

INF01151 – Sistemas Operacionais II

Web services (SHORT VERSION)

Prof. Alberto Egon Schaeffer Filho



Web services

- Definição W3C
 - “Um **web service** é um sistema **identificado por uma URI** cujas interfaces são definidas e **descritas usando XML**. Sua definição **pode ser descoberta** por algum outro sistema ou software. Esses sistemas devem **interagir** com um serviço web de forma presente em sua definição utilizando **mensagens baseadas em XML** transmitidas **através de protocolos da Internet**.”
- Porém, com a popularização da arquitetura REST, pode-se definir web service como “qualquer sistema desenvolvido para suportar operações entre computadores através de uma rede



Introdução

- Necessidade de comunicação cliente/servidor entre sistemas heterogêneos, com necessidade de interoperabilidade
 - Ex: DCOM, Java RMI, Sun RPC, etc
 - Problema: “conversam naturalmente” apenas consigo mesmo, isto é, apenas com software homogêneo
- Solução: explorar formato e/ou arquitetura padrão para troca de informações
 - XML-RPC (XML sobre HTTP)
 - SOAP (XML + protocolo de comunicação)
 - REST (métodos HTTP + XML/JSON)

evolução



Web service

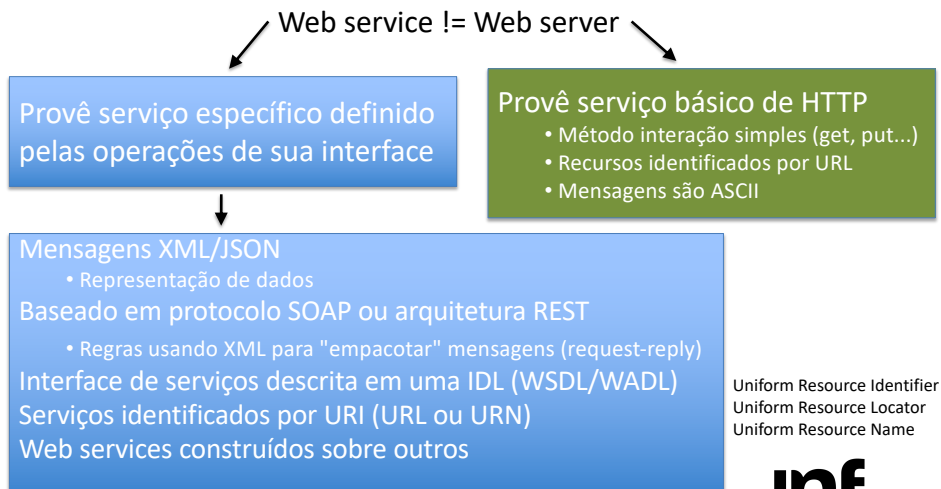


Objetivos dos web services

- Substituir os protocolos atuais (RPC, DCOM e RMI) por uma abordagem aberta:
 - Interoperabilidade baseada na web
 - Reusável
 - Independente de plataforma/linguagem
- Permitir a interação de componentes heterogêneos, remotos e independentes através de um protocolo (SOAP) ou arquitetura (REST) padrão
- Voltado para organizações
 - Integração B2B (Business to Business)
 - EAI (Enterprise Application Integration)
 - P2P (Peer-to-Peer)



Web services vs. Web server



5

INF01151 - Sistemas Operacionais II

Web services vs. Web server

- HTTP é um protocolo request-reply
 - Web server
 - São acessados por um browser
 - Provê um serviço HTTP básico (métodos padrões HTTP)
 - Acessados por URL
 - Web service
 - Sem interação humana
 - Provê serviços baseados nesta interface
 - Acessados por URI
- URI (Uniform Resource Identifier)
 - URL (Uniform Resource Location)
 - Fornece a localização de um recurso
 - URN (Uniform Resource Name)
 - Independe de localização
 - Serviço de lookup para transformar URN em uma URL

6

INF01151 - Sistemas Operacionais II

Web services

- Característica chave:
 - Processamento de mensagens XML empacotadas em SOAP
 - Alternativa: Arquitetura REST (*Representational State Transfer*)
 - Clientes usam URL e HTTP (get, put, delete e post) para manipular recursos
 - Ênfase em manipulação de dados ao invés de interfaces
- Implantação de serviços web
 - Gerenciado por um servidor web com páginas web; ou
 - Aplicação totalmente separada de um servidor web
 - Utiliza seu próprio *service description* para lidar com características específicas das mensagens recebidas
 - Por exemplo, web services providos pela amazon.com (SOAP ou REST)
 - Permitir que clientes obtenham características sobre um produto
 - Adicionar produto ao carrinho
 - Verificar status de uma transação

7

INF01151 - Sistemas Operacionais II

Composição de web services

- Combinar operações de uma interface com interfaces de outros serviços para oferecer novas funcionalidades (agente de viagens)
 - Reservar vôos, hotéis, aluguel de carros usando diferentes web services

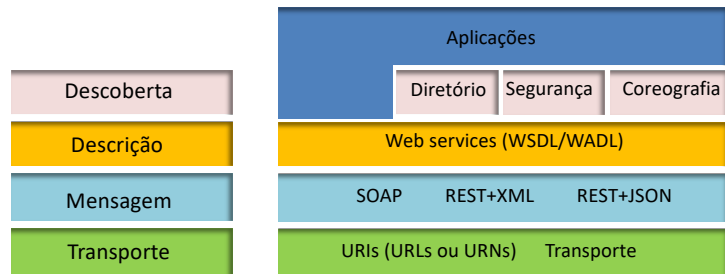


- Recursos disponíveis na web
- Serviços básicos são como *commodities*
- Obtenção de serviços com valor agregado

8

INF01151 - Sistemas Operacionais II

Web services: infra-estrutura e componentes



Detalhes:

- Representação externa de dados e marshalling feito em XML ou JSON (readability/debugging)
- Transporte é um protocolo de aplicação da Internet (**HTTP**, SMTP, FTP, etc)
- SOAP: regras para usar XML no empacotamento de mensagens (em request-reply protocols)
 - Encapsula mensagens e transmite elas sobre HTTP ou outro protocolo
- Web services construídos sobre outros, podem prover funcionalidades mais gerais
 - Descoberta envolve serviços de diretórios e de nome (UDDI, ebXML, ...)

9

INF01151 - Sistemas Operacionais II

E mais concretamente?

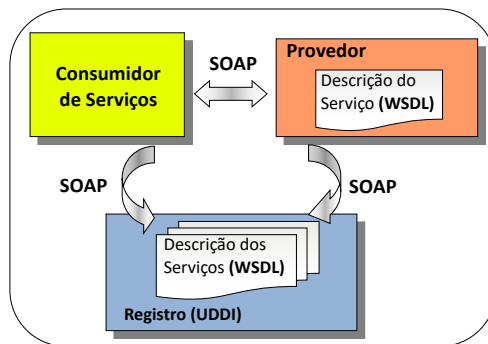
- Uma nova tecnologia para objetos distribuídos
 - Invocação remota de WS: SOAP (≈ RMI, RPC)
 - Descrição de WS: WSDL (≈ IDL)
 - Registro e descoberta de WS: UDDI (≈ rmiregistry, portmapper)
- Baseado em padrões XML
 - W3C: XML, SOAP, WSDL
 - Industriais: UDDI, ebXML
 - Proprietários: Disco, WSDD, WSFL, ASMX,...
- Implementações atuais:
 - Microsoft .NET
 - Sun JavaONE: J2EE+WebServices (JAXP, JAX-RPC, JAXM...)
 - Apache XSOAP
 - Oracle, Bea, Iona, Enhydra...

10

INF01151 - Sistemas Operacionais II

Arquitetura de web services

- Baseado em arquiteturas orientadas a serviço (*service oriented architecture – SOA*)
- Três componentes:
 - Consumidor
 - Registro
 - Provedor
- Tecnologias
 - SOAP
 - WSDL
 - UDDI



11

INF01151 - Sistemas Operacionais II

XML – eXtensible Markup Language

- Prover mecanismo para identificação de dados visando a troca de informações
 - Formatação e filtragem de dados
 - Manipulação de informações
- XML está presente nos principais níveis da pilha tecnológica
 - Formatação de mensagens SOAP
 - Descrição de serviços (WSDL)
 - Publicação de serviços (UDDI)
- XML *schema* é um conjunto de regras para definição de elementos
 - Modelo abstrato de dados

12

INF01151 - Sistemas Operacionais II

JSON – JavaScript Object Notation

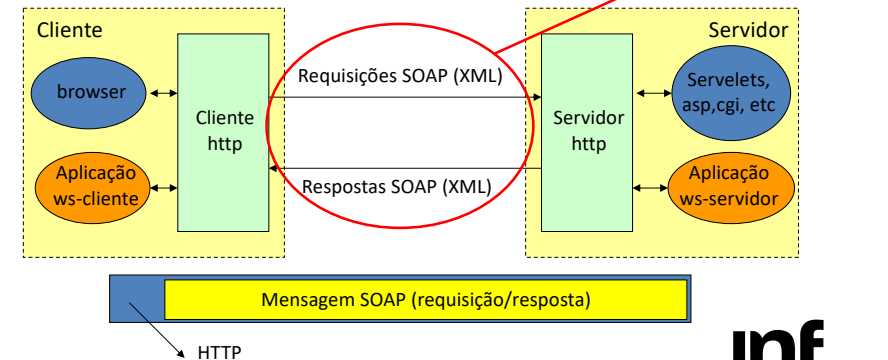
- Alternativa a XML para troca de mensagens em web services
 - Formato originalmente derivado da linguagem JavaScript
 - Requer menos recursos computacionais e de rede comparado a XML
 - Não necessita de validação/*schema*
 - Facilmente incorporado a linguagens populares (JS, Java, Python..)
 - Legibilidade boa (debugging)
 - Suporta hierarquia e tipos de dados compostos (i.e. Listas)
- XML ainda é necessário para descrição de serviços REST (WADL)

13

INF01151 - Sistemas Operacionais II

Simple Object Access Protocol (SOAP)

- Projetado para suportar interação entre web services
 - Empacotar mensagens em XML
 - [Tipicamente] usa HTTP para comunicar mensagens



14

INF01151 - Sistemas Operacionais II

Especificação SOAP

- O que define o padrão SOAP?
 - Como XML é usado para representar conteúdo de mensagens
 - Como mensagens podem ser combinadas para fazer request-reply
 - Regras de processamento do XML pelo receptor da mensagem
 - Como HTTP (e outros) são usados para comunicar mensagens SOAP
 - Mensagem SOAP definida em um envelope
 - **header**: contexto para um serviço
 - E.g., transaction ID, username, public key
 - **body**: XML para um web service em particular
 - Documento a ser comunicado diretamente no corpo (assíncrono)
 - Mensagem de request ou reply (síncrono)
- ✓ Mensagens SOAP NÃO contém referência ao destinatário

15

INF01151 - Sistemas Operacionais II

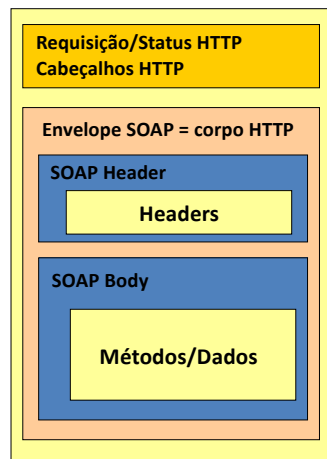
Transporte de mensagens SOAP

- Mensagem SOAP não contém referência sobre seu destino
 - Necessidade de “alguém” fornecer o destino
 - Protocolo de transporte
- Protocolo de transporte
 - Indicar quem deve tratar a URI (URN ou URL)
 - HTTP é tipicamente empregado
- SOAP request
 - Método HTTP GET:
 - Para obter informações (sem argumento, sem ações)
 - Método HTTP POST:
 - Para realizar ações no servidor ou enviar argumentos

16

INF01151 - Sistemas Operacionais II

Mensagem SOAP sobre HTTP



- Uma comunicação por mensagem
- Estrutura de mensagem
 - Envelope
 - Moldura base
 - Fornece o *namespace*
 - Cabeçalho (Header)
 - Opcional
 - Suporte a transações, sessões, segurança
 - Corpo (Body)
 - Elementos da mensagem
 - Métodos, argumentos, valores de retorno
 - Elementos de erro (*fault*)

17

INF01151 - Sistemas Operacionais II

Mensagem SOAP sobre HTTP

```
POST /examples/stringer
Host: www.cdk4.net
Content-Type: application/soap+xml; charset=utf-8
Content-Length: 299
SOAPAction: "http://www.cdk4.net/examples/stringer#GetStockPrice"
```

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
  </soap:Header>
  <soap:Body>
    <m:GetStockPrice xmlns:m="http://www.example.org/stock">
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  </soap:Body>
</soap:Envelope>
```

18

INF01151 - Sistemas Operacionais II

REpresentational State Transfer

- Arquitetura REST visando web services
 - Principal foco é em como prover acesso a recursos de maneira interoperável, fácil e *lightweight*
 - Faz uso dos métodos HTTP para realizar operações (GET, POST, DELETE)
 - Questões de segurança cabem ao protocolo de transporte (HTTPS)
 - Diferentemente de SOAP, permite representação de dados em formatos distintos (texto puro, XML, JSON, YAML)
 - Não necessita guardar estado entre operações (cada requisição é autocontida)
- URI utilizada para expor lógica de serviços
 - Pode conter informações como namespace, nome do serviço e opcionalmente parâmetros
 - Informações adicionais passadas via HTTP POST

19

INF01151 - Sistemas Operacionais II

Web services x objetos distribuídos

- Comparação web services vs. modelo de objetos distribuídos
 - Superficialmente, interação cliente/servidor é muito parecida
 - RMI: cliente utiliza referência remota para invocar método no objeto remoto
 - Web services: cliente usa URI para invocar operação no recurso correspondente
- Referências remotas x URI
 - Web service não pode criar novas instâncias e passá-las como parâmetros
 - Web service consiste em um único objeto remoto
 - Garbage Collection não se aplica
 - Referência remota não se aplica
- Organizações usam firewalls para proteger suas redes locais
 - Protocolos usados por Java RMI e CORBA normalmente não poderiam passar pelo firewall
 - Por outro lado, normalmente firewalls permitem tráfego HTTP e SMTP

20

INF01151 - Sistemas Operacionais II

SOAP API

- Como proteger o programador de detalhes?
 - Marshalling, unmarshalling
 - Sintaxe para invocação remota
- SOAP API: disponível em várias linguagens (Java, Python, C++)
- SOAP API para Java: JAX-RPC
 - Interface do web service deve estender Remote (como em RMI)
 - Mas, objeto que implementa interface remota **NÃO** pode ser passado como parâmetro
 - Compiladores usados para gerar stubs e descrição do WS (em WSDL)
 - wscompile & wsdeploy
 - Arquitetura similar aquela utilizada no middleware RMI
 - Porém, não existe módulo de referência

Nada disso é provido por infraestrutura de web services

Leituras adicionais

- Couloris, G.; Dollimore, J.; Kindberg, T. and Blair, G.–
“*Distributed Systems: Concepts and Design*” (5th edition),
Addison Wesley, 2012
 - Capítulo 9