

WEB SERVICES

OUTUBRO/2003

Definição

Um web service é um componente, ou unidade lógica de aplicação, acessível através de protocolos padrões de Internet. Como componentes, esses serviços possuem uma funcionalidade que pode ser reutilizada sem a preocupação de como é implementada. O modo de acesso é diferente de alguns modelos anteriores, onde os componentes eram acessados através de protocolos específicos, como o DCOM, RMI ou IIOP. *Web Services* combinam os melhores aspectos do desenvolvimento baseado em componentes e a Web.

Há algumas especificações e tecnologias definidas para a construção ou utilização de Web services. Essas especificação e tecnologias endereçam para os seguintes requisitos para o desenvolvimento baseado em serviços:

- uma forma comum de representar dados
- um formato de mensagens comum e extensível
- uma linguagem de descrição do serviço, comum e extensível
- um mecanismo para localizar os serviços localizados em um Web site específico
- um mecanismo para descobrir os provedores de serviço

O XML é a escolha natural para o modo de representação dos dados. Muitas especificações utilizam o XML para representação dos dados, assim como os *XML Schemas* para descrever os tipos dos dados.

O SOAP (Simple Object Access Protocol) é um protocolo leve para troca de informações. Parte da sua especificação é composta por um conjunto de regras de como utilizar o XML para representar os dados. Outra parte define o formato de mensagens, convenções para representar as chamadas de procedimento remoto (RPCs) utilizando o SOAP, e associações ao protocolo HTTP.

O WSDL (Web Services Description Language) é uma linguagem baseada em XML, com a finalidade de documentar as mensagens o Web service aceita e gera. Esse mecanismo padrão facilita a interpretação dos contratos pelos desenvolvedores e ferramentas de desenvolvimento.

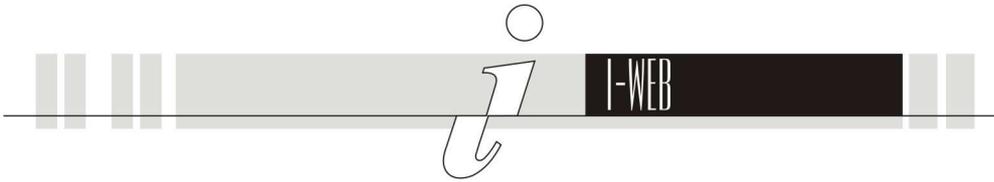
Também é necessária uma forma de localização dos Web services. O protocolo Disco (Discovery

Protocol) define um formato para o documento *discovery* e um protocolo para devolver esse documento, possibilitando a localização dos serviços em um web site conhecido. No entanto, é comum que não se saiba as URLs onde os serviços podem ser encontrados. O UDDI (Universal Description, Discovery, and Integration) é um mecanismo para os fornecedores anunciarem a existência de seus serviços, e para os consumidores localizarem os serviços de seu interesse.

Podemos definir, resumidamente, um XML *Web service* como um serviço de software publicado na Web através do SOAP, descrito com um arquivo WSDL e registrado em UDDI.

SOAP (Simple Object Access Protocol)

O SOAP é uma especificação para requisitar métodos de negócio, como documentos XML, e suporta outros protocolos como HTTP e SMTP. É o protocolo de comunicação para os Web Services; quando descrito como um protocolo de comunicação, muitos pensam no DCOM ou no CORBA e se perguntam "Como o SOAP realiza a ativação de objetos?" ou "Qual serviço de nomeação o SOAP utiliza?". Apesar da implementação SOAP provavelmente incluir esses itens, o padrão SOAP não os especifica. O SOAP é uma especificação que define o formato das mensagens XML. Uma mensagem SOAP nada mais é que um fragmento XML bem formado, encapsulado por um par de elementos SOAP. Existem outras partes da especificação SOAP que descrevem como representar os dados do programa em XML e como usar o SOAP para fazer as chamadas de procedimento remoto. Essas partes opcionais da especificação são usadas para implementar as aplicações no estilo RPC, onde a mensagem SOAP, contendo a chamada e os parâmetros da função, é enviada pelo cliente, e o servidor retorna uma mensagem com os resultados da função executada. As implementações correntes do SOAP suportam aplicações RPC porque os programadores acostumados com as aplicações COM ou CORBA aplicam muito bem esse estilo. O SOAP também suporta aplicações no estilo documento, onde a mensagem é apenas um envólucro de um documento XML. Esse estilo é muito flexível e alguns Web services o utilizam para implementar



WEB SERVICES

OUTUBRO/2003

serviços que seriam difíceis de desenvolver usando RPC.

A última parte opcional da especificação SOAP define como é uma mensagem HTTP que contém uma mensagem SOAP. A ligação com o HTTP é opcional, mas quase todas as implementações SOAP suportam, pois é o único protocolo padronizado para o SOAP. Por essa razão, há uma concepção errada de que o SOAP requer o HTTP. Algumas implementações suportam MSMQ, MQ Series, SMTP ou TCP/IP, mas quase todos os Web services usam HTTP por ser largamente difundido. Como o HTTP é o protocolo central da Web, muitas organizações possuem uma infraestrutura de rede que suporta HTTP e pessoas que entendem como gerenciá-lo. A segurança, monitoração e balanceamento de carga para HTTP são bem eficazes hoje em dia.

A maior fonte de confusão na utilização do SOAP é a diferença entre a especificação e muitas implementações. Muitas pessoas que usam o SOAP não escrevem as mensagens diretamente, mas usam um SOAP *toolkit* para criar e analisar as mensagens SOAP. Esses *toolkits* geralmente traduzem chamadas de funções de alguma linguagem para uma mensagem SOAP. Por exemplo, o Microsoft SOAP *Toolkit* 2.0 transforma uma chamada de função COM para o SOAP, e o Apache *Toolkit* transforma uma chamada de função JAVA para o SOAP. Os tipos de chamadas e tipos de parâmetros suportados variam com cada implementação, assim uma função que funciona com um *toolkit* pode não funcionar com outra. Essa não é uma limitação do SOAP mas sim da implementação particular que se está usando.

De longe a funcionalidade mais importante do SOAP é que está sendo implementado em diversas plataformas de *hardware* e de *software*. Isso significa que o SOAP pode ser usado para ligar sistemas muito diferentes, dentro e fora da organização. Muitas tentativas desse tipo foram feitas no passado, propondo um protocolo de comunicação padrão para ser usado na integração de sistemas, mas nenhuma delas teve a mesma repercussão que o SOAP, devido a sua simplicidade e facilidade de implementação.

WSDL (*Web Services Description Language*)

WSDL For a business to discover a service it wants to use, it needs to understand the call syntax and semantics prior to actually making a call. The WSDL (*Web Services Description Language*) specification is an XML document which describes the interface, semantics, and administrivia of a call to the web service. This allows for simple services to be quickly and easily described and documented.

Simplificadamente, pode-se dizer que o arquivo WSDL é um documento XML que descreve um conjunto de mensagens SOAP e a forma como essas mensagens são trocadas. Em outras palavras, o WSDL é para o SOAP o que o IDL é para o CORBA ou COM. Como o WSDL é XML, ele é legível e editável mas na maioria dos casos, ele é gerado e consumido pelo software.

Para enxergar o valor do WSDL, imagine que você quer invocar um método SOAP que é fornecido por um de seus parceiros de negócios. Você pode pedir alguns exemplos de mensagens SOAP e escrever sua aplicação para produzir e consumir mensagens que se parecem com os exemplos, mas isso pode gerar muitos erros. Por exemplo, você pode assumir que um campo é um inteiro, quando de fato é uma string. O WSDL especifica o que a mensagem de requisição deve conter e como vai ser a resposta, em uma notação não ambígua.

A notação que o arquivo WSDL usa para descrever o formato das mensagens é baseado no padrão XML, o que significa que é uma linguagem de programação neutra e baseada em padrões, o que a torna adequada para descrever as interfaces dos Web services, que são acessíveis por uma grande variedade de plataformas e linguagens de programação. Além de descrever o conteúdo das mensagens, o WSDL define onde o serviço está disponível e quais protocolos de comunicação são usados para conversar com o serviço. Isso significa que o arquivo WSDL define tudo que é necessário para escrever um programa que utilize o XML *Web service*. Há várias ferramentas disponíveis para ler o arquivo WSDL e gerar o código para comunicar com o XML *Web service*.

UDDI (*Universal Discovery Description and Integration*)

Para fazer uma chamada a um web service, é necessário localizá-lo, descobrir a interface e semântica da sua chamada, escrever e configurar o software local para colaborar com o serviço. Assim, é necessário um mecanismo para publicar os serviços.

O UDDI são as páginas amarelas dos *Web services*. Como as páginas amarelas tradicionais, você pode procurar por uma companhia que ofereça os serviços que você precisa, ler sobre o serviço oferecido e contatar alguém para obter mais informações. Você pode, naturalmente, oferecer um web service sem registrá-lo na UDDI, assim como você pode abrir um negócio no porão da sua casa e contar com a propaganda boca a boca, mas se você quiser alcançar um mercado significativo, você precisará do UDDI, assim seus clientes poderão encontrá-lo.

Um diretório UDDI é um arquivo XML que descreve o negócio e os serviços. O registro tem três partes: as "páginas brancas" descrevem a companhia: nome, endereço, contatos, etc. as "páginas amarelas" incluem as categorias, baseada em taxonomias padrões. As "páginas verdes" descrevem a interface para o serviço, em nível de detalhe suficiente para se escrever uma aplicação que use o *Web service*. A forma como os serviços são definidos no documento UDDI é chamado *Type Model* ou tModel. Em muitos casos, o tModel contém o arquivo WSDL que descreve a interface SOAP do *Web service*, mas o tModel é flexível o suficiente para descrever quase todo tipo de serviço.

O diretório UDDI também inclui várias maneiras de procurar os serviços. Por exemplo, pode-se procurar por fornecedores de um serviço em uma localização geográfica específica ou por negócios de um tipo específico.

Vantagens

Flexibilidade

Uma das principais vantagens dessa arquitetura é permitir que programas escritos em diferentes linguagens, em diferentes plataformas, comuniquem-se uns como os outros de uma forma padronizada.

Facilidade e Aceitação

Outras iniciativas nesse sentido já foram concebidas, e pode-se questionar qual a deferência dessa implementação com o CORBA ou com o DCE. A força do SOAP está na sua simplicidade; é fácil de entender e implementar. O SOAP é significativamente menos complexo que as abordagens anteriores, assim a barreira para a mudança para uma implementação SOAP é menor. Hoje existem implementações de grandes companhias de software, mas também existem muitas implementações construídas e mantidas por um único desenvolvedor. Outra diferença significativa é trabalhar com protocolos padrões de Web - XML, HTTP e TCP/IP. Um grande número de companhias já possuem uma infraestrutura *Web*, pessoas com conhecimento e experiência na sua manutenção, e novamente, o custo para uma mudança para o XML *Web services* é menor que para as propostas anteriores.

Aplicações

Os serviços mais simples são fontes de informação, que podem ser facilmente incorporados às aplicações - cotações de preços, cotas de estoque, previsões de tempo e temperatura, resultados de partidas esportivas, etc.

É fácil imaginar uma grande classe de aplicações que podem ser feitas para analisar e agregar informações importantes, e apresentá-las de várias formas; por exemplo, uma planilha que resume sua situação financeira - títulos, contas bancárias, empréstimos, etc. Se essa informação estiver disponível através de *Web services*, a planilha pode ser atualizada continuamente.

A publicação de aplicações existentes através de *Web Services* permitirão a construção de aplicações novas e poderosas. Por exemplo, pode-se desenvolver uma aplicação de aquisições que automaticamente obtém os preços de vários fornecedores, possibilitando que o usuário selecione um fornecedor, faça o pedido e acompanhe o carregamento até o recebimento. A aplicação do fornecedor, pode também utilizar *web services* para checar o crédito do cliente, cobrar da conta e solicitar o transporte da carga para uma transportadora.



Recycle