



# History

---

- Structured programming
- Object-oriented programming
- Distributed computing
- Electronic data interchange
- World Wide Web
- Web services



# Who Was First?

---

- What company first proposed the web services concept?
  - Hewlett-Packard's e-Speak in 1999
  - was an enabler for e-services
  - Microsoft introduced the name "web services" in June 2000
  - MS "bet the company" on its web services strategy
  - now every major vendor is a player



# Open, Standard Technologies

---

- XML – tagging data such that it can be exchanged between applications and platforms
- SOAP – messaging protocol for transporting information and instructions between applications (uses XML)



# Open, Standard Technologies

---

- WSDL – a standard method of describing web services and their specific capabilities (XML)
- UDDI – defines XML-based rules for building directories in which companies advertise themselves and their web services



# Advantages

---

- Open, text-based standards
- Modular approach
- Inexpensive to implement (relatively)
- Reduce the cost of enterprise application integration
- Incremental implementation



# Real Web Services

---

- UC-Berkeley
  - Unified Communications Technical Project
  - unify email, voice, and fax into in-boxes accessible from cell phones, PDAs, or e-mail clients
- Eastman Chemical Company
  - distributors access chemical catalog in real-time and push info to customers



# Real Web Services

---

- Accenture
  - Live Information Models
  - stock traders access real-time information from a single terminal
- Dollar Rent-a-Car + Southwest Airlines
  - Southwest runs Unix
  - Dollar runs MS Windows
  - Dollar turned its system into a web service



# More Examples

---

- Web service broker sites
  - [www.xmethods.net](http://www.xmethods.net)
  - [www.salcentral.com](http://www.salcentral.com)





# Online Resources

---

- [www.deitel.com](http://www.deitel.com)
  - web services, C#
- [www.w3.org](http://www.w3.org)
  - World Wide Web Consortium (W3C)
  - recommendations, news, mission, FAQs
- [www.w3.org/History.html](http://www.w3.org/History.html)
  - history of computing and internet from 1945 to now



# Online Resources

---

- [www.webservices.org](http://www.webservices.org)
  - news, standards, vendors, platforms, products, applications, case studies, security mechanisms
- [www.webservicesarchitect.com](http://www.webservicesarchitect.com)
  - online journal for web service developers; tools, vendors, business models, additional resources



# Online Resources

---

- [www.ws-i.org](http://www.ws-i.org)
  - web service interoperability organization (WS-I); promote interoperability among services created in different languages and platforms; white papers, news, FAQs

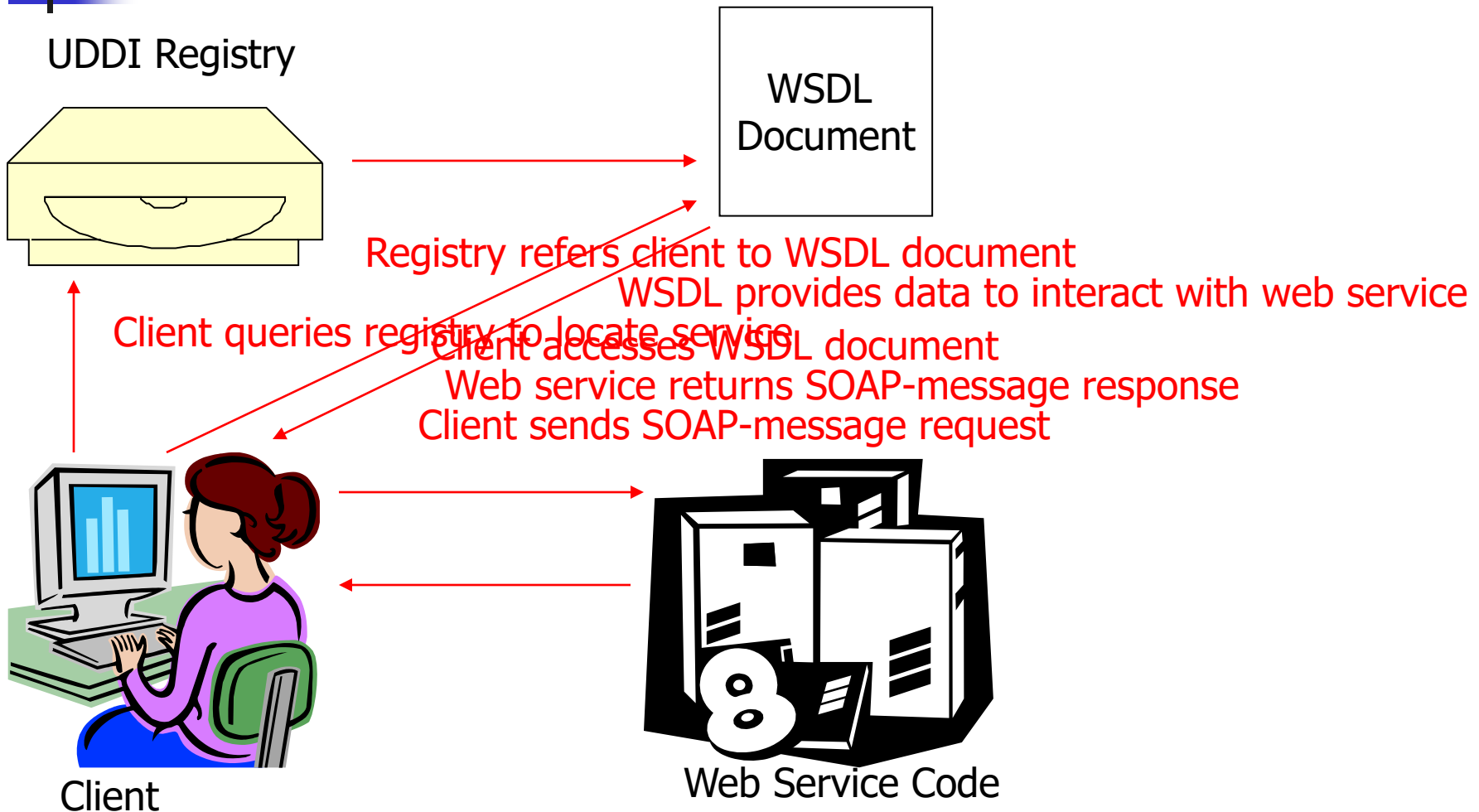


# Online Resources

---

- [www.xml.com/lpt/a/2002/02/06/webservices.html](http://www.xml.com/lpt/a/2002/02/06/webservices.html)
  - "Web Service Pitfalls": limitations, unresolved security issues
- [www.webservices.org/print.php?sid=201](http://www.webservices.org/print.php?sid=201)
  - "Web Services—A Reality Check": transactions, security, QoS

# The Big Picture





# XML

---

- Developed from Standard Generalized Markup Method (SGML)
- XML widely supported by W3C
- Essential characteristic is the separation of content from presentation
- XML describes only data
- Any application that understands XML can exchange data



# XML

---

- XML parser checks syntax
- If syntax is good the document is *well-formed*
- XML document can optionally reference a *Document Type Definition (DTD)*, also called a *Schema*
- If an XML document adheres to the structure of the schema it is *valid*



# SOAP

---

- SOAP enables between distributed systems
- SOAP message has three parts
  - *envelope* – wraps entire message and contains header and body
  - *header* – optional element with additional info such as security or routing
  - *body* – application-specific data being communicated





# WSDL

---

- Web services are self-describing
- Description is written in WSDL, an XML-based language through which a web service conveys to applications the methods that the service provides and how those methods are accessed
- WSDL is meant to be read by applications (not humans)



# UDDI

---

- UDDI defines an XML-based format that describes electronic capabilities and business processes
- Entries are stored in a UDDI registry
- UDDI Business Registry (UBR)
  - "white pages" – contact info, description
  - "yellow pages" – classification info, details
  - "green pages" – technical data
  - [uddi.microsoft.com](http://uddi.microsoft.com)

The logo graphic consists of a yellow square, a red square, and a blue square, each with a black crosshair-like pattern overlaid on it. The word "OASIS" is written in a blue, sans-serif font to the right of the graphic.

# OASIS

---

- Not competition to W3C
- Ensure that businesses acquire e-business tools that meet their needs
- United Nations Centre for Trade Facilitation and Electronic Business produced Electronic Business XML (ebXML)



# More Info

---

- [www.w3.org/2002/ws](http://www.w3.org/2002/ws)
  - web services activity
- [www.uddi.org](http://www.uddi.org)
  - explanation; business benefits
- [www.oasis-open.org](http://www.oasis-open.org)
  - technical work and standards
- [www.ebxml.org](http://www.ebxml.org)
  - technology and business benefits



# Web Services Security Architecture

---

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-Security

---

Describes how to attach signature and encryption headers to SOAP messages; how to attach security tokens such as X.509 certificates and Kerberos tickets

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-Policy

---

Describes the capabilities and constraints of the security and business policies on intermediaries and endpoints

WS-Secure  
Conversation

WS-Federation

WS-Authorization

**WS-Policy**

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-Trust

---

Framework for trust models that enables web services to interoperate securely.

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP





# WS-Privacy

---

Model for how web services and requesters state privacy preferences and organizational privacy practice statements

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-SecureConversation

---

Manage and authenticate message exchanges between parties, including security context exchange and establishing and deriving session keys

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-Federation

---

Manage and broker trust relationships in a heterogeneous federated environment, including support for federated identities.

WS-Secure  
Conversation

**WS-Federation**

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP



# WS-Authorization

---

Manage authorization data and authorization policy.

WS-Secure  
Conversation

WS-Federation

WS-Authorization

WS-Policy

WS-Trust

WS-Privacy

WS-Security

SOAP