SISTEMAS DISTRIBUÍDOS

Aula 4

Web Services em Java

SUMÁRIO

Web services

Exemplo

WEB SERVICES

Um Web service é um sistema especificamente ***desenhado para suportar a interoperação e a interação*** *máquina-para-máquina sobre uma rede.*

A interface é descrita num formato que é processável por uma máquina: ***WSDL***

Outros sistemas podem interagir com um Web service seguindo a descrição da sua interface, por exemplo através do ***mensagens SOAP****, tipicamente sobre* ***HTTP****, utilizando* ***serialização em XML*** *e outros standarts Web.*

Protocolo: HTTP e HTTPS (eventualmente SMTP)
Referências: URL/URI
Representação dos dados: XML

WSDL – INTERFACE DESCRIPTION LANGUAGE PARA *WEB SERVICES*

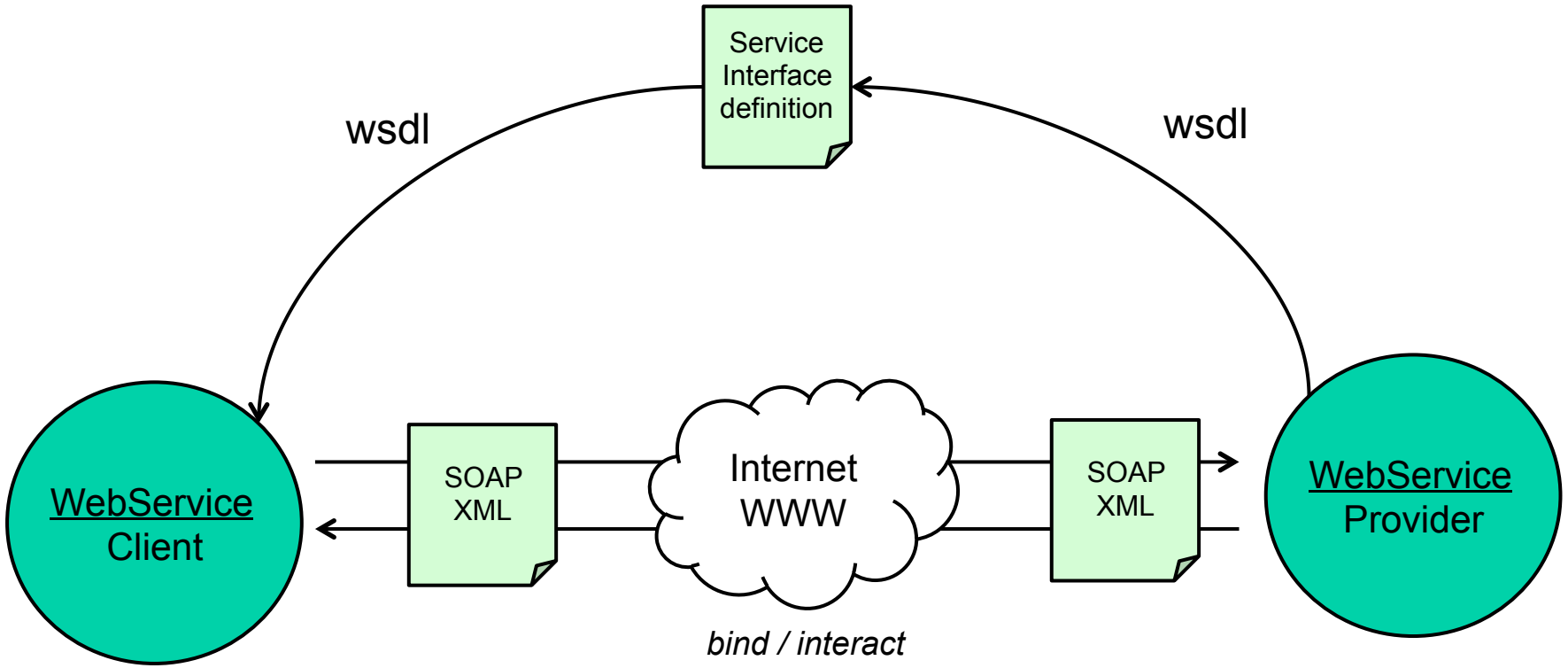
Define a interface do serviço, indicando **quais as operações disponíveis**

Define as mensagens trocadas na interação (e.g. na invocação duma operação, quais as mensagens trocadas)

Permite definir a forma de representação dos dados e a forma de aceder ao serviço

Especificação WSDL bastante verbosa – pode ser criada **automaticamente** a partir de interface ou código do servidor

WSDL – Modelo Abstrato



WEB SERVICES EM JAVA - SERVIDOR

SERVIDOR:

- Começa pela criação da interface e a sua implementação
- Utilizam-se **anotações** para indicar: os métodos disponibilizados remotamente; a classe que implementa o serviço propriamente dito.
- Instancia-se, criando um *endpoint* HTTP e fazendo a sua ligação (binding) à instância da classe principal.

WEB SERVICES EM JAVA - CLIENTE

CLIENTE:

- Com base no URL de uma instância do serviço, no cliente é possível instanciar um "middleware" que expõe as operações do serviço localmente, acessíveis por invocação direta de métodos.
- Havendo múltiplas instâncias do serviço, o cliente pode aceder a cada uma delas através do URL respetivo.

Web Services em Java – Aspetos Adicionais

A passagem de parâmetros e o retorno dos métodos é feita por cópia dos valores (não há passagem de referências)

O tipo dos parâmetros e do retorno pode sofrer alterações na tradução (e geração dos stubs).

- Por exemplo, String[] -> List<String>

EXEMPLO – SERVIDOR RENDEZVOUS

- Interface do Serviço
- Servidor
- Cliente

EXEMPLO: INTERFACE

```
@WebService
public interface RendezVousAPI {

        final String NAME = "RendezVousService";
        final String NAMESPACE = "http://sd2017";
        final String INTERFACE = "api.soap.RendezVousAPI";

        @WebMethod
        Endpoint[] endpoints();

        @WebMethod
        void register(String id, Endpoint endpoint);
}
```

EXEMPLO: SERVIDOR (1)

```
@WebService( serviceName = RendezVousAPI.NAME,
        targetNamespace = RendezVousAPI.NAMESPACE,
        endpointInterface = RendezVousAPI.INTERFACE)
public class RendezVousServiceImpl implements RendezVousAPI {
    private Map<String, Endpoint> db = new ConcurrentHashMap<>();

    @Override
    public Endpoint[] endpoints() {
        return db.values().toArray( new Endpoint[ db.size()]);
    }

    @Override
    public void register(String id, Endpoint endpoint) {
        System.err.printf("register: %s <%s>\n", id, endpoint);
        db.put(id, endpoint);
    }
}
```

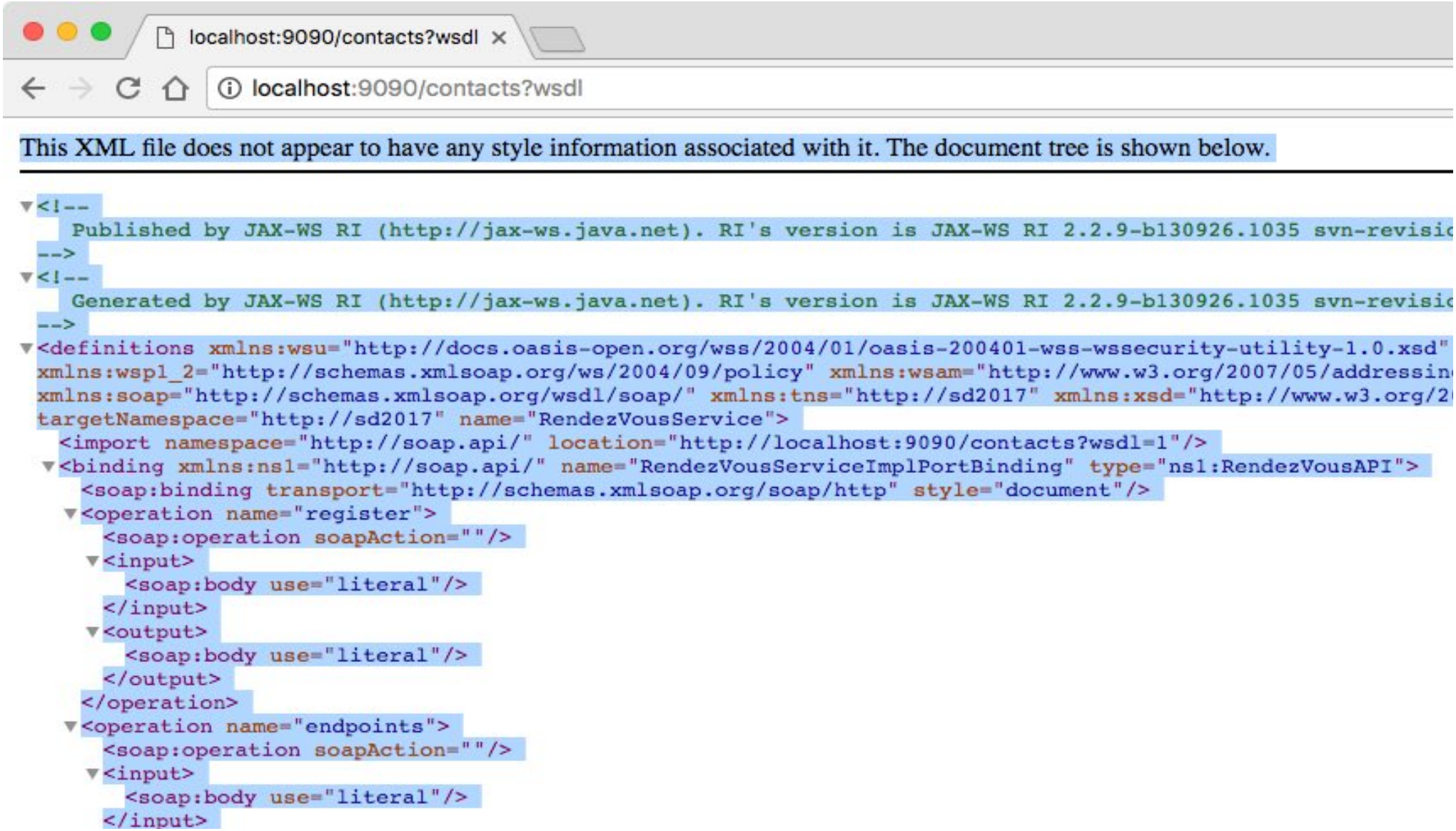
EXEMPLO: SERVIDOR (2)

```
public class RendezVousServer {
   public static void main(String[] args) throws Exception {
        int port = 9090;
        if (args.length > 0)
                port = Integer.parseInt(args[0]);

        String baseURI = String.format("http://0.0.0.0:%d/contacts", port);

        javax.xml.ws.Endpoint.publish(baseURI, new RendezVousServiceImpl());

        System.err.println("SOAP RendezVous Server ready @ " + baseURI);
   }
}
```

EXEMPLO: CLIENTE

```
public class ListEndpoints {
    static final QName QNAME = new QName(RendezVousAPI.NAMESPACE, RendezVousAPI.NAME);

    public static void main(String[] args) throws Exception {
        URL wsURL = new URL("http://localhost:9090/contacts?wsdl");
        Service service = Service.create( wsURL, QNAME);
        RendezVousAPI contacts = service.getPort( RendezVousAPI.class );
        System.out.println( Arrays.asList(contacts.endpoints() ) );
    }
}
```

RENDEZVOUS WSDL



EXERCÍCIO

No contexto do trabalho, o serviço de indexação terá uma versão suportada por webservices SOAP.

Baseando-se na implementação SOAP do serviço RendezVous, crie uma versão SOAP do seu serviço de Indexação.