

Contract first vs Contract last

Vimos duas abordagens de criar um Web Service SOAP no treinamento: **Contract first** e **Contract last**

Produzir um serviço a partir de um WSDL é chamado de **Contract first**. Gerar o WSDL a partir de uma classe Java, ou seja implementar primeiro o serviço (e o resto é gerado) é chamado do **Contract last**.

O *Contract first* pode ser usado quando temos um serviço já exposto e queremos migrá-lo para outra linguagem/plataforma, por exemplo, ou quando você tiver escrito o WSDL do zero, sem ser a partir de uma classe como fizemos no exercício anterior.

Uma vantagem do *Contract first* é que equipes diferentes podem começar a trabalhar ao mesmo tempo, uma trabalhando na implementação do servidor, e outra no cliente. Uma vez que o contrato (WSDL) está definido os dois lados estão desacoplados.

Outra vantagem do *Contract first* é uma definição mais clara do serviço. Ou seja, como não estamos escrevendo alguma implementação, o foco é o contrato e seus detalhes. Tópicos como expressividade, versionamento do serviço, granularidade e os tipos expostos são discutidos muito antes da implementação.

A API exposta é sempre o WSDL e os tipos (o schema), ambos os documentos devem ser legíveis e bem escritos. A implementação é apenas um detalhe.

Sabendo disso, você consegue imaginar alguma desvantagem do *Contract First*?