

02

O que é WCF

Serviço

O conceito de serviço já existe há um bom tempo! Serviços assim como componentes são blocos de construção independentes que em conjunto representam o ambiente da aplicação, sendo que um serviço é independente a outro e cada serviço é responsável pelo seu próprio domínio. Ou seja, é responsável por uma função de negócio específica.

Por exemplo, se temos um sistema de compra de passagens aéreas, quais serviços seriam possíveis oferecer? Poderíamos ter um serviço responsável por selecionar um cliente, disponibilizar a lista de passagens disponíveis para terceiros ou um serviço que efetua a compra de uma passagem. Veja que, para comprar uma passagem, provavelmente eu vou precisar selecionar um cliente, mas como eu já tenho um serviço para isso, poderíamos reaproveitar essa funcionalidade!

Para fazer um serviço consumir outro é necessário estabelecermos um *protocolo de comunicação* padrão para eles. A maneira mais comum de um serviço enviar e receber dados é utilizando XML.

Esse tipo de serviço tem dois requisitos fundamentais:

- Comunicar-se via protocolos de internet (geralmente HTTP)
- Enviar e receber dados via XML

Alem de definir o formato das mensagens precisamos definir um padrão de como a mensagem é enviada. Se pensarmos no mundo web, já temos um padrão de comunicação bem comum que é o *URI*. Por exemplo, quando vamos fazer uma busca no google precisamos acessar um link como:

http://www.google.com.br?q=sua_busca (http://www.google.com.br?q=sua_busca).

Neste formato estamos solicitando uma informação, ou seja, estamos fazendo um GET. Mas qual operações são possíveis? No protocolo HTTP temos as 4 principais operações que são `POST` , `GET` , `PUT` e `DELETE` . Já que temos toda essa arquitetura bem definida, porque não utiliza-la? Quando criarmos um serviço nesse padrão nós o chamamos de **REST**.

Poderíamos ter o recurso `clientes` que seria acessado como `http://meuendereco.com.br/clientes` e nesse endereço seria possível realizar as 4 operações do protocolo HTTP.

Um outro protocolo de comunicação bem conhecido é o SOAP onde estabelecemos um padrão para as mensagens, e onde obrigatoriamente todas precisam ter um cabeçalho e um corpo dentro de um elemento chamado de “envelope”.

As vantagens do SOAP é que qualquer linguagem consegue gerar e ler mensagens facilmente, porem, por ser em XML, isso gera um tempo considerável na hora de serializar/ deserializar as mensagens.

Veja que só para pensar em um Web Service temos que considerar diversas possibilidades. Além disso, caso precisássemos mudar o formato da resposta para JSON, por exemplo, seria necessário mexer em diversas partes do código - definir rotas, programar autenticação etc. Programar um web service é bem mais trabalhoso do que parece, mas para resolver esse problemas temos o WCF que vamos conhecer no próximo capítulo!

