1.264 Lecture 23

Web services, concluded

Next class: No readings. Exercise due before class (case study on course Website)

Web Services Description Language: WSDL

- WSDL 1.1 (current version) contains:
  - Description of XML data (types) that can be passed in <types>
  - List of messages in <message>
    - Request and response messages
  - Direction and pairing of message passing in <portType>:
    - Request-only, request-response, response-only
  - Message encoding in <binding> element (literal, etc.)
  - Location where service is offered in <service> element
- eBay example in download
  - eBay deploys 300 features per quarter and 600,000 lines of code per quarter
- WSDL 2.0 in limited use

WSDL document

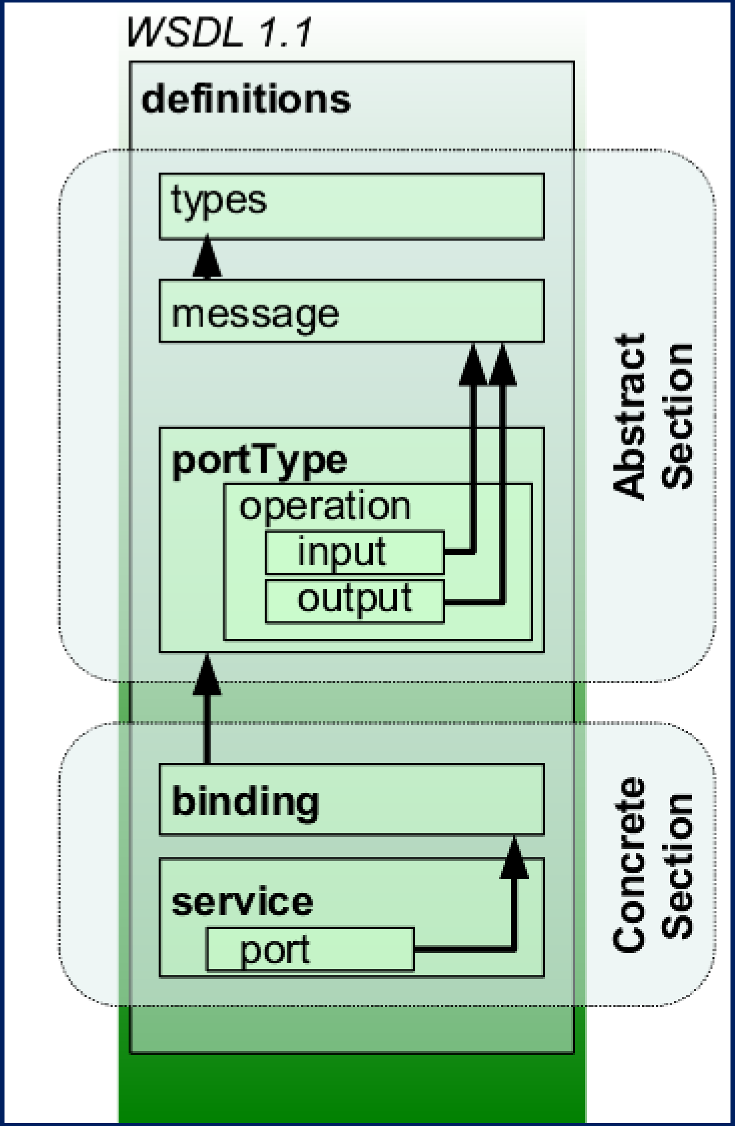


Image in public domain. Source: Wikimedia Commons

- type: rate: int
  - Complex types defined w/ XSD
- message: RateRequest()
  - Requests, responses
- portType: matches request/response
  - input message: RateRequest(),
  - output message: RateResponse()
- binding:
  - operation: request-response
- service: TruckService
  - address: http://www.aaa.com/

Web service discovery

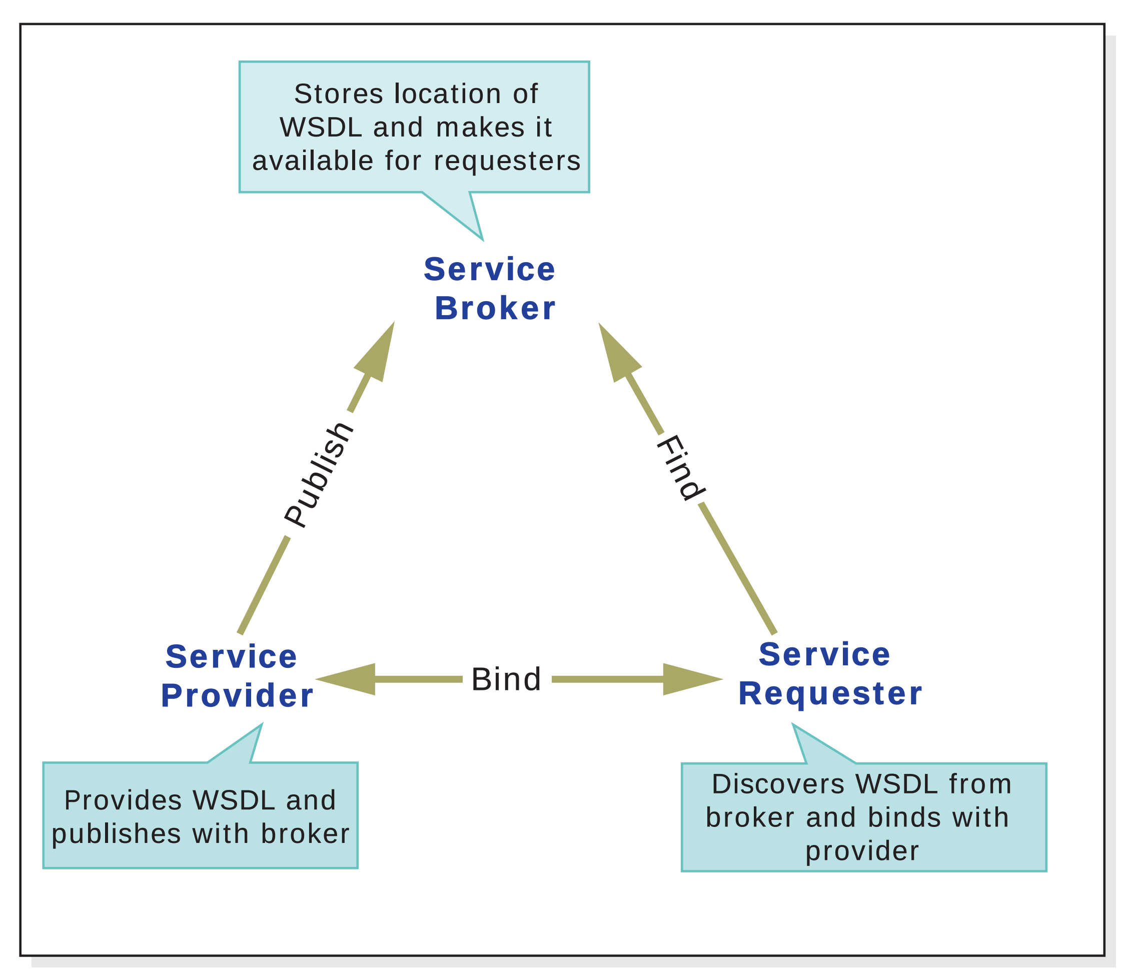
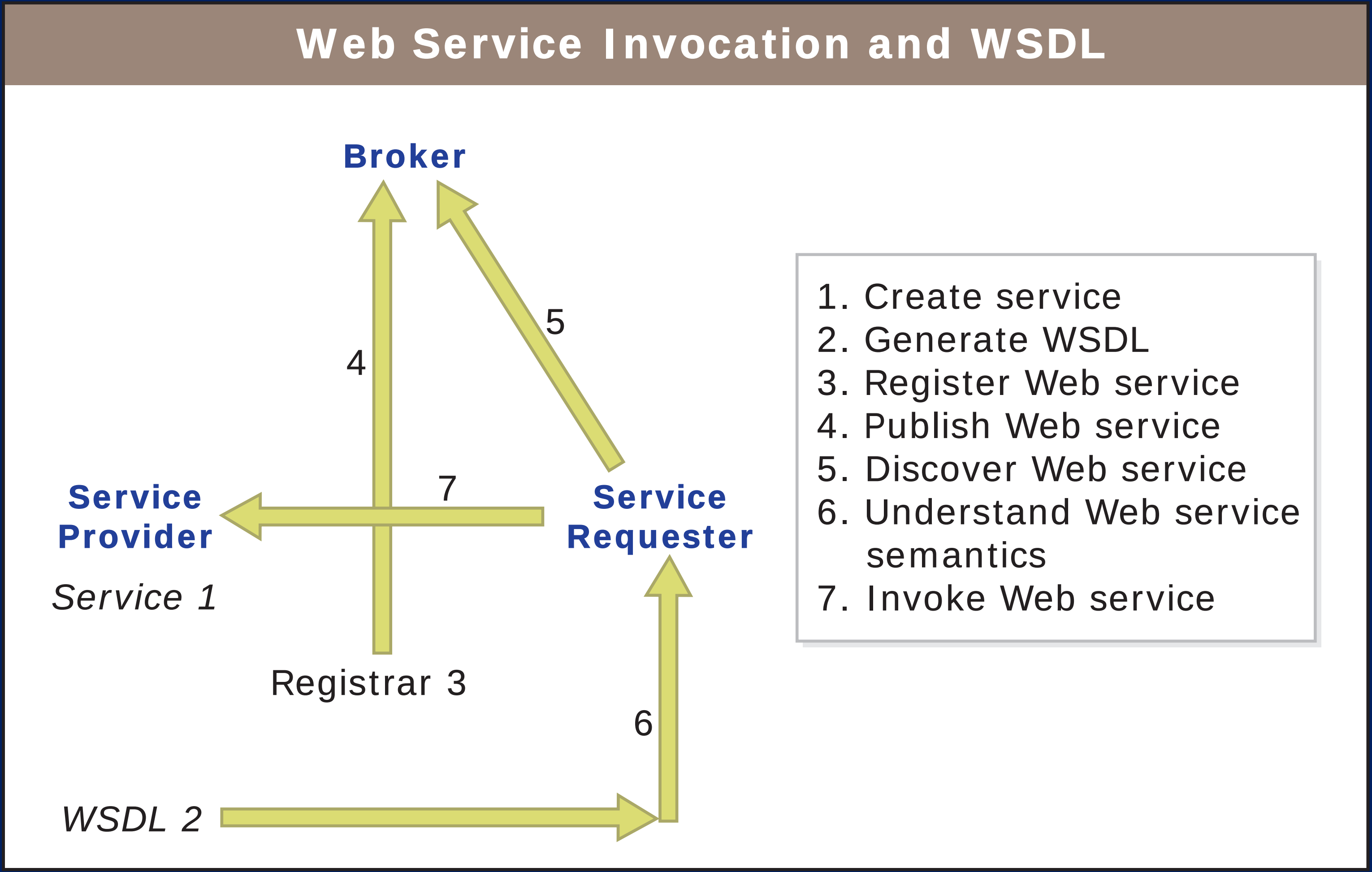

Stores location of WSDL and makes it available for requesters
Service Broker
Publish
Find
Service Provider
Bind
Service Requester
Provides WSDL and publishes with broker
Discovers WSDL from broker and binds with provider

Image by MIT OpenCourseWare.

WSDL discovery



Web Service Invocation and WSDL

1. Create service
2. Generate WSDL
3. Register Web service
4. Publish Web service
5. Discover Web service
6. Understand Web service semantics
7. Invoke Web service

Image by MIT OpenCourseWare.

Exercise: Registering a Web service on a Web server

- Use Web site from lecture 22
- Set up the Web service on the server, as follows:
- Right click Web site name in Solution Explorer
  - Select Add Service Reference
  - Click ‘Discover’
  - Click link to ProductService Web service
  - Use default namespace (ServiceReference)
  - Click OK
- Web service reference placed in folder
  - Look at the WSDL file, which describes the service

Exercise: Using a Web service on a client

- Now use/consume the service on the client:
- Create Parts.aspx page (File->New File-> Web Form)
  - Make sure Parts.aspx is in the main folder of the Web site
  - Drag ObjectDataSource from toolkit onto page
  - Configure data source: select Product Service, and GetProducts() method
  - Set default value= A
  - Drop Gridview control on page, set its data source. Test
- We usually have a vendor GridView and use the selected field instead of a default value—this is a shortcut example
- Your customer would normally build this Parts.aspx page on its Web site. Not you.
  - Customer would drop several ObjectDataSources on the page, one for each of the parts/products manufacturers/distributors (including yours) that it uses. It would then have a query or logic to choose the vendor based on availability, price, quality, etc.

Registries: UDDI, ebXML

- UDDI: Universal Description, Discovery and Integration
  - XML-based registry for businesses and Web services
  - UDDI contains information about the business that publishes Web services, and the Web services themselves
  - Example: http://soapclient.com/uddisearch.html
- ebXML: Competing standard (ISO)
- Individual organizations or industries typically use private registries
- Registries provided by MS Biztalk, IBM WebSphere, Software AG WebMethods, others

Exercise

- List the Web services you would include in your homework Web site
  - These would be registered and available for your customers to use in their systems that check your catalog, place orders, check status, and make payments
  - Most, if not all, will essentially be database queries
- What problems would you anticipate with these Web services?

Solution

- Web services:
  - Browse catalog, with many parameters/arguments to narrow by product type, etc.
  - Place order, with parameters on carriers/modes, split/backorder preferences, etc.
  - Order payment
  - Order/shipment status
  - Cancel order
  - Order/shipment/payment history
- Problems:
  - Letting users find the services, making sure they are almost always running, transactions/rollback, performance, security, etc.

Summary

- Web services:
  - Send XML documents as request and response between Web servers and/or clients using HTTP. (SOAP)
  - Are text-based, and avoid differences in Web server hardware, operating system, programming language, etc.
  - Are validated using XSD (XSchema Definition) (or DTD)
  - Can be discovered through WSDL registries
    - How they are called and what they return is defined in WSDL
  - Can be used as data sources (ObjectDataSource) from remote databases
  - Are relatively easy to implement
  - Allow data to be liberated from databases and shared
- Current developments:
  - JSON (JavaScript Object Notation) is being substituted for XML: less verbose. Still standards-based, text-based.
  - WCF (Windows Communications Framework) supersedes simple Web services (.asmx), more flexible (Web 2.0)