HPI Hasso Plattner Institut

IT Systems Engineering | Universität Potsdam

Informationsintegration
Beispiel

17.4.2007

Felix Naumann

Web Service A

```
<pub>
   <Titel> Federated Database
           Systems </Titel>
   <Autoren>
      <Autor> Amit Sheth </Autor>
      <Autor> James Larson </Autor>
    </Autoren>
</pub>
```

Web Service B

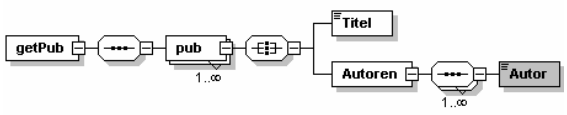
```
<publication>
   <title> Federated Database
           Systems for Managing
           Distributed, Heterogeneous,
           and Autonomous
           Databases </title>
   <auth> Scheth & Larson </auth>
   <year> 1990 </year>
</publication>
```

Integration | Identifikation | Fusion | Optimierung | Visualisierung

Beispiel – Web Service A

3

- Standort: Trier
- Operation:
  - getPubByAuthor(firstName, lastName)
  - getPubByTitle(title)
- Output-Struktur:



Beispiel – Web Service A Output

4

```
<xs:element name="pub" maxOccurs="unbounded">
    <xs:complexType>
        <xs:all>
            <xs:element name="Titel" type="xs:string" nillable="true"/>
            <xs:element name="Autoren">
                <xs:complexType>
                    <xs:sequence maxOccurs="unbounded">
                        <xs:element name="Autor" type="xs:string" nillable="false"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:all>
    </xs:complexType>
</xs:element>
```

Beispiel – Web Service B

5

- Standort: Humboldt-Universität
- Operation: myPubs(Autor, Jahr)
- Struktur:

Beispiel – Web Service B Output

6

```
<xs:element name="publication" maxOccurs="unbounded">
    <xs:complexType>
        <xs:all>
            <xs:element name="Title" type="xs:string" nillable="true"/>
            <xs:element name="Auth" type="xs:string" nillable="false"/>
            <xs:element name="Year" type="xs:string" nillable="false"/>
        </xs:all>
    </xs:complexType>
</xs:element>
```

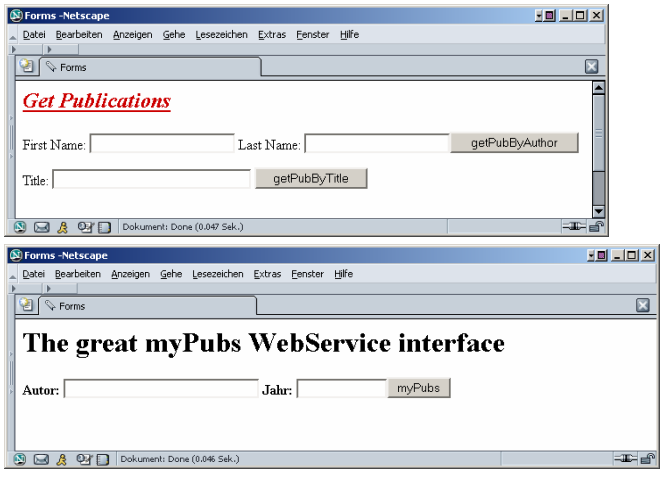
Integration von Web Services A & B

7

1. Nutzerschnittstelle
2. Schema Integration / Schema Mapping
3. Anfrage-Umwandlung
4. Zeit abschätzen (Optimierung)
5. Requests an beide Services abschicken
6. Antworten einholen
7. Objektidentifikation
8. Integrationsschritte
    1. Konfliktlösung etc.
    2. Entscheidung kleinster gemeinsamer Nenner?
    3. Durchführung (deklarativ, prozedural)
9. Anzeige beim Nutzer

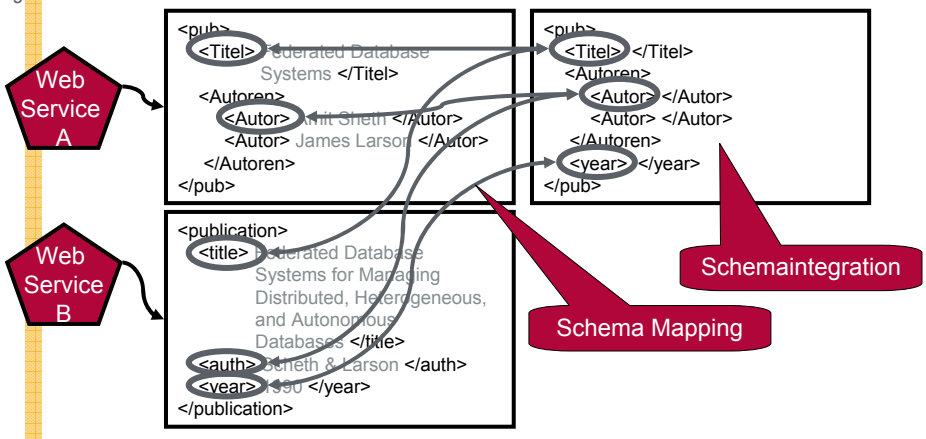
Nutzerschnittstellen

8



Informationsintegration

9



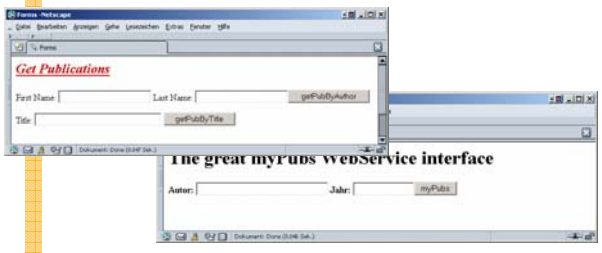
Modellierung durch eine Menge von Anfragen (Views)

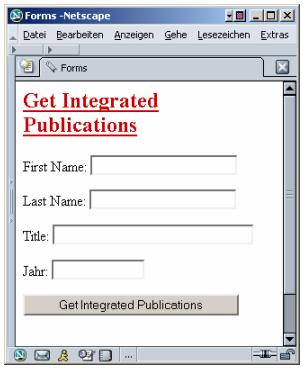
Anfrage Umwandlung

10

Integration der Anfrage durch Mediator:

- Integrierte Schnittstelle
- Z.B.
  Concat(First Name, Last Name)
  = Autor





Anfrageoptimierung

11

- Was ist besser: Eine schnelle Antwort oder vollständige Antwort?
  - Web Service A in Trier (remote)
  - Web Service B in Adlershof (local)
  - Web Service A hat mehr Attribute und mehr Objekte.
  - Web Service B hat weniger Attribute.
- Außerdem:
  - Eine Suche nach „year“ kann nur durch Web Service B beantwortet werden.
  - Transformationen können teuer sein.

Zwei Resultate

12

Web Service A

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- edited with XMLSPY v2004 rel. 2 U (http://www.xmlspy.com) by Felix Naumann
Universitat zu Berlin)  -->
<!-- Sample XML file generated by XMLSPY v2004 rel. 2 U (http://www.xmlspy.com)
- <getPub xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="C:\Dokumente und Einstellungen\naumann\Eigene
  Dateien\Lehre\Winter03_04\VL InfoInt I WS03_04\getPub.xsd">
  - <pub>
    <Titel>Real-world Data is Dirty: The Merge/Purge Problem</Titel>
    - <Autoren>
      <Autor>Mauricio Hernandez</Autor>
      <Autor>Salvatore Stolfo</Autor>
    </Autoren>
  </pub>
  - <pub>
    <Titel>MAC: Merging Autonomous Content</Titel>
    - <Autoren>
      <Autor>Felix Naumann</Autor>
      <Autor>Jens Bleiholder</Autor>
    </Autoren>
  </pub>
</getPub>
```

Web Service B

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Sample XML file generated by XMLSPY v2004 rel. 2 U (http://www.xmlspy.com)
- <myPub xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="C:\Dokumente und Einstellungen\naumann\Eigene
  Dateien\Lehre\Winter03_04\VL InfoInt I WS03_04\myPubs.xsd">
  - <publication>
    <Title>Merging Autonomous Content</Title>
    <Auth>Naumann</Auth>
    <Year>2003</Year>
  </publication>
  - <publication>
    <Title>Object Mathcing for Information Integration</Title>
    <Auth>Doan</Auth>
    <Year>1999</Year>
  </publication>
</myPub>
```

Schema Matching

13

```
<xs:all>
  <xs:element name="Titel" type="xs:string" nillable="true"/>
  <xs:element name="Autoren">
    <xs:complexType>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="Autor" type="xs:string" nillable="false"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:all>
```

?

```
<xs:all>
  <xs:element name="Title" type="xs:string" nillable="true"/>
  <xs:element name="Auth" type="xs:string" nillable="false"/>
  <xs:element name="Year" type="xs:string" nillable="false"/>
</xs:all>
```


Objektidentifikation

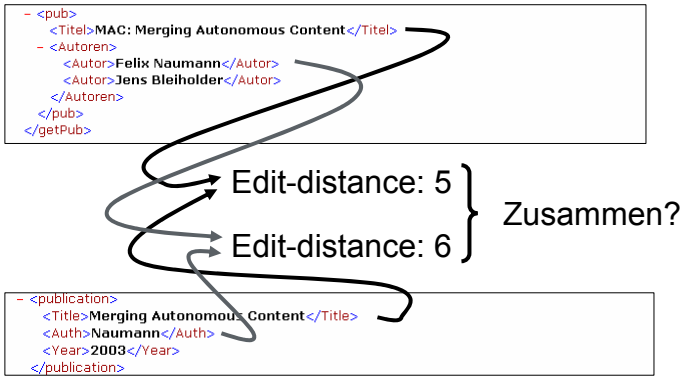
14

```
- <pub>
    <Titel>Real-world Data is Dirty: The Merge/Purge Problem</Titel>
  - <Autoren>
      <Autor>Mauricio Hernandez</Autor>
      <Autor>Salvatore Stolfo</Autor>
    </Autoren>
  </pub>
- <pub>
    <Titel>MAC: Merging Autonomous Content</Titel>
  - <Autoren>
      <Autor>Felix Naumann</Autor>
      <Autor>Jens Bleiholder</Autor>
    </Autoren>
  </pub>
</getPub>
```

```
- <publication>
    <Title>Merging Autonomous Content</Title>
    <Auth>Naumann</Auth>
    <Year>2003</Year>
  </publication>
- <publication>
    <Title>Object Mathcing for Information Integration</Title>
    <Auth>Doan</Auth>
    <Year>1999</Year>
  </publication>
</myPub>
```

Objektidentifikation

15



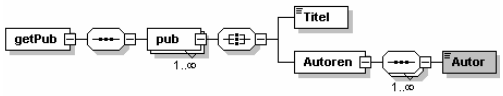
Stand der Dinge

16

- Wir haben die heterogenen Informationen.
- Wir wissen, was wir integrieren wollen.
- Aber noch nicht wie:
  - Integriertes Schema
  - Integrierte Daten

Angestrebtes Integrationsergebnis

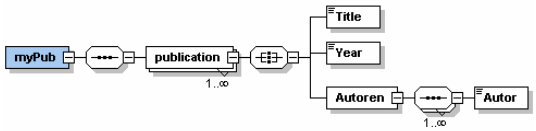
17



+

=

Integriertes Schema:



Angestrebtes Integrationsergebnis

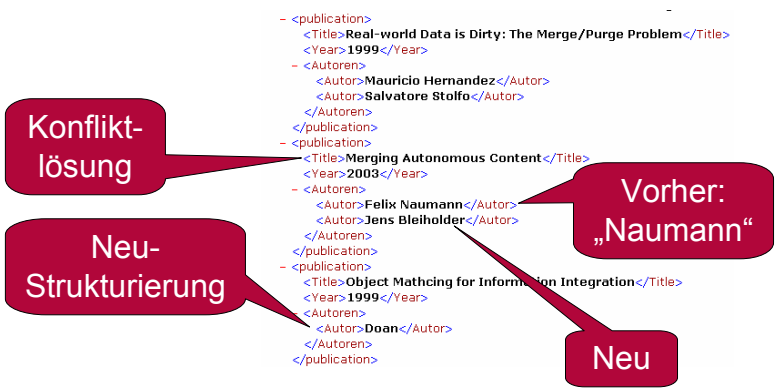
18

Integrierte Daten:

```
- <publication>
    <Title>Real-world Data is Dirty: The Merge/Purge Problem</Title>
    <Year>1999</Year>
  - <Autoren>
      <Autor>Mauricio Hernandez</Autor>
      <Autor>Salvatore Stolfo</Autor>
    </Autoren>
  </publication>
- <publication>
    <Title>Merging Autonomous Content</Title>
    <Year>2003</Year>
  - <Autoren>
      <Autor>Felix Naumann</Autor>
      <Autor>Jens Bleiholder</Autor>
    </Autoren>
  </publication>
- <publication>
    <Title>Object Mathcing for Information Integration</Title>
    <Year>1999</Year>
  - <Autoren>
      <Autor>Doan</Autor>
    </Autoren>
  </publication>
```

Integrierte Daten – was ist passiert?

19



Implementierung

20

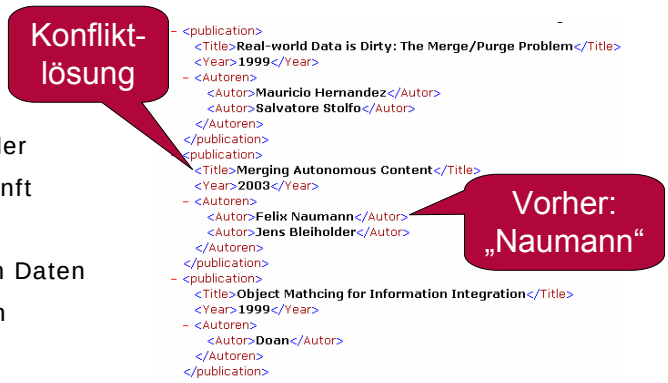
- Auf Folien ist alles klar, aber wie implementieren?
- Deklarativ?
  - SQL, XQuery, XSLT
  - Oft nicht alles möglich
  - Langsam
- Prozedural?
  - Java, C++
  - Schlecht wartbar
  - Schnell

Anzeige beim Nutzer

21

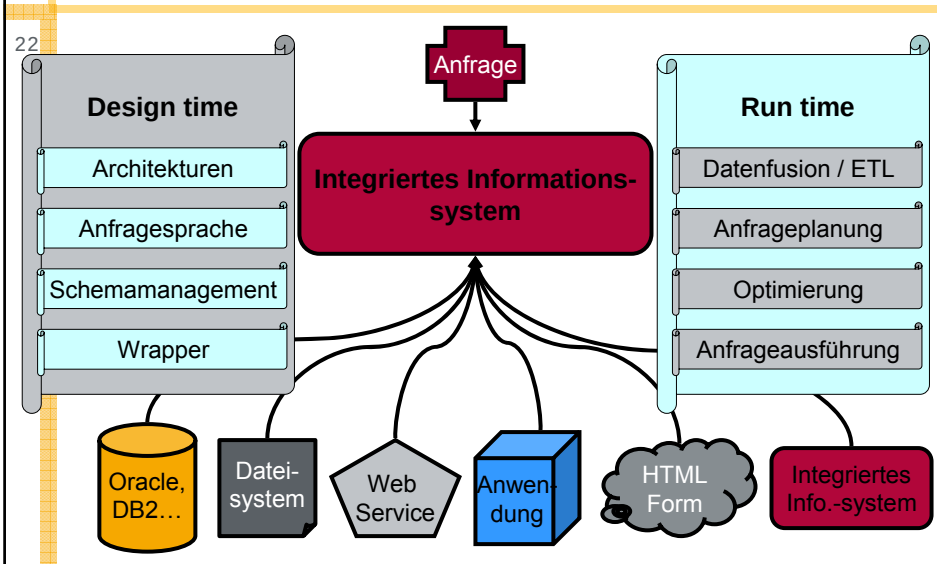
Visualisierung der

- Datenherkunft
- Qualität
- veränderten Daten
- Operationen



Integrierte Informationssysteme

22



Erste Hälfte

23

Einführung in die Informationsintegration
Szenarien der Informationsintegration
Verteilung und Autonomie
Heterogenität

Problemstellung

Materialisierte und virtuelle Integration
5-Schichten Architektur
Mediator/Wrapper-Architektur / PDMS

Architekturen

Schema Mapping
Schema Matching
SchemaSQL

Mapping

Global-as-View und Lokal-as-View Modellierung
Global-as-View Anfragebearbeitung

Modellierung

Zweite Hälfte

24

Containment & Local-as-View Anfragebearbeitung
Bucket Algorithmus

Anfragen

Verteilte Anfragebearbeitung
Dynamische Programmierung in verteilten Datenbanken
Top-N Anfragen

Optimierung

Duplikaterkennung
Datenfusion - Union & Co.
DWH, ETL & Data Lineage
Informationsqualität

Konflikte

Hidden Web
Semantic Web

Anwendungen

Fragen, Wünsche und Vorstellungen

- Jetzt, oder...
- Raum: A.1-13
- Sprechstunden: Dienstags 15-16 Uhr
  oder n.V.
- Email: naumann@hpi.uni-potsdam.de
- Telefon: (0331) 5509 280