost of traditional client/server software, or choose the low-cost, slower performance and lower productivity of pseudo-software HTML pages. In a world of declining resources and budget, the near-term savings often won.

> ***"Smart client technologies deliver the high productivity features of client/server software at the lower TCO offered by Web-delivered applications"***

Changing the balance of this equation have been advancements over the past few years in Web services, which, when paired with "smart clients," (a.k.a "rich clients," or "X-Internet clients") eliminate the tradeoff between simple browser-based applications and full-featured client-server software. Smart client technologies deliver the high productivity features of client/server software at the lower TCO offered by Web-delivered applications by combining Internet delivery of just-in-time application code with the computational power of the workstation CPU. Various companies have followed differing technical strategies toward this objective. Technical approaches similar to those offered by Droplets focus on pumping up the browser's capabilities by asking the user to add plug-ins or extra Java runtimes and components. Approaches like that of Curl seek to supplant the browser altogether by providing an alternative client to install and a special server to communicate with that client. Companies such

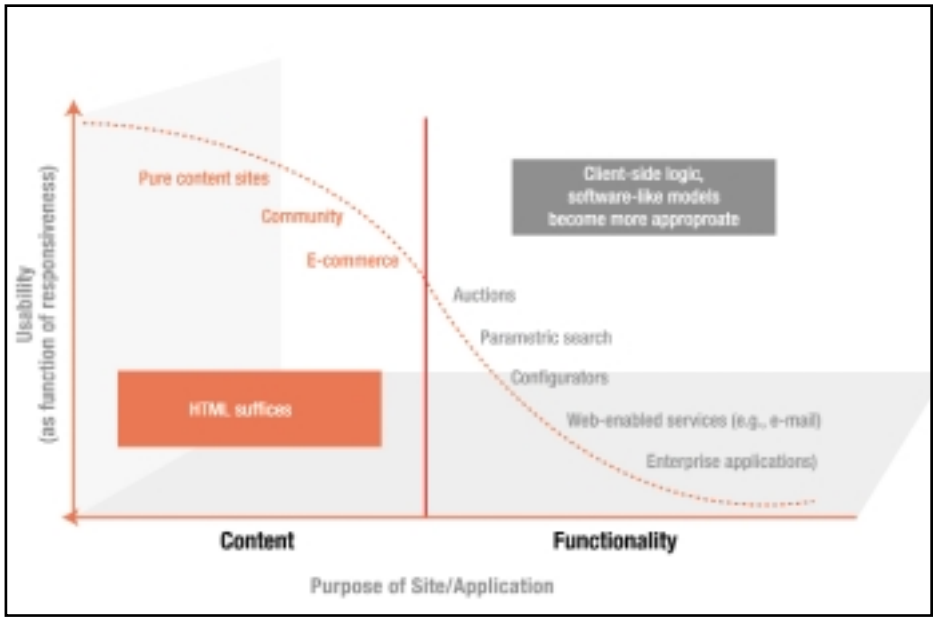


FIGURE 1 **Appropriate Scenarios for Pushing Client-Side Logic (Source: Jupiter Research [11/02] copyright 2002 Jupiter Research, a division of Jupitermedia Corporation)**

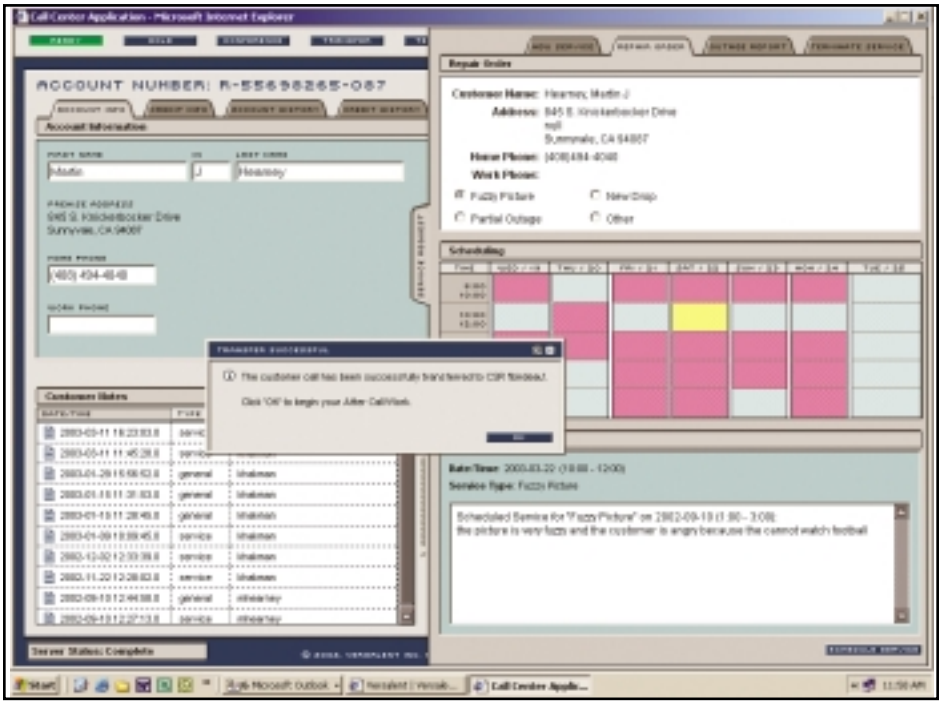


FIGURE 2 **PhotoWorks' new print wizard for Window XP users may look like an installed application, but in fact is delivered instantly over the Internet using Versalent's smart client technology to interact directly with Web services**

as Versalent have sought to better utilize the technologies that already exist within the browser to eliminate the need for extra software installation and leverage capabilities the browser does well, such as security, and communicate directly with any number of Web servers, XML data sources, or Web services.

Web service implementations today face the same constraints that end-user applications have faced historically: the two options remain either lower-productivity Web pages or high-cost installed software clients. A simple analogy makes this problem clear. Web services have often been described as independent stereo system components working together to deliver great sound processing on behalf of the listener. It was the invention of the RCA jack that provided universal interoperability among stereo components that previously had proprietary interfaces. SOAP essentially is to Web services what the RCA jack was to stereo components. But there is the problem of getting the sound these nicely connected components produce to the user. Imagine only having the option to listen through either a low-fidelity gramophone or a set of expensive high-performance speakers. So that enterprises can garner the most value from Web service implementations, new client technology is needed that is aligned with the fundamental benefits of Web services.

Why are Web services and smart clients a perfect pair? Web services can deliver structured data that is independent and agnostic of how (or if) that data ever gets presented to an end user. This makes for the best use of server resources and lets the service interoperate with the maximum possible number of other systems. At the same time, smart clients leverage the computational power of the workstation CPU and in most cases reduce or even eliminate the need to generate HTML on the server, delivering instead the full feature functionality of a thick client by simply communicating via messages and data transfers between the client and the server. This architecture delivers the best performance for the enterprise by making the most of client, bandwidth, and server resources.

The Right Tool for the Right Job

This is not to say that smart clients are always the right tools for the job. HTML page applications and thick clients alike will not soon be extinct by any measure. Instead. the universal "it depends" maxim certainly applies when selecting the client technology that's best for a given need. As Figure 1 (adapted from a 2002 Jupiter Research report) shows, HTML frequently suffices, but for more sophisticated applications, "client-side logic and software-like models become more appropriate".

Case in Point: PhotoWorks Inc.

PhotoWorks Inc., a leading innovator of high-quality photofinishing services for more than 24 years, provides a great case study as they are implementing three classes of client

applications, powered by Web service infrastructure.

The Web site, www.photoworks.com, is the company's main e-commerce site. The site is delivered by a number of servers running ASP.NET processes that communicate with the underlying layer of .NET Web services, then in turn generate the HTML pages that get sent to the user's browser. For the purposes for which the site is intended, the solution delivers a good experience for the end user.

In addition, PhotoWorks offers its users free desktop photo editing and management software. Users can wait for the custom C++ application to download and install, then get access to its rich features. Of course upgrades to the software must be downloaded and installed again and PhotoWorks must concurrently maintain back-end systems that will work with all the various versions of installed software that have been downloaded by users.

In late August 2003, PhotoWorks will launch a print wizard for Windows XP users. Using the online print function of the My Pictures feature on Windows XP-based computers, the print wizard will make it easy for consumers to order real, professional-quality prints on Kodak paper from PhotoWorks – directly from their computers. This service will be available as a free download from the PhotoWorks Web site in August and on some PCs in the Fall.What's unique is that the software, powered by Versalent's smart client technology, looks, feels, and performs as if it were preinstalled, but in reality the software code arrives over the Internet in a matter of seconds. What's more, the Versalent technology that simply uses the pre-existing capabilities of a standard Web browser enables the client application to talk directly with the Web services layer, eliminating the need for any intermediary servers. The result is a premium, powerful, and easy-to-use service from PhotoWorks that outperforms their competitors' offerings (see Figure 3).

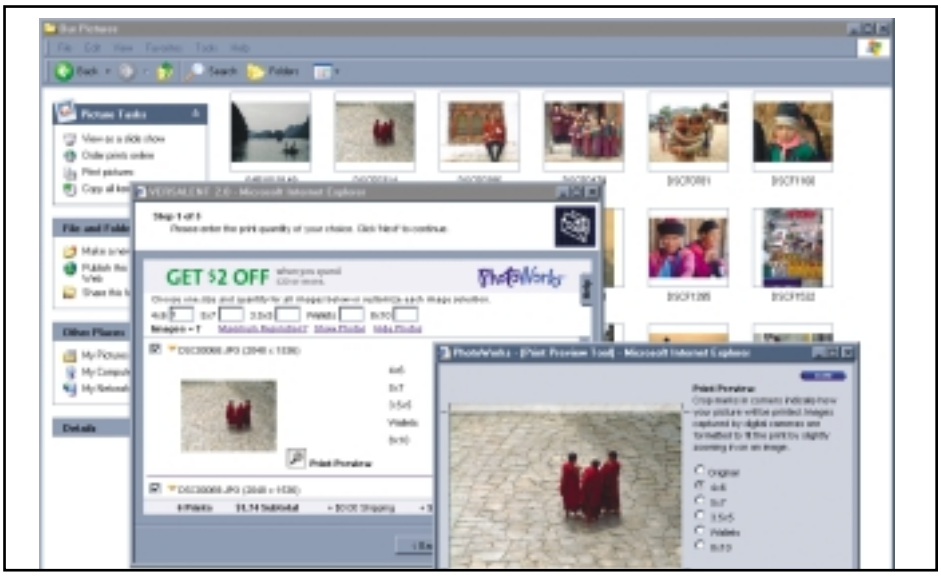


FIGURE 3 | **PhotoWorks running with Versalent Technology**

TABLE 1: Comparing the Client Options

Model Aspect:	Web Page	Thick Client	Smart Client
What the user experiences	Click and refresh pages, hyperlinks, images, forms	Windows, dialogs, tabs, instant responsiveness, drag and drop	Windows, dialogs, tabs, instant responsiveness, drag and drop
Client components	Browser	Installed software	Browser, or plug-in
Server components	HTTP application server	Database server	XML / Web service, (sometimes with intermediary "presentation server"
Development issues	Adherence to standards delivers multibrowser compatibility.	Duplicative development for multiple operating systems	Adherence to standards delivers multiservice compatibility
Distribution Issues	Instant distribution	Costs of installations	Instant distribution
Upkeep issues	None; The browser receives the latest code every time	Updates, patches, installations	Smart client runtime or plug-in updates from time to time; otherwise, runtime receives latest code every time
Method of scaling	More servers	More server and client installations	More servers if you choose a smart client requiring a "presentation server"; Fewer servers to scale if you choose a serverless smart client.
Applicability	Content Web sites, community Web sites, e-commerce Web sites	High-performance analytics, 3-D graphics, photo, sound, video editing. Ultrafast real-time data requirements	Enterprise applications, business process portals
User productivity	Satisfactory	Excellent	Excellent
Overall TCO	Excellent	Poor	Excellent

Compared to the traditional alternative of building and maintaining three front ends and three back ends, PhotoWorks maximizes the return on its Web services investment, distributing those costs across channels. Of interest here is that different client technologies were used for different objectives, and each client technology carries with it a substantially different total cost of development and ownership.

CONCLUSION

Web services don't need to be relegated to the system integration and messaging tiers. Smarter, richer client technology that leverages the power of the Internet for instant distribution of code provides a means to deliver superior functionality to the end user desktop at costs that make the technology a great way to get more from investments in service-oriented architectures.

DELIVERING .NET, JAVA, MAC OS X, AND XML TECHNOLOGIES

International Web Services Conference & Expo

Web Services Edge

SEPT. 30 – OCT. 2, 2003

3rd Annual

web services conference EDGE &expo

- *Take in tutorials covering .NET & Web services*
- *Listen to success stories*
- *Evaluate case studies & best practices*
- *Experience hands-on labs*

REGISTER TODAY!
CALL 201-802-3058
www.sys-con.com/edge
Register by September 5th and
SAVE Up To $200

2003 WEST

Santa Clara, California

KEYNOTE SPEAKERS

Vermeulen
CTO
amazon.com

Magee
VP, Oracle 9i
ORACLE

Litwack
Senior VP
Novell

Schmidt
VP, Systems Integration
BEST BUY

Event Sponsors: ORACLE Novell. Education Sponsors: Microsoft Sun

Media Sponsors: WebServices Journal, Java Developer's Journal, WebSphere Developer's Journal, XML Journal, .NET Journal, WebLogic Developer's Journal, LinuxWorld Magazine, wireless Business Technology, Linux Business Week, CE Advisor, ColdFusion Developer's Journal, PowerBuilder Developer's Journal, market WIRE, ISP Planet, CMS, SD Times, XMLWorld.com, SAMS, Java Industry Connection

Owned and Produced by: SYS-CON EVENTS

SPECIAL INSERT:

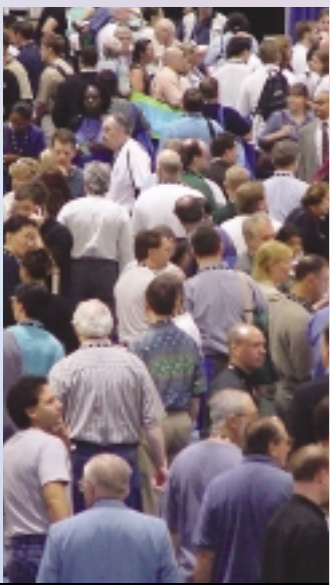
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

web services conference EDGE &expo

Conference & Expo

THE LEADING EVENT for i-Technology Professionals



SEPT. 30 – OCT. 2, 2003

Santa Clara Convention Center

FEATURES & ATTRACTIONS

- 3 Days Packed with Education and Training
- Keynotes & Panel Discussions from Industry Leaders
- 60 Hard-hitting and Informative Seminars
- **FREE** Web Services Workshop Presented by Oracle
- **FREE** .NET Tutorial with Microsoft's Russ' Tool Shed
- Java University Certification Training
- Industry-Leading Certification Programs
- **FREE** IBM Web Services Tutorial
- "Birds of a Feather" Discussions
- Round Table Discussions
- Opening Day Welcome Reception
- SAMS Meet the Authors Hot Topics Lounge
- Compelling Case Studies & Best Practices
- Hands-On Labs
- Featured Product Demonstrations
- Exhibit Floor featuring more than 40 companies and hundreds of products
- Real-time SYS-CON Radio Interviews

For more information visit www.sys-con.com or call 201 802-3069

WHO SHOULD ATTEND

- CEO
- CTO
- CIO
- IT Director
- Project Manager
- ▸ Software Developer
- ▸ Software Engineer
- ▸ Development Manager
- ▸ Application Developer
- ▸ Technical Director
- ▸ Analyst/Programmer
- ▸ IT Manager
- ▸ Technical Architect
- ▸ Team Leader
- ▸ Software Consultant

KEYNOTES & HIGHLIGHTED SPEAKERS

Allan Vermeulen
CTO, Amazon.com
Sept. 30 10:00 a.m.
"Web Services Foundations"
Allan Vermeulen, CTO and vice president at Amazon.com, directly oversees the Platform Technologies group. This group is responsible for guiding Amazon.com's technology architecture, including building and acquiring foundational components. Prior to his move to Amazon.com, Vermeulen was CTO and vice president of development at Rogue Wave Software. He holds a PhD in systems design engineering from the University of Waterloo.

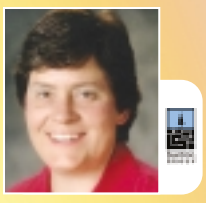
John Magee
Vice President, Oracle9i Application Server, Oracle
Oct. 1 10:00 a.m.
"J2EE Development on the Grid"
John Magee is vice president of Oracle9i Application Server and Oracle9i Developer Suite at Oracle. Mr. Magee has over 14 years of experience in the enterprise software industry and has held positions in product development, product management, and product marketing. In his current role, he manages technical product marketing for Oracle's application server and development tools products, and is responsible for evangelizing Oracle technology initiatives around J2EE, XML, and Web services.

David Litwack
Senior Vice President, Web Application Development Products, Novell
Sept. 30 2:00 p.m.
"Business Integration and IT" Keynote Panel
David A. Litwack is senior vice president of Web Application Development Products, responsible for the development and advancement of Novell's secure Web services strategy. Mr. Litwack assumed his current position in July 2002 following Novell's acquisition of SilverStream Software, a company for which Litwack had served as president and CEO since 1997.

John Schmidt
Leader of Systems Integration and Middleware, Best Buy Co.
Sept. 30 2:00 p.m.
"Business Integration and IT" Keynote Panel
John Schmidt is the chairman of the Methodology Committee for the EAI Industry Consortium and leader of systems integration and middleware at Best Buy Co., a leading specialty retailer of consumer electronics, personal computers, entertainment software, and appliances.

Jon Bosak
Distinguished Engineer, Sun Microsystems
Jon Bosak organized and led the W3C working group that created the XML specification and then served for two years as chair of the W3C XML Coordination Group. At Sun, where he holds the title of Distinguished Engineer, Mr. Bosak sponsors projects intended to advance XML technology. He is currently chair of the Universal Business Language (UBL) Technical Committee of OASIS.

Dave Chappell
VP, Chief Technology Evangelist, Sonic Software
Dave Chappell is the vice president and chief technology evangelist for Sonic Software. He has more than 18 years of industry experience building software tools and infrastructure for application developers, spanning all aspects of R&D, sales, marketing, and support services. Dave has also been published in numerous technical journals, and is currently writing a series of contributed articles for *Java Developer's Journal*.



Anne Thomas Manes
Research Director, Burton Group
Anne Thomas Manes is a research director at Burton Group, a research, consulting, and advisory firm. Anne leads research for the Application Platform Strategies service. Named one of NetworkWorld's "50 Most Powerful People in Networking" in 2002, and one of Enterprise Systems Journal's "Power 100 IT Leaders" in 2001, Anne is a renowned technologist in the Web services space. Anne participates in standards development at W3C and OASIS.

Marc Fleury
President, JBoss
Marc Fleury, PhD, is chief technical officer for Telkel, Inc. He is the leader of the JBoss project (www.jboss.org), which is an open source EJB server. Marc is based out of Silicon Valley and founded the project upon leaving Sun Microsystems. He was one of the main developers behind JBoss 1.0 and 2.0. Marc is the "keeper" of the project. He founded the Jboss Group, a company regrouping the elite developers of Jboss to consult around Jboss.

Hotel & Travel

Reserve Your Hotel Room Now
At The Westin Santa Clara!
The Official Conference Hotel of Web Services Edge West 2003

The Westin Santa Clara
5101 Great America Parkway
Santa Clara, CA 95054

Arrangements have been made with the Westin Santa Clara, which is conveniently located at The Santa Clara Convention Center. Specially reduced rates have been secured at this luxury, full-service hotel.

Single Occupancy Room: $165.00

Double Occupancy Room: $165.00

All rooms are quoted exclusive of applicable state and local taxes which are currently 9.5% as well as the California State Tourism Tax of 0.045%. The above rates are group rates and are available for Web Services Edge 2003 delegates, over the show dates of September 28 – October 3, 2003, only.

To learn more about The Westin Santa Clara you can contact the hotel directly or you can make your reservations by calling Expo Travel International at (800) 829-2281 or (201) 444-0060 (direct). Fax reservations to (201) 444-0062. Credit card information is required to guarantee reservations and expedite confirmation. Confirmations will be mailed directly from the hotel, time permitting. All changes and cancellations should be made directly through Expo Travel International.

To make online reservations:
www.expotravel.com by September 12, 2003.

Reservations received after September 12, 2003, will be accepted on a space-available basis only, at the special rate, if available.

Contact Information:
The Westin Santa Clara Reservations:
Tel: 408 986-0700
Fax: 408 980-3990

Hotel Arrangements Are Easier Than Ever! You have your choice – contact the hotel directly or call us. The Official Conference travel agent, Expo Travel International.

Official Conference Travel Agent:
Expo Travel International
Toll Free: (800) 829-2281
Tel: (201) 444-0060
Fax reservations to (201) 444-0062

Driving Directions to Westin Santa Clara from San Jose Airport:
Highway 101 North. Exit at Great America Parkway/Bowers. Turn Right onto Great America Parkway; hotel is about 1.5 miles down on the right side.

SPECIAL DISCOUNTS AVAILABLE

Take advantage of the Early Bird and Preregistration values available right now, or save even more with a group of 5 or more. For special group discounts contact Michael Lynch at mike@sys-con.com, or by phone at (201) 802-3058.

PRODUCED BY SYS-CON EVENTS

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

Conference at-a-Glance

		JAVA	.NET	WEB SERVICES
TUESDAY, SEPTEMBER 30	8:00AM – 4:00PM	**REGISTRATION**		
	9:00AM – 9:50AM	The Next Phase in Evolution of J2EE	Using WSE 2.0	Web Services Management
	10:00AM – 10:50AM	Keynote - "Web Services Foundations" - Allen Vermeulen, CTO and Vice President, Amazon.com		
	11:00AM – 6:00PM	**EXPO OPEN**		
	2:00PM – 2:50PM	Keynote Panel Discussion - Business Integration and *i*-Technology		
	3:00PM – 3:50PM	Ant Applied in "Real World" Web Services	Smart Devices in the Enterprise	Building Interoperable Web Services Using WS-I Basic Profile
	4:00PM – 4:50PM	Developing Applications with SWT	Using the Mobile Internet Toolkit	Web Services Orchestration
	5:00PM	**OPENING NIGHT RECEPTION**		
WEDNESDAY, OCTOBER 1	8:00AM – 4:00PM	**REGISTRATION**		
	9:00AM – 9:50AM	Empowering Java and RSS for Blogging	Introduction to ROTOR	ID, Please. The Case for Giving Web Services an Identity
	10:00AM – 10:50AM	Morning Keynote - "J2EE Development on the Grid" - John Magee, Vice President, Oracle9i, Oracle		
	11:00AM – 4:00PM	**EXPO OPEN**		
	2:00PM – 2:50PM	Keynote Panel Discussion - Interoperability: Is Web Services Delivering?		
	3:00PM – 3:50PM	JUnit: Testing Your Java with JUnit	Using Portable .NET	WS-BPEL
	4:00PM – 4:50PM	JDK1.5: The Tiger	ASP.NET with Mono	UDDI: Dead or Alive?
	5:00PM – 6:00PM	Squeezing Java	Using WSE with IBM's Web Services Tool Kit	Web Services Choreography, Management, and Security - Can They Dance Together?
THURSDAY, OCTOBER 2	8:00AM – 4:00PM	**REGISTRATION**		
	9:00AM – 9:50AM	Leveraging AOP in JBoss	Success Story: Eiffel, .NET, and Design by Contract for the Financial Industry	Strategies for Securing Web Services
	10:00AM – 10:50AM	Technical Keynote		
	11:00AM – 11:50AM	Apache Axis	.NET IDE's	Web Services Progress Report
	12:00PM	**LUNCH**		
	1:00PM – 1:50PM	Meeting the Challenges of J2ME Development	Windows SharePoint Services	The Seven Habits of Highly Effective Enterprise Service Buses (ESBs)
	2:00PM – 2:50PM	Keynote Panel Discussion - Summit on Web Services Standards		
	3:00PM – 3:50PM	Simplifying J2EE Applications	BizTalk 2004	See www.sys-con.com for more information
	4:00PM – 5:00PM	Integrating Java + .NET	See www.sys-con.com for more information	See www.sys-con.com for more information

XML	MAC OS X
Introduction to Xforms	Introducing OS X (Panther) What's New?
Securing Your XML and Web Services Infrastructure	Programming Rich User Interfaces Using Cocoa
UBL - The Universal Business Language	Quick Applications Using AppleScript
Standards-Based Enterprise Middleware Using XML/Web Services	Java and OS X: A Perfect Marriage
XML and Enterprise Architecture: Technology Trends	Enterprise Java and OS X
Using XML Schemas Effectively in WSDL Design	Developing Web Services Using WebObjects
Canonical Documents for Your Business: Design Strategies	Cocoa, Carbon, Java: Application Frameworks for OS X (When to Use What)
XML and the Fortune 500	Securing OS X Applications
XML at Work in 'Fortune 500' Companies	Xserve: Ease of OS X and Power of Unix
XML Schema Best Practices	OS X for the Unix Developer
See www.sys-con.com for more information	Introducing Quartz: 2D Graphics for Apple
See www.sys-con.com for more information	See www.sys-con.com for more information

PROGRAM SUBJECT TO CHANGE

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

Java Technology Track

Java Technology Track

JDJ EDGE conference&expo

The Java Track features presentations aimed at the beginner, as well as the seasoned Java developer. Sessions will explore the whole spectrum of Java, focusing on J2EE, application architecture, EJB, and J2ME. In addition the track will cover the latest in SWT, Ant, JUnit, open source frameworks, as well as an in-depth look into the vital role that Java is playing in building and deploying Web services.

Sessions will focus on:

- Enterprise Java 1.4
- Ant Applied in "Real World" Web Services
- Developing Application Frameworks with SWT
- Empowering Java and RSS for Blogging
- JUnit: Testing Your Java with JUnit
- JDK1.5: The Tiger
- Simplifying J2EE Applications
- Using IBM's Emerging Technologies Toolkit (ETTK)
- Apache Axis
- Meeting the Challenges of J2ME Development
- Integrating Java + .NET
- Squeezing Java

(JV1) The Next Phase in the Evolution of J2EE

BILL ROTH, SUN MICROSYSTEMS

Tuesday, September 30, 9:00 a.m. - 9:50 a.m.

Did you know that J2EE has developed to the point where companies can bring in millions of dollars a year selling J2EE-based application systems? This session will discuss the original plans for the evolution of the J2EE marketplace, and the companies that are capitalizing on completing one of the final stages of this process.

BIO: Bill Roth is currently technology evangelist for E.piphany. He previously was group marketing manager for J2EE at Sun Microsystems, and is a member of the *Java Developer's Journal* editorial board.

(JV2) Ant Applied in "Real World" Web Services

KYLE GABHART, GABHART CONSULTING

Tuesday, September 30, 3:00 p.m. - 3:50 p.m.

A defined and easily repeatable process is one of the most necessary but often least used aspects of good software development. A defined build process ensures that your project's software is built, deployed, and tested identically every time. Without this type of control and predictability, valuable time is often lost chasing down bugs that don't exist or rejecting solutions that were only partially implemented.

Apache's Ant is a powerful scripting tool that enables developers to define and execute routine software development tasks using the simplicity and extensibility of XML. Ant provides a comprehensive mechanism for managing software development projects, including compilation, deployment, testing, and execution. In addition, it is compatible with any IDE or operating system.

BIO: Kyle Gabhart is an independent consultant, mentor, and published author, specializing in J2EE, XML, and Web services technologies. He is a prolific writer, with his most recent work displayed on IBM's developerWorks Web site in the 'J2EE Pathfinder' column. Kyle is highly regarded as a dynamic and enthusiastic public speaker with an innovative perspective on technology.

(JV3) Developing Applications with SWT

Tuesday, September 30, 4:00 p.m. - 4:50 p.m.

The Standard Widget Toolkit (SWT) provides a common, OS-independent, Java-based API for widgets and graphics implemented in a way that allows tight integration with the underlying native window system. The Eclipse project and the various tools that plug in to it use SWT for presenting information to the user. This session will provide a general overview of SWT and introduce its basic concepts and classes. See www.sys-con.com/edge for further details on this session.

(JV4) Empowering Java and RSS for Blogging

JASON BELL, IT DEVELOPMENT MANAGER

Wednesday, October 1, 9:00 a.m. - 9:50 a.m.

One of the fastest growing areas over the last few years is the blogging community. The ease with which you can post and publish information has enabled everyone to become his or her own publisher. One of the powers of blogs has been the syndication of data via the RSS (XML) protocol. Discover how you can easily produce and consume RSS feeds within your Java applications for wider appeal and hook into JavaBlogs.

BIO: Jason Bell is Java developer and IT development manager for a B2B portal in the UK. He also contributed to a number of open source projects and is an advocate of everyone reading the API docs.

(JV5) JUnit: Testing Your Java with JUnit

Wednesday, October 1, 3:00 p.m. - 3:50 p.m.

A critical measure of the success of software can be found in whether or not it executes successfully. Equally important, however, is whether or not that software does what it was intended to do. JUnit is an open source testing framework that provides a simple means for developers to define their intentions for how their software should work. JUnit then provides test runners that process your intentions and verify that your code performs as intended. The result is software that not only works, but works in the correct way. See www.sys-con.com/edge for further details on this session.

(JV6) JDK 1.5: The Tiger

CALVIN AUSTIN, SUN MICROSYSTEMS

Wednesday, October 1, 4:00 p.m. - 4:50 p.m.

Java 1.5 is the next major release of Java and with it comes a whole host of new enhancements and additions to the language. 1.5 promises a lot. Attend this session and discover the wonders that await the Java community.

BIO: Calvin Austin is the lead engineer on Sun Microsystems' port of the Java 2 Platform, Standard Edition (J2SE) to the Linux OS. He has been with Java Software since its inception six years ago. Calvin is the specification lead for JSR-176, which defines the J2SE 1.5 ("Tiger") release contents.

(JV7) Squeezing Java

ALAN WILLIAMSON, JAVA DEVELOPER'S JOURNAL

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

Java is a very powerful language, and while it offers the developer a rich array of tools, the fundamentals mustn't be overlooked. Improving your code at the core layer will result in great improvements in efficiency and produce (hopefully) fewer bugs. We'll look at the "do's and don'ts" of programming and learn lots of hints and tips that will accelerate your Java coding.

BIO: Alan Williamson is editor-in-chief of *Java Developer's Journal*. In his spare time he holds the post of chief technical officer @ n-ary (consulting) Ltd, one of the first companies in the UK to specialize in Java at the server side. Reach him at alan@n-ary.com (www.n-ary.com) and rumor has it he welcomes all suggestions and comments!

(JV8) Leveraging AOP in JBoss

MARC FLEURY, PH.D., JBOSS

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

This session will explore the JBoss Aspect Oriented Programming (AOP) framework. It will define AOP and discuss its implementation in the JBoss application server, showing

how a microkernel combined with simple AOP technology can enable the creation of generalized containers. From small single purpose embedded controls to large enterprise systems, JBoss middleware enables easy system assembly by AOP and aspects. This session will focus on the aspects themselves, moving beyond the logging examples to cover all the standard aspect technology that has been present in JBoss for the past 3 years, including persistence, caching, invocations, transactions and acidity as aspectized components that can be reused in applications.

BIO: Marc Fleury, PhD. is chief technical officer for Telkel, Inc. He is the leader of the JBoss project (www.jboss.org/), which is an open source EJB server. Marc is based out of Silicon Valley and founded the project upon leaving Sun Microsystems. He was one of the main developers behind JBoss 1.0 and 2.0., and is the "keeper" of the project. Marc is a graduate of the French Ecole Polytechnique with a degree in mathematics and holds a PhD in physics for work done at MIT as a visiting scientist.

(JV9) Apache Axis

CHRIS HADDAD, COBIA COMMUNICATIONS

Thursday, October 2, 11:00 a.m. - 11:50 a.m.

Apache Axis is the very popular SOAP engine that includes everything you need to start producing Web services. Discover just what Axis is, and how you can utilize the power of this free engine to kick start your Web services.

(JV10) Meeting the Challenges of J2ME Development

Thursday, October 2, 1:00 p.m. - 1:50 p.m.

Synchronization is an important component in building mobile applications that can operate offline to review, capture, or change data that will later be reconciled with enterprise data on a central server. Today, distributed clients with limited memory and intermittent connectivity, such as cellular phones and PDAs, are increasingly used to access multiple server-side enterprise applications and data. This presentation will demonstrate a flexible yet powerful framework for managing data and synchronization in the Java environment (J2EE, J2ME/pJava). See www.sys-con.com/edge for further details on this session.

(JV11) Integrating Java + .NET

DEREK FERGUSON, .NET DEVELOPER'S JOURNAL

Thursday, October 2, 3:00 p.m. - 3:50 p.m.

Contrary to popular belief, you can still use Java (J#) on Microsoft's .NET platform. While the flavor may not be the latest JDK we know and love, J# offers other goodies and trinkets to accelerate your .NET solution. Find out what they are.

BIO: Derek Ferguson is chief technology evangelist for Expand Beyond Corporation (www.xb.com), the worldwide leader in mobile software for enterprise management. He is also editor-in-chief of *.NET Developer's Journal* and author of the book Mobile .NET (Apress).

(JV12) Simplifying J2EE Applications

Thursday, October 2, 4:00 p.m. - 4:50 p.m.

TODD LAUINGER, BEST BUY CO., INC.

J2EE is a large, complex specification for server-side, Web-enabled application development. Over the past few years, I have led many teams through the J2EE jungle, trying to steer them away from the hype and keep them focused on delivering rock-solid end user applications. The purpose of this tutorial is to discuss a variety of tips, tricks, and lessons I have learned so you and your teams can develop J2EE applications better, faster, and simpler than before.

BIO: Todd Lauinger is currently employed as a software construction fellow at Best Buy Co., Inc. He has over 10 years of experience developing large, mission-critical software systems for engineering and business organizations. Todd is also an experienced instructor, mentor, conference speaker, and published author, and has a master's degree in software engineering.

Java University Program℠

Sun microsystems

October 1

October 2

SUN MICROSYSTEMS

Architecting Web Services with Java 2 Platform, Enterprise Edition (J2EE)

Wednesday, October 1, 2003

Who Should Attend

Developers and software architects with experience using the Java programming language, and some knowledge of XML who are interested in discovering how Java technology can deliver Web services for multi-tier applications.

Overview

Web services provide an excellent mechanism to create integrated architectures for B2B solutions. The Java 2 Platform, Enterprise Edition (J2EE) is an excellent API for creating and deploying Web services. When used in conjunction with the Java technology APIs for XML (JAX family of APIs), powerful and adaptable architectures can be created that offer all the advantages of J2EE technology including portability, scalability and interoperability. The core of this seminar deals with investigating the JAX family of APIs and how they can be used to create Web services.

Benefits

- Understand Web services and their use in the enterprise
- Identify how servlets and JavaServer Pages (JSP) can be used with Web services
- Recognize the use of the Simple Object Access Protocol (SOAP)
- Be able to utilize the Java technology API for XML Messaging (JAXM) with SOAP
- Recognize the Java technology API for XML Binding (JAXB) and its use in the middle tier
- Learn about the Java technology API for XML Registries (JAXR)
- Investigate the Java technology API for XML-based Remote Procedure Calls (JAX-RPC)

Java 2 Platform: Architect Certification Fast Path

Thursday, October 2, 2003

Who Should Attend

This session is designed for enterprise application architects, system analysts, experienced technologists and developers using Java technology seeking certification as an architect for the Java 2 Platform, Enterprise Edition (J2EE).

Overview

Many of the solutions in today's "Net economy" are, or soon will be, developed using the Java 2 Platform, Enterprise Edition (J2EE) architecture. Gaining recognized competency architecting J2EE platform-based solutions is vital to your success as an architect, reaffirms your value and increases your career opportunities.

Benefits

- Receive an intensive review of the topics covered on the Sun Certified Enterprise Architect for the Java 2 Platform, Enterprise Edition Exam.
- Increase your understanding and knowledge of successfully architecting solutions using J2EE technology.
- Understand the system qualities: scalability, availability, extensibility, performance and security.
- Understand trade-offs of different architectural choices as they pertain to system qualities.
- Describe the benefits and weaknesses of potential J2EE technology-based architectures.
- State benefits and costs of persistence management strategies.
- Review real-world case studies of J2EE technology-based architecture.
- Prepare for the exam by reviewing practice tests and questions.

Go to www.sys-con.com/edge for more details about the Java University Program

Register Online at ***www.sys-con.com/edge***

SPECIAL INSERT: Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

.NET Track

.NET Track

.NET EDGE conference&expo

Presentations will explore the Microsoft .NET platform for Web services. To the average developer, it represents an entirely new approach to creating software for the Microsoft platform. What's more, .NET development products – such as Visual Studio .NET – now bring the power of drag-and-drop, GUI-based programming to such diverse platforms as the Web and mobile devices.

Sessions will focus on:

- ASP.NET
- Security
- VB.NET
- .NET and XML
- Smart Device Extensions for VS.NET
- Best Practices
- Shared Source CLI
- .NET Remoting
- Smart Devices in Health Care Settings
- Mobile Internet Toolkit
- ROTOR
- Portable .NET
- ASP.NET Using Mono
- Using WSE with IBM's WSTK
- GUI Applications Using Mono
- Portals – Windows SharePoint Services/SharePoint Portal Server
- Windows Server 2003 and IIS 6
- .NET and Java Interoperability
- Distributed .NET for Financial Applications
- Developing C# with Eclipse

(NT1) Using WSE 2.0

JOHN BRISTOWE, EMPOWERED SOFTWARE SOLUTIONS

Tuesday, September 30, 9:00 a.m. - 9:50 a.m.

Web Services Enhancements for Microsoft .NET (WSE) v2.0 is a class library that augments the .NET Framework to provide an implementation of advanced Web service specifications such as WS-Addressing, WS-Policy, WS-Security, WS-SecurityPolicy, WS-SecureConversation, and WS-Trust. This session highlights support for these specifications while examining many new features, which include an extensible framework for policy enforcement and SOAP messaging. Take advantage of the future by learning how to incorporate WSE v2.0 in your applications quickly and easily.

BIO: John Bristowe is a senior solution developer architect with Empowered Software Solutions (ESS), a .NET consulting firm based in Burr Ridge, Illinois. An active speaker and author of several .NET and XML Web service–related articles, John is also a volunteer for the International .NET Association (INETA) Infrastructure Committee and various GotDotNet workspaces. John is presently focused on the .NET Framework and XML Web service plumbing and protocols.

(NT2) Smart Devices in Health Care Settings

BRAD MCCABE, INFRAGISTICS

Tuesday, September 30, 3:00 p.m. - 3:50 p.m.

Mobile devices are gaining popularity and functionality every day in the enterprise. When you combine this with .NET s unparalleled support for mobile computing, via Tablets, PDAs, and phones, you know that this is a topic not to be missed. In this session, Infragistics' Brad McCabe will discuss all of these .NET mobility technologies and how they are being used in organizations around the globe today and how you can apply these technologies to your enterprise.

BIO: Prior to joining Infragistics, Brad McCabe served as systems architect for the network solutions development team at Verizon Communications. Concurrently, Brad held the position of lead .NET evangelist within Ajilon Consulting responsible for content and delivery of material for the national Inside .NET tour. Brad has been working as a senior developer and a software engineer, and his current responsibilities include developing reference applications and working with enterprise customers on project implementation.

(NT3) Using the Mobile Internet Toolkit

JON BOX, QUILOGY

Tuesday, September 30, 4:00 p.m. - 4:50 p.m.

The Microsoft Mobile Internet Controls Runtime (previously known as the Microsoft Mobile Internet Toolkit) is one of the most exciting technologies in mobile computing today. By adapting to the unique characteristics of a mobile Web client at runtime, this tool provides unparalleled flexibility for the creation of mobile Web applications. In this session, Quilogy's Jon Box will guide you through the fundamentals of using this technology to create your own .NET applications.

BIO: Jon Box, a solutions architect at Quilogy (www.quilogy.com), has advanced experience in multiple technologies with a solid background in infrastructure, application development, data access, and a host of other technologies. He has served in diverse roles as an architect, trainer, author project manager and general manager at Quilogy, where he is currently a part of Quilogy's Atomic team. His current activities include authoring and developing advanced .NET training courses and technical content for the Atomic Web site. Jon is a Microsoft regional director for Memphis and serves on the MSDN Customer Council. He is a noted speaker on Microsoft emerging technologies, an active participant in the Memphis technology community, and founded the Memphis .NET User Group (www.memphisdot.net). In addition, Box has served as a technical editor on several books, including *An Introduction to Object-Oriented Programming with Visual Basic .NET.*

(NT4) Introduction to ROTOR

TED NEWARD, DEVELOPMENTOR

Wednesday, October 1, 9:00 a.m. - 9:50 a.m.

Microsoft's Shared Source CLI (also known as ROTOR) is the only officially supported CLI from Microsoft that will run on non-Windows platforms. For those interested in truly understanding the inner workings of the .NET platform, DevelopMentor's Ted Neward will present this introduction to the technology that lets you "look under the hood."

BIO: Ted Neward is an independent software development architect and mentor in the Sacramento, California area. He is the author of a number of books, including *Server-Based Java Programming* and the forthcoming *Effective Enterprise Java and SSCLI Essentials*, and coauthor of *C# In a Nutshell* with Peter Drayton and Ben Albahari. He is also an instructor with DevelopMentor, where he teaches and authors both the Java and .NET curriculum. He speaks frequently for technology user groups, and is the writer of technical papers for www.javageeks.com and www.clrgeeks.com. He currently labors on behalf of the University of California, Davis, architecting a rebuild of the Davis Accounting and Financial Information Services software system.

(NT5) Using Portable.NET

ADAM BALLAI, DOTGNU DEVELOPER

Wednesday, October 1, 3:00 p.m. - 3:50 p.m.

DotGNU Portable.NET forms the basic .NET compatibility support in the DotGNU framework. The goal of the project is to allow existing .NET developers to smoothly migrate to DotGNU and still remain compatible. The platform includes compilers, a runtime engine, and class libraries. Portable.NET has a number of extras to offer, like TreeCC, Web services support, generic unrollers, and so on. The compiler suite currently supports C#, C, Java, and VB.NET. A JScript interpreter, portable WinForms, system. Drawing, and ADO.NET are in usable stages. And all this is portable to a variety of hardware and operating systems.

(NT6) ASP.NET with Mono

DEREK FERGUSON, EXPAND BEYOND

Wednesday, October 1, 4:00 p.m. - 4:50 p.m.

One of Mono's most compelling features is its ability to run ASP.NET Web sites on platforms other than Windows and IIS. In this session, you'll learn how to create and host such sites, as well as the unique advantages that Mono offers in this space.

BIO: Derek Ferguson is chief technology evangelist for Expand Beyond Corporation (www.xb.com), the worldwide leader in mobile software for enterprise management. He is also editor-in-chief of *.NET Developer's Journal* and author of the book *Mobile .NET* (Apress).

(NT7) Using WSE with IBM's WSTK

DAVID WHITE, MICROSOFT

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

Achieving WS-I style interoperability between Java and .NET is not yet as easy as it should be. Standards that have only recently been ratified have not been completely implemented and documented by all vendors. In this session, you will learn how to create .NET clients that can communicate securely with Java Web services and vice versa.

BIO: David White is the architectural evangelist for the Great Lakes District for the Automotive Vertical. He has more than 13 years' enterprise experience across many disparate platforms. After spending time working on enterprise mainframe applications on MVS he moved to the Unix/VAX arena and then on to the Microsoft arena. He has used almost all the major development paradigms over time (J2EE, DNA, CORBA, etc.) and because of this advises customers on technology decisions related to development and architecture with regards to Microsoft products and how they fit into the customers overall architecture. He has been with Microsoft now for over 5 years.

(NT8) Success Story: Eiffel, .NET, and Design by Contract for the Financial Industry

BERTRAND MEYER, EIFFEL SOFTWARE & ETH ZURICH

Thursday, October 2, 9:00 a.m. - 9:50 a.m.

The financial industry requires ever higher guarantees of quality and productivity to produce the applications that the business demands, and deliver them on time. Using a combination of leading-edge technologies, AXA Rosenberg was able to capitalize on the best of software engineering to satisfy demanding market needs. A combination of .NET, the Eiffel language and method, and the techniques of Design by Contract enables the company to stay ahead of its user needs. This presentation discusses the challenges encountered and how the retained technical solutions enable the company to meet them; it concludes with an outlook of the most promising avenues for future development.

BIO: Bertrand Meyer is founder and CTO of Eiffel Software (Santa Barbara), professor of software engineering at the Swiss Federal Institute of Technology (ETH), and an adjunct professor at Monash University in Australia. He is one of the pioneers of object technology and author of several award-winning books, including *Object-Oriented Software Construction*, one of the best sellers in the field.

(NT9) .NET IDEs

Thursday, October 2, 11:00 a.m. - 11:50 a.m.

Visual Studio .NET is, by far, the best-known IDE for creating .NET-based software. There are, however, several alternatives out there for .NET developers looking to keep their options open. In this session, *.NET Developer's Journal* editor-in-chief Derek Ferguson will guide you through the many choices you can make. See www.sys-con.com/edge for further details on this session.

(NT10) Windows SharePoint Services

Thursday, October 2, 1:00 p.m. - 1:50 p.m.

SharePoint services allow a new level of collaboration for workers in information-focused pursuits. The newest version of the SharePoint Portal server is, in fact, actually built 100% in managed .NET code. In this session, you will learn how to create sites using SharePoint Services and C#. See www.sys-con.com/edge for further details on this session.

(NT11) BizTalk 2004

Thursday, October 2, 3:00 p.m. - 3:50 p.m.

BizTalk 2004 is the key Microsoft solution for business orchestration. If your organization has a multitude of processes that must be coordinated, and/or if you have many automated processes that you share with your business partners, then you must attend this session to learn what BizTalk can do for you! See www.sys-con.com/edge for further details on this session.

Microsoft .net

FREE Microsoft® Tutorial

RUSS' TOOLSHED

TINKERING WITH VISUAL STUDIO

September 30, 2003

Join Russ as he shows you how to use Visual Studio .NET

INTRO TO WEB SERVICES USING VS.NET

One of the key ideas behind the .NET strategy is the concept of software as a service, or in short, Web services. This session will explain what a Web service is and provide an overview of its related technologies like XML, SOAP, and UDDI. We will demonstrate how the .NET Framework makes it easy to implement them for new and existing applications. This session will also provide concrete best practices for building XML Web services using Visual Studio .NET. We'll answer many common questions like: How will my Web service scale? How can my XML Web services enable interoperability with Web services from other vendors as well as within my own organization? We'll delve into building highly reliable and secure Web services. Also, we will discuss issues such as dealing with complex data types using WSDL (Web Services Description Language), as well as securing SOAP messages using encryption. We'll see how developers can use enterprise-level XML Web services to simplify customer solutions.

ADVANCED WEB SERVICES USING ASP.NET

This session will explore some of the more advanced areas of SOAP in ASP.NET's support for Web services. ASP.NET Web services are the preferred way for Web developers to expose Web services on the Internet. The goal is quick, easy, and high-performing SOAP services. We will look at how to use the SOAP extension classes to create some very interesting applications on top of the core SOAP architecture found within the .NET Framework. For instance, you can implement an encryption algorithm or screen scraping on top of the Web service call. We'll dig into more advanced topics, explore the SOAP headers, and see ways to ensure security in our Web services.

.NET REMOTING ESSENTIALS

Microsoft .NET Remoting is the .NET technology that allows you to easily and quickly build distributed applications. All of the application components can be on one computer, or they can be on multiple computers around the world. .NET Remoting allows client applications to use objects in other processes on the same computer or on any other computer to which it can connect over its network. During this presentation we will discuss what you need to know to get started with .NET Remoting. We will talk about how .NET Remoting compares with DCOM, how to host remoted objects in a variety of applications, how to call remoted objects from a client application, how to control the lifetime of remoted objects, and how to secure remoting applications.

FREE Microsoft Tutorial when you register for a VIP Pass
Guarantee your seat when you register for a Full Conference Pass

Register Online at www.sys-con.com/edge

SPECIAL INSERT: Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

Web Services Track

Web Services Track

web services EDGE conference&expo

Presentations will include discussions of security, interoperability, the role of UDDI, progress of the standards-making bodies, SOAP, and BPM. Case studies cover the design and deployment of Web services in the marketplace.

Sessions will focus on:

- Interoperability
- Enterprise Networks
- Web Services Management
- Web Services Standards
- Web Services Orchestration
- Security (WS-Security, SAML)
- BPEL4WS
- UDDI: Dead or Alive?
- ebXML & Web Services
- EAI & Web Services
- RPC vs Documents: Uses and Differences
- User Interfaces for Web Services
- Web Services Best Practices
- Service-Oriented Architecture

(WS1) Web Services Management: Managing the Impact of Change

JAMES PHILLIPS, ACTIONAL

Tuesday, September 30, 9:00 a.m. - 9:50 a.m.

Adoption of Web services technology is well underway in the Global 2000. It is impossible to ignore the numerous articles and speeches from CIOs declaring their adoption of Web services as high-priority strategic IT initiatives. Whereas the service-oriented approach to application architecture was meant to accelerate the responsiveness of the IT organization, the impact of the changes to the service network and the ripple effects that ensue can lead to precisely the opposite result. Change is easy and affordable, but the impact of change can be unmanageable and expensive. To truly reap the benefits of the service-oriented approach to application architecture, customers must have a way to manage the impact of change in their enterprise service network.

BIO: James Phillips has worldwide responsibility for Actional's product and market strategy and market execution. Prior to joining Actional, James served as chief strategy officer and vice president of product marketing and business development with Ensim Corporation, and has held senior marketing management, software engineering, and business development roles with Intel, Intuit, Synopsys, and Central Point Software.

(WS2) Building Interoperable Web Services Using WS-I Basic Profile

KEVIN LIU, SAP LABS, LLC

Tuesday, September 30, 3:00 p.m. - 3:50 p.m.

The promise of Web services interoperability is based on a set of standards, including XSD, SOAP, WSDL, and UDDI. However, early interoperability experiments such as the SOAP Builder activity have proven that making these technologies seamlessly work together, inter- or intra-enterprises, is a bigger challenge than most of us expected. Most of the Web services technologies are still in the standardization process, and Web services platforms have to be based on prestandard draft specifications. The inconsistencies and ambiguities left in these initial drafts can easily lead to different interpretations for the same feature and substantially compromise the promised interoperability.

The presentation explains how WS-I Basic Profile addresses these challenges, and what the key techniques are that you can get from the profile that can make your Web services interoperable. It explains why the profile disallows SOAP encoding and why RPC style should give way to document style.

BIO: Kevin Liu is an architect of XML Web services technologies at SAP Labs in California. Kevin helps drive the adoption of Web services technologies in SAP's strategic products, and represents SAP in various standard bodies. Kevin has over 10 years of experience in software engineering and the financial industry. He holds multiple master's degrees in information management, finance, and economics.

(WS3) Web Services Orchestration

CHRIS PELTZ, HP

Tuesday, September 30, 4:00 p.m. - 4:50 p.m.

The real value of Web services will come as we start finding standard ways to connect these distributed components in a more reliable fashion. One of the key enablers to this will be web services orchestration. This session introduces web services explains some of the requirements for connecting we services together. A history of key standards will be provided, including XLANG, and WSFL. A comparison will be drawn between WSCI and BPEL4WS, and where each might be appropriate in an architecture. The presentation will conclude with a case study demonstrating the use of orchestration technologies to connect web services together.

BIO: Chris Peltz is a senior software consultant within HP's Developer Resources Organization. He provides technical and architectural consulting to enterprise customers in the areas of J2EE, web services, and mobile development. Chris has over 10 years of software experience in object-oriented technologies, 4GL development, GIS, and web applications design.

(WS4) ID, Please. The Case for Giving Web Services an Identity

ASHISH LARIVEE, NOVELL

Wednesday, October 1, 9:00 a.m. - 9:50 a.m.

Without identity management, Web services can be consumed by anyone. The challenge for Web services developers is to provide appropriate access based on the user's identity. As identity management moves into the forefront of technology, directory services will evolve from simple LDAP repositories used for authentication and storage to robust engines that provide identity integration, access management, and policy enforcement. This presentation will discuss how identity management and directory services provide a robust solution for Web services authentication, authorization, and single sign-on.

BIO: With more than nine years of experience in the software industry, Ashish Larivee has designed and developed many enterprise applications across a variety of platforms including Microsoft, Lotus Notes/Domino, and the J2EE platform. In 1999, Ashish joined SilverStream Software, acquired by Novell in July 2002, and has served in various roles in consulting, development, and technical marketing. In her current role, she helps define the strategy and product direction across Novell's Web application development products.

(WS5) WS-BPEL

JOHN EVDEMON, MICROSOFT

Wednesday, October 1, 3:00 p.m. - 3:50 p.m.

BIO: John Evdemon is program manager, industry standards, industry solutions enablement, for Microsoft, and is cochair of OASIS' Business Process Execution Language TC. He is an XML and e-business expert, having served as CTO/director of XML-related products for both a large integration platform vendor and a small XML-centric start-up, and has been designing and deploying enterprise systems on a wide variety of platforms for over 15 years. He is an invited expert with the W3C XML Core Syntax Working Group and has chaired several industry-specific XML initiatives. An associate editor of *XML-Journal*, John is also a regular contributor to journals, books, and online forums and is writing a book on XML. See www.sys-con.com/edge for further details on this session.

(WS6) UDDI: Dead or Alive?

ANNE THOMAS MANES, THE BURTON GROUP

Wednesday, October 1, 4:00 p.m. - 4:50 p.m.

When UDDI was first announced in September 2000, IBM and Microsoft predicted that it would be a fundamental component of the Web services infrastructure. Three years later, UDDI has still seen only very modest adoption. So what is the future of UDDI? Will it ever live up to the early predictions? Will it enable dynamic discovery and utilization of Web services? Why should you even consider using it? Although it's had a slow start, UDDI is, in fact, a core component of the Web services infrastructure. This session explores the Web services registry specification and examines user guidelines and best practices.

BIO: Anne Thomas Manes is a research director at Burton Group, a research, consulting, and advisory firm. Anne leads research for the Application Platform Strategies service. Named one of NetworkWorld's "50 Most Powerful People in Networking," in 2002 and one of *Enterprise Systems Journal's* "Power 100 IT Leaders," in 2001, Anne is a renowned technologist in the Web services space. Anne participates in standards development at W3C and OASIS. She is a frequent speaker at trade shows and author of numerous articles and the book *Web Services: A Manager's Guide*.

(WS7) Web Services Choreography, Management, and Security – Can They Dance Together?

PAUL LIPTON, COMPUTER ASSOCIATES

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

Web services choreography, management, and security are among the principal challenges implementers of service-oriented architectures face in their efforts to create a more cost-effective and agile IT infrastructure, despite the fact that significant progress has been made in various standards organizations such as the W3C and OASIS. There is still much confusion in the IT community about the standards, which are at various stages of maturity. Also, their relevance to enterprise IT and how they might work together is often unclear.

This session begins with a useful overview of standards in these three critical areas of Web services: choreography, management, and security. We will then discuss the role of each in the enterprise, and conclude with an examination of how these three areas might work together to solve the real challenges facing IT in its efforts to bring greater utility, flexibility, and agility to the enterprise.

BIO: Paul Lipton is Computer Associates' Web services technology leader for the field services organization and a technology strategist in the office of the CTO. He has been an architect and developer of enterprise systems for more than 20 years, and has worked closely with key CA customers to architect distributed solutions using J2EE, .NET, wireless, and Web services technology. Paul has represented CA in various standards organizations, and has participated in the Java Community Process. He has published magazine articles on many technologies including Web services, Java, .NET, EAI, wireless technology, and distributed systems.

(WS8) Strategies for Securing Web Services

MARCK SECRIST, HP

Thursday, October 2, 9:00 a.m. - 9:50 a.m.

Web services standards for security are just beginning to emerge and stabilize, yet lack of security standards is listed as the top barrier to Web services adoption. Many project teams are reluctant to adopt Web services due to the risk associated with immature and changing standards. This session will look at the issues associated with securing Web services, and cover the important standards for Web services security. It will conclude by exploring strategies for securing Web services in ways that allow developers to get started with Web services while isolating themselves from standards that are still in a state of flux.

BIO: Mark Secrist is a senior consultant for the HP Developer Resource Organization with more than 10+ years of experience involving distributed object technologies and building N-tier, Web-based applications. He currently consults with enterprise customers on J2EE and Web services development. Mark has also published technical white papers on J2EE, mobile, and Web services development.

(WS9) Web Services Progress Report

MICHAEL CHAMPION, SOFTWARE AG

Thursday, October 2, 11:00 a.m. - 11:50 a.m.

Web services have been the buzz for the last couple of years, but the concept remains confusing to many. New "standards" are proposed on a regular basis, but they overlap and seem to form the same fault lines as previous industry politico-strategic controversies. Now analysts and writers are talking about "service-oriented architectures," further bewildering most nonspecialists. Throughout all this, a small but passionate group has been arguing that many of the ideas coming from the Web services community are antithetical to the principles of the Web and are unlikely to ever work on an Internet scale. The W3C Web Services Architecture Working Group is trying to determine a consensus position on key architectural issues concerning Web services and their relationship to XML and the Web. This presentation provides a progress report after an approximately 18-month effort to distinguish the Web services architectural principles from the marketing agenda of individual companies.

BIO: Michael Champion is a research and development specialist at Software AG, working out of Ann Arbor, Michigan. He has been a software developer for 20 years, working primarily in the area of middleware for client/server document and image management systems. He has been active in the World Wide Web Consortium's Document Object Model (DOM) Working Group for more than three years and was an editor of the core XML portion of the DOM Level 1 Recommendation. Michael is now cochair of the Web Services Architecture Working Group.

(WS10) The Seven Habits of Highly Effective Enterprise Service Buses (ESBs)

DAVE CHAPPELL, SONIC SOFTWARE

Thursday, October 2, 1:00 p.m. - 1:50 p.m.

Gartner, Inc., has defined a new class of application integration infrastructure called an enterprise service bus (ESB). An ESB is a standards-based integration product that reliably connects and orchestrates the interaction of hundreds of application endpoints spanning a global organization. Because of their significant differences from application server products and proprietary integration brokers, ESBs represent a compelling third option for businesses in the market for integration technology.

What is an ESB? How can you properly distinguish this new product category and compare it with other offerings on the market? Dave Chappell, chief technology evangelist for Sonic Software, will examine the seven key characteristics of an ESB in depth.

BIO: Dave Chappell is vice president and chief technology evangelist for Sonic Software, the leading provider of integration products and services for the real-time enterprise. Dave is coauthor of *Java Web Services*, *Professional ebXML Foundations*, and *The Java Message Service*, and a frequent contributor to *Web Services Journal*.

XML Track

XML Technology Track

XML EDGE conference&expo

Presentations will focus on the various facets of XML technologies as they are applied to solving business computing problems. Sessions will include emerging standards in XML Schemas, XML repositories, industry applications of XML, applying XML for building Web services applications, XML/XSLT/ XQuery-based programming using Java/.NET, XML databases, XML tools and servers, XML-based messaging, and the issues related to applying XML in B2B/EAI applications. The XML Track is geared for audiences ranging from beginners to system architects and advanced developers.

Sessions will focus on:

- XML Standards and Vocabularies
- Introduction to XForms
- Securing Your XML and Web Services Infrastructure
- XQuery Fundamentals: Key Ingredients to Enterprise Information Integration
- XML and Enterprise Architecture: Technology Trends
- Standards-Based Enterprise Middleware Using XML/Web Services
- XML and Financial Services
- Canonical Documents for Your Business: Design Strategies
- XPath/XSLT 2.0: What's New?
- XML Schema Best Practices
- XML in EAI, Enterprise Portals, Content Management

(XM1) Introduction to XForms

T.V. RAMAN, IBM RESEARCH

Tuesday, September 30, 9:00 a.m. - 9:50 a.m.

This presentation will introduce W3C XForms, including details about how XForms enables the last mile of connecting the end user to Web services. XForms user agents provide an easy-to-use, browser-based interface that enables the end user to interact with information technologies that have been published as Web services. As the Web moves from being a desktop-only phenomenon to a means of ubiquitous electronic access, Web transactions need to be available from a variety of end-user access devices ranging from desktop computers to smart phones. The XForms authored interface is well suited for delivery to a variety of interaction modalities and end-user devices, thus assuring content developers of the widest audience for their transaction-based applications. From the user's perspective, XForms revolutionizes the way business-critical information is collected and published on the Web, enabling information technologists to continue to model business data using abstract structures that are amenable to machine processing. XForms binds a user-friendly Web browser interface to such abstract XML models, thereby empowering the end user to edit and update these abstract structures. In this sense, XForms enables a standard Web browser to associate editable views to the underlying XML models. This ability to view and edit XML documents from within a standard Web browser is likely to prove a key empowering technology.

BIO: T.V. Raman works in IBM Research on multimodal user interfaces and is the author of *Auditory User Interfaces*. He is the editor of the user interface chapter of the XForms 1.0 specification and is an active participant in a number of W3C working groups including XForms, voice browser, and XHTML. His research interests are primarily auditory user interfaces and structured electronic documents, and his previous work includes AsTeR – Audio System For Technical Readings, Aural CSS, and Emacspeak – the complete audio desktop.

(XM2) Securing XML and Web Services Applications

PHIL STEITZ, AMERICAN EXPRESS

Tuesday, September 30, 3:00 p.m. - 3:50 p.m.

Securing Web services is a big challenge for companies moving toward service-oriented architectures. We will discuss some practical strategies for meeting this challenge using standards and technologies available today. Key technical tradeoffs in the areas of performance, scalability, manageability, availability, and integration cost will be considered as we work through the details of a typical integration use case. We will also discuss business and organizational issues related to trust and identity management. The session will conclude with a brief survey of the standards and technology landscape and some strategies for ensuring that investments made today will continue to pay benefits into the future.

BIO: Phil Steitz is vice president of e-commerce applications development at American Express. Phil has more than 20 years of experience as a developer, architect, and technology leader involved in distributed systems development. Before joining American Express, Phil served as a middleware architecture consultant, designing large-scale distributed systems for enterprise customers. Phil holds a PhD in mathematics from the University of Maryland.

(XM3) UBL: The Universal Business Language

JON BOSAK, SUN MICROSYSTEMS

Tuesday, September 30, 4:00 p.m. - 4:50 p.m.

Web services technologies promise to revolutionize electronic business, but global interoperability of business processes cannot occur without the semantic standardization of the messages exchanged in business transactions. This presentation will describe the OASIS UBL project to create standard XML Schemas for basic business documents, explore the relationship of UBL-based business to traditional EDI, and note the explosive potential of standard markup combined with reliable XML messaging.

BIO: Jon Bosak organized and led the W3C working group that created the XML specification and then served for two years as chair of the W3C XML Coordination Group. At Sun, where he holds the title of distinguished engineer, Jon sponsors projects intended to advance XML technology. He is currently chair of the Universal Business Language (UBL) Technical Committee of the Organization for the Advancement of Structured Information Standards (OASIS).

(XM4) Standards-Based Enterprise Middleware Using XML/Web Services

ANDY ASTOR, WEBMETHODS

Wednesday, October 1, 9:00 a.m. - 9:50 a.m.

Some say that standards-based enterprise middleware is not yet mature enough for industrial-strength usage, and that proprietary vendor products are the only practical alternative. Others insist that standards-based middleware is "ready for prime-time," and that proprietary vendors will be out of business within a year. The truth is that each of these extremes contain dubious hype, and the truth falls somewhere between these schools of thought. This session seeks to separate fact from fiction, and to offer a model for understanding both the current and future states of standards-based middleware.

BIO: Andy Astor joined webMethods in 2002 as vice president of enterprise Web services. In this role, he is responsible for driving the company's Web services strategy and execution. Prior to joining webMethods, Andy was a vice president at D&B, where he led worldwide customer-facing products, including all Web- and Internet-based applications. His work at D&B also included the development and launch of one of the earliest commercial Web services. Prior to his tenure at D&B, Andy held leadership positions at American Management Systems and Ernst & Young, both large systems integration and consulting firms.

(XM5) XML and Enterprise Architecture: Technology Trends

HITESH SETH, IKIGO

Wednesday, October 1, 3:00 p.m. - 3:50 p.m.

XML is a key ingredient of a number of technology solutions. Whether it's serving up enterprise portals, integrating diverse systems (EAI), serving Web services, or driving content management, support for XML has been a key trend in major off-the-shelf packaged applications. With Web services, XML support has been extended to mission-critical ERP and CRM systems. This presentation takes a look at some of the key trends in using XML in these applications.

BIO: Hitesh Seth, editor-in-chief of *XML-Journal*, is the chief technology officer of ikigo, Inc., a provider of XML-based Web services monitoring and management software. A freelance author and known speaker, he regularly writes for technology publications on VoiceXML, Web services, J2EE and Microsoft .NET, wireless computing, and enterprise/B2B integration.

(XM6) Using XML Schema Effectively in WSDL Design

CHRIS PELTZ, HP

Wednesday, October 1, 4:00 p.m. - 4:50 p.m.

Developers building Web services today are beginning to see the value of using the document-style approach over RPC. Recent experience shows that taking full advantage of document-style Web services requires a strong knowledge of XML Schemas and related XML standards. This session will present a number of important tips and techniques for properly using XML Schemas in the design of a Web services interface, including the importance of XML-based development tools, considerations for binding between XML and underlying objects, WSDL reusability through XML Schemas, and XML Schema naming best practices.

BIO: Chris Peltz is a senior software consultant within HP's Developer Resources Organization. He provides technical and architectural consulting to enterprise customers in the areas of J2EE, Web services, and mobile development. Chris has more than 10 years of software experience in object-oriented technologies, 4GL development, GIS, and Web application design.

(XM7) Canonical Documents for Your Business: Design Strategies

DAVID CONNELLY, OPEN APPLICATIONS GROUP, INC.

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

The ability to change rapidly when business processes change is key for successful enterprises in today's world. A factor critical to enabling this change is the integration of all of an organization's business software applications in a way that is much faster and less expensive than current methods. Many organizations are taking advantage of the open standards–based protocols underlying Web services to integrate systems quickly and inexpensively. But before Web services can achieve its full potential, it needs a rich and mature business language that can be used for both e-business and application integration. The OAGIS XML language is the technology-neutral, horizontal business language that enables enterprises to fully leverage Web services protocols. This presentation will discuss the OAGIS CANONCIAL Model, why and how it can make a difference, and strategies for using the OAGIS XML documents within the CANONICAL Model in your business environment.

BIO: David Connelly is president and CEO of the Open Applications Group, Inc. Before joining the OAGI, David worked in various capacities at customer organizations and software vendors. He graduated from George Tech in 1975 with a BS in industrial engineering.

(XM8) XQuery Fundamentals: Key Ingredients of Enterprise Information

PAUL COTTON, MICROSOFT

Thursday, October 2, 9:00 a.m. - 9:50 a.m.

The W3C XML Query Working Group was chartered in September 1999 to develop a query language for XML documents. The goal of the XML Query Working Group is to produce a formal data model for XML documents with namespaces based on the XML Infoset and XML Schemas, a set of query operators on that data model, and then an XQuery language with a concrete canonical syntax based on the proposed operators.
In May 2002 the XML Query WG and XSL WGs published a complete new set of XQuery 1.0/XPath 2.0/XSLT 2.0 documents. This presentation will provide an overview of the XQuery and XPath languages and their current status. It will also outline the relationship of the work of the XML Query WG to other W3C XML standards, especially XML Schema. The status of the WG's efforts to add support to XQuery for full-text retrieval and an update language will also be discussed.

BIO: Paul Cotton is program manager of XML standards with Microsoft Canada. Paul has been active within the W3C XML activity since 1998 and has been the chairman of the W3C XML Query WG since it was formed in 1999. Paul was elected to the first W3C Technical Architecture Group (TAG) in December 2001 and re-elected in December 2002. The W3C TAG is responsible for defining the W3C's view of the architecture for the Web. Paul is also a member of the WS-I Board of Directors and chair of the WS-I Basic Security Profile Working Group. Paul has more than 30 years of experience in the IT industry.

(XM9) XML at Work in the Fortune 500

Thursday, October 2, 11:00 p.m. - 11:50 p.m.

The objective of this session is to understand how XML has been a strategic investment with a Fortune 500 company. This session highlights the process of how support for XML was cultivated within the company and its partners and how XML was transformed from a cool technology trend to a necessity and a corporate standard. See www.sys-con.com/edge for further details on this session.

(XM10) Designing XML Schema Best Practices

AYESHA MALIK, OBJECT MACHINES

Thursday, October 2, 1:00 p.m. - 1:50 p.m.

XML Schemas constrain and formalize the vocabulary and grammar of XML documents. As XML is fast becoming the data transport format of the future, organizing the structure of the XML, as outlined by schemas, is becoming key to successful interoperability and implementation. Developers experienced in object-oriented design know that a flexible architecture ensures consistency throughout the system and helps to accommodate growth and change. This presentation uses an object-oriented framework to show you how to design XML Schemas that are extensible, flexible, and modular. Three principles of object-oriented design – encapsulation, inheritance, and polymorphism – are discussed in light of XML Schemas.

BIO: Ayesha Malik is a senior consultant at Object Machines, a software engineering firm providing Java technology and XML solutions to businesses. Ayesha has worked extensively on large XML and messaging systems for companies such as Deutsche Bank and American International Group (AIG). Most recently, she has been researching new ways to make schemas extensible and object-oriented.

OS X Track

OS X Track

OS X represents a new wave of operating systems. It combines the ease of use of a Mac with the power of Unix. Sessions in this track will highlight the use of the Mac OS X platform in applications and Web services development, deployment, and management.

Sessions will focus on:

- Introducing OS X Panther: What's New?
- Quick Applications Using AppleScript
- Enterprise Java and OS X
- Developing Web Services Using WebObjects
- Xserve: Ease of OS X and Power of Unix
- Introducing Quartz: 2D Graphics for Apple
- OS X for the Unix Developer
- Securing OS X Applications
- Java and OS X: A Perfect Marriage
- Programming Rich User Interfaces Using Cocoa

(OS1) Introducing OS X Panther: What's New?

Tuesday, September 30, 9:00 a.m. - 9:50 a.m.

The recently announced Mac OS X version 10.3, commonly known as Panther, boasts a number of key enhancements to the features and technologies introduced by OS X. This presentation looks at some of these new and exciting features and how they can benefit individuals and businesses considering deploying OS X.

(OS2) Programming Rich User Interfaces Using Cocoa

Tuesday, September 30, 3:00 p.m. - 3:50 p.m.

An evolution from NeXTStep APIs, Cocoa is a collection of advanced, object-oriented APIs for the development of OS X applications using Objective C and Java. Included in OS X is a set of Java classes that allows Java developers to build enriched multimedia applications based on the Cocoa framework. This presentation highlights the key benefits of the Cocoa application development model w.r.t. It also points out some areas where the Cocoa framework can provide the much-needed edge.

(OS3) Quick Applications Using AppleScript

Tuesday, September 30, 4:00 p.m. - 4:50 p.m.

A peer to the Aqua GUI, AppleScript is the language interface for Mac OS X. It can control and communicate between applications, databases, networks, and even remote Web services. AppleScript provides the technology needed for applications to automate workflow processes and extend the life of off-the-shelf products. Learn to build cross-applications using AppleScript Studio in this fast-paced introduction to AppleScript… scripting like you've never seen before.

(OS4) Java and OS X: A Perfect Marriage

Wednesday, October 1, 9:00 a.m. - 9:50 a.m.

Mac OS X sports a new look – not just a new look-and-feel on the outside, but also a new look on the inside. Mac OS is built on top of a BSD Unix–based core. An exciting aspect of the Mac OS X release is the integration of the Java 2 Platform with OS X. This presentation introduces OS X to users, focusing on Apple's Java implementation, platform support, Web browser support, and the tools that are available to help developers build great-looking Java applications for OS X.

(OS5) Enterprise Java and OS X

Wednesday, October 1, 3:00 p.m. - 3:50 p.m.

With built-in support for the Java 2 Platform in OS X operating systems and the evolution of the OS X Server platform, OS X will be considered a platform for not only developing and deploying rich-client applications but also for server-side applications. A number of options exist in this area. First and foremost is Apple's own WebObjects platform. In addition, OS X Server introduced bundling Apache Tomcat Server with the OS X Server and it was recently announced that Apple will bundle the JBoss Application Server with the new version of OS X Server. In addition to these "bundled" choices, a number of third parties have announced support for their own application servers for the OS X platform. This session previews some of the supported enterprise Java–based server-side technologies for the OS X platform.

(OS6) Developing Web Services Using WebObjects

Wednesday, October 1, 4:00 p.m. - 4:50 p.m.

WebObjects is Apple's framework and tools for rapid application development and deployment of server-side Java applications. Version 5.2 brings support for standards-based Web services. It allows developers to build or use standards-based Web services without getting into the details of writing low-level SOAP, XML, or WSDL. The environment includes a set of tools that enable code-free generation, configuration, and testing of Web services from existing data assets. This session previews some of these changes and walks attendees through the "really rapid" Web services development environment – WebObjects.

(OS7) Cocoa, Carbon, Java: Application Frameworks for OS X

Wednesday, October 1, 5:00 p.m. - 6:00 p.m.

From a development perspective, OS X provides multiple application frameworks. Choice is good, but it's important to know when to choose what. This presentation will provide guidelines on when to use which framework. Carbon is the traditional Mac OS API and emphasizes complete backward compatibility. Cocoa is a development environment for OS X "native" applications and represents a new framework for the development of OS X applications. Another key consideration is Apple's support for the latest version of the Java platform. Java is treated as a first-class citizen in OS X, and developers are expected to build and deploy cross-platform applications on this key framework.

(OS8) Securing OS X Applications

Thursday, October 2, 9:00 a.m. - 9:50 a.m.

Whether you're developing a banking application or just a graphics-based engineering application, security is an integral aspect of design and development. When applications are available beyond the firewalls, you must consider the additional challenges of securing your enterprise and customer data. Internally, you have to worry about potential misuse of rights. When it comes to OS X development, securing applications is no different. The operating system provides built-in support for LDAP directory services, making it easier to maintain consistent user profiles, and provides security APIs to a broad set of applications. This presentation provides a technology overview and a security checklist.

(OS9) Xserve: Ease of OS X and Power of Unix

Thursday, October 2, 11:00 a.m. - 11:50 a.m.

Xserve combines the ease of Mac OS X, the power of Unix, and the reliability of a server-grade solution. A one-of-a-kind deployment environment, Xserve promises to do for the production server's landscape what iBooks, iMacs, Power Macs, and PowerBooks have done for the desktop landscape – make it simple to use and easier to deploy. This presentation highlights the various platform-level capabilities of Xserve, including OS X Server.

(OS10) OS X for the Unix Developer

Thursday, October 2, 1:00 p.m. - 1:50 p.m.

The heart of a great-looking OS X operating system is a Unix Core. Unix developers should find themselves at home with the OS X platform. The same command-line utilities still work; however, you get ease-of-use as well. While you'll always have the flexibility of reverting to scripts and command-line tools, in a number of cases you won't need to. And even if you do, you can then make it simple for others by making the script available as a tool to others. This session looks at OS X from a Unix developer's perspective, highlighting key similarities and differences between the two environments.

(OS11) Introducing Quartz: 2D for Apple

Thursday, October 2, 3:00 p.m. - 3:50 p.m.

A key highlight of OS X architecture is Quartz, the lightweight window server and PDF-based 2D graphics-rendering library. Quartz provides crisp graphics, and anti-aliased fonts, and blends 2D, 3D, and QuickTime content together with transparency and drop shadows. Quartz Extreme boosts the performance of the Quartz rendering system with hardware-accelerated graphics. Developers get access to the rich functionalities available in Quartz through the Quartz 2D Core Graphics API for the Carbon framework and NSView, NSImage, and NSBezierPath classes for Cocoa developers. This session provides a technical introduction to Quartz and highlights application scenarios in which developers can utilize these rich capabilities.

REGISTRATION FORM

CONFERENCE: Sept. 30 – Oct. 2, 2003 EXPO: Sept. 30 – Oct. 1, 2003

Santa Clara Convention Center • Santa Clara, CA

THREE WAYS TO REGISTER FOR CONFERENCE

1) On the Web: Credit Cards or "Bill Me." Please make checks payable to SYS-CON Events.
2) By Fax: Credit Cards or "Bill Me" 201-782-9651
3) By Mail: 135 Chestnut Ridge Road, Montvale, New Jersey 07645, Attention: Registration

Please note: Registrations are not confirmed until payment is received.

Please complete sections 1, 2, 3 and 4

1 YOUR INFORMATION (Please Print) ☐ Mr. ☐ Ms.

First Name _____ Last Name _____

Title _____

Company _____

Street _____

Mail Stop _____

City _____

State _____ Zip _____ Country _____

Phone _____

Fax _____ E-Mail _____

2 PAYMENT METHOD: (Payment in full due with registration)

☐ Check or Money Order Enclosed (Registration confirmed upon receipt of payment)

Check #_____ Amount of Check $ _____

Charge my ☐ Visa ☐ MasterCard ☐ American Express ☐ Discover

Name on card_____

Card #_____ Exp. Date_____

Signature_____

Billing Address (if different from mailing address)

_____

3 PLEASE INDICATE YOUR CONFERENCE CHOICE

Total Registration fee $_____

	By 9/5/03	Before 9/26/03	Onsite
❑ **GP Gold Passport** Good for all three days of the .NET, Web Services, XML, Java, and Mac OS X Tracks, including preferred seating for the Oracle, IBM and Microsoft Russ' Toolshed Tutorials, Keynotes, Panel Discussions, and your choice of One Sun Microsystems Java University℠ Class Select one: ❑ Architecting Web Services Using J2EE (Oct. 1) ❑ Java 2 Platform: Architect Certification Fast Path (Oct. 2)	$1,295.00	$1,395.00	$1,495.00
❑ **3D Three Day Conference** (Does not include Sun Java Education)	$1,195.00	$1,295.00	$1,395.00
❑ **2D Two Day Conference** (Does not include Sun Java Education) (select any two days: ❑ Tue. ❑ Wed. ❑ Thurs.)	$1095.00	$1,195.00	$1,295.00
❑ **1D One Day Conference** (Does not include Sun Java™ Education) (select any one day: ❑ Tue. ❑ Wed. ❑ Thurs.)	$595.00	$595.00	$695.00
❑ **JU1 Sun Java™ University Class** Select one: ❑ Architecting Web Services Using J2EE (Oct. 1) ❑ Java 2 Platform: Architect Certification Fast Path (Oct. 2)	$695.00	$695.00	$795.00
❑ **JU2 Sun Java University Class** Attend both Architecting Web Services Using J2EE (Oct. 1) and Java 2 Platform: Architect Certification Fast Path (Oct. 2)	$1,195.00	$1,295.00	$1,395.00
❑ **VIP PASS** Good for access to the Exhibit Floor, Keynotes and Panel Discussions, Product Demonstrations, and your choice of (Select one): ❑ Microsoft Russ' Tool Shed (Sept. 30) ❑ How to Develop, Deploy, and Manage Web Services Using IBM Tools (Sept. 30) ❑ Web Services Workshop presented by Oracle (Oct. 1)	FREE	FREE	$50.00
❑ **EO Expo Only**	FREE	FREE	$50.00

4

A. Your Job Title

❑ CTO, CIO, VP, Chief Architect
❑ Software Development Director/Manager/Evangelist
❑ IT Director/Manager
❑ Project Manager/Project Leader/Group Leader
❑ Software Architect/Systems Analyst
❑ Application Programmer/Evangelist
❑ Database Administrator/Programmer
❑ Software Developer/Systems Integrator/Consultant
❑ Web Programmer
❑ CEO/COO/President/Chairman/Owner/Partner
❑ VP/Director/Manager Marketing, Sales
❑ VP/Director/Manager of Product Development
❑ General Division Manager/Department Manager
❑ Other (please specify)_____

B. Business/Industry

❑ Computer Software
❑ Computer Hardware and Electronics
❑ Computer Networking & Telecommunications
❑ Internet/Web/E-commerce
❑ Consulting & Systems Integrator
❑ Financial Services
❑ Manufacturing
❑ Wholesale/Retail/Distribution
❑ Transportation
❑ Travel/Hospitality
❑ Government/Military/Aerospace
❑ Health Care/Medical
❑ Insurance/Legal
❑ Education
❑ Utilities
❑ Architecture/Construction/Real Estate
❑ Agriculture
❑ Nonprofit/Religious
❑ Other (please specify) _____

C. Total number of employees at your location and entire organization (check all that apply):

	Location	Company
10,000 or more	01 ❑	01 ❑
5,000 – 9,999	02 ❑	02 ❑
1,000 – 4,999	03 ❑	03 ❑
500 – 999	04 ❑	04 ❑
100 – 499	05 ❑	05 ❑
100 or less	06 ❑	06 ❑

D. Please indicate the value of communications and computer products and services that you recommend, buy, specify, or approve over the course of one year:

❑ $10 million or more
❑ $1 million – $9.9 million
❑ $500,000 – $999,999
❑ $100,000 – $499,999
❑ $10,000 – $99,999
❑ Less than $10,000
❑ Don't know

E. What is your company's gross annual revenue?

❑ $10 billion or more
❑ $1 billion – $9.9 billion
❑ $100 million – $999 million
❑ $10 million – $99.9 million
❑ $1 million – $9.9 million
❑ Less than $1 million
❑ Don't know

F. Do you recommend, specify, evaluate, approve or purchase wireless products or services for your organization?

01 ❑ Yes 02 ❑ No

G. Which of the following products, services, and/or technologies do you currently approve, specify or recommend the purchase of?

❑ Application Servers
❑ Web Servers
❑ Server Side Hardware
❑ Client Side Hardware
❑ Wireless Device Hardware
❑ Databases
❑ Java IDEs
❑ Class Libraries
❑ Software Testing Tools
❑ Web Testing Tools
❑ Modeling Tools
❑ Team Development Tools
❑ Installation Tools
❑ Frameworks
❑ Database Access Tools / JDBC Devices
❑ Application Integration Tools
❑ Enterprise Development Tool Suites
❑ Messaging Tools
❑ Reporting Tools
❑ Debugging Tools
❑ Virtual Machines
❑ Wireless Development Tools
❑ XML Tools
❑ Web Services Development Toolkits
❑ Professional Training Services
❑ Other [Please Specify]

If you require special assistance covered under the Americans with Disabilities Act, please call 201-802-3058 by September 16, 2003.

CANCELLATIONS, SUBSTITUTIONS, REFUNDS

Fax written request to SYS-CON Registration 201-782-9651. Requests for refunds received prior to August 29, 2003, will be honored, less a 10% handling charge; requests received after August 29, 2003, and before September 12, 2003, will be honored less a 20% handling charge. No requests for refunds will be honored after September 12, 2003. Requests for substitutions must be made in writing prior to September 26, 2003. No one under 18 is permitted to attend. No warranties are made regarding the content of sessions or materials.

Speakers, sessions, and schedule are subject to change without prior notice.

No solicitation by anyone other than official exhibitors, sponsors or marketing partners is permitted. Such behavior is cause for expulsion without refund.

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

SPECIAL INSERT:
Web Services Edge West Conference & Expo Sept. 30 - Oct. 2, 2003

Register by September 5th and SAVE up to **$200**

web services conference EDGE & expo

SEPT. 30–OCT. 2, 2003
Santa Clara, CA

VISIT US AT
SYS-CON.COM/EDGE

WEB SERVICES EDGE CONFERENCE PAST SPONSORS & EXHIBITORS

WSJ ADVERTISER INDEX

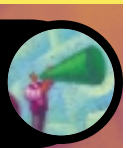
ADVERTISER	URL	PHONE	PAGE
BEA dev2dev 2003	bea.com/dev2devdays2003	925-287-5156	15
Borland	www.borland.com		23
Choreology	www.choreology.com		8, 9
Comdex	www.comdex.com		39
Ektron	www.ektron.com/ws		59
HostMySite.com	www.hostmysite.com/WSJ	877-248-HOST	27
JDJ Store	www.jdjstore.com	888-303-5282	29
LinuxWorld Magazine	www.sys-con.com	201-802-3020	33
Mindreef	www.mindreef.com	603-465-2204	5
Mindreef News	www.mindreef.com/nl0309		4
Parasoft	www.parasoft.com/wsj08	888-305-0041	2
Quest Software	www/java.quest.com/jcsr/ws	800-663-4723	11
Strike Iron	www.strikeiron.com		13
Strike Iron	www.strikeiron.com		60
SYS-CON Media	www.sys-con.com/suboffer.cfm	888-303-5282	37
Web Services Edge West 2003	www.sys-con.com	201-802-3069	41-56
WowGao, Inc.	www.wowgao.com	416-292-2364	21

Advertiser is fully responsible for all financial liability and terms of the contract executed by their agents or agencies who are acting on behalf of the advertiser. This index is provided as an additional service to our readers. The publisher does not assume any liability for errors or omissions.

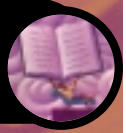
IN THE NEXT ISSUE OF WSJ...

Focus on UDDI & Standards

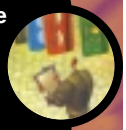
A Conversation with the WS-I
Sean Rhody talks to Tom Glover and Andy Astor about the WS-I's mission, membership, and plans for the future



What You Should Know About UDDI V3
A review of upcoming features in UDDI V3 from one of UDDI's original authors. This article will explore how to best make use of UDDI inside the enterprise.

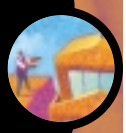


The WSDM Protocol
The OASIS interoperability consortium plans to define a standard way of using Web services architecture and technology to manage distributed resources – work that will be extremely important to business integrators who use Web services, management system vendors, and Web service platform vendors.



Plus

The Agile Enterprise: Migration to a Service Oriented Architecture
SOAs tend to be long-lived, need to be agile, and should be treated as one of your company's most valuable assets. However, a number of challenges arise in the migration of this strategy.



Web Services in the Real World: Avnet Computer Marketing
Companies have begun deploying significant Web Services-Based Integration (WSBI) architectures. We'll explore Avnet CM's business objectives, why they chose Web services for some parts of their architecture (and not others), and the results they achieved when they implemented WSBI.

WebServices JOURNAL .NET J2EE XML

Be Creative Implementing Web Services

Tyson Hartman

One of Avanade's senior technologists, Tyson Hartman is responsible for guiding Avanade's .NET vision and go-to-market solutions. He works closely with the company's sales, engineering, and delivery teams around the world to ensure that Avanade's solutions and services meet customer demand. Tyson is also an experienced public speaker and author, with particular expertise in Web services, application integration, and high-volume transactional Web sites.

tysonh@avanade.com

A recent Accenture survey ("Web Services: IT Efficiency Today…Powerful Business Solutions Tomorrow," January 2003) revealed that most C-level executives expect standards-based technology to deliver on the promise of Web services – and only 16% think Web services are "cutting edge" technology. So it would appear Web services are moving beyond pure hype and into mainstream enterprises.

"Good news," say IT pros, "but what's to be done while Web services standards are still in development?"

In fact, we don't have to wait for all the standards to start taking advantage of them. We simply have to be creative until they are finalized. Even as protocols evolve, innovative thinking and inventive use of tools can fill the void and produce alternatives to fulfill customers' business and technology needs using Web services technology available today. Here are a number of creative alternatives.

Security and validation services will be ubiquitous when Web services become widespread. In the meantime, a Web services gateway can be built today to perform authentication, validation, transformation, and exception handling for Web service–based solutions.

This was our approach for a federal unemployment agency that works in coordination with other government entities. A mainframe had previously handled all navigation and business logic processing in response to job-seekers' online queries. The agency sought to improve service by allowing more access to information, centralizing responsibility for data and related processes, and creating a foundation for expansion.

Rather than wait until all the necessary products were available as open, standards-based components, we recommended using those that *were* available, and reusing or rewriting legacy technology and applications for which solutions were not yet available.

In this case, the customer retained the mainframe database and chose the Microsoft .NET development platform for Web services–based integration and security. We built a Web services gateway to provide security and validation services on the edge of the enterprise, implementing Web services standards externally, while reusing the existing mainframe internally. This provided a solid, flexible foundation for interaction with internal and external audiences and systems.

Another interim approach is to provide new enterprise service aggregations to mask underlying complexity. This model exposes Web service interfaces internally for many consumers of a common domain of information. For example, mandating use of a single central CRM package might be too disruptive for companies that already have legacy customer management systems in production. A CRM Web services layer presenting a single interface to both old and new systems could smooth over complexity, as well as extend an application's lifetime utility and extract greater value from all of them.

Building gateways and services for aggregation are just two ways to implement Web services using existing protocols, even as additional standards evolve. Companies also may consider using products that provide alternatives to nonexistent standards. Major platform vendors will likely ensure users a migration path as standards evolve, too.

For example, Microsoft BizTalk Server 2004 is earning early praise for management and consumption of Web services and business process integration. Based on Web service standards XSD, the product will support using SOAP, WSDL, and BPEL. The business services in BizTalk Server 2004 and many other traditional EAI tools can perform process coordination, routing, reliable messaging, and more, in place of the standards while they evolve. By exposing Web service interfaces externally, and implementing the functional equivalent behind the scenes with another product, you can take advantage of current standards and implement new ones without inconveniencing customers.

As Gartner analysts Roy Schulte and Yefim Natis point out in "Most Composite Applications Will Need an Integration Layer" (April 2003), mismatched technology and mismatched information are primary obstacles to the Web services ideal.

The analysts propose that an ESB combine Web services, messaging middleware, intelligent routing, and transformation to serve as a lightweight, ubiquitous integration backbone. A Web service would make a request to the ESB layer, which would engage with multiple systems and respond.

Even ESBs are not a silver bullet; software vendors still need to expose their products as Web services. But ESBs are a valuable model, and existing elements in the IT environment can supply ESB functions such as message queuing and workflow management. Add a Web services gateway for validation, authentication, transformation, and exception handling, and you've got a creative solution to many business problems today.

If there is reason to act now, creative thinking can produce solutions that fill the void while the technology industry agrees on and implements standards. IT professionals should weigh the needs of the business and assess whether they can afford to wait for standards to implement Web services or proceed now to reap some of the benefits immediately.

Strike Iron

www.strikeiron.com