Introducing Web Services

• [ACKM04] Ch 5, 6

Definition: Web Services

• A standardized way of integrating Web-based applications, using
  • XML to tag data
  • SOAP to transport data
  • Simple Object Access Protocol
  • WSDL to describe available services
  • Web Services Description Language
  • UDDI to list available services
  • Universal Description, Discovery and Integration

Not Web Services

• Generic client (browser)

• RPC or messaging protocol between client and web server is not the focus here
Web Services

But it makes more sense here

Manual Integration Between Companies

B2B interactions occur by accessing Web pages, filling Web forms, or via email.

In Principle - Middleware a Service

The combination of message broker and adapters enables interoperability.
Why not?

• Autonomy?
• Trust?
• Confidentiality?

How About Point-to-Point?

• Note middleware must interoperate

Middleware Explosion ...

...
Goals for B2B Integration

- Service-Oriented Architecture
- Redesign of middleware protocols
- Standardization

Service-Oriented

- Functionality always exposed as services
- Loosely coupled
- Invoked by programs as well as users

Middleware Redesign

- Peer-to-Peer
- Between trust domains
- Compatibility with Internet
Standardization

- In praise of motherhood...
- BUT - Web standards seem to have achieved much more traction than previous standards

Web Services Integrating Between Companies

interactions based on protocols redesigned for peer to peer and B2B settings
interactions based on protocols standardized, eliminating need for many different middleware infrastructures (need only the Web services middleware)

A Sophisticated Wrapper - Expose Apps on Web

Company A (provider) Company B (client)
Can Use WS Protocols Internally

Company A (or a LAN within Company A)

Web service-enabled broker

SmartQuotation
DBMS applications
SmartForecasting
XYZ

assumes all back-end systems are accessible as Web services

Two Facets of Web Services Architecture

- Internal
  - run conventional apps
  - expose them as web services
- External
  - global services (like DNS)

Internal Architecture

Company A (service provider)

Web service interface
access to internal systems

Web services middleware (internal)

Conventional middleware (includes middleware services)

clients from other companies

Conventional middleware provides lots of services (load balancing, transaction support, etc). Current Web services middleware is quite poor in this respect.
External Architecture

Company A (service requester)
- Web service client
- Web services middleware (internal)
- other tiers

Company B (service provider)
- Web service
- Web services middleware (internal)
- other tiers

Company C (directory service provider)
- service descriptions

1. publish the service description
2. find
3. interact

the abstraction and infrastructure provided by the registry are part of the external middleware

External Architecture

- Is service discovery the only component of Web Services middleware?
- No but external middleware needs to run in peer-to-peer fashion with minimal trust requirements ...

External Architecture With Middleware

Company A (service requester)
- Web service client
- internal middleware
- transaction engine
- other protocol infrastructure
- composition engine
- other tiers

Company B (service provider)
- Web service
- internal middleware
- transaction engine
- other protocol infrastructure
- composition engine
- other tiers

Company C (directory service provider)
- service descriptions
Basic Web Services

- Minimalist definition/requirements:
  - communicate (SOAP)
  - describe services - IDL (WSDL)
  - directory service (UDDI)

Minimalist Infrastructure ...

Using WSDL Specification

Note all WSDL "processing" happens at development time.
Using UDDI Registry

Service descriptions

SOAP-based middleware

Service provider

Application object

Consumer

SOAP messages

To look for services

Service descriptions

UDDI registry

Service requestor

Application object

Consumer

SOAP messages

To publish service description

Simple Object Access Protocol

- Specifies:
  - Message format for one-way communications
  - Specification for SOAP RPC
  - Rules for processing SOAP messages
  - Rules for transport - HTTP and SMTP

A SOAP Message

SOAP envelope

SOAP header

Header block

SOAP body

Body block
Document vs RPC

(a) Document-style interaction

(b) RPC-style interaction

Intermediate Processing - Roles

- Header blocks only ...
- none: no node processes this block
- next: every node processes the block
- ultimateReceiver: only last node in path

Intermediate Processing - Roles

- Note "next" role in header
SOAP over HTTP

Simple Implementation

WSDL Specifications

- Abstract part
- signatures, operations
- like traditional IDL
- Concrete part
- bindings / services / ports
Concrete Part

- Interface Bindings
- Message encoding & protocol bindings
- Ports
- Interface binding + network address
- Services
- Logical groupings of ports

A WSDL Service Specification

Automatic Generation of WSDL
Universal Description Discovery and Integration

- The Web Services Directory Service
- Service Registry
- for browsing by developers
- for dynamic binding
- Business registry
- original goal: global registry, every business and exported service registered there
- now: support interaction between private and public UDDI registries

Information in UDDI

- Business entity
- an organization that provides Web services
- Business Service
- group of Web Services that cooperate in performing some business process
- bindingTemplate
- technical information needed to use the service
- tModel
- general container for a specification

Information in UDDI
tModel Description

- Note overviewDoc has URL - refers to specification stored outside UDDI server
- Multiple overviewDocs are allowed

Sharing of tModels

- Each tModel is assigned a GUID when it is registered
- The tModels may be shared by multiple bindingTemplates
- Software can use UDDI to search for all instances of a given GUID

UDDI Registry API

- Inquiry
- search
- Publisher
- register and unregister operations
UDDI Registry API - II

- Security
- Custody and Ownership Transfer
- create/destroy/move objects
- Subscription
- monitor for changes in UDDI data
- Replication
- readonly except at "owner" site

Architecture of Distributed UDDI

WSDL in UDDI
Private UDDI Registries

- Public Universal Business Registry not required for Web apps that are intra-enterprise or between trusted partners
- Three categories
  - public
  - private
  - shared
- Must support replication between registries where appropriate

“Advances” in SOAP

- BLOBs
  - now Web Services description yields interaction sequence but not data formats, which are agreed to by communicating applications
- Document exchange
- Constructing WSDL requires semantic understanding of the documents

Dynamic Binding

- Fully general
  - e.g. CORBA DII
- Now we have run-time types, introspection
- Restricted
  - e.g. all services using a given tModel
- But enterprises usually won’t accept a program effectively signing a contract...
Grid Computing

- Sharing supercomputer cycles and huge volumes of data in scientific computing
- Open Grid Services Architecture effort
- IBM, Microsoft, Sun, Oracle, ...
- Extensions to Web Services standard