

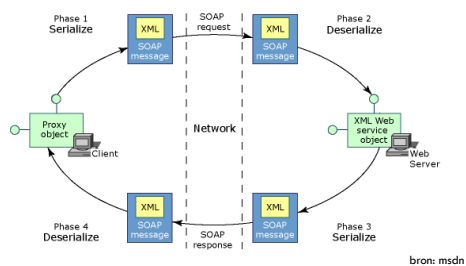
WEBSERVICES IN JAVA

Veerle Ongenae

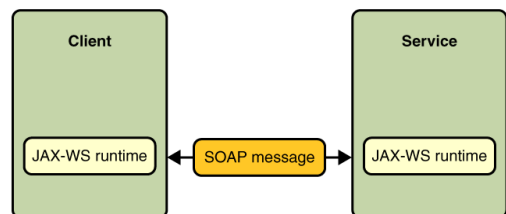
Informatie over webservices

- The JavaEE 8 Tutorial Part VI Web Services (<https://javaee.github.io/tutorial/partwebsvcs.html>)
- Webservices in Netbeans (<http://www.netbeans.org/kb/trails/web.html>)

Overzicht



JAX-WS Principe



Voorbeeld

- Boek-object opvragen aan de hand van ISBN

Web Services

Endpoint	Information
Service Name: [http://web.boeken.iii.be/]Catalogus	Address: http://localhost:8080/WebCatalogus/Catalogus
Port Name: [http://web.boeken.iii.be/]CatalogusPort	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 isbn);
```

<http://localhost:8080/WebCatalogus/CatalogusTester>

SOAP-aanvraag

```
<?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>
```

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>
```

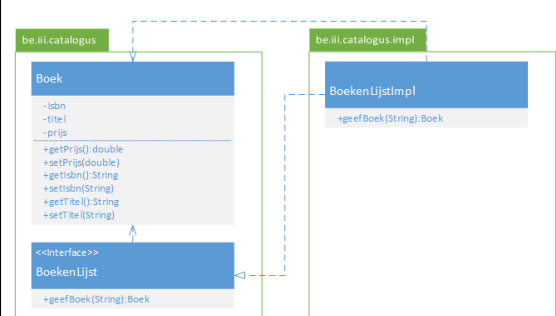
WSDL

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

Werkwijze in J2EE

- 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

Eigenlijke functionaliteit



Webservice

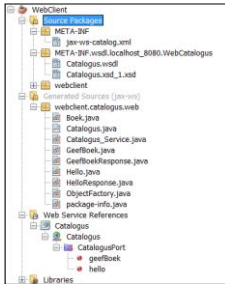
```
@WebService(serviceName = "Catalogus")
public class Catalogus {
    @WebMethod(operationName = "geefBoek")
    public Boek geefBoek(@WebParam(name = "isbn") String isbn) {
        try {
            BoekenLijst boeken = new BoekenLijstImpl();
            return boeken.geefBoek(isbn);
        } catch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

Webservice publiceren

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

Genereren proxy-klassen

13



WebClient

 Java Servlet Programming
 Java Server Programming

14

```

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);
}

```

WebClientServlet

15

```

import be.iii.boeken.web.Catalogus_Service;
...
import javax.xml.ws.WebServiceRef;

@WebServlet(urlPatterns = {"/BoekServlet"})
public class BoekServlet extends HttpServlet {

    @WebServiceRef(wsdlLocation =
        "http://localhost:8080/WebCatalogus/Catalogus?WSDL")
    Catalogus_Service service;

    protected void processRequest(...) throws ... {
        ...
        Catalogus port = service.getCatalogusPort();
        Boek boek = port.geefBoek(isbn); ...}

```