Se bem percebi, vocês conseguem usar e ver as mensagens SOAP com o LoggingHandler nos servidores que respondem aos web services (servidores transporter-ws em execução), mas não conseguem ver nem a mensagem SOAP inicial antes dos stubs do cliente (broker-ws neste caso é o cliente dos web services dos transporter-ws) invocarem o web service nem a mensagem SOAP que o cliente recebe como resposta à invocação inicial do web service?

Noto aqui que configurar os handlers para um endpoint do servidor é diferente de configurar os handlers para os stubs num cliente que invoca os web services.

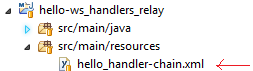
Peço para estudarem o exemplo do laboratório de handlers de estafetas para verem as diferenças:  
  
<http://disciplinas.tecnico.ulisboa.pt/leic-sod/2015-2016/labs/10-soap-handlers/index.html>

No final dessa página, têm links para o cliente e servidor de estafetas.

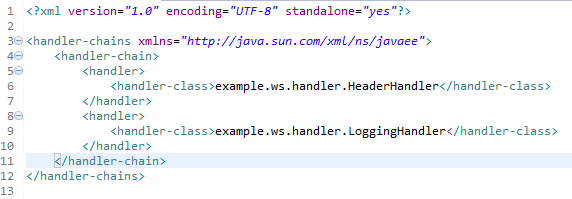
Nota: para compilar o cliente, lembrem-se que este exemplo é **implementation-first**, pelo que o servidor tem de estar em execução para conseguir compilar o cliente.

**Servidor:**  
<http://disciplinas.tecnico.ulisboa.pt/leic-sod/2015-2016/labs/10-soap-handlers/hello-ws_handlers_relay.zip>

Tem o ficheiro de configuração src/main/resources:

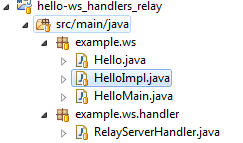


Este ficheiro tem "chain" no nome sem ser por acaso. Como o nome indicia, contém uma lista de handlers a usar no servidor e a ordem pela qual são usados:

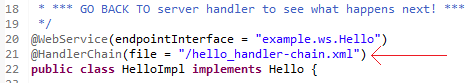


Na implementação da interface dos web services do servidor, coloca-se manualmente uma anotação a indicar que é para usar um ficheiro de handler.

Neste exemplo, a implementação é o ficheiro HelloImpl.java:



Se abrirem o ficheiro e virem a definição da classe HelloImpl, encontrarão a tal anotação:



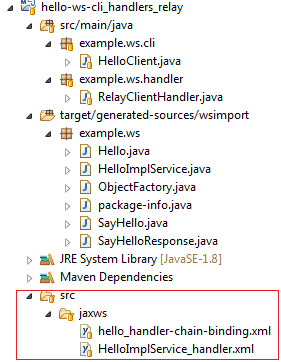
É com estas configurações que os handler desejados são configurados para funcionar no servidor.

**Cliente:**  
<http://disciplinas.tecnico.ulisboa.pt/leic-sod/2015-2016/labs/10-soap-handlers/hello-ws-cli_handlers_relay.zip>

Neste caso a configuração é ligeiramente diferente.

Em vez de anotar uma classe que implemente uma interface remota, é necessário anotar o stub que é usado para invocar o web service.

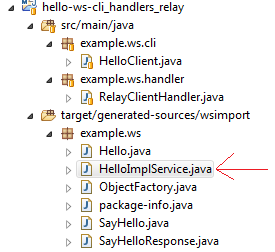
Mas... O stub é gerado automaticamente! Se anotarem o ficheiro gerado automaticamente, a anotação irá desaparecer se fizerem mvn clean e compilarem novamente.

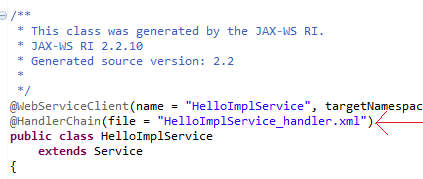
Para resolver isto, tem de existir o diretório src/jaxws no projeto cliente. Esta pasta tem dois ficheiros XML:  
  
  


O primeiro ficheiro, hello\_handler-chain-binding.xml tem dentro de si uma especificação do nome do ficheiro HelloImplService\_handler.xml, que é justamente o segundo ficheiro.

Este segundo ficheiro é o que especifica a configuração de handlers para **para o stub do cliente**. É assim que se configuram handlers para intercetar as mensagens SOAP a partir e a vir do cliente.

Com estes ficheiros presentes, quando se compila o projeto cliente, o stub é gerado automaticamente com a anotação que indica o ficheiro HelloImplService\_handler.xml:



Se for aberto esse ficheiro, encontrarão a tal anotação:  
  


É assim que conseguem associar handlers ao lado cliente de web services.

Se removerem a pasta jaxws desse projeto cliente e recompilarem, a anotação deverá desaparecer do HelloImplService (gerado automaticamente) e os handlers deixarão de funcionar no cliente.

Espero que este exemplo vos ajude. Obviamente, no vosso projeto, terão provavelmente de mudar o nome dos ficheiros, uma vez que não têm nenhuma interface remota ou stub chamado "Hello".

Bom trabalho,