WebServices - an Introduction

David Artus - (based on the original by Mark Colan)
Technical Consultant, Software Services, IBM UK Ltd.

Agenda

- Introduction
- WebServices overview
- WebServices adoption
- WebServices technology futures
- Q&A

Introduction

Indicators(1) - Gartner

- Predicts 30% increase in developer productivity using Web Services
- More companies going "outside" themselves
- Through 2H02, 75% Large Companies will use Web Services
- Through 1H03, 50% Medium/Small Companies will use Web Services

October 2001

Indicators(2) - John McKinley, CTO, Merrill Lynch & Co.

"We've already made the bet that Web services will be probably one of the most fundamental pillars of our overall technology architecture ...

"Do I think it will be as fundamental to us as mobile computing? Yes, you bet. I think this is one of the fundamental changes in our overall computing architecture that will affect us over the next two decades."

"There's a huge amount of value just for embracing this robustly across all of Merrill... The biggest opportunity beyond that, as we expose our own services to clients really becomes an inter-enterprise infrastructure, as well.

Computerworld, March 2002

Inidicators(3) - Our customers

- We have customers who are already realizing the cost-savings and flexibility offered by Web services
  - visit ibm.com/software/jstart for some case studies

WebServices - why the interest?

Simpler linking between distributed systems

Dynamic e-business:

- Cannot control end-point technologies
  - will be heterogeneous,
  - must be platform neutral
- Constant evolution and extension
  - need agility
  - loosely coupled
  - scalable
- Participants have varying degrees of sophistication
  - Simple to set up
  - Minimal education and support
  - Toolable
  - Low cost-of-entry

but, all the above can apply to **internal** systems integration

A Universal Internet Programming Model

- Share functionality and information on the Web, regardless of
  - Operating system
  - Hardware or delivery device
  - Programming language
  - Distributed object system
- Direct program-to-program integration for
  - Business-to-business applications
  - Enterprise Application Integration
  - Reusable components for interactive applications
  - ...and it's general enough to handle anything else that requires integration across a network
- If XML defines a platform-independent way of **<u>representing</u>** data,
  - making <u>data integration</u> easy and standard
- ...then Web services defines a platform-independent way of **<u>exchanging</u>** that data.
  - <u>process-level integration</u> becomes easy
- This is all moving very quickly because, basically, it is a really good idea.

WebServices Defined

What is a Web Service?

"Web services are software components described via WSDL which are capable of being accessed via standard network protocols such as SOAP over HTTP."

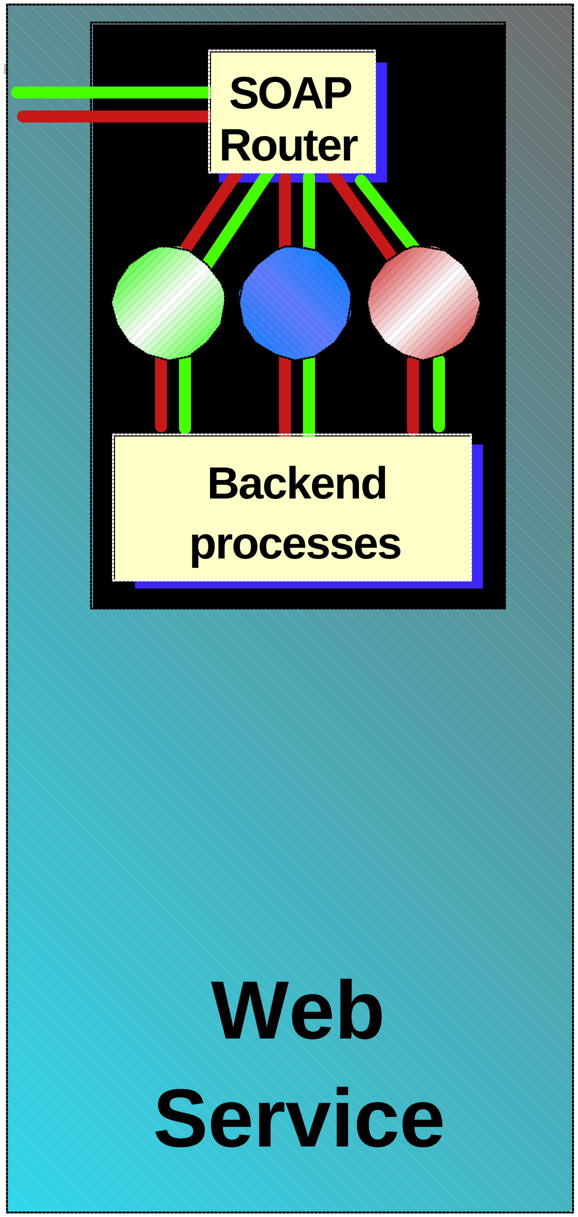
?

Web Service

What is a Web Service?

"Web services are software components described via WSDL which are capable of being accessed via standard network protocols such as SOAP over HTTP."

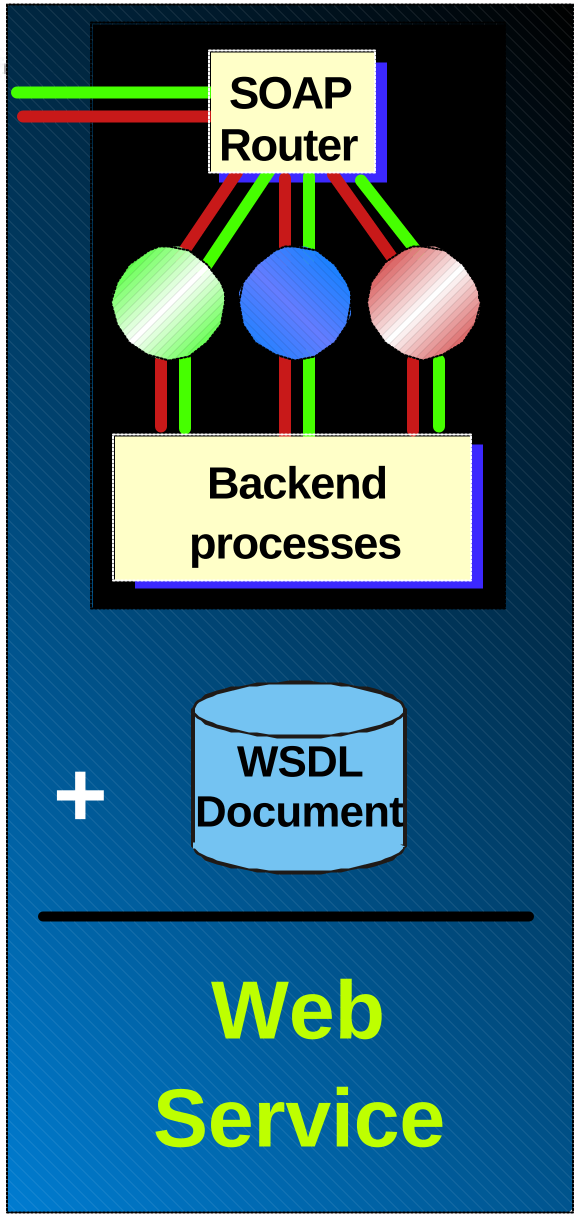
- Today, SOAP over HTTP is the common protocol for Web services.
- The architecture will still be called Web services even if the protocol changes.
- For now, a SOAP interface connected to application processes can be thought of as a minimum...
  - ...but by itself does not address rapid integration.



What is a Web Service?

"Web services are software components **described via WSDL** which are capable of being accessed via standard network protocols such as SOAP over HTTP."

- WSDL descriptions can be used to drive assembly tools, code generators, and other tools to speed integration.
- For now, SOAP+WSDL can be thought of as the base technologies for any Web service.
  - UDDI, other technologies can be considered optional, to add on as makes sense for the application



A SOAP Request Message

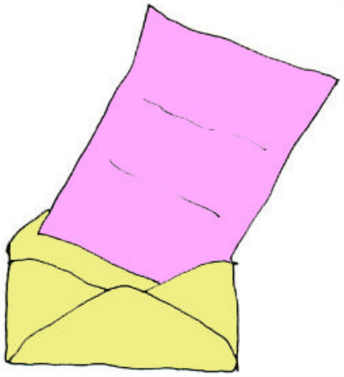
```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV=
     "http://www.w3.org/2001/06/soap-envelope"
    SOAP-ENV:encodingStyle=
     "http://www.w3.org/2001/06/soap-encoding">

   <SOAP-ENV:Body>
       <m:GetLastTradePrice xmlns:m="Some-URI">
           <symbol>IBM</symbol>
       </m:GetLastTradePrice>
   </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

message

SOAP envelope



Why SOAP Will Succeed

Other distributed technologies failed on the Internet because they strongly coupled the endpoints and therefore could not become pervasive:

- ✘ Unix RPC - requires binary-compatible Unix implementations at each endpoint
- ✘ CORBA - requires compatible ORBs
- ✘ RMI - requires Java at each endpoint
- ✘ DCOM - requires Windows at each endpoint

✓ SOAP is the platform-neutral choice

- ➡ simply an XML wire format
- ➡ places no restrictions on the endpoint implementation technology choices

SOAP and Standardization

- SOAP was submitted to W3C for consideration as a standard (summer 2000)
  - W3C "XML Protocol" working group: Sept 13, 2000
    - work is done in public
    - http://www.w3.org/2000/xp
    - chair: David Fallside, IBM
  - What's available now:
    - SOAP Version 1.2 Part 0: Primer (Working Draft: 17 December 2001)
    - XML Protocol Abstract Model (Working Draft: 9 July 2001)
    - XML Protocol Requirements Document
  - Quite similar to SOAP 1.1, some additions
  - SOAP 1.2 Specification is the name of the new spec (not "XML Protocol")
  - SOAP does not stand for anything

SOAP Resources

- SOAP 1.1 Specification
  - http://www.w3.org/TR/SOAP/
- Apache SOAP4J: xml.apache.org
  - SOAP4J version 2.2, stable, ready for use
  - AXIS (Alpha 3 available)
- W3 standardization: w3.org/2000/xp
  - SOAP 1.2 specification (draft)
  - XML Protocol working group requirements and charter
- SOAP - WebServices Resource Center
  - http://www.soap-wrc.com/webservices/default.asp
  - MANY resources - e.g., link to SOAP::Lite for Perl
- Xmethods lists publicly-accessible web services
  - http://www.xmethods.net
- Articles and tutorials:
  - http://ibm.com/developerworks/webservices
- Microsoft's SOAP toolkit (!Java)

Apache SOAP 2.2

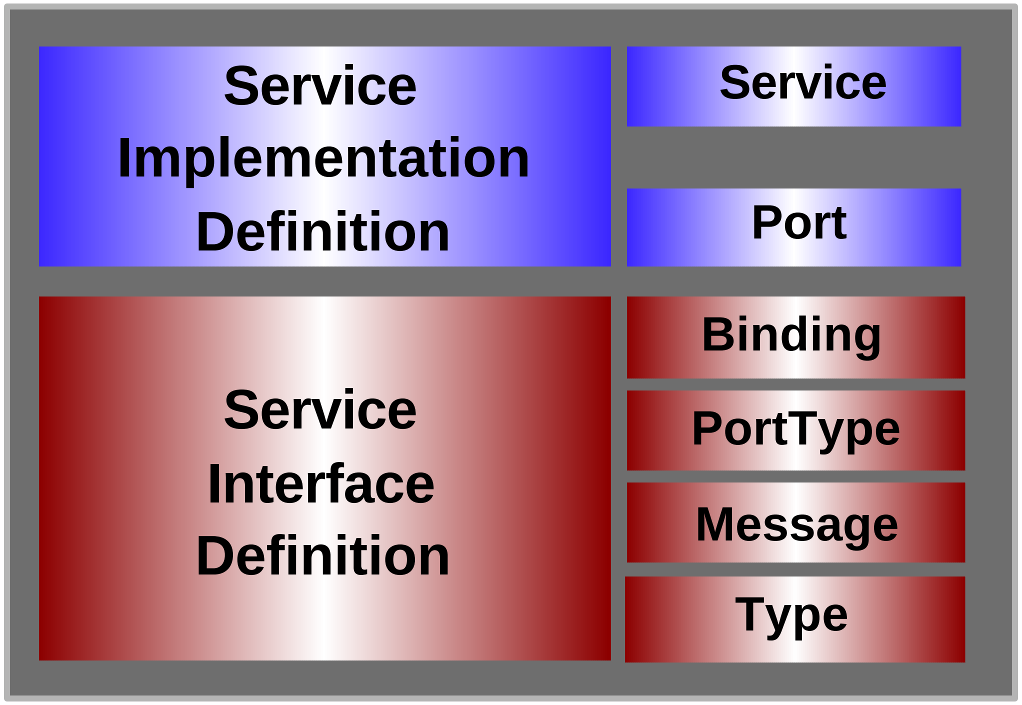
- History:
  - SOAP4J posted to IBM alphaWorks, April 2000
  - Contributed by IBM to the Apache Software Foundation, June 1, 2000
- SOAP from Apache:
  - Solid implementation of SOAP v1.1 Specification, supporting HTTP and SMTP protocols
  - platform-independent Java
  - Developed by IBM and others
  - Free download from xml.apache.org with source
- SOAP distribution includes:
  - User's Guide
  - API documentation
  - a tool for debugging SOAP
  - three samples

AXIS: SOAP 3.0

- New codebase implementing W3C SOAP 1.2 Specification
  - will feature full support for the spec
  - flexible chains of message handlers
  - speed: SAX events for parsing SOAP messages
  - adds WSDL support
  - http://xml.apache.org/axis/index.html
- Alpha 3 release is available
  - code is still under development
  - adds JAX RPC support, WSDD, more WSDL support
- Implementation team spans many companies
  - including IBM (of course!)

WSDL: Web Services Description Language

- WSDL - The key to interoperability
  - where the service is located (service implementation definition)
  - what the service does (service interface definition)
  - machine readable, generated and used by IDEs
  - similar in purpose to IDL, but in XML form


Service Implementation Definition
Service
Port
Service Interface Definition
Binding
PortType
Message
Type

WSDL Interface Example

Messages

```
<message name="getRateReq">
  <part name="country1" type="xsd:string"/>
  <part name="country2" type="xsd:string"/>
</message>

<message name="getRateResp">
  <part name="result" type="xsd:float"/>
</message>
```

Port

```
<portType name="Exchange">
  <operation name="getRate">
    <input name="getRateReq" message="getRateRequest"/>
    <output name="getRateResp" message="getRateResponse"/
  </operation>
</portType>
```

WSDL Interface Example ...

```
  <binding name="ExchangeBinding" type="Exchange">
    <soap:binding style="rpc"
       transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="getRate">
      <soap:operation soapAction="Exchange" style="rpc"/>
      <input>
        <soap:body use="encoded"
         encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/
      </input>
      <output>
        <soap:body use="encoded"
         encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/
      </output>
    </operation>
  </binding>
</definitions>
```

Binding

Encoding

WSDL Implementation Example

```
<service name="ExchangeService">
  <port name="ExchangePort" binding="binding:ExchangeBinding">
    <soap:address
         location="http://www.exch.com/servlet/rpcrouter"/>
  </port>
</service>
```

Target location

Who Writes WSDL Interface?

- Many Plausible Scenarios, examples:
  - The Service author - "Use my RouteFinder service - **here's** the WSDL"
  - The Service user - "All suppliers must provide an order tracking service according to **this** WSDL"
  - An authority - "The Widget Council provide **this** standard WSDL for a Widget Specification service"
- Each scenario implies different approaches
  - to obtaining the WSDL interface
  - to obtaining the WSDL **implementation** (where to find the service)

WSDL Resources

- WSDL 1.1 Specification
  - http://w3.org/TR/wsdl
- WSDL4J
  - http://oss.software.ibm.com/developerworks/projects/wsdl4j
- WSDL Toolkit (part of WSTK)
  - http://ibm.com/alphaworks (look under xml on left)
- WS Application Developer (beta available soon):
  - http://ibm.com/software/webservers/studio/preregister.html
- WSDE (early version of WSAD available now):
  - http://ibm.com/alphaworks (look under xml on left)
- Articles and tutorials:
  - http://ibm.com/developerworks/webservices

WSDL4J: an Open-source Java API

- Allows the creation, representation, and manipulation of WSDL documents describing services
- Service descriptions can be treated by a client in a uniform manner, regardless of the origin of the description:
  - parsing a WSDL document
  - constructed programmatically by direct invocation of the APIs
  - built using information provided by a user via a command-line or graphical interface
  - built using information retrieved from a network source
- Reference implementation of JSR110
- Visit:
  - oss.software.ibm.com/developerworks/projects/wsdl4j

How does the Requestor get the WSDL?

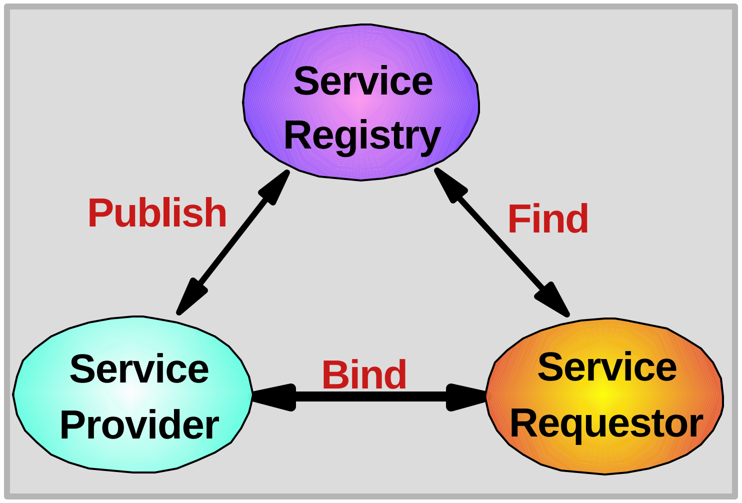
- What are the ways a requestor can get the WSDL?
  - email, sneaker-net ...
  - Web site, perhaps downloadable with sample client code
  - ...or use UDDI "find" methods to look it up in the UDDI Business Registry

What is UDDI?

- Universal Description, Discovery, and Integration
- A project to speed interoperability and adoption for web services
  - Standards-based specifications for service description and discovery
- A set of Internet-based **implementations**
  - UDDI Business Registry
  - Interoperating to share registrations
- Partnership among industry and business leaders
  - Now over 300 UDDI community members
  - Initiated by IBM, Microsoft, and Ariba
- UDDI has two pieces:
  - the UDDI Business Registry (hosts the data)
  - the API and data model (provides access to the data)

UDDI Roles and Operations

- Service Registry
  - provides support for publishing and locating services
  - like telephone yellow pages



- Service Provider
  - provides e-business services
  - **PUBLISH**es availability of these services through a registry

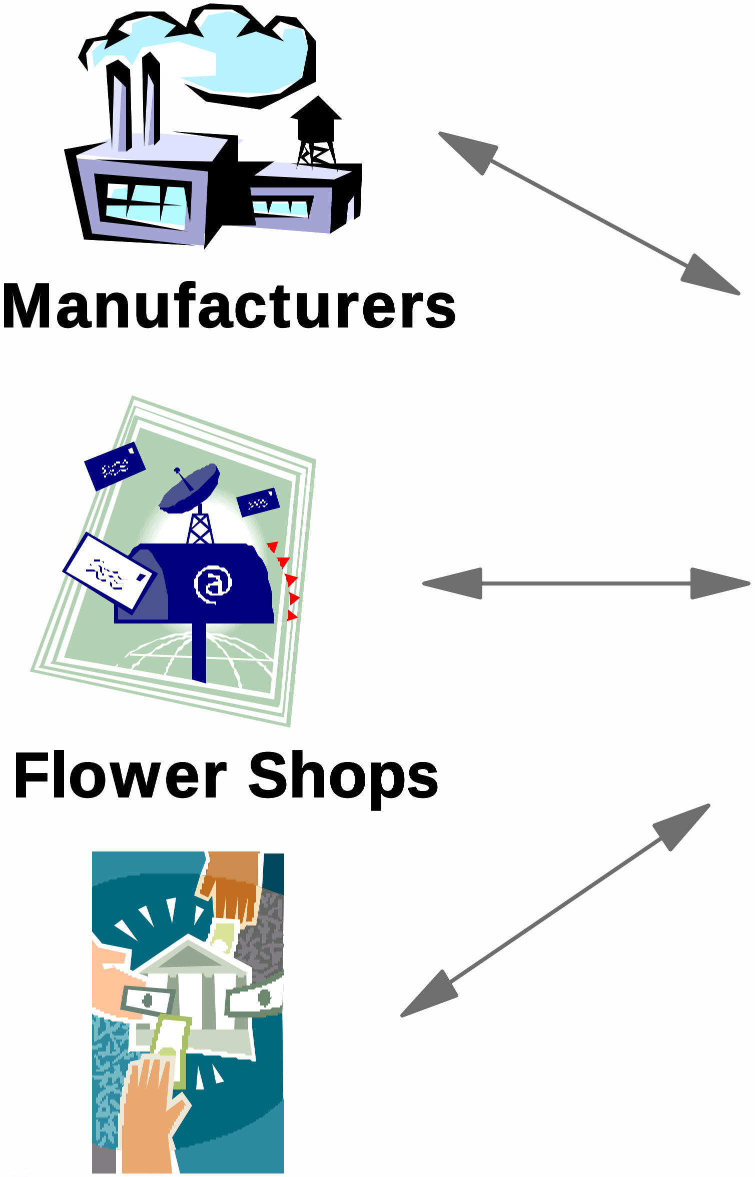
- Service Requestor
  - **FIND**s required services in the Service Registry
  - **BIND**s to services from Service Provider

- UDDI defines publish and find messages ("API")
  - the service provider defines **bind** operations according to application-specific service requirements

The UDDI Specification Roadmap

- UDDI version 1
  - Specification published 9/2000
  - Public registry sites now in production
    - Two V1 production operators: IBM & Microsoft
- UDDI version 2
  - Specification published 6/2001
  - V2 UDDI Business Registry sites now in first beta
    - Non-replicating, 2nd beta w/replication is planned
    - Now 4 V2 operators: HP, IBM, Microsoft & SAP
- UDDI version 3
  - Specifications to be published mid-2002
  - Beta registry sites – fall, 2002
  - Transition specification to industry standards body

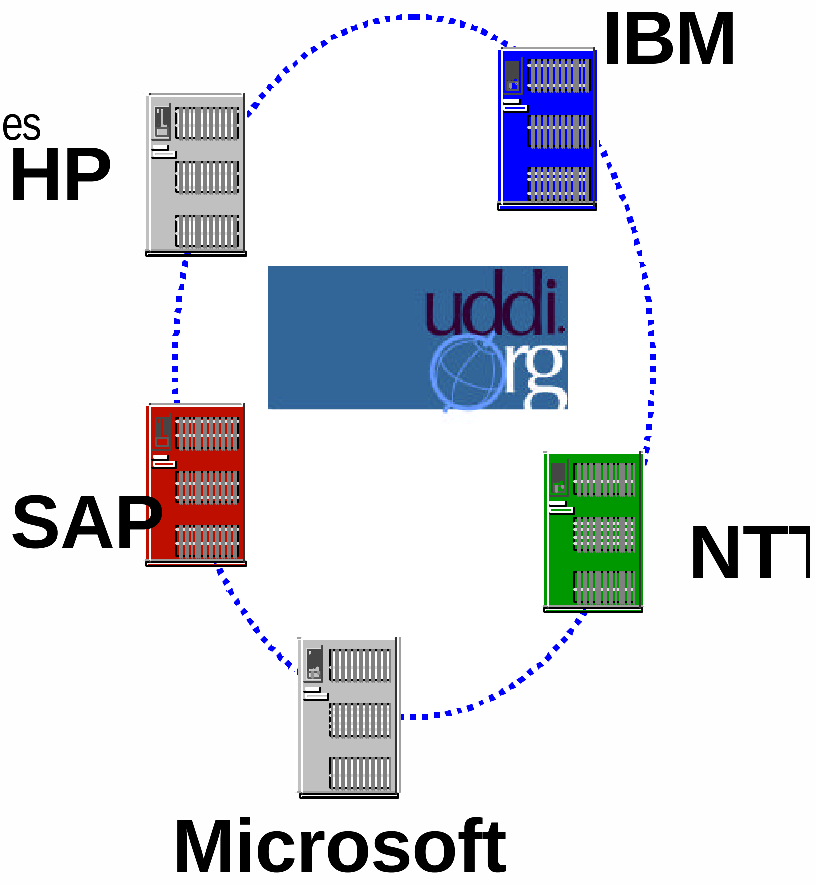
UDDI Implementation



- UDDI Business Registry
  - ►Programmatic descriptions of businesses and the services they support
  - ►Programmatic descriptions of Web service specifications
  - ►Programming model and schema
  - ►Platform & language neutral
  - ►Uses XML, HTTP, and SOAP
  - ►Free on the Internet

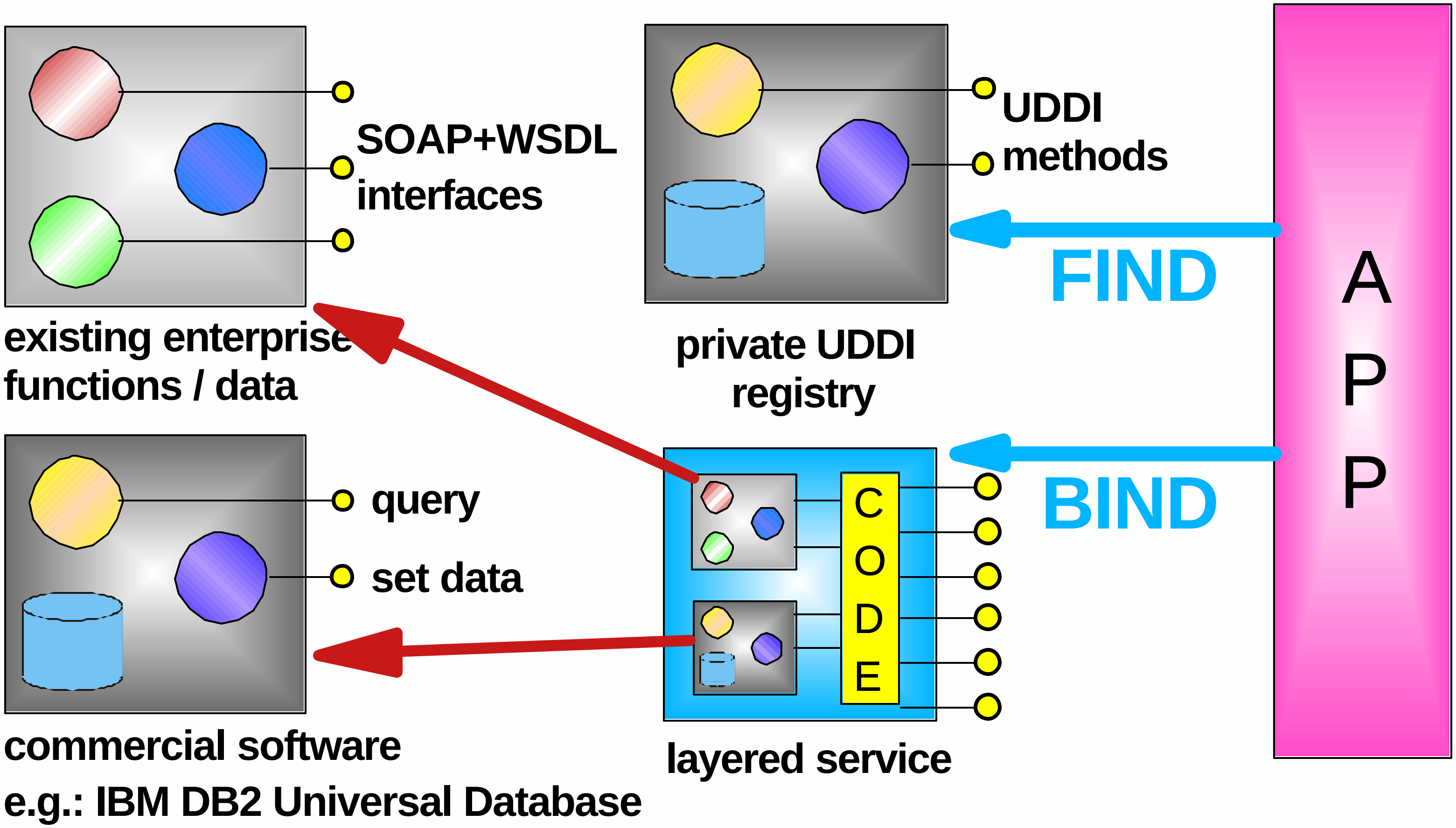
Registry Operation

- Peer nodes (websites)
  - Companies register with any node
  - Registrations replicated on a daily basis
  - Complete set of “registered” records available at all nodes
- Common set of SOAP APIs supported by all nodes
- Compliance enforced by business contract



Private UDDI: Inside the Enterprise

SOAP+WSDL+UDDI is useful for an application or data integration strategy: offers loose coupling and late binding



Service Requestor algorithm

- **Implement code** to interact with Web services which support a particular specification (standard)
- **find_business:** pass it a tModelBag referencing the tModel(s) that define the desired specification
  - Returns serviceInfos that have BindingTemplates that reference these tModel(s)
- **find_binding** using the serviceKey from one of the serviceInfos
- **get_bindingDetail** to obtain the access point
- **Invoke** the Web service using its access point or hosting redirector

- Each of these steps need code that
  - builds and sends request message, then
  - parses response message
  - processes response data

UDDI4J: an Open-source Java API

- Open-source Java bindings for UDDI messages
  - Creates SOAP messages via Java method calls with an API that maps to UDDI message elements
  - Other housekeeping chores to make your UDDI implementation work easier
  - works with any UDDI Registry
- UDDI4J source and binaries available
  - oss.software.ibm.com- IBM's open source software site
  - OSI-approved open-source licence
  - UDDI version 2 support added by HP
  - version 2 included in IBM WSTK 3.1 (ibm.com/alphaworks)
- Supported by HP and SAP
- Read Doug Tidwell's "UDDI4J: Matchmaking for Web services" to get started
  - ibm.com/developerWorks/library/ws-uddi4j.html

WebSphere Private UDDI version 1.1

- A complete implementation of a UDDI registry
  - based on IBM's UDDI Business Registry
  - underlying database: IBM DB2 Universal Database
  - supports complete UDDI version 2 message spec
- Technology preview available for free download
  - http://www7b.software.ibm.com/wsdd/downloads/UDDIregistry.html
  - FREE electronic support available
  - Requires WebSphere App Server version 4, and DB2 Universal Database version 7.2 FP5
    - these are also available for free download
    - the URL above makes all required pieces available
- Use locally (e.g. for testing and experimentation)
  - Production license available

UDDI Resources

- White papers, product offerings
  - http://www.ibm.com/webservices
- Software:
  - UDDI4J - open-source Java API to access UDDI
    - code: http://oss.software.ibm.com
  - Private UDDI preview for developers edition
  - http://www7b.software.ibm.com/wsdd/downloads/UDDIregistry.html
  - Web Services ToolKit (WSTK)
    - http://www.alphaworks.ibm.com/tech/webservicestoolkit
- Articles, tutorials: http://ibm.com/developerworks/webservices
  - Steve Graham: Role of private UDDI nodes in Web services
    - Part 1: Six species of UDDI
    - Part 2: Private nodes and operator nodes
  - Doug Tidwell: Introduction to UDDI4J
    - ibm.com/developerWorks/library/ws-uddi4j.html
-
-

Adoption

Technology Readiness?

- Evolutionary
  - typically exploits existing technology - App servers, ip-sprayers ...
- Base technologies ready
  - XML, SOAP, WSDL, tools, Java (and other) APIs,deployment platforms
- Some areas still evolving
  - UDDI - private registries are available, they are relatively new
  - Security - HTTPS sufficient is some cases, finer granularity may need hand-rolled code
  - Metering, provisioning ...

Proof of Concept

- Objectives
  - Education and skills development
  - Technology validation
  - Tool selection
- Exercise technologies
  - XML, SOAP, WSDL, UDDI
  - security
  - reliability
- Intra-organisation

Pilot Project

- Objectives
  - Real-world experience
  - Exercise development process
  - Understand deployment platform and procedures
  - Technology validation
- Keep to core technologies
  - Scenarios not requiring UDDI
  - Wrap an existing service
- Prefer intra-organisation
- Consider evolution to a 2nd generation service

Essentials

- Tooling
  - Creation of WSDL, services and proxies all must be automated
- Establish Best Practices
  - Ease of problem determination - log and trace points
  - Testing
  - Data types
- Focus on process and non-functional requirements

Technology Futures

IBM's Web Services Strategy

- Support and help lead open standards efforts in the W3C, OASIS, ebXML, and elsewhere
- Get code to developers quickly via alphaWorks and support open source efforts in Apache and elsewhere
- Add product support in WebSphere, MQ Series, DB2, Lotus Notes, etc
- Build customer solutions

IBM and Web Services Standards

- IBM is leading the industry in developing the specifications needed to create Web Services standards
  - worked with Microsoft to make SOAP vendor-neutral
  - created first platform-neutral SOAP 1.1 implementation
    - then gave it away to Apache Software Foundation
  - worked with Microsoft to develop WSDL
  - worked with Microsoft and Ariba to create UDDI.org
  - created WSFL for composition and workflow coordination of web services
  - proposed changes to HTTP to provide a reliable protocol as an underpinning to SOAP messaging
  - co-founder of WS-I.org (interoperability)
- Note: there are not yet any finished **standards**...
  - SOAP and WSDL have been **proposed** to the W3C

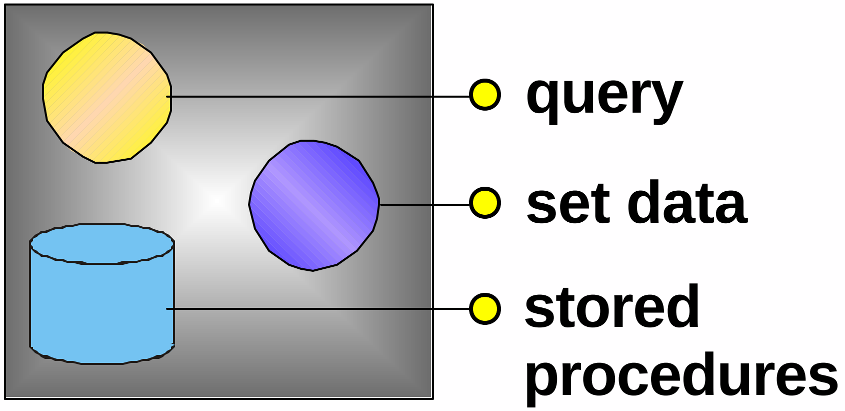
Web Services in
WebSphere Application Server 4.0

- WAS4 is the industry's premier production-ready Web app server for deploying Web Services solutions for dynamic e-business

- Integrated support for Web Services
  - SOAP - Simple Object Access Protocol
  - UDDI - Universal Description, Discovery, Integration
  - WSDL - Web Services Description Language
  - enables powerful interoperability between Web Services and J2EE applications
- Security:
  - HTTPS support
  - Implementations of XML Signature and Encryption

DB2 Universal Database / XML Extender

- XML Extender current version:
  - XML import and shred across tables
  - XML export and recomposition
  - XPath to SQL query transformation
  - flexible configuration using XML-syntax Document Access Definition
- Web Services features coming in next version:
  - query and set data functions available via web services
  - database management functions via web services
  - stored procedures accessible thru web services



Web Services in other IBM Products

- WebSphere MQ Series
  - technology preview of SOAP in WSTK 2.3
  - integrated into next release of the product
- Lotus:
  - enable Domino Services as Web Services
  - incorporate SOAP interfaces, XML-based messaging
  - other products will explore web services features for collaborative products like instant messaging
- Tivoli:
  - management and security for Web Services deployment

Extensions

- WebServices Invocation Framework
  - Dynamic invocation of WebServices
  - Provides a standard API for invoking services described in WSDL, no matter how/where services are provided
  - Allows new bindings to be added at run time
- WebServices Gateway
  - Runtime configurable proxy for existing WebServices
  - Exploits WSIF
  - One purpose: controlled internet presentation of intranet services
- Available now on alphaworks

WSFL: Web Services Flow Language

An XML language to describe Web Services compositions. Two types:

1. Usage pattern of a collection of Web Services
   - describes how to achieve a particular business goal as a business process
   - flow composition, orchestration, or choreography
   - defines the flow of control and data
2. Interaction pattern of a collection of Web Services
   - describes the overall partner interactions
   - no specification of an execution sequence is provided

- WSFL has extensive support for the recursive composition of services
  - support for top-down progressive refinement design
  - support for bottom-up aggregation

WSFL Resources

Specification: ibm.com/software/solutions/webservices/pdf/WSFL.pdf

Introductory articles: visit ibm.com/developerworks/webservices, search for "Snell" for four articles by James Snell ("Web Services Insider" series):

- Introducing WSFL
- Business process modeling with WSFL
- Implementing roles in WSFL
- WSFL and recursive composition

HTTPR

- Reliable RPC over HTTP
- Uses persistence and correlation ids to ensure delivery
- Submitted for Certification to IETF

Resources:

- ibm.com/developerworks/webservices/library/ws-phtt/?dwzone=webservices

Web Services Interoperability

- WS-I.org announced Feb 6, 2002
- Industry initiative for Web services
  - Open to any organization committed to Web services
  - Promote and accelerate adoption, deployment
- Focused on promoting Web service interoperability
  - Across platforms, applications, and programming languages
  - Promote a common, clear definition for Web services
- Promote customer adoption & deployment
  - Integrate specifications from standards bodies
  - Implementation guidance & tools for customers building and deploying Web services

ws-i.org deliverables:

- Profiles
    - named groups of specifications at given version levels with conventions about how they work together
- Implementation Scenarios
    - based on customer requirements
- Test suites and supporting materials
    - Sample solutions
    - Implementation aids
    - Conformance testing tools
    - Supporting documentation and white papers

Free!
Tools and Resources for Web Services

IBM alphaWorks

http://ibm.com/alphaWorks

- Hundreds of tools for Web Services, XML, Java
  - early versions of features that may be in products
  - some are solid production-code (XML4J, LotusXSL)
  - some are experimental, prototypes
  - free download and use
- Some recent Web Services downloads:
  - Web Services Toolkit 3.0 and demos
  - Web Services Hosting Technology
  - Web Services Process Management Toolkit
  - Web Services Invocation Framework
  - Web Services Gateway
  - WSDL Toolkit

IBM's Web Services Toolkit version 3.01

- An implementation of the Web Services architecture for creating, locating and invoking web services
  - Utility business services
  - HTTPR Demo
  - Integrated W3C Digital Signatures and Encryption
  - XKMS prototype
  - WSDL Toolkit
  - IBM MQSeries transport for SOAP (technology preview)
  - UDDI4J -- support for UDDI version 2
  - UDDI4B (UDDI for Browser plugin)
  - COM object support
  - Lotus Domino enablement kit
- Needs only a JDK to run
- Versions available for Windows and Linux
- Available for free download from http://ibm.com/alphaworks

UDDI4J: an Open-source Java API

- Open-source Java bindings for UDDI messages
  - Creates SOAP messages via Java method calls with same "API" as UDDI messages
  - Other housekeeping chores to make your UDDI implementation work easier
- Read Doug Tidwell's "UDDI4J: Matchmaking for Web services" to get started
  - ibm.com/developerWorks/library/ws-uddi4j.html
- UDDI4J source and binaries available
  - oss.software.ibm.com
    (IBM's open source software site)
  - OSI-approved open-source licence
  - originally part of the WSTK; now available on our open source site
  - now supports UDDI version 2 API

WebSphere Business Components

- A set of tools and Enterprise JavaBean components that improve developer productivity, speed product cycles, and simplify the process of building solutions
- Integrates with IBM Web Services offerings, enabling you to create robust enterprise solutions which also offer services on the Internet
- Invoke Web services faster
  - Ready Java technology-based components to jumpstart development
  - A natural stepping stone to Web services technology
  - Use of open standards and complementary architectures
  - Offer solutions both inside and outside the company
- More info:
  - Product site: ibm.com/software/components
  - White paper: ibm.com/software/components/library.html
  - Demo: www7b.boulder.ibm.com/wsdd/zones/wsbc/wsbcdemo.html

WSDL4J: an Open-source Java API

- Allows the creation, representation, and manipulation of WSDL documents describing services
- Service descriptions can be treated by a client in a uniform manner, regardless of the origin of the description:
  - parsing a WSDL document
  - constructed programmatically by direct invocation of the APIs
  - built using information provided by a user via a command-line or graphical interface
  - built using information retrieved from a network source
- Reference implementation of JSR110
- Visit:
  - oss.software.ibm.com/developerworks/projects/wsdl4j

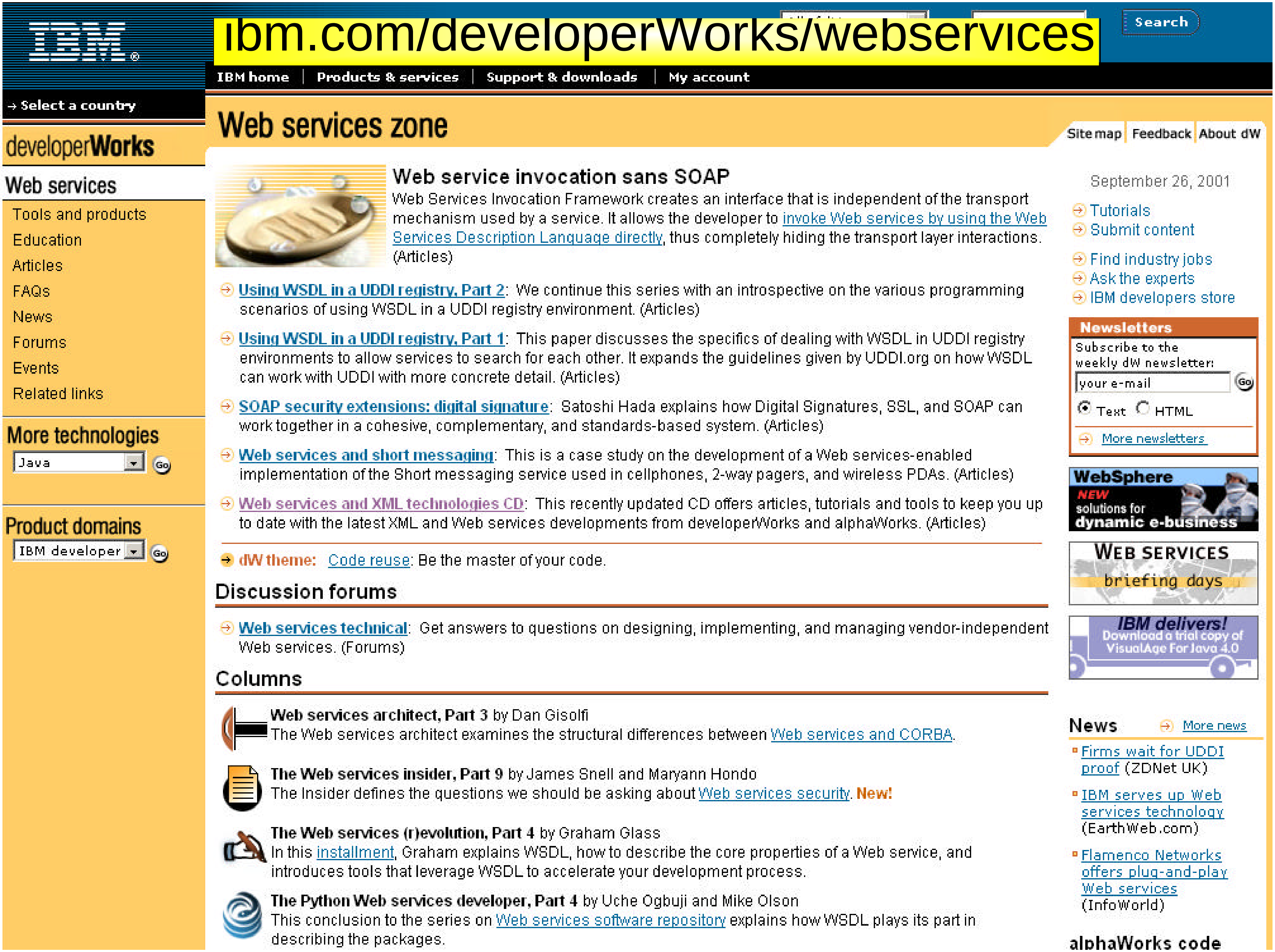
XML Security Suite

- Complete implementation of W3C specifications:
  - Digital Security 1.0 proposed recommendation
  - Encryption 1.0 working draft
  - Experimental access control implementation
- Integrated into Websphere App Server 4.0
- Included in Web Services Toolkit

Schema support in XML Parsers

- Apache Xerces/J version 3.4
  - Open Source from xml.apache.org, free redistribution
  - Developed by IBM and others in the open source community
  - The same code we use in Websphere Application Server and other IBM products
- Complete implementation of XML Schema 1.0 recommendation from W3C, beta release
  - a few documented limitations (reasonably obscure)
  - a few known bugs
  - these will be addressed in the next release (schema support is gold)

ibm.com/developerWorks/webservices


Search
IBM home | Products & services | Support & downloads | My account
→ Select a country
developerWorks

Web services zone

Site map | Feedback | About dW

Web services

- Tools and products
- Education
- Articles
- FAQs
- News
- Forums
- Events
- Related links

More technologies

Product domains

Web service invocation sans SOAP

Web Services Invocation Framework creates an interface that is independent of the transport mechanism used by a service. It allows the developer to invoke Web services by using the Web Services Description Language directly, thus completely hiding the transport layer interactions. (Articles)

- Using WSDL in a UDDI registry, Part 2: We continue this series with an introspective on the various programming scenarios of using WSDL in a UDDI registry environment. (Articles)
- Using WSDL in a UDDI registry, Part 1: This paper discusses the specifics of dealing with WSDL in UDDI registry environments to allow services to search for each other. It expands the guidelines given by UDDI.org on how WSDL can work with UDDI with more concrete detail. (Articles)
- SOAP security extensions: digital signature: Satoshi Hada explains how Digital Signatures, SSL, and SOAP can work together in a cohesive, complementary, and standards-based system. (Articles)
- Web services and short messaging: This is a case study on the development of a Web services-enabled implementation of the Short messaging service used in cellphones, 2-way pagers, and wireless PDAs. (Articles)
- Web services and XML technologies CD: This recently updated CD offers articles, tutorials and tools to keep you up to date with the latest XML and Web services developments from developerWorks and alphaWorks. (Articles)

- dW theme: Code reuse: Be the master of your code.

Discussion forums

- Web services technical: Get answers to questions on designing, implementing, and managing vendor-independent Web services. (Forums)

Columns

Web services architect, Part 3 by Dan Gisolfi
The Web services architect examines the structural differences between Web services and CORBA.

The Web services insider, Part 9 by James Snell and Maryann Hondo
The Insider defines the questions we should be asking about Web services security. **New!**

The Web services (r)evolution, Part 4 by Graham Glass
In this installment, Graham explains WSDL, how to describe the core properties of a Web service, and introduces tools that leverage WSDL to accelerate your development process.

The Python Web services developer, Part 4 by Uche Ogbuji and Mike Olson
This conclusion to the series on Web services software repository explains how WSDL plays its part in describing the packages.

September 26, 2001

- Tutorials
- Submit content
- Find industry jobs
- Ask the experts
- IBM developers store

Newsletters

Subscribe to the weekly dW newsletter:

- More newsletters

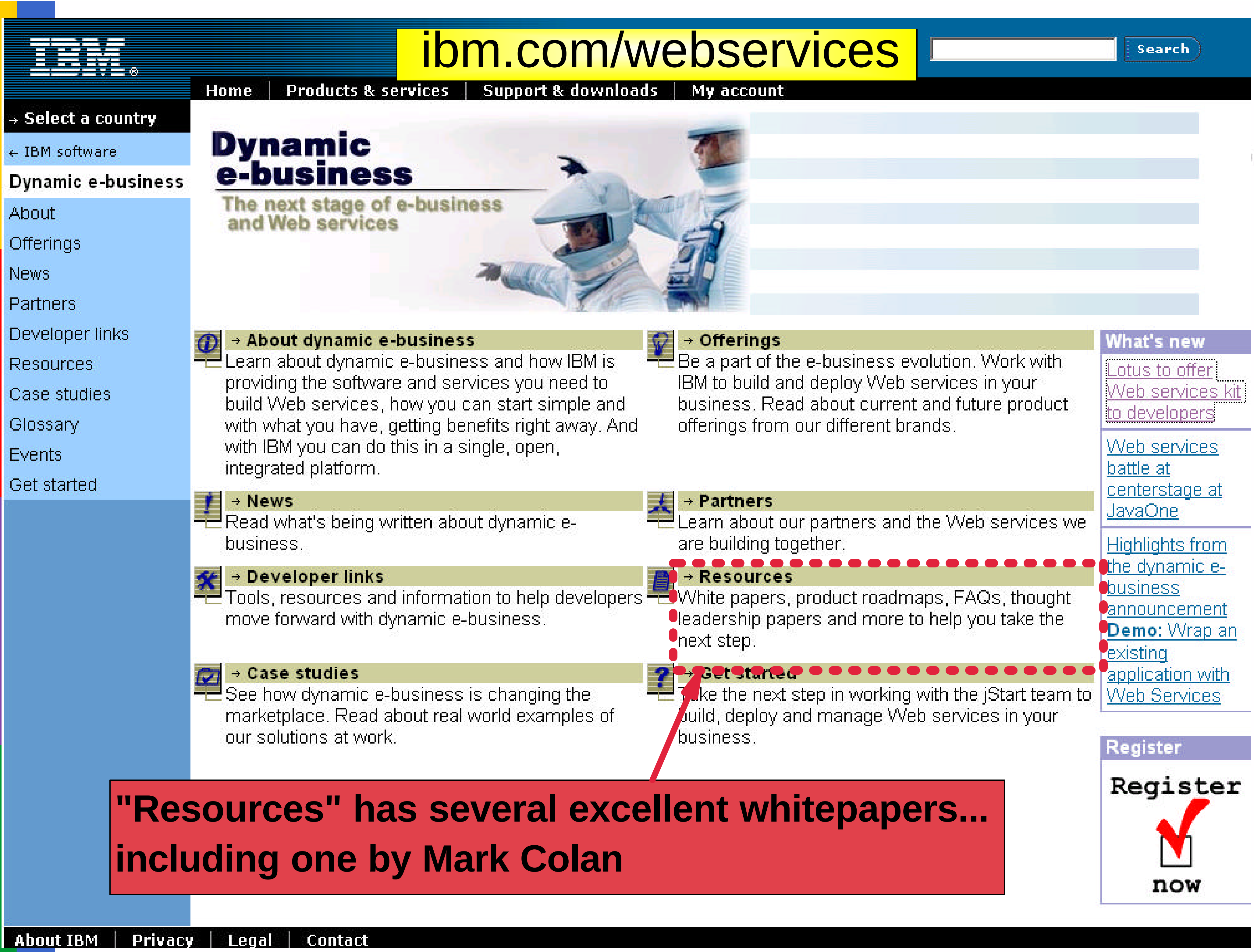
WebSphere
NEW solutions for dynamic e-business
WEB SERVICES briefing days
IBM delivers!
Download a trial copy of VisualAge For Java 4.0

News More news

- Firms wait for UDDI proof (ZDNet UK)
- IBM serves up Web services technology (EarthWeb.com)
- Flamenco Networks offers plug-and-play Web services (InfoWorld)

alphaWorks code

ibm.com/webservices



→ About dynamic e-business
Learn about dynamic e-business and how IBM is providing the software and services you need to build Web services, how you can start simple and with what you have, getting benefits right away. And with IBM you can do this in a single, open, integrated platform.

→ Offerings
Be a part of the e-business evolution. Work with IBM to build and deploy Web services in your business. Read about current and future product offerings from our different brands.

→ News
Read what's being written about dynamic e-business.

→ Partners
Learn about our partners and the Web services we are building together.

→ Developer links
Tools, resources and information to help developers move forward with dynamic e-business.

→ Resources
White papers, product roadmaps, FAQs, thought leadership papers and more to help you take the next step.

→ Case studies
See how dynamic e-business is changing the marketplace. Read about real world examples of our solutions at work.

→ Get started
Take the next step in working with the jStart team to build, deploy and manage Web services in your business.

What's new

Lotus to offer Web services kit to developers

Web services battle at centerstage at JavaOne

Highlights from the dynamic e-business announcement

Demo: Wrap an existing application with Web Services

Register

Register now

"Resources" has several excellent whitepapers... including one by Mark Colan

Information sources

ibm.com/webservices

whitepapers on IBM's vision of dynamic e-business enabled by web services

ibm.com/developerWorks/webservices

Web Services Zone on developerWorks - resources for customers and developers on the use of XML

ibm.com/alphaworks

site for free emerging tools and technologies from IBM

oss.software.ibm.com

UDDI4J, WSDL4J open source Java class libraries

xml.apache.org

Apache SOAP and other open source XML tools

uddi.org

UDDI consortium - information and resources

xml.org

XML standard vocabularies repository

ebxml.org

electronic business in XML initiative

- PDFs of this and other presentations are available at this site

ibm.com/developerworks/speakers/colan

UDDI.org Specifications

- Programmer's API
  - Inquiry & publication APIs for discovery and publishing
  - Request/response structure semantics, error handling
  - XML & SOAP details
  - Schemas for all APIs
- Data structures
  - Describes the data structures used in the API document
- Operators
  - Information on operating a UDDI node
- Replication
  - Details on how to replicate data between nodes

More UDDI.org papers

- White Papers:
  - Executive White Paper
  - Technical White Paper
- Best Practices papers:
  - Using WSDL in a UDDI Registry
  - Providing a Taxonomy for use in UDDI version 2.00

Current Production Operators

- IBM UDDI Business Registry V1 Sites:
  - Official Registry (official business only please):
    - http://www.ibm.com/services/uddi
  - User Test Registry (sandbox):
    - http://www.ibm.com/services/uddi/testregistry
- Microsoft UDDI Business Registry V1 Sites:
  - Official Registry (official business only please):
    - http://uddi.microsoft.com
  - User Test Registry (sandbox):
    - http://test.uddi.microsoft.com
- Each supports both SOAP and Web Page access
-

Current V2 Beta Operator Sites

- Hewlett Packard:
  - http://uddi.hp.com
- IBM:
  - http://www.ibm.com/services/uddi/v2beta/registry.html
- Microsoft:
  - https://uddi.rte.microsoft.com/register.aspx
- SAP:
  - http://udditest.sap.com
-