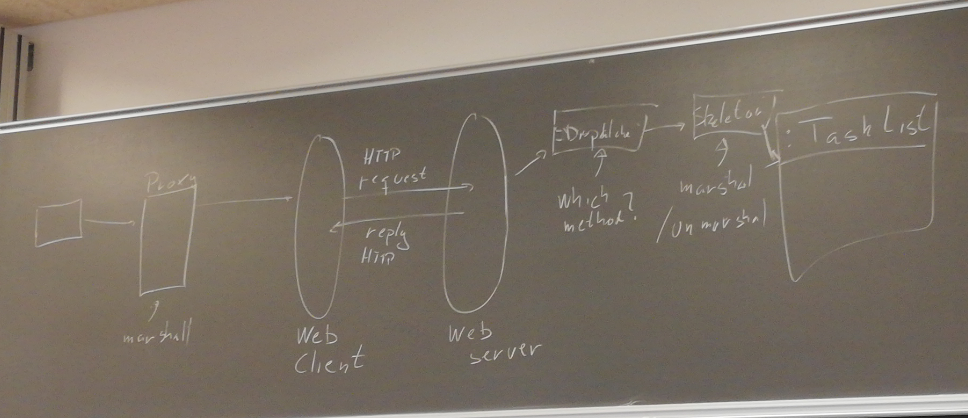
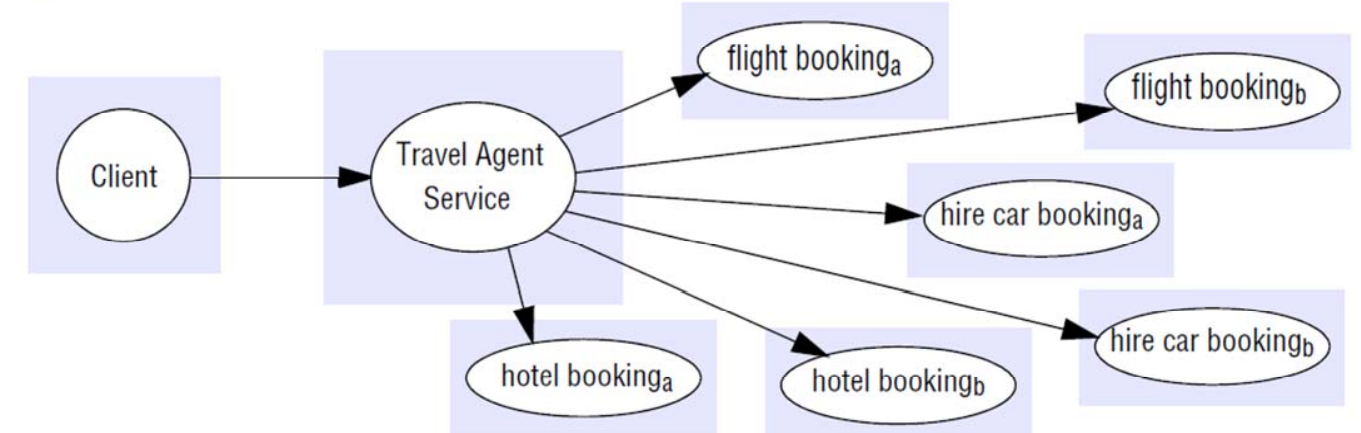
4. Web Services client/server architecture

Explain the ideas behind the Web Services client/server architecture. Describe some implementation details and how it works.





Web service = a collection of operations that can be used by a client over the Internet

Provides a service description: interface def and other info

They process SOAP messages or use a REST approach

- Designed to be independent of programming languages, to facilitate cooperation Loose coupling: to minimize dependencies between services, rely rather on interfaces and standard protocols

Are referred to via an URI or URL

- doesn't matter what language it was written in

- WSDL is a recipe of how the service can be consumed

What all web services have in common that they are all building on top of this model

Web server means pretty much that it speaks HTTP, and web client is anything that cab speaks HTTP.

Client proxy: A proxy (or stub) method knows the URL of the service and marshals its own method name and its arguments, together with a reference to the XML schema for the service, into a SOAP request envelope. Unmarshalling the reply consists of analyzing a SOAP envelope in order to extract the results, return value or fault report. The client’s request method call is sent to the service as an HTTP request.

Dispatcher and skeleton: As mentioned above, the dispatcher and skeletons live in the servlet container. The dispatcher extracts the name of the operation from the Action header in the HTTP request and invokes the corresponding method in the appropriate skeleton, passing the SOAP envelope to it.

A skeleton method carries out the following tasks: it analyzes the SOAP envelope in the request message and extracts its arguments, calls the corresponding method and assembles a SOAP reply envelope containing the results.