# Unit 7 Remote Procedure Calls using Web Services

**Unit Outcomes**. Here you will learn

- how remote procedure calls compare with REST principles and the distributed objects model as a basis for developing a DS
- about the important Web Services standards SOAP and WSDL; their purpose, design, operation and good practices
- to use the Axis tools via Eclipse to develop Java clients and servers for Web Services based on SOAP and WSDL

**Further Reading:** CDK 2005 Ch.19



## Contents



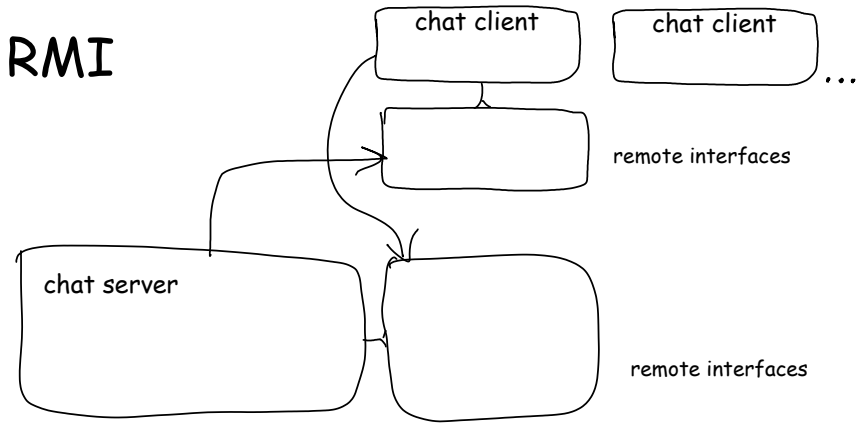


## Introduction to RPC WS
### Steps of RPC

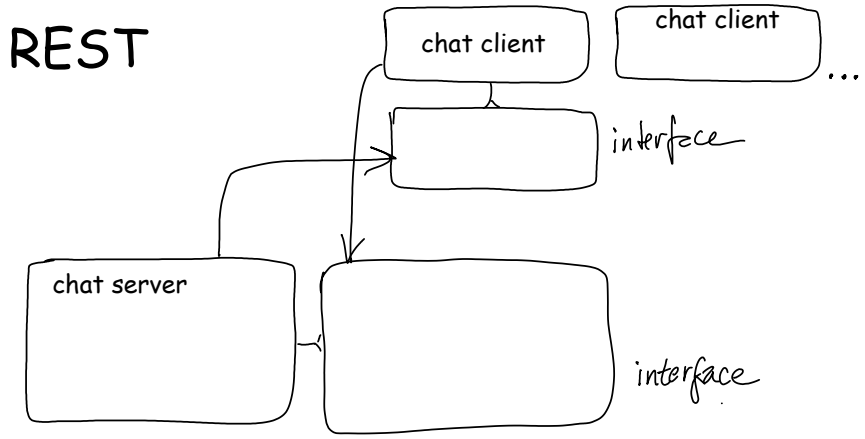
- remote procedure call:
  - node A locates a "port" or "remote procedure" on distributed node B;
  - node A sends a call to this port together with appropriate parameters and waits for a response;
  - node B receives the parameters, does some work and sends a response to node A.
- all parameters and return values *serialised*



## Conceptual comparison RPC–REST–RMI (RMI)

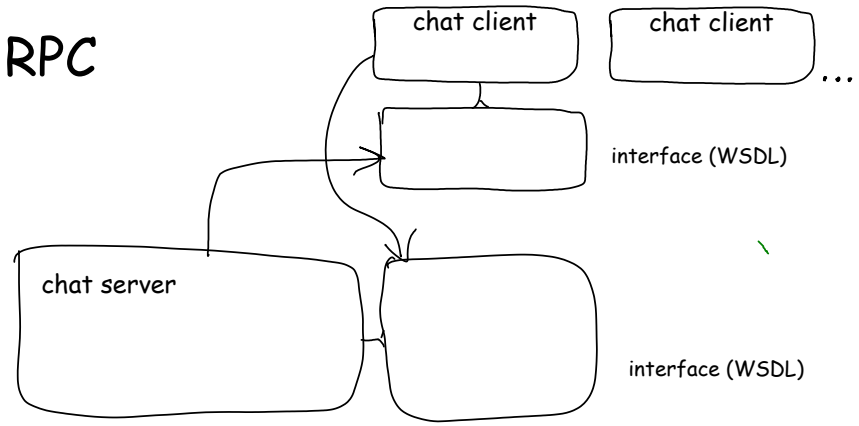


## Conceptual comparison RPC–REST–RMI (REST)





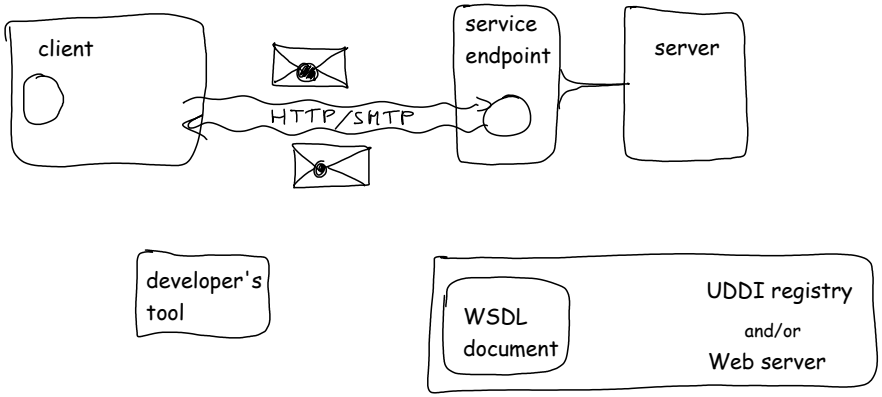

## Conceptual comparison RPC–REST–RMI (RPC)






## Introduction to RPC WS
### Overview of WS RPC standards

- how to fully specify a RPC service?
- standards help...

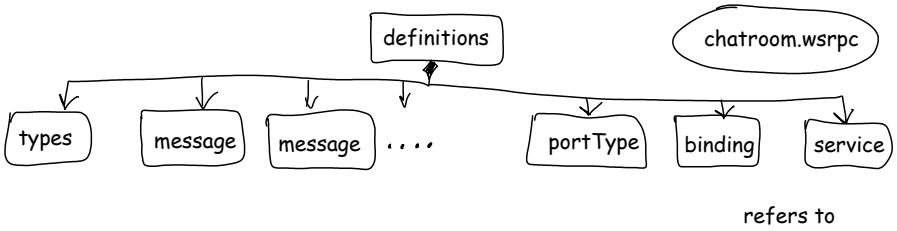





## Introduction to RPC WS
### Generic clients

- offer a human interface to any WSDL specified service
- eg:
  - Web Services Explorer in Eclipse
  - free web-based tools such as `http://www.soapclient.com/soaptest.html`
- good for:
  - testing one's own service
  - understanding other people's services
    - WSDL: RPC format, but no meaning or intentions

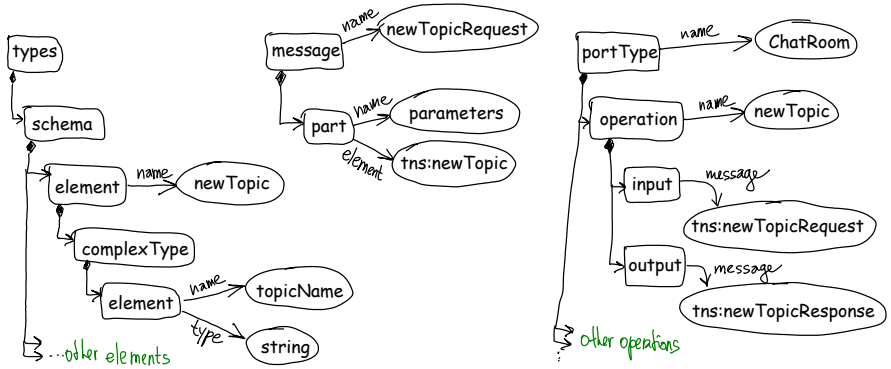
# Inroduction to WSDL
## Structure overview — root element




- WSDL defines new:
  - data types, message types, port types, binding types

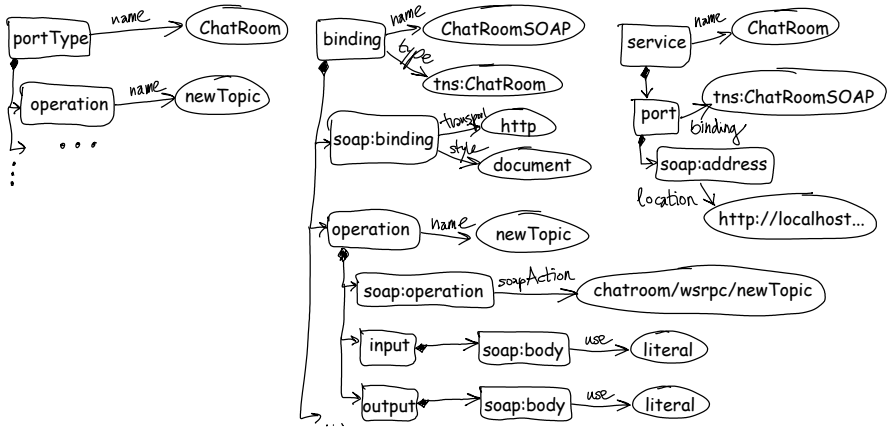


## Structure overview — port type






## Structure overview — service






## Structure overview — exercise

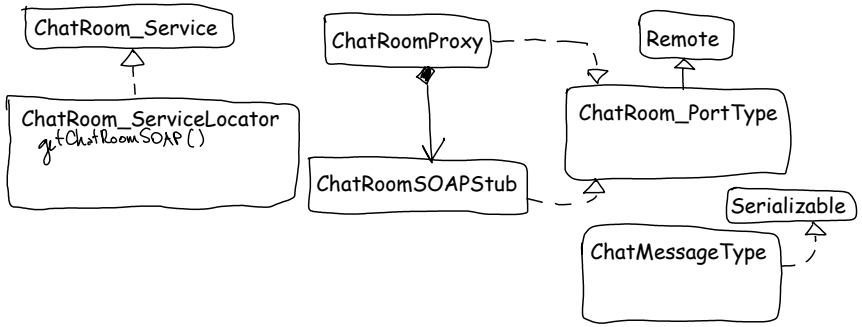
- draw a diagram for the 'quote of the day' service

## Composing WSDL from multiple documents

- embedded schema can *include* or *import* another schema
  - include: same effect as copy & paste, adding to target namespace
  - import: importing another namespace
- imported schema local or remote (use URL)
  . . . good for sharing in DS
- WSDL can import another WSDL
  - importing document can use imported:
    data types, message types, port types and binding types
  - imported document can be incomplete, eg:
    - only data types
    - only data, message and port types

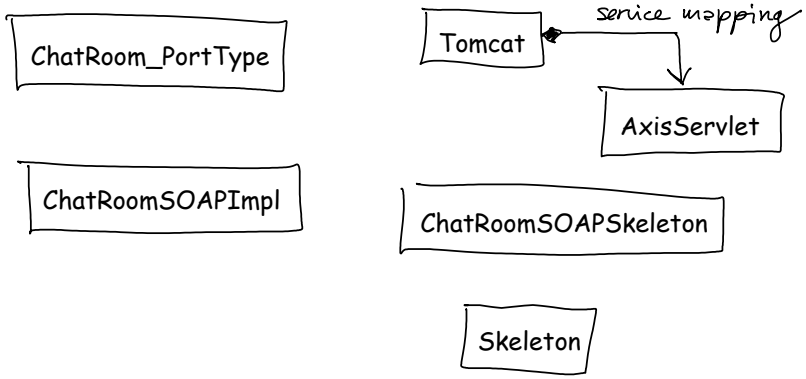


## Generating Axis client proxy






## Generating Axis server skeleton






## Learning Outcomes

**Learning Outcomes**. You should now be able to

- describe the conceptual differences between WS RPC and RESTful WS and RMI as DS development methodologies
- describe the roles of SOAP and WSDL in WS
- correctly interpret simple given WSDL document and represent the document by an appropriate diagram
- list the usual components of both a server and a client of a Web Service generated from a WSDL document and explain the function of each component
- generate a Java client proxy for a service described by a given WSDL document
- develop a Java program providing a service described by a given WSDL document