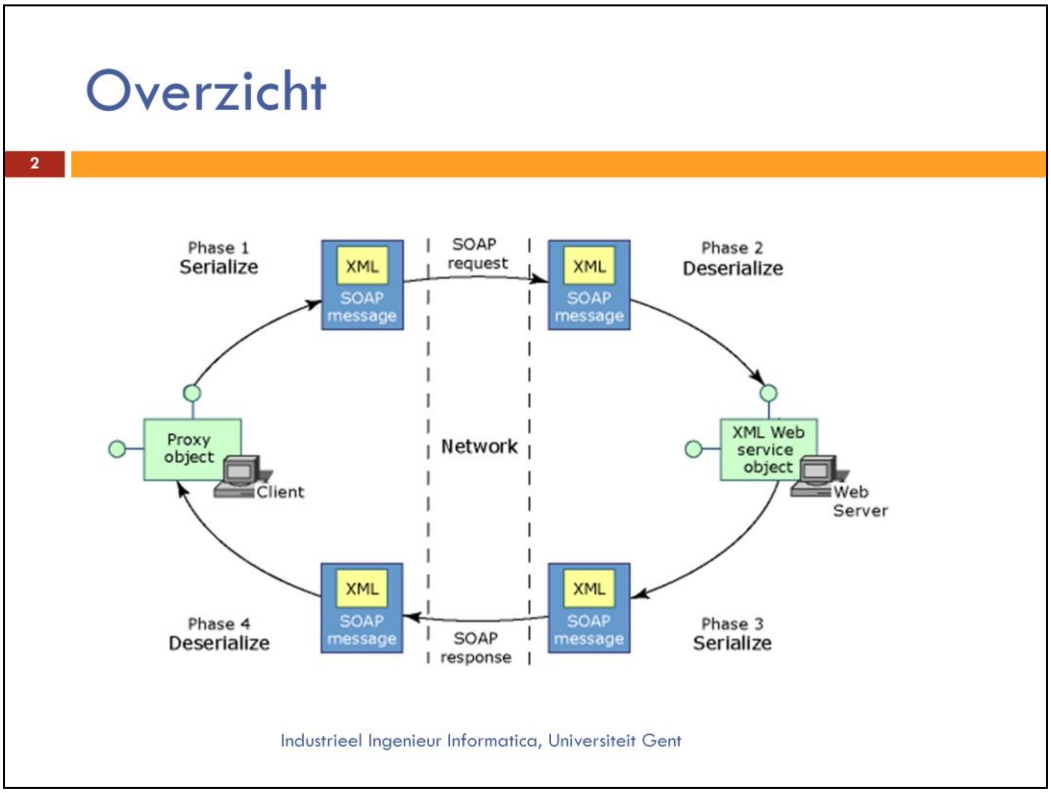
WEBSERVICES IN JAVA

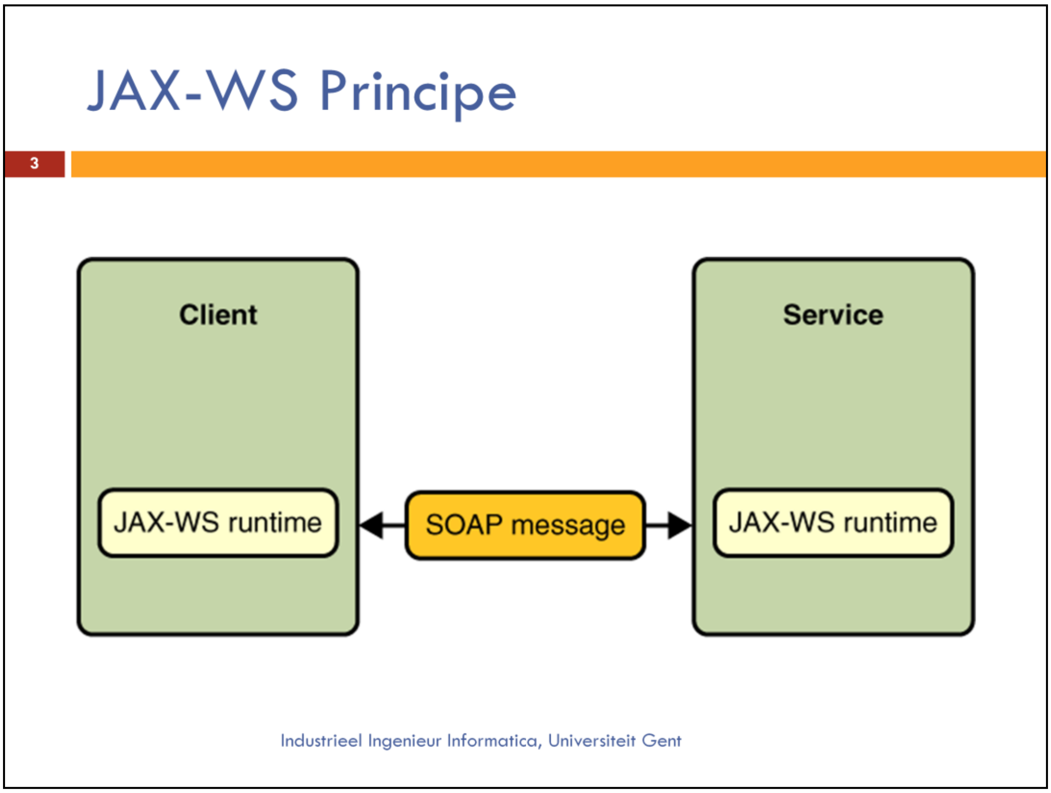
Veerle Ongenae



Overzicht

2

Industrieel Ingenieur Informatica, Universiteit Gent



JAX-WS Principe

3

Industrieel Ingenieur Informatica, Universiteit Gent

Java API for XML Web Services (JAX-WS) is a technology for building web services and clients that communicate using XML. JAX-WS allows developers to write message-oriented as well as Remote Procedure Call–oriented (RPC-oriented) web services.

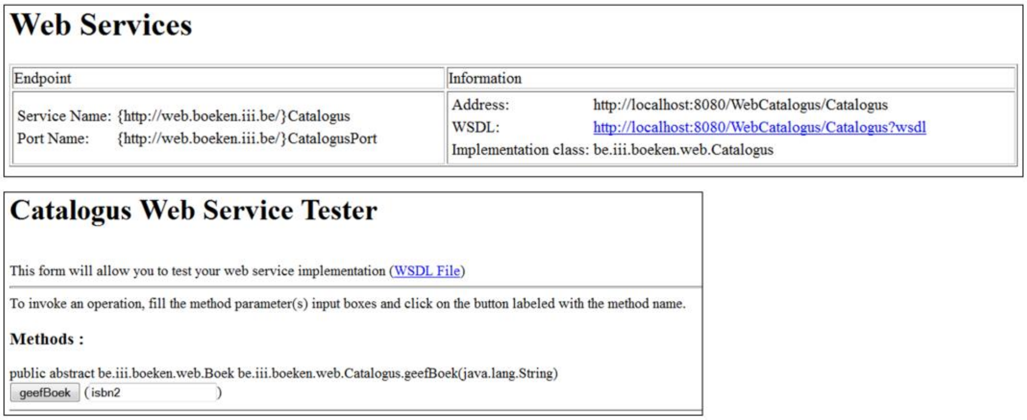
In JAX-WS, a web service operation invocation is represented by an XML-based protocol, such as SOAP. The SOAP specification defines the envelope structure, encoding rules, and conventions for representing web service invocations and responses. These calls and responses are transmitted as SOAP messages (XML files) over HTTP.

Although SOAP messages are complex, the JAX-WS API hides this complexity from the application developer. On the server side, the developer specifies the web service operations by defining methods in an interface written in the Java programming language. The developer also codes one or more classes that implement those methods. Client programs are also easy to code. A client creates a proxy (a local object representing the service) and then simply invokes methods on the proxy. With JAX-WS, the developer does not generate or parse SOAP messages. It is the JAX-WS runtime system that converts the API calls and responses to and from SOAP messages.

With JAX-WS, clients and web services have a big advantage: the platform independence of the Java programming language. In addition, JAX-WS is not restrictive: A JAX-WS client can access a web service that is not running on the Java platform and vice versa. This flexibility is possible because JAX-WS uses technologies defined by the W3C: HTTP, SOAP, and WSDL. WSDL specifies an XML format for describing a service as a set of endpoints operating on messages.

Voorbeeld

- □ Boek-object opvragen aan de hand van ISBN



Web Services

Endpoint	Information
Service Name: {http://web.boeken.iii.be/}Catalogus Port Name: {http://web.boeken.iii.be/}CatalogusPort	Address: http://localhost:8080/WebCatalogus/Catalogus WSDL: http://localhost:8080/WebCatalogus/Catalogus?wsdl Implementation class: be.iii.boeken.web.Catalogus

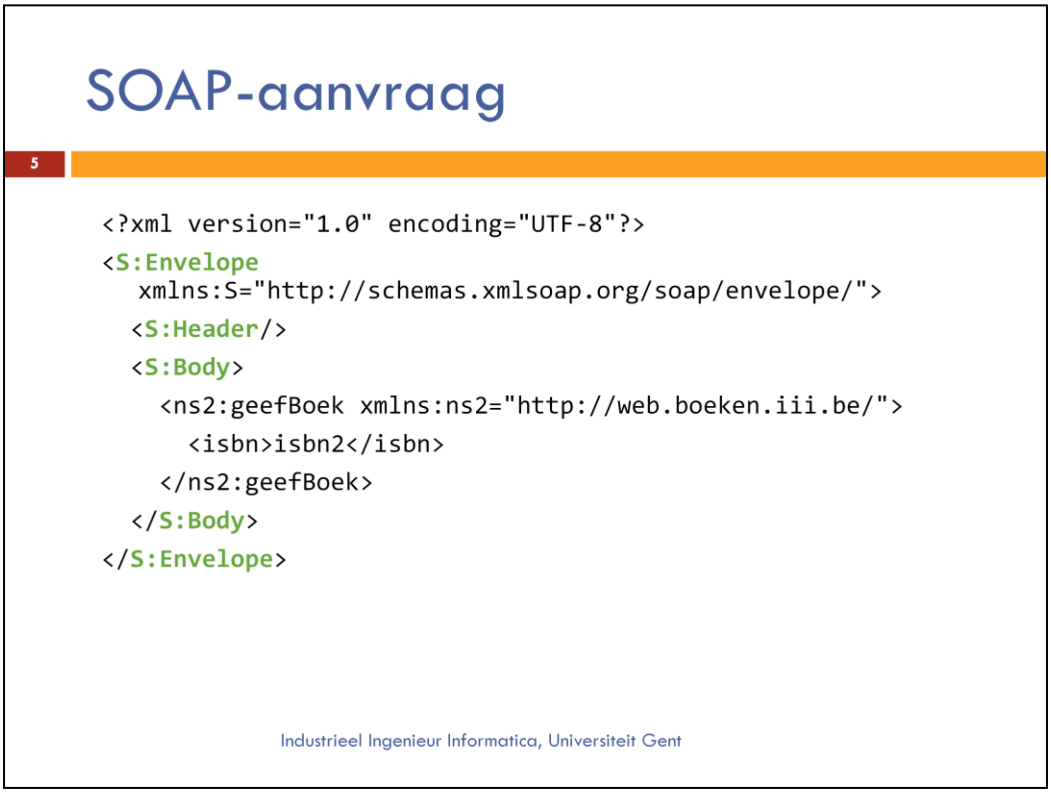
Catalogus Web Service Tester

This form will allow you to test your web service implementation (WSDL File)

To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.

Methods :

public abstract be.iii.boeken.web.Boek be.iii.boeken.web.Catalogus.geefBoek(java.lang.String)



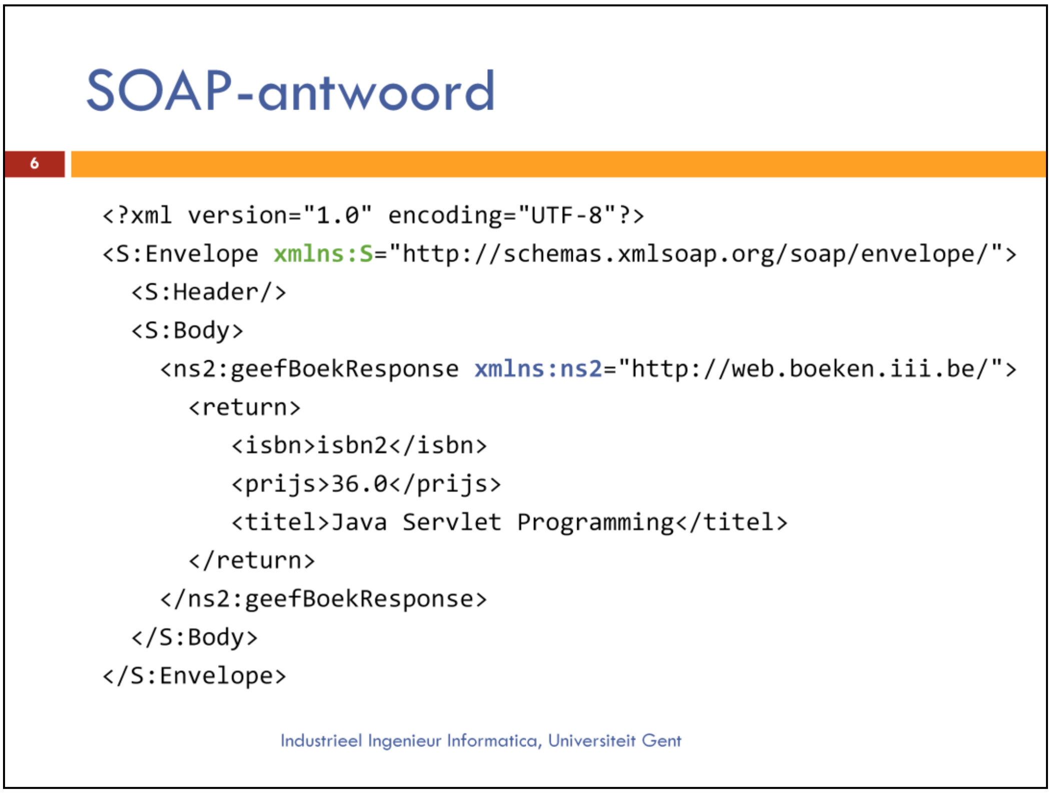
SOAP-aanvraag

5

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
   xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns2:geefBoek xmlns:ns2="http://web.boeken.iii.be/">
      <isbn>isbn2</isbn>
    </ns2:geefBoek>
  </S:Body>
</S:Envelope>
```

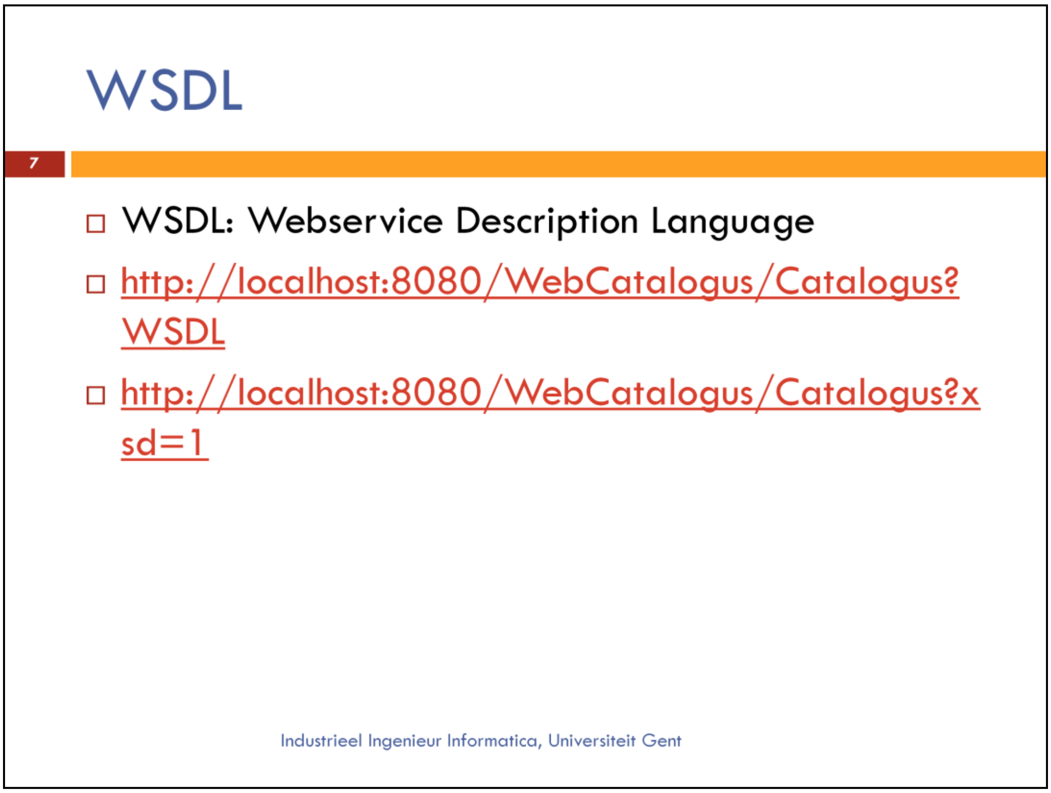
Industrieel Ingenieur Informatica, Universiteit Gent

SOAP 1.1



SOAP-antwoord

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns2:geefBoekResponse xmlns:ns2="http://web.boeken.iii.be/">
      <return>
         <isbn>isbn2</isbn>
         <prijs>36.0</prijs>
         <titel>Java Servlet Programming</titel>
      </return>
    </ns2:geefBoekResponse>
  </S:Body>
</S:Envelope>
```



WSDL

7

- WSDL: Webservice Description Language
- http://localhost:8080/WebCatalogus/Catalogus?WSDL
- http://localhost:8080/WebCatalogus/Catalogus?xsd=1

Industrieel Ingenieur Informatica, Universiteit Gent

WSDL 1.1

Informatie over webservice → genereren proxy-object

De structuur van de SOAP-berichten

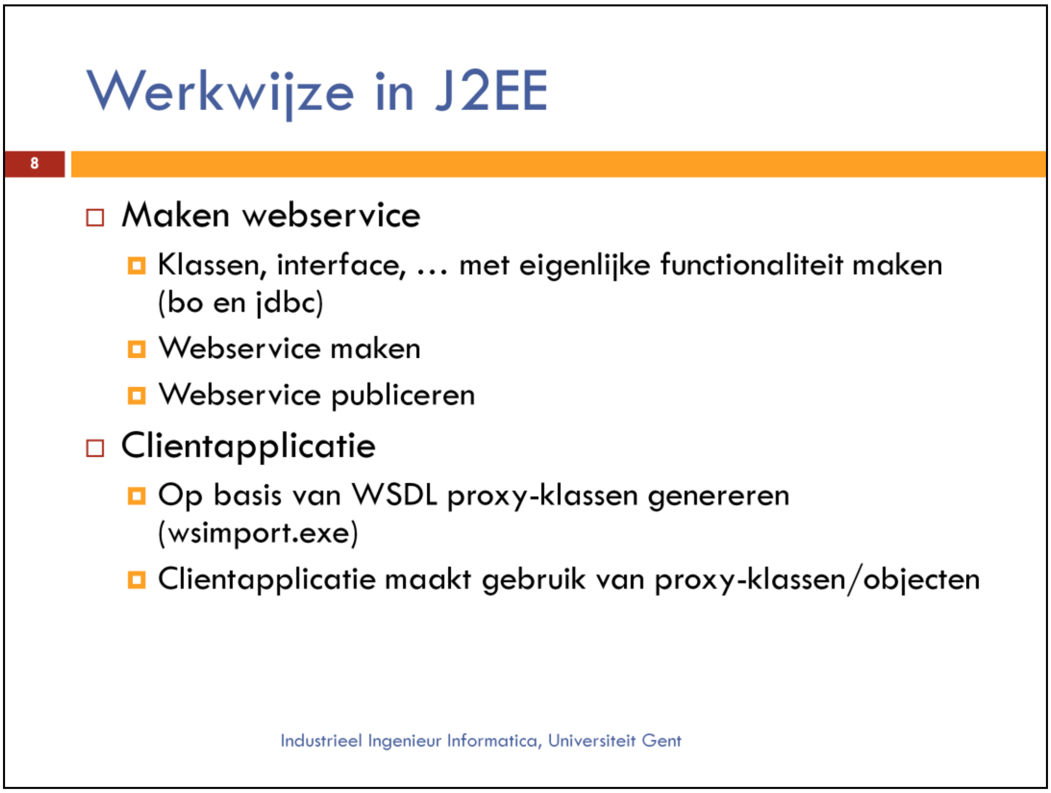
Methodes

Argumenten

Teruggeefwaarden

Locatie webservices

Transportprotocol



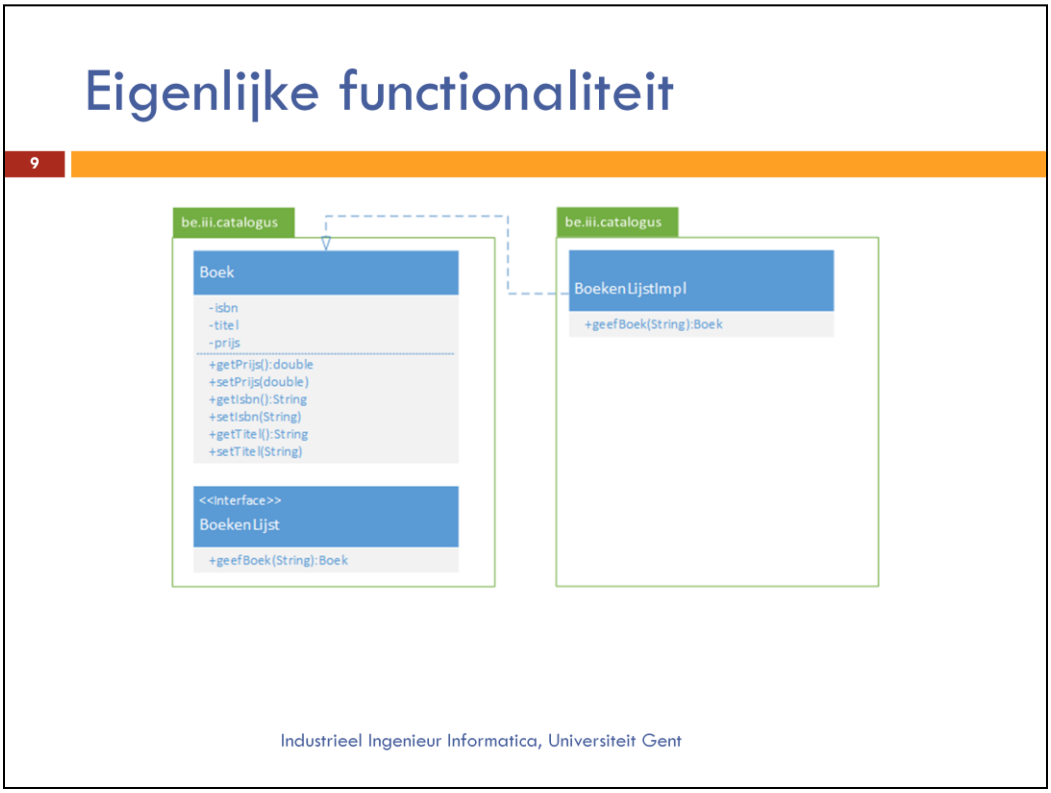
Werkwijze in J2EE

8

- □ Maken webservice
  - Klassen, interface, … met eigenlijke functionaliteit maken (bo en jdbc)
  - Webservice maken
  - Webservice publiceren
- □ Clientapplicatie
  - Op basis van WSDL proxy-klassen genereren (wsimport.exe)
  - Clientapplicatie maakt gebruik van proxy-klassen/objecten

Industrieel Ingenieur Informatica, Universiteit Gent

J2EE = Java 2 Entreprise Edition



Eigenlijke functionaliteit

9

Industrieel Ingenieur Informatica, Universiteit Gent

be.iii.catalogus.Boek

Moet geserialiseerd worden

Default-constructor

get/set-methodes voor id, titel en prijs □ anders NIET in XML

“eenvoudige” types

be.iii.catalogus.BoekenLijst

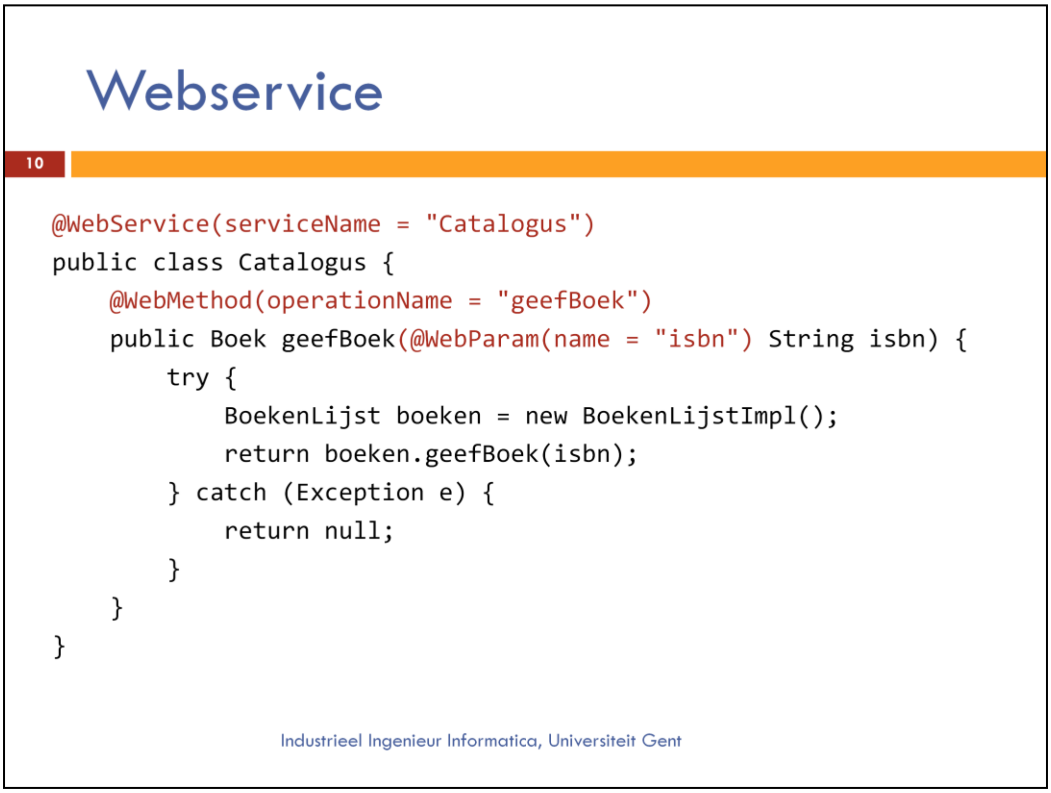
Interface

Methode geefBoek

be.iii.catalogus.impl.BoekenLijstImpl

Implementatie interface BoekenLijst

In jar catalogus.jar



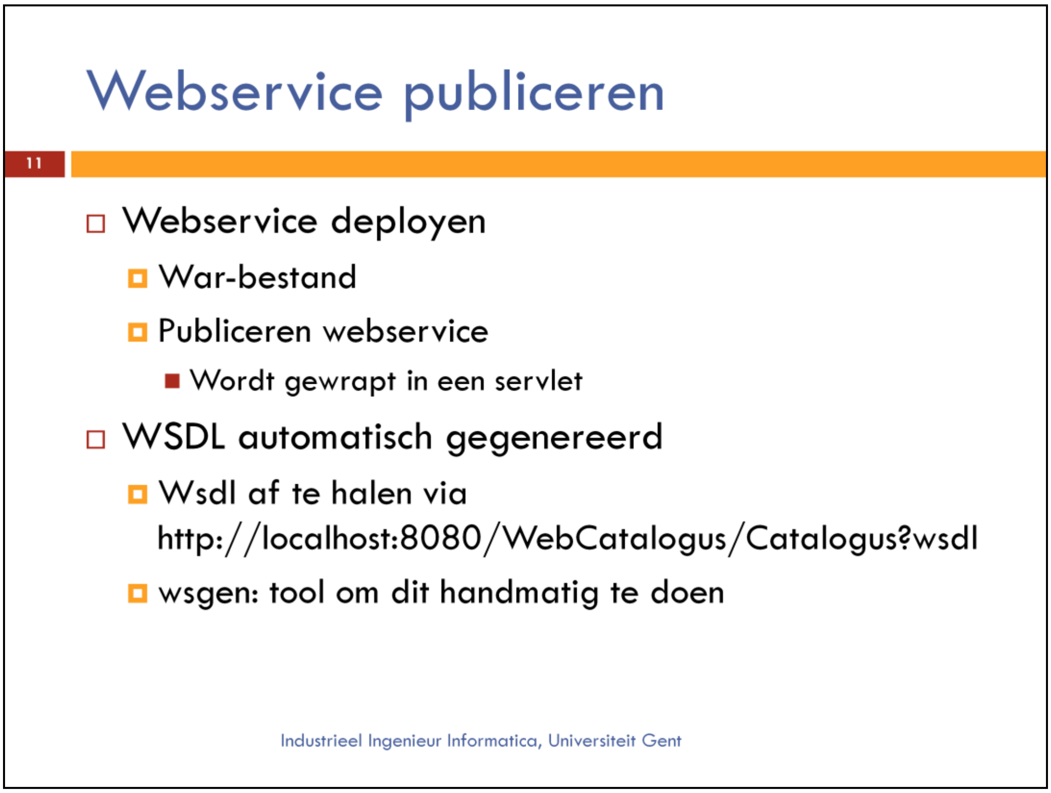
Webservice

```
@WebService(serviceName = "Catalogus")
public class Catalogus {
    @WebMethod(operationName = "geefBoek")
    public Boek geefBoek(@WebParam(name = "isbn") String isbn) {
        try {
            BoekenLijst boeken = new BoekenLijstImpl();
            return boeken.geefBoek(isbn);
        } catch (Exception e) {
            return null;
        }
    }
}
```

Klasse met annotatie @Webservice()

Methode met annotatie @WebMethod

Parameters met annotatie @WebParam(…)

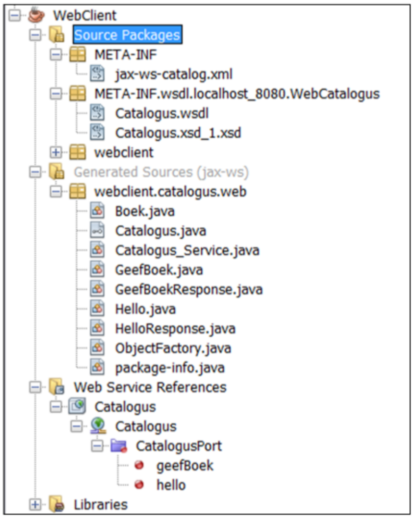


Webservice publiceren

- Webservice deployen
  - War-bestand
  - Publiceren webservice
    - Wordt gewrapt in een servlet
- WSDL automatisch gegenereerd
  - Wsdl af te halen via http://localhost:8080/WebCatalogus/Catalogus?wsdl
  - wsgen: tool om dit handmatig te doen

Genereren proxy-klassen

12



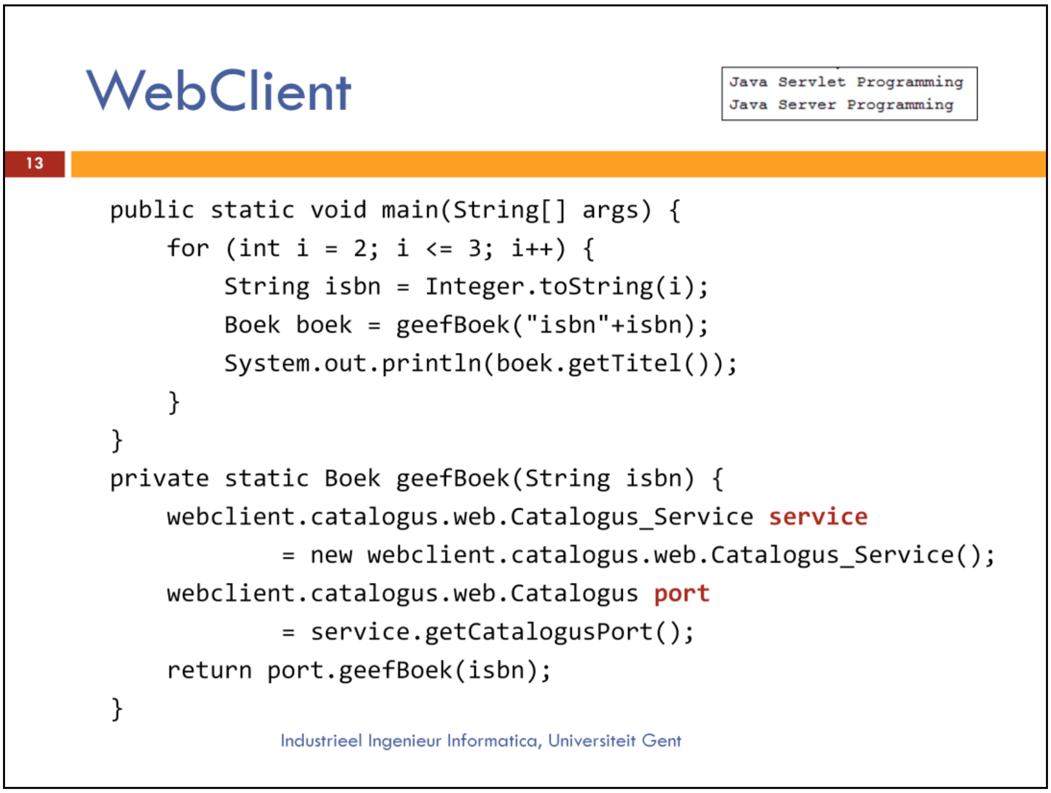
```
WebClient
  Source Packages
    META-INF
      jax-ws-catalog.xml
    META-INF.wsdl.localhost_8080.WebCatalogus
      Catalogus.wsdl
      Catalogus.xsd_1.xsd
    webclient
  Generated Sources (jax-ws)
    webclient.catalogus.web
      Boek.java
      Catalogus.java
      Catalogus_Service.java
      GeefBoek.java
      GeefBoekResponse.java
      Hello.java
      HelloResponse.java
      ObjectFactory.java
      package-info.java
  Web Service References
    Catalogus
      Catalogus
        CatalogusPort
          geefBoek
          hello
  Libraries
```

Industrieel Ingenieur Informatica, Universiteit Gent

Automatisch in Netbeans in een project
- New → Web Service Client
- Url van de WSDL doorgeven

Gegeneerd
- Package META-INF
  - jax-ws-catalog.xml
    - Locatie WSDL
- META-INF.wsdl.localhost_8080.WebCatalogus
  - Catalogus.wsdl
  - Catalogus.xsd_1.xsd
- Klassen voor proxy
  - webclient.catalogus.web

In Projectsvenster
- Map Web Service References
  - Catalogus



WebClient

Java Servlet Programming
Java Server Programming

13

```
public static void main(String[] args) {
    for (int i = 2; i <= 3; i++) {
        String isbn = Integer.toString(i);
        Boek boek = geefBoek("isbn"+isbn);
        System.out.println(boek.getTitel());
    }
}
private static Boek geefBoek(String isbn) {
    webclient.catalogus.web.Catalogus_Service service
            = new webclient.catalogus.web.Catalogus_Service();
    webclient.catalogus.web.Catalogus port
            = service.getCatalogusPort();
    return port.geefBoek(isbn);
}
```

Industrieel Ingenieur Informatica, Universiteit Gent

Project WebClient

webclient.WebClient.java

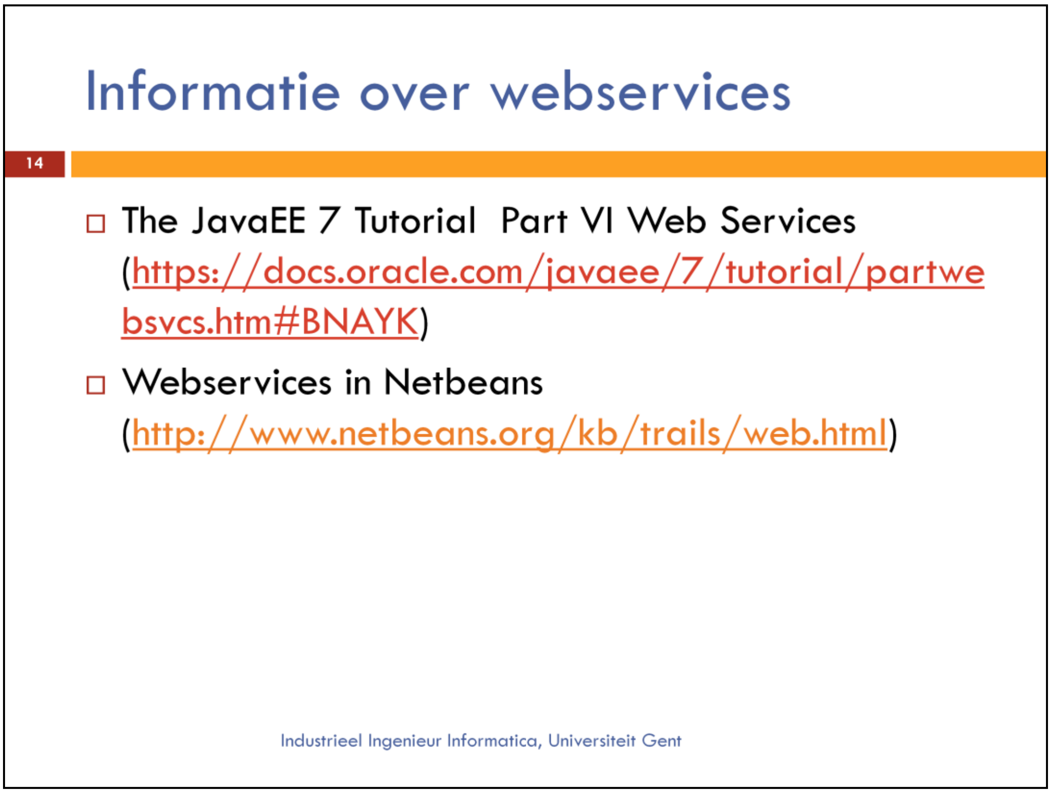
Oproepen methode webservice

Rechts klikken in editeervenster

Insert Code

Call Web Service Operation

Genereert de nodige code om de proxyobjecten te gebruiken



Informatie over webservices

- □ The JavaEE 7 Tutorial Part VI Web Services (https://docs.oracle.com/javaee/7/tutorial/partwebsvcs.htm#BNAYK)
- □ Webservices in Netbeans (http://www.netbeans.org/kb/trails/web.html)