Module 1 note

from human-centric web to application-centric web

humans are actors

conversations between browsers and servers

helpful in numerous areas

need technology and mechanisms to support

- service discovery
- service description
- common standards

Web Service.

① What is a Web service?

network; XML; not tied (different { operating system / programming language })

② Web service are based on standards.

(some information about standardization)

↓

Web service standards!

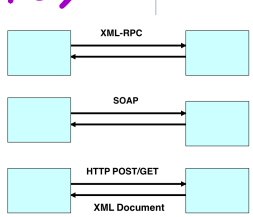
(XML, XML-RPC, SOAP, WSDL, UDDI ···)

③ XML Messaging there options

1) XML Remote Procedure Calls (XML-RPC)

2) SOAP

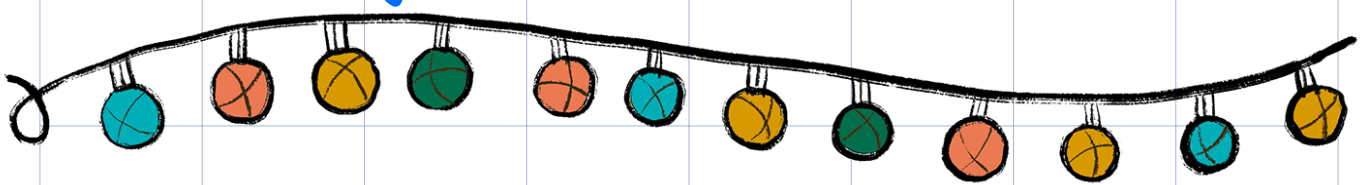
3) Regular XML transported over HTTP.



④ Web services Defined

{ self describing
  discoverable

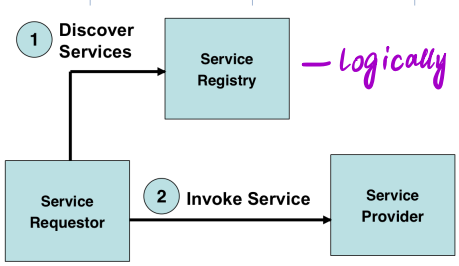
⑤ summary : include ① and ④



Web Services Architecture

① two ways to view the web service arch framework

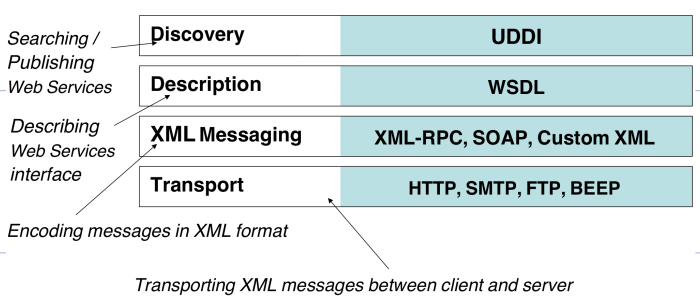
1) Web services Roles



— logically centralized directory of services

any consumer of the web services

2) Web Services Protocol Stack



Part II Web Service Protocols

① XML Messaging (request, response)

1) XML-RPC : the easiest way!

2) SOAP

XML-RPC VS SOAP ✓. (XML-RPC is simpler)

② Description - WSDL (web services description Language)

WSDL in a Nutshell

<type>, <message>, <portType>, <binding>, <service>

③ Discovery - UDDI

1) definition

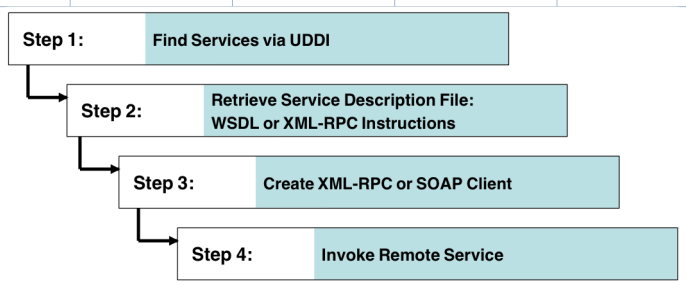
2) two part: technical specification, implementation.
技术规范 实现.

3) UDDI Data: White Pages, Yellow Pages, Green Pages

④ Transport. 略

⑤ All together two kinds of view.

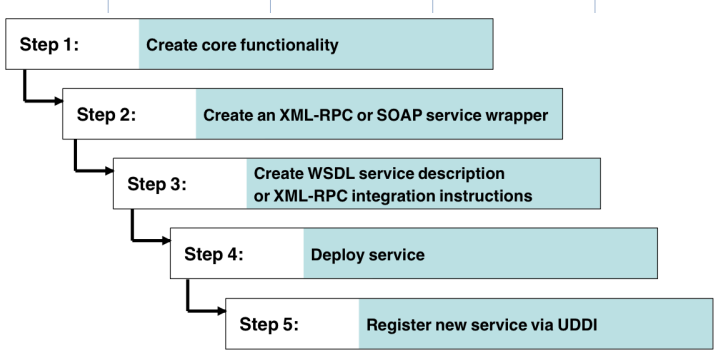
1) use the protocol together - service request perspective



retrieve 检索

invoke 引发

2) use the protocol together - service provider perspective



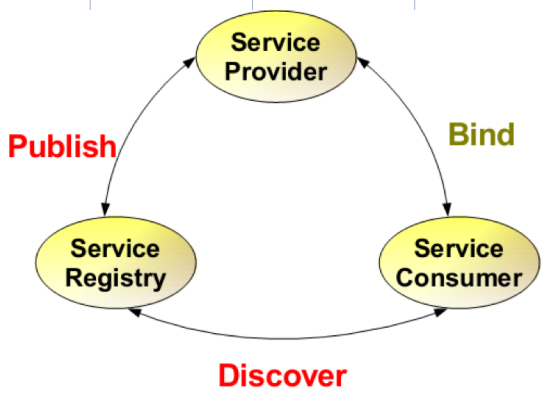
⑥ Web services example

1) Amazon SimpleDB

2) Amazon S3 (simple storage service)

can combine these different service.

⑦ another version of the roles view. (1)



(service consumer can also be called service requestor.)

Then an example for this part, and some review.

Part III XML-RPC Essentials

further talk about XML-RPC

① (我总结的) why choose XML-RPC for messaging in web service?

web services not tied to different system or language.

connect disparate (不同的) system (further more will be talked later)

example. why, how -> (RPC, XML simple introduction).

② XML-RPC usage and benefit (simplicity)

- on private network - glue code
- on public network - publish

② XML-RPC details

1) how XML-RPC works (HTTP request and HTTP response)

2) XML-RPC parts ① data model

(further details needs to be found in slides)

② request structures

③ response structures

④ Developing with XML-RPC 略

⑤ Criticism of XML-RPC

someone compare XML-RPC with XML.

they do have the same function, but XML-RPC is 4 times larger.

⑥ Benefits of XML-RPC (in summary, simple!)

⑦ Connect disparate systems — how

1) XML is data format, not protocol.

2) XML over HTTP POST request

3) Use standardized vocabularies

4) REST.

5) BEEP.