

WebServices -reloaded-

Jan Krüger

Bielefeld Bioinformatics Service
Institute of Bioinformatics
CeBiTec Bielefeld University

jkrueger@techfak.uni-bielefeld.de

3 Juli 2007

Übersicht

- Motivation
 - Was sind WebServices ?
 - Grundlegendes zu WebServices
 - WebServices am BiBiServ
- WebServices
 - Axis 1.x
- WebServices – reloaded
 - JWSDP 2.x
 - JAX-WS
 - JAXB

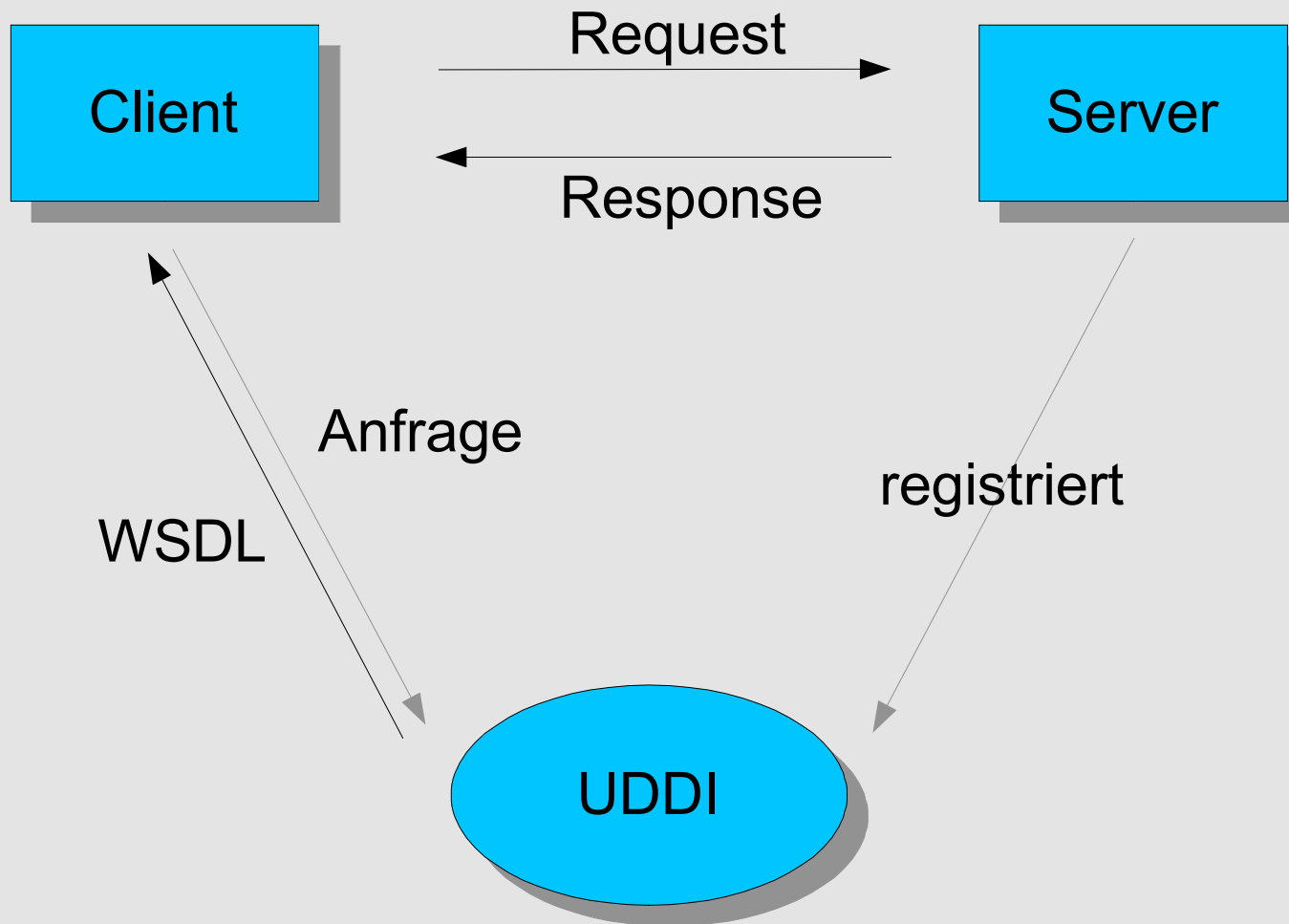
Was sind Webservice ?

Ein Webservice ist ein Stückchen (Programm-) Logik, das sich irgendwo im Internet befindet und auf das man mit standardisierten Protokollen zugreifen kann.

WebService Eigenschaften

- XML basiert
- Lose gekoppelt
- Grobkörnig
- Synchron oder Asynchron
- RPC- oder Document-Styled

WebService Technologien



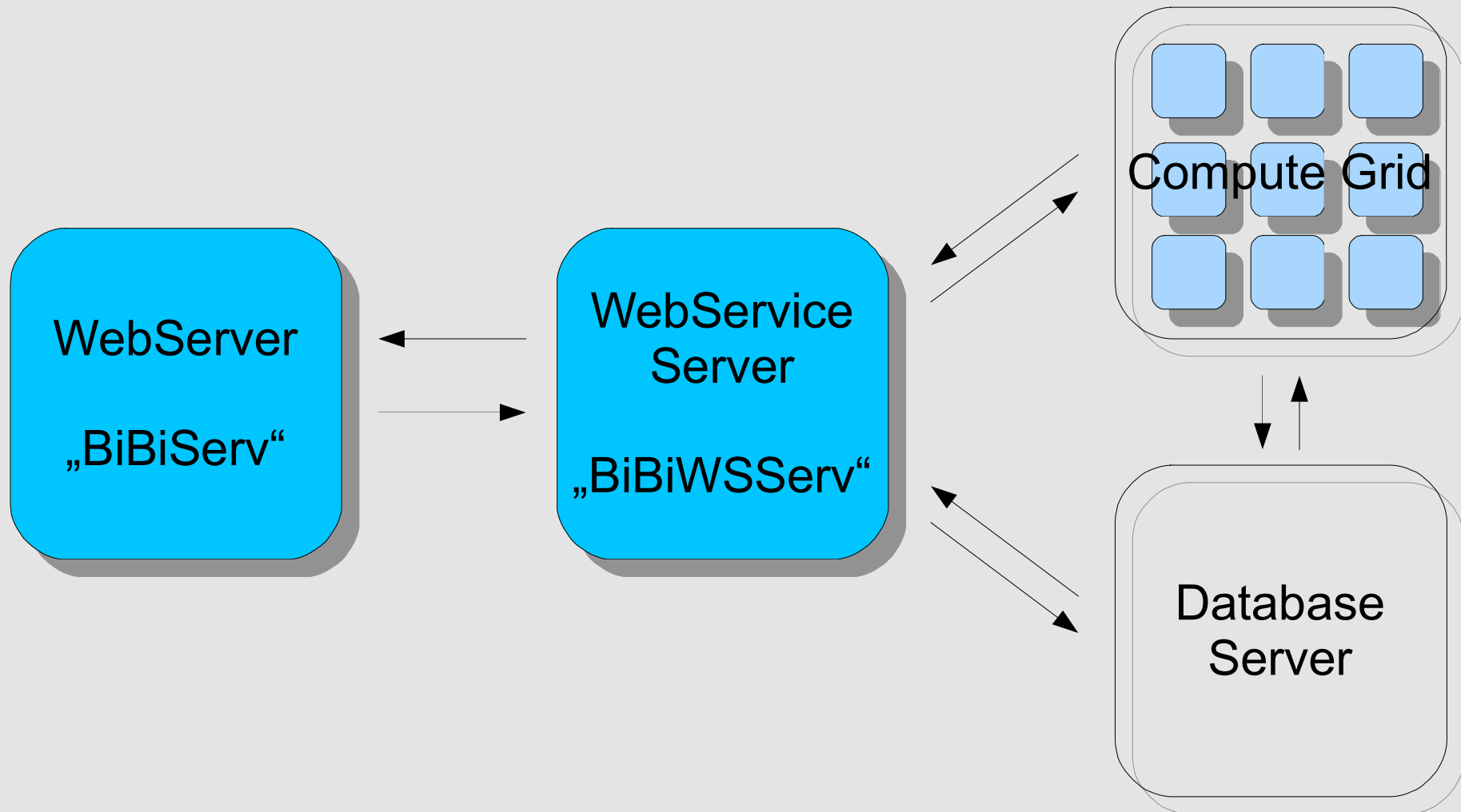
WebService Technologien

DEMO 1

Asynchrone Ansätze

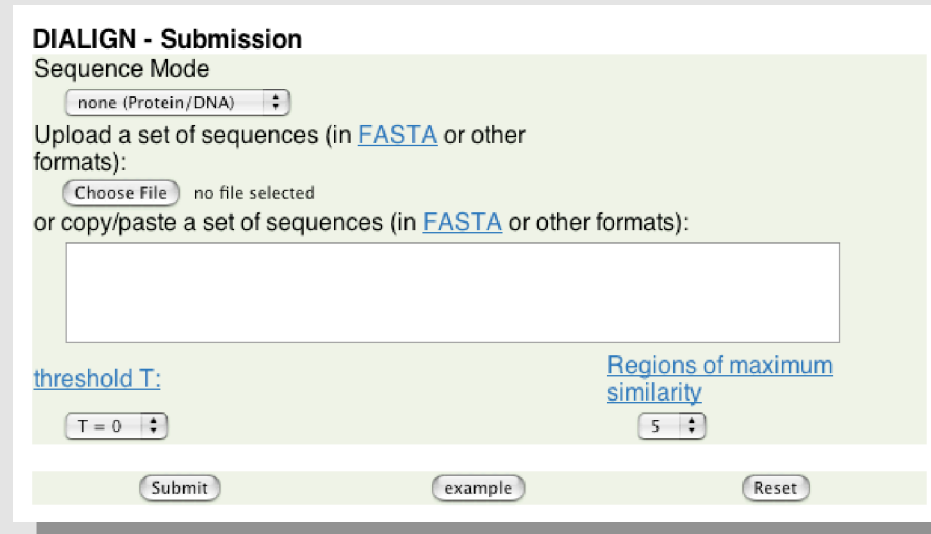
- One-Way and notification operations
- Request/Reply operation
- Request/Reply with polling
- Request/Reply with posting

WebServices am BiBiServ



Daten in der BioInformatik

- einfache Daten
 - Parameter
 - eingeschränkt
 - optional
- komplexe Daten
 - Sequenzdaten (z.B. FASTA)
 - DNA/RNA, AMINOACID, FREE
 - Metainformation
 - Strukturdaten (z.B. DotBracket)



The screenshot shows the 'DIALIGN - Submission' web form. It includes a 'Sequence Mode' dropdown menu set to 'none (Protein/DNA)'. Below this, there are instructions to 'Upload a set of sequences (in FASTA or other formats):' with a 'Choose File' button and 'no file selected' text. There is also a text area for 'or copy/paste a set of sequences (in FASTA or other formats):'. At the bottom, there are two dropdown menus: 'threshold T:' set to '0' and 'Regions of maximum similarity' set to '5'. The form has 'Submit', 'example', and 'Reset' buttons.

→ **Problem der Spezifikation und Validierung**

Ansätze

- Programm (-gerüst) aus WSDL
 - schnell und einfach
 - Programmlogik wird hinzugefügt
 - Bei größeren WSDL Beschreibung schnell unübersichtlich
 - Unwartbar, wenn das Programm in der Entwicklung ist!
- WSDL von dem Programm
 - Aufwendiger, aber einfacher zu verstehen
 - Komplexe Datentypen sind oftmals schwierig zu implementieren

WebService Frameworks (1st Generation)

Sprache	Name
Java	SOAP[4J] (IBM) Apache Axis 1.x JWS DP 1.x (SUN)
C/C++	GSOAP Apache Axis 1.x
C#, VB,NET 1.x und 2.x
Perl	SOAP::Lite

BiBiWS und Apache Axis 1.x

- WSDD (WebService Deployment Descriptor)
- Properties
- Serializer / Deserializer für einfache Datentypen
- Interfaces / Classes / Exception

—————▶ **DEMO 2**

Warum was Neues ?

- WSDL wird nicht zu 100% unterstützt
- eigene Datentypen
- besch... Dokumentation
- Speicherlecks
- Qualität des Sourcecodes
-

WebService Frameworks (2nd Generation)

Sprache	Name
Java	Apache Axis 2.x JWSDP 2.x (SUN) XFire
C#,VB,...	.NET 3.x

Exkurs : Java Annotations

- Bis Java 5 : Interfaces, Properties
- Ab Java 5 : Annotations

- @Annotation
- @Annotationtype

```
@WebService
public class Calculator
{
    @WebMethod
    public int add( int x, int y ) {
        return x + y;
    }
}
```

- Typisch : XML-Mapping, WebServices
- Seit Java 6 : Commons Annotation

JWSDP 2.x

- **Java Architecture for XML Web Services (JAX-WS)**
- **Java Architecture for XML Binding (JAXB)**
- Streaming API for XML (StAX)
- SOAP with Attachments API for Java (SAAJ)
- XML Registries
- XML Digital Signature
- Security in the Web Tier

JAX-WS - Beispiel

1. Klasse implementieren

2. Kompilieren

3. WSGen aufrufen

4. WAR bauen

5. WAR deployen

```
@WebService()  
public class Calculator  
{  
    @WebMethod()  
    public int add( int x, int y ) {  
        return x + y;  
    }  
}
```

JAX-WS - @WebService

Name	Type	Default
name	String	Name der Klasse / des Interface
targetNamespace	String	Name des Package
serviceName	String	„WebService.name“ + Service
endpointInterface	String	
portName	String	„WebService.name“ + Port
wsdlLocation	String	

```
@WebService(name="AddNumbers",  
            targetNameSpace="de:unibi:techfak:bibiserv:DEMO",  
            serviceName="AddNumbersService")  
public class Calculator  
{  
    @WebMethod()  
    public int add( int x, int y ) {  
        return x + y;  
    }  
}
```

JAX-WS - @WebMethod

Name	Type	Default
operationName	String	Name der Java Methode
action	String	“”
exclude	Boolean	FALSE

```
@WebService()  
public class Calculator  
{  
    @WebMethod(operationName="add", action="urn:addNumbers")  
    public int add( int x, int y ) {  
        return x + y;  
    }  
}
```

JAX-WS - @WebResult

Name	Type	Default
name	String	result
targetNamespace	String	NS of WebService
header	Boolean	FALSE
partName	String	@WebResult.name

```
@WebService()  
public class Calculator  
{  
    @WebMethod()  
    @WebResult(name="result")  
    public int add( int x, int y ) {  
        return x + y;  
    }  
}
```

JAX-WS - @WebParam

Name	Type	Default
name	String	Name der Java Methode
targetNamespace	String	“
mode	String	IN bzw. INOUT
header	boolean	FALSE
partName	String	@WebParam.name

```
@WebService()  
public class Calculator  
{  
    @WebMethod()  
    public int add(@WebParam(name="x")int x,  
                  @WebParam(name="y")int y ) {  
        return x + y;  
    }  
}
```

JAX-WS - Exception

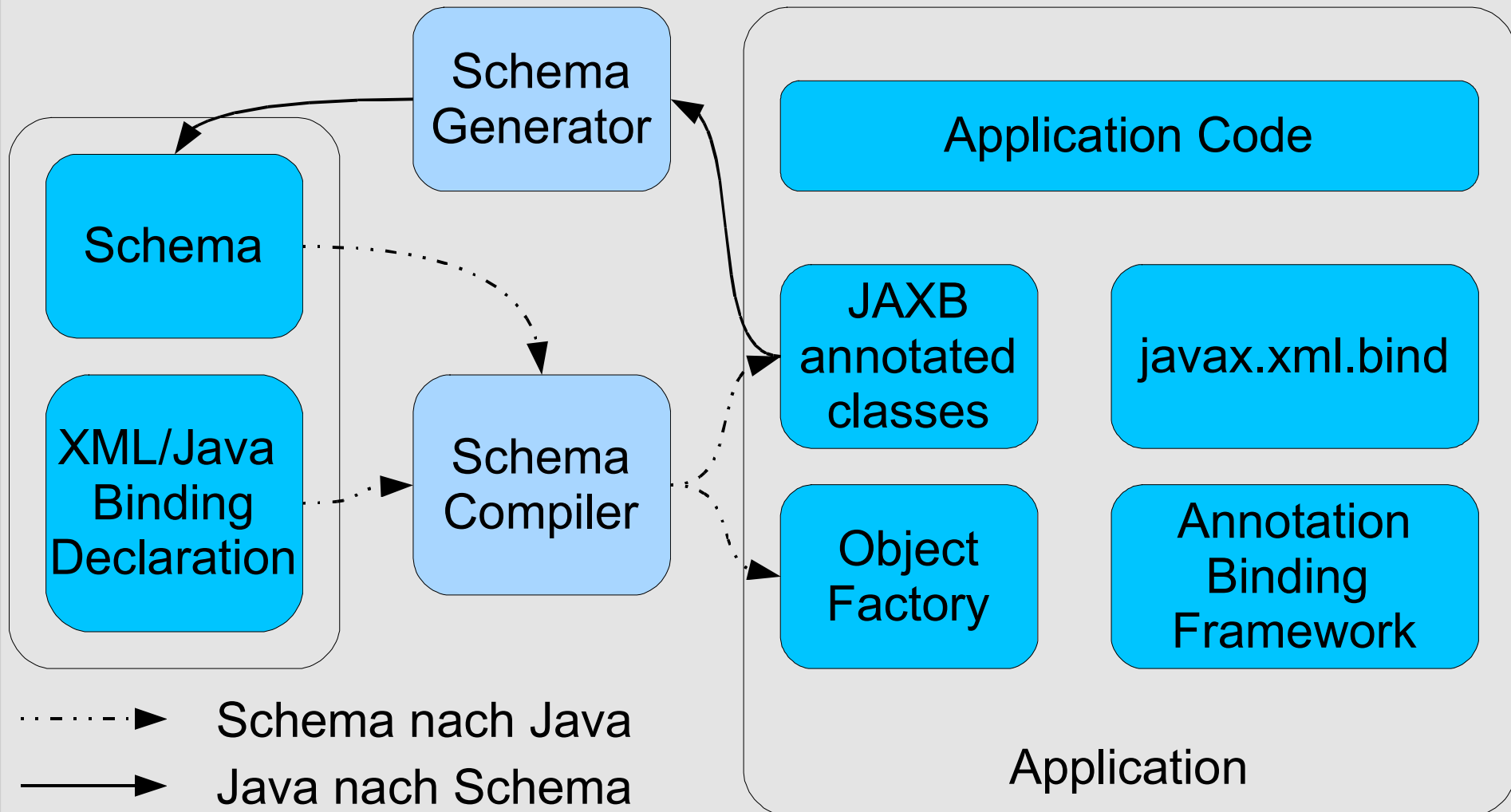
```
public class MyException extends Exception {  
    private String message;  
    public MyException(String message) {  
        this.message = message;  
    }  
    public String getMessage() {  
        return message;  
    }  
}
```

```
@WebService()  
public class Calculator {  
    @WebMethod()  
    public int add( int x, int y ) throws MyException {  
        if ( x < 0 || y < 0 ) {  
            throws new MyException („Negative numbers are not supported!“);  
        }  
        return x + y;  
    }  
}
```

JAXB : DataBinding

XML Schema Type	Java Data Type
xsd:string	java.lang.String
xsd:integer	java.math.BigInteger
xsd:int	java.lang.Integer bzw. int
xsd:long	java.lang.Long bzw. long
xsd:float	java.lang.Float bzw. float
xsd:double	java.lang.Double bzw. double
xsd:boolean	java.lang.Boolean bzw. boolean
xsd:QName	javax.xml.namespace.QName
xsd:anySimpleType	java.lang.Object
xsd:anySimpleType	java.lang.String

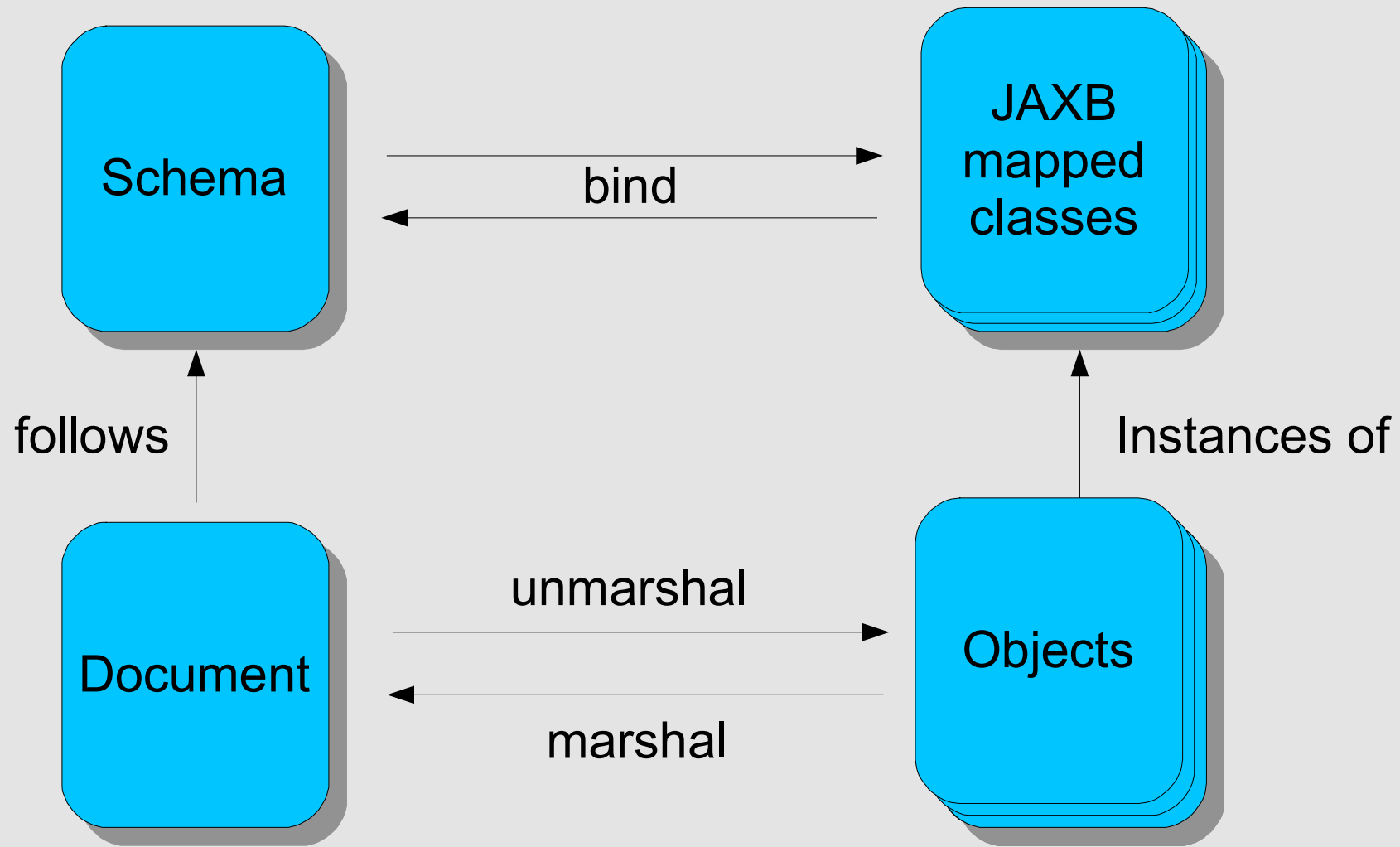
JAXB - Übersicht



JAXB - Fallstricke

- Schema nach Java
 - Constraints werden nicht automatisch in Java Klassen übertragen
 - Strukturen können defacto abgebildet werden, automatisiert klappt das jedoch i.d.R. Nicht
 - Beispiel : `biotypes:aminoacidsequence`
 -
- Java nach Schema
 - Ohne (JAXB-) Annotationen oft zu allgemeine XML Schema Datentypen

JAXB Binding Process



JWSDP - Demo

DEMO 3

Zusammenfassung

- Aus Sicht der Entwickler:
 - Hürde für den Einstieg ist größer
 - XML Schema
 - Java Annotations
 - Unterstützende Software
 - Netbeans (ab 5.x) / Eclipse (ab 3.1x)
 - Oxygen / XMLSpy
 - Übersichtlicher = weniger Fehlerquellen!
- Aus Sicht der Administratoren:
 - Sehr gut dokumentiert (Tutorial, API, Blogs)
 - S.O.