

Java, XML und Servlets zur Integration
datenbankbasierter Applikationen im Web

Albrecht Schmidt
TU München

Überblick

- digitale Bibliotheken (OMNIS)
- (Multi, Hyper)-media-Datenbanken (MultiMAP)
- unser Ansatz
- unsere Implementierung

Digitale Bibliotheken – OMNIS

- digitale Dokumente
- gescannte Dokumente
- Suchen auf den Dokumenten
- Bestellung/Auslieferung
- Bezahlung
- Legacy-Applikationen

Digitale Bibliotheken aus Benutzersicht

- Suchen
- Bestellen
- Abholen

Digitale Bibliotheken aus Bibliothekssicht

- Bestellungen abarbeiten
- Dokumente archivieren

(Multi, Hyper)-media-Datenbanken – MultiMAP

- Variante des graphentheoretischen Dexter-Modells
- Komponenten
- verschiedene Medientypen (Text, Bild, Ton, Video, ...)
- (getypte) Links zwischen den Komponenten
- Linkintegrität
- Erweiterbarkeit mittels Anmerkungen

Kombination und Erweiterung von digitalen Bibliotheken und multimedialen Datenbanken

- verschiedene Medientypen
- Links zwischen elektronischen Dokumenten
- Integrität der Links
- Suchen
- Personalisierung
- Autorensystem

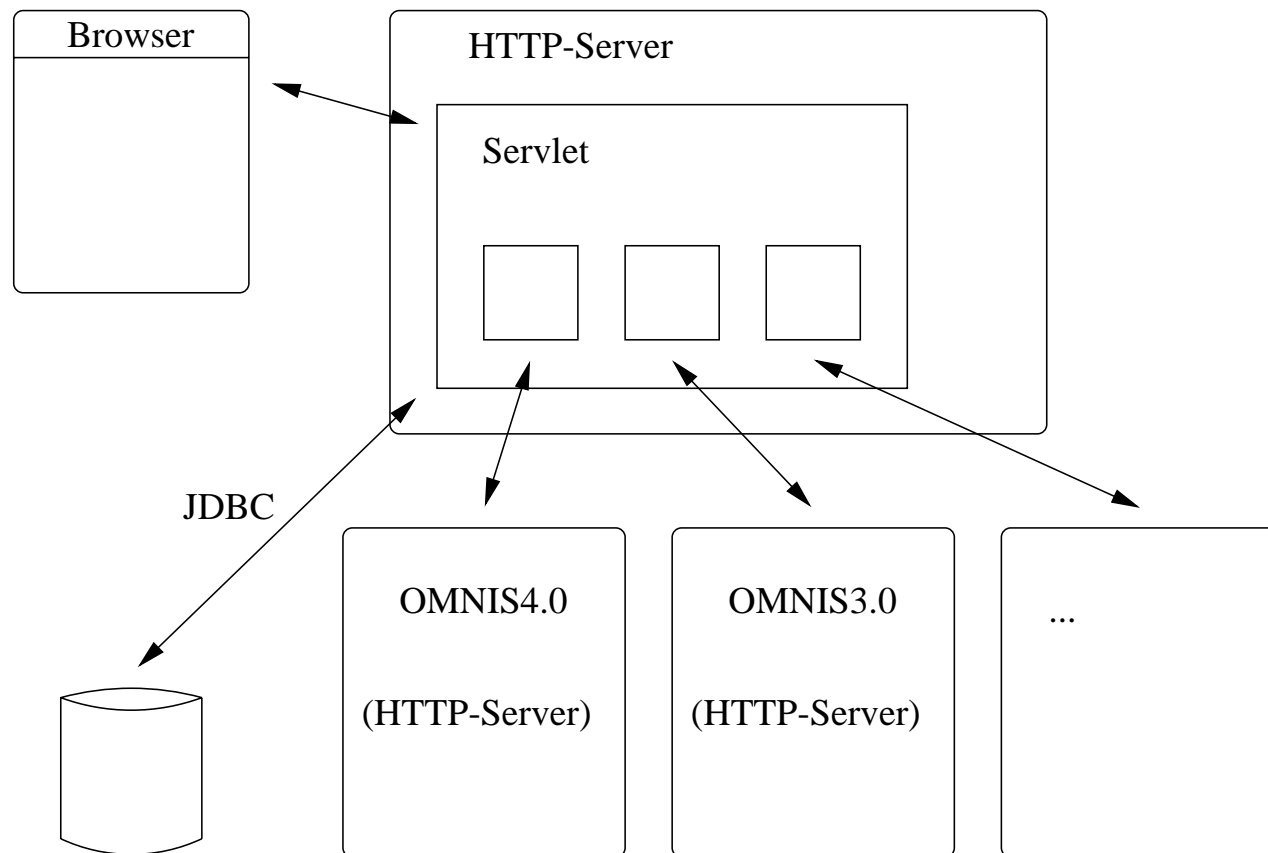
Technische Herausforderungen

- Integration von Legacy-Applikationen und deren Datenbanken
- Kommunikation zwischen verschiedenen Servern
- Parallelismus

Schlüssel-Technologien

- XML
- Java
- Servlets
- JDBC

Die Architektur von OMNIS/2



Kommunikation

- Infobus
- HTTP-Server geben HTML-Text aus
- Übergang von HTML zu XML
- Server-Parsed-HTML-Templates werden geändert
- bestehende und sich herausbildende XML-Standards

Volltext-Suche: HTML → XML

```
<A HREF="URL">
```

```
  A. W. Riter, Java -- How to Program
```

```
</A>
```

... wird zu ...

```
<BOOK>
```

```
  <URL>URL</URL>
```

```
  <AUTHOR>A. W. Riter</AUTHOR>
```

```
  <TITLE>Java -- How to Program</TITLE>
```

```
</BOOK>
```

Parallelismus

- Implementation in Java
- Servlet erweitert den HTTP-Server
- Servlets können jederzeit aktiviert werden
- mehrere Servlets können gleichzeitig aktiv sein
- Multithreading bedarf besonderer Vorsicht

Servlets

- Nebenläufige Prozesse in der virtuellen Maschine
- nutzen mehrere CPUs aus
- `init()`-Routine
- `destroy()`-Routine

Datenbank-Login

```
class DataBase {  
    ...  
    public synchronized Connexion getConnection()  
        throws ServletException {  
        if (Connexions.isEmpty()) {  
            throw new ServletException("Überlastung");  
        }  
        else {  
            Connexion c = (Connexion) Connexions.elementAt(0);  
            Connexions.removeElementAt(0);  
            return c;  
        }  
    }  
}
```

Zusammenfassung

- Java gut geeignet für WWW-Anwendungen
- geringe Entwicklungskosten
- Standard-XML leicht aus HTML konvertierbar
- XML zur Beschreibung multimedialer Dokumente