

Report: WSDL/SOAP Web Service

Group 16: You Hu 12036234, Wenchen Lai 12482625

Bottom Up Approach

In bottom-up approach, we create our web service from Java files using JAX-WS. We implement a Calculator Java Class and annotate the class as a web service endpoint using `@WebService` annotation. Then, we declare four public methods, add, sub, mul and div, with `@WebMethod` annotation. Each method has two double type inputs and return the calculation result which is also a double. With this file, we can build, package and deploy our service and a WSDL file will be automatically generated.

We also write a client to call our service. We import our service endpoint and specify the WSDL file location with `@WebServiceRef` annotation. Then four private methods are declared to invoke add, sub, mul and div methods on the port respectively.

Top Down Approach

As for top-down approach, we create our web service from a WSDL file. We use Eclipse to finish this task. In the WSDL file, four operations (add, sub, mul and div) are defined with tag `<xsd:element>` and their two inputs and one output are defined as double type. Messages and PortType are also described. Binding element is used to declare that our service communicate using SOAP protocol. Finally, we define our service with tag `<wsdl:service>`. With the WSDL file, we can generate all needed files to deploy our service. Some codes also need to be added to an implementation Java file (CalculatorTopDownServiceSOAPImpl.java) so that four operations can function as we define. For example, the return value of add method should be modified to the result of addition of two inputs. After that we can deploy our service. Eclipse can create a client to call our web service automatically. Then we can test our service with client.

Problem 3: make the Calculator stateful

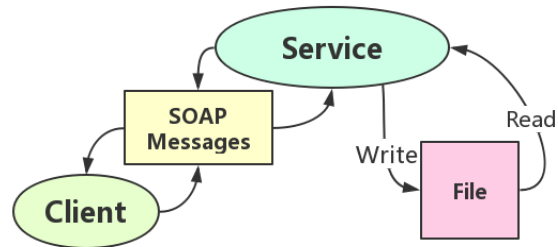


Figure: stateful calculator service

In order to make the calculator stateful, a file or a database can be created to store the history instructions in the server for each client. When the service receives a message, it will write the operation (add, sub, mul or div), two inputs and the return value to the file. If we want to know our last calculation, the service can read the file and get history calculations.