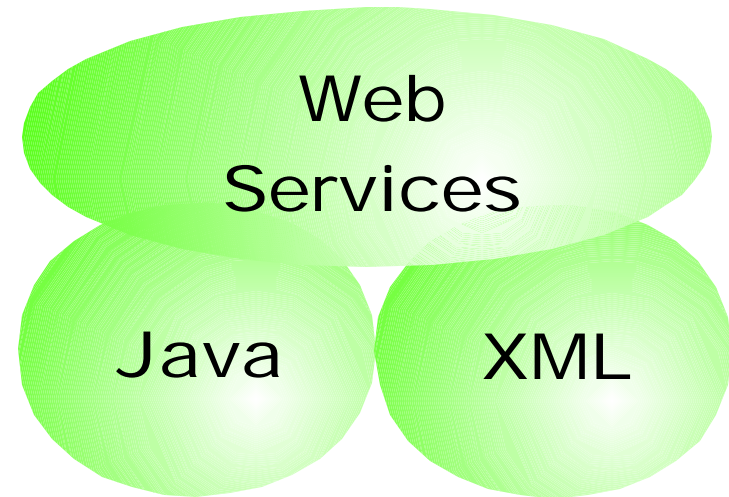


SESSION	<b>Java, XML, and Web Services - Keys to Your Future</b>
NUMBER	<b>E432711</b>
SPEAKER	<b>Michael Conner</b>
TITLE	<b>Chief Technical Officer for Web Services</b>
AFFILIATION	<b>IBM, <a href="mailto:mconner@us.ibm.com">mconner@us.ibm.com</a></b>

In this session, we will give an overview of IBM's technical strategy with respect to Java, XML, and Web Services. We will discuss the implementation of these e-business technologies in IBM products and also describe IBM's strategy with respect to industry and middleware standards for these technologies. The focus will be on web services and how Java and XML can be used together to create dynamic web services that can easily be advertised, discovered and invoked over the Internet.

# Dynamic e-business

The Networked economy is driving an evolution of e-business



- 1-to-1 => many-to-many collaborations
- Packaged software applications => self-contained, interoperable, modular components
- Rigid, point-to-point application integration => JIT sourcing and “software assemblies”
- Software as a product => software as service (utility)

# What are Web Services

- **Web services....**

- ▶ self-describing, self-contained modular applications
- ▶ use open standards and common infrastructure for description, discovery and invocation (e.g. WSDL, UDDI, SOAP, HTTP)

- **Web services perform encapsulated business functions: From simple request-reply to full business process interactions**

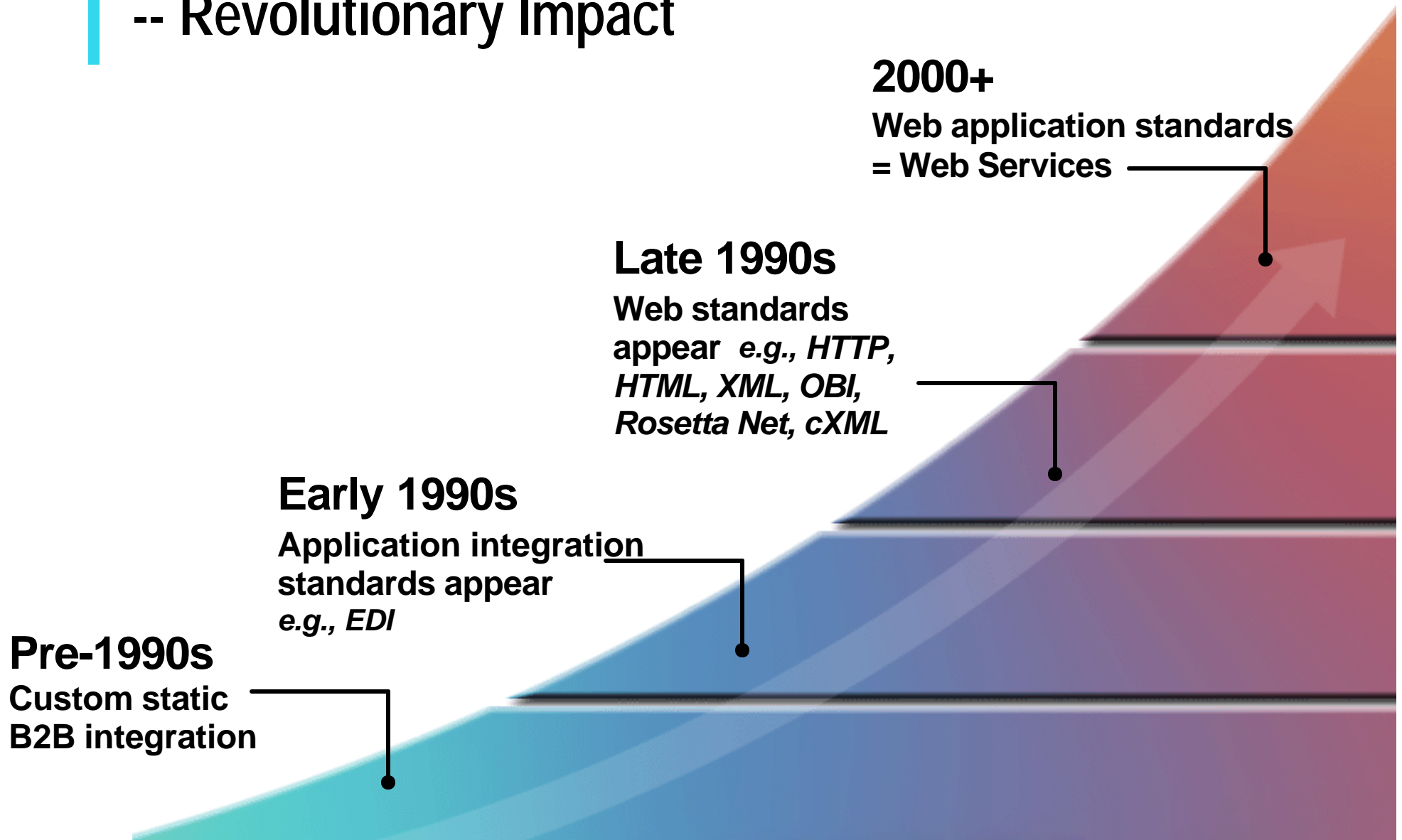
- ▶ stock quotes/stock charting
- ▶ credit card verification/payment processing
- ▶ integrated travel planning
- ▶ RFQ/bid process/auctions

- **Can be mixed and matched to create complete process, product**

- **Enable dynamic integration with decreased human interaction**

- **Both new & extensions to existing applications**

# Evolutionary Technology -- Revolutionary Impact



# Web Services: A revolution for B2B

- **The Web revolutionized business-to-customer**
    - ▶ Massive extension of business opportunity
    - ▶ New business models
    - ▶ Dramatic reduction in infrastructure costs and complexity
  - **The key was a universal server-to-client model based on standards and industry support**
- 

- **Web Services promises to do the same thing for business-to-business**
  - ▶ Massive extension of business opportunity
  - ▶ New business models
  - ▶ Dramatic reduction in infrastructure costs and complexity
- **The key will be a universal program-to-program communication model based on standards and industry support**

# The Value of Web Services



## Businesses benefit by using Web Services to:

- Connect applications to applications in other businesses quickly and easily
- Establish interaction with marketplaces more efficiently
- Deliver business functions to a broader set of customers and partners
- Pursue new business models by combining applications in new dynamic ways
- Reduce costs of internal integration, while increasing reuse

# When Do Web Services Make a Difference?



When you are:

- **Providing service to your customers**
  - ▶ e.g., How long & how much does it cost an Insurance company to link its systems to the systems of a new institutional customer?
- **Accessing services from your partners and suppliers**
  - ▶ e.g., Dynamically link to new partners and suppliers to offer their services to complement the value you provide?
- **Integrating existing and new function in your internal systems**
- **Providing a better end-user experience**

# How Can Web Services Be Used?

## ■ **Between businesses**

- ▶ **Standards and common infrastructure reduce the barriers**
- ▶ **Simplicity accelerates deployment**
- ▶ **Dynamics opens new business opportunities**

## ■ **Within an enterprise**

- ▶ **Accelerate and reduce the cost of integration**
- ▶ **Save on infrastructure deployment and management costs**
- ▶ **Reduce skill requirements**
- ▶ **Improve reuse**

## ■ **Between a business and end-users**

- ▶ **Deliver a better user experience**
- ▶ **Integrate diverse content**
- ▶ **Reduce the cost of content delivery**

# The Web Service Model

- **Simple:** XML, HTTP
- **Dynamic:** Publish, Find, Bind
- **Standard:** XML, XML Schema, SOAP, WSDL, UDDI, ...



# The Magic of Web Services

- **Standards**
  - ▶ TCP/IP, HTTP, XML, ...
- **Industry Support**
  - ▶ The Internet, routers, load balancing, firewalls, web servers, ...
- **Good Choices**
  - ▶ Loose coupling, simple, solution driven

# The Web Services Stack

- HTTP
- XML
- XML Schema
- SOAP - Simple Object Access Protocol
- WSDL - Web Services Description Language
- UDDI - Universal Description, Discovery and Integration
- Java

# XML Schema

- **Defines XML messages**

- ▶ **Syntax: ordering, required/optional,**
- ▶ **Types: numbers, dates, intervals -> all compatible with Java**
- ▶ **Uses an XML based notation**
- ▶ **Replaces XML DTD**
  
- ▶ **Beta support available in the XML parser**

# SOAP

- **Standard for XML based requests and responses**

- ▶ **Can work with many transport protocols, but HTTP(S) is already defined and deployed today (MQ binding in alpha, HTTPR(s) being defined)**
- ▶ **Defines many header elements and a basic payload envelope.**
- ▶ **Also, provides a refinement for RPC.**
  
- ▶ **Draft proposal for public key based signing and encryption - this will be the primary security mechanism**
- ▶ **Draft proposal for attachments (non-XML content)**
- ▶ **Work underway to define reliable, 1-time delivery (based on HTTP, but could leverage MQ as well)**
  
- ▶ **Support available today for Java, Apache, WebSphere**
- ▶ **In commercial use today**
- ▶ **Will be formally standardized in W3C XML Protocol effort**

# WSDL

- **Standard for defining Web Services**

- ▶ **Defines an abstract interface and bindings to particular message formats (e.g., SOAP) and protocols (e.g., HTTP)**
- ▶ **Open, but only SOAP 1.1, HTTP GET/POST, and MIME binding are documented**
- ▶ **Written in XML, leverages XML Schema**
  
- ▶ **Tools available to generate WSDL from a number of Java service implementations (bean, servlet, EJB)**
- ▶ **Tools available to generate client binding**
  
- ▶ **Used to publish services in UDDI**

# UDDI

## ■ Standard for publishing Web Services

- ▶ Services can be found by searching or by unique id
- ▶ Servers provide both a browser and a SOAP based web service interface for publishing and finding services
- ▶ Can be used to register any service (e.g., 800 number, web site, web service, EDI service, ...)
  
- ▶ Free, public, interconnected UDDI servers are deployed today!
- ▶ A private UDDI implementation is available today in the WSTK

# Coming: Web Services Flow Control

- **Business process definition and integration**
  - ▶ XML Notation for business process definition
  - ▶ Manage internal processes
  - ▶ Sequence constraints at the service boundary



# Web Services Support

# Provide best of breed support for Web services

## ■ Today

- ▶ Web Services Toolkit on alphaWorks
- ▶ Web Services Development Environment

## ■ Later this year

- ▶ WebSphere Application Server
- ▶ WebSphere Studio
- ▶ WebSphere Business Components
- ▶ DB2
- ▶ MQ Family support for SOAP, (HTTPR support coming)
- ▶ Tivoli Web Services Manager
- ▶ Lotus Domino

# Web Services Toolkit

## ■ Contents

- ▶ A Service Broker that uses WSDL to support the Publish, Unpublish and Find operations against a UDDI registry, and also provides a set of Java APIs for accessing UDDI directly;
- ▶ An implementation of a UDDI registry;
- ▶ A Web Services Browser Tool that aids in the Publish, Unpublish and Find operations;
- ▶ A Web Services Creation Tool for assisting in encapsulating legacy code as web services;
- ▶ Java bindings for UDDI, WSDL service broker, and web services
- ▶ An ADS web service for discovering services that are advertised via ADS;
- ▶ The Apache SOAP 2.x implementation;
- ▶ A sample application called Gourmet2Go, as well as many smaller samples;
- ▶ A prototype showing how to take wrap EJBs as web services;
- ▶ A demo showing how digital signatures can be used in SOAP messages;
- ▶ A demo illustrating aggregation of several web services
- ▶ Micro-WebSpher

# WebSphere and VisualAge Tools

Tool support for SOAP, UDDI and other Web Services standards to be delivered in 1H2001 as a part of WebSphere Studio (alpha release available now)

## ■ Web Services Development Environment

- ▶ The first development environment that enables developers to create Web Services for deployment across multiple platforms
- ▶ Supports open standards: XML, HTTP, UDDI, SOAP, WSDL, and Java™ technology
- ▶ Supports publishing and finding Web service descriptions in public or private UDDI registries
- ▶ Generates Java code that implements the interface to a Web service described in WSDL
- ▶ Extends existing applications and turns them into e-business applications, allowing programmers to create Web Services
- ▶ Leverages the complete range of WebSphere servers and supporting tools

# WebSphere Application Servers

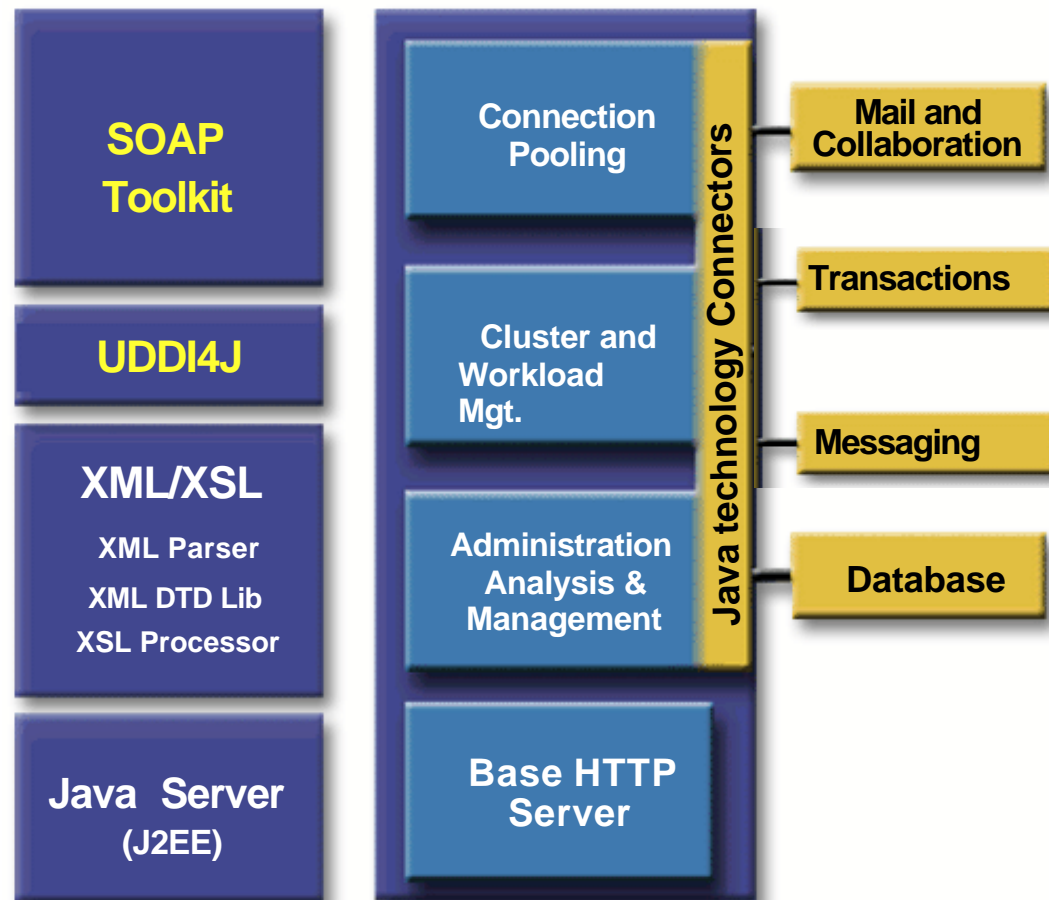
UDDI and SOAP enhancements to be delivered in 2001

## Integration of SOAP4J Toolkit

- Apache SOAP4J environment
- Enables WebSphere applications to send and receive SOAP and WSDL messages

## Integration of UDDI4J

- Java interface to UDDI registries
- Enables WebSphere applications to communicate with UDDI-compliant registries to publish and find Web services



# Java - The Platform for Enterprise Applications

## Continue to Improve for Enterprise Applications

- **Smarter tools that scale to enterprise class development teams**
- **Execution: Meet high performance requirements of fast, short-lived transactional applications and very large address spaces**
  - ▶ Reduce overhead in initialization and termination
  - ▶ Efficient garbage collection for very large memories
  - ▶ Isolation across transactions - fast reset
  - ▶ Maintain compatibility and operational characteristics of existing Java applications
  - ▶ To be made available as enhancement to JVM technology from IBM on OS/390, Windows, AIX & Linux platforms

# More Web Services Support

- **DB2**
  - ▶ Improved XML content and results support
  - ▶ DB2 to support stored Web Services procedures
  - ▶ XML Query
- **WebSphere Business Components**
  - ▶ Enabled for UDDI, SOAP and other Web Services standards in 2001
- **MQ Series Family**
  - ▶ SOAP support in 1H2001
- **Lotus**
  - ▶ Direction to enable Domino services as Web services, by incorporating SOAP interfaces and XML-based messaging
- **Tivoli Web Services Manager**
  - ▶ Tools to manage Web application servers
  - ▶ Plan to add SOAP support in 2001

# Help customers and business partners take advantage of Web services

- **jStart consulting and deployment program for Web Services**
- **Advanced Technology Services**
- **New developerWorks Web Services Zone**
- **Patterns for e-business adding web service and private registry patterns**
- **IGS Competency Center**

# Web Services Resources

- **jStart:** [www.ibm.com/framework/jstart](http://www.ibm.com/framework/jstart), [jstart@us.ibm.com](mailto:jstart@us.ibm.com)
- **Web Services Toolkit**  
[www.alphaworks.ibm.com/tech/webservicestoolkit](http://www.alphaworks.ibm.com/tech/webservicestoolkit)
- **Web Services Development Environment:**  
[www.alphaworks.ibm.com/tech/wsde](http://www.alphaworks.ibm.com/tech/wsde)
- [www.ibm.com/developerWorks](http://www.ibm.com/developerWorks)
- **XML:** [www.w3.org/XML/](http://www.w3.org/XML/)
- **XML Schema:** [www.w3.org/XML/Schema](http://www.w3.org/XML/Schema)
- **SOAP:** [www.w3.org/TR/SOAP/](http://www.w3.org/TR/SOAP/)
- **XML Protocol:** [www.w3.org/2000/xml/](http://www.w3.org/2000/xml/)
- **WSDL:** Spec is in the WSTK
- **UDDI:** [www.uddi.org](http://www.uddi.org), [www.ibm.com/services/uddi](http://www.ibm.com/services/uddi)
- **Apache: Server, Servlets, JSP, XML tools:**  
[www.apache.org/](http://www.apache.org/)



# IBM's Leadership

# IBM Leadship in Web services standards

## ■ XML Schema

- ▶ Author of the primer
- ▶ Contributor in all phases

## ■ SOAP - Simple Object Access Protocol

- ▶ Co-author of specification (with Microsoft)
- ▶ First implementation
- ▶ Chair of XML Protocol working group in W3C
- ▶ Contributor of SOAP4J to Apache open source project

## ■ WSDL - Web Services Description Language

- ▶ Co-author of specification (with Microsoft)
- ▶ First WSDL toolkit implementation on alphaWorks

## ■ UDDI - Universal Description, Discovery and Integration

- ▶ Co-designer (with Microsoft and Ariba)
- ▶ Leader in creation of UDDI project
- ▶ UDDI Business Registry beta host

# Web services is a natural evolution of IBM's e-business leadership

- Builds on experience with thousands of e-business engagements
- Continues our commitment to developing and adopting open standards
- Supports cross-platform, multi-language solutions
- Continues business integration leadership

# Web Services

## *An IBM-wide initiative*

- IBM's Web services support is a natural evolution of its e-business strategy
- IBM is first to deliver an open, integrated platform to build, test, deploy and publish Web Services
- IBM is extending WebSphere software platform so that customers, partners and developers can develop, assemble and deploy Web services

End

# Java, XML, and Web Services - Keys to Your Future