

# INTEROPERABILITY

Univ. Prof. Dr. Stefanie Rinderle-Ma  
Workflow Systems and Technology Group  
Faculty of Computer Science  
University of Vienna  
stefanie.rinderle-ma@univie.ac.at

## Outline



### 1.1 Motivation and Terminology

### 1.2 XML and Databases

### 1.3 Information Integration

### 1.4 Service-oriented Architectures

### 1.5 Ontologies

### General references

## 1.1 Motivation



- ❑ **Situation today:**
  - **Enterprise = many (distributed) systems form a single system**
- ❑ **Challenges for IT:**
  - Business/IT Alignment → Interface between departments and IT
  - Flexibility → Reaction on changing requirements
  - Complexity → Effects of steadily changing requirements
  - Heterogeneity → Platforms, programming languages and paradigms
  - Maintenance → scalability (central solutions are restricted with respect to scalability)
- ❑ **Modern IT has to deal with these challenges**  
**→ INTEROPERABILITY**

## 1.1 Terminology



Interoperability – let us try some definition:

- ❑ “... the ability of two or more systems or components to **exchange** information and to use the information that has been exchanged.” (IEEE Standard Computer Dictionary)
- ❑ „... The ability of software and hardware on multiple machines from multiple vendors to **communicate**.” (www.computer-online-dictionary.org)
- ❑ In summary interoperability means the ability to
  - Exchange **information**
  - Share **information**
  - **Communicate** with each other
  - Use each others **services**
  - Without creating effort for the partners

## 1.1 Terminology



- It is all about information and services
- → Main topics of this lecture
  - Data exchange between XML and databases
  - Information integration
  - Service-oriented architectures
  - Ontologies

## Outline



1.1 Motivation and Terminology

1.2 XML and Databases

1.3 Information Integration

1.4 Service-oriented Architectures

1.5 Ontologies

General references

## 1.2 XML and Databases



### □ XML

- Exchange format
- very important in practice

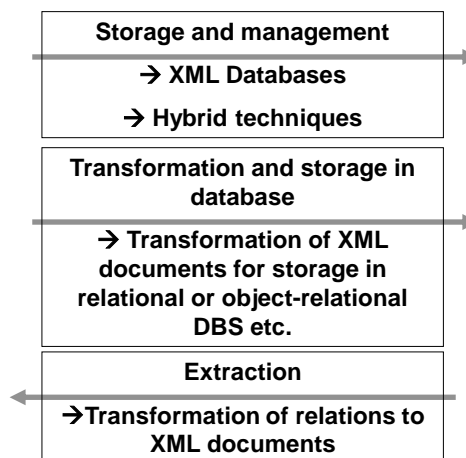
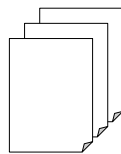
### □ Databases

- are the most important implementation platform for realizing information systems
- are used as development and Integration tools
- As starting question for approaching interoperability:  
*How to bring XML and databases together?*
- **Note:** We do not want to repeat XML basics (→ Web Engineering)

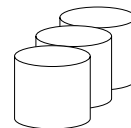
## 1.2 XML and Databases



XML documents



Databases





### Teaching objectives:

- ❑ Show challenges of extracting database content into XML
  - Database techniques
  - SQL/XML standard
- ❑ Show possibilities to store XML documents in databases
  - Different mappings
  - Native XML databases
  - SQL/XML standards

## Outline



1.1 Motivation and Terminology

1.2 XML and Databases

1.3 Information Integration

1.4 Service-oriented Architectures

1.5 Ontologies

General references

### 1.3 Information Integration



- ❑ Information integration is a difficult problem → more than 50% of IT costs are due to integration of existing systems
- ❑ WHY?
  - Data extraction from multiple heterogeneous data sources
  - Data integration conflicts [SPD92]:
    - ◆ Semantic conflicts
    - ◆ Description conflicts
    - ◆ Heterogeneity conflicts
    - ◆ Structural conflicts
    - ◆ Additionally multiple kinds of conflicts
- ❑ Structure → schema integration
- ❑ Data → data cleaning and integration

### 1.3 Information Integration



#### Teaching objectives:

- ❑ Show difficulties and challenges of information integration
- ❑ Discuss different kinds of conflicts
  - Structural conflicts
  - Semantic conflicts
- ❑ Show specific challenges along Data Warehouse Systems
- ❑ Introduce different approaches to deal with conflicts (selection):
  - Data transformation
  - Data cleaning
  - Schema matching

## Outline



1.1 Motivation and Terminology

1.2 XML and Databases

1.3 Information Integration

1.4 Service-oriented Architectures

1.5 Ontologies

General references

## 1.4 Service-oriented Architectures



### SOA levels

**Orchestration and Composition**

**Services**

**Enterprise Service Bus**

**Application Systems**

## 1.4 Service-oriented Architectures



### Teaching objectives:

- ❑ Understanding concepts of service-oriented architectures and current web service technologies
- ❑ Understanding challenges of enterprise-wide application integration
- ❑ Understanding methods, techniques, and concepts for service orchestrations and choreographies in a process-oriented way
- ❑ Summary of core questions for a service-oriented architecture:
  - How quickly can business processes and services be designed, implemented, and enacted (i.e., *deployed*)?
  - How expensive is it to change the Process- and Service-oriented Information system after their introduction (i.e., *rollout*)?
  - How flexible can we react on new requirements (e.g., when changing the business process)?
  - How comprehensive is the life cycle support of business process and services?

## Outline



1.1 Motivation and Terminology

1.2 XML and Databases

1.3 Information Integration

1.4 Service-oriented Architectures

1.5 Ontologies

General references



## 1.5 Ontologies

Partly based on  
[LEKS]



### Two buzzwords: Ontologies and Semantic Web

- ❑ Why are they useful with respect to interoperability?
- ❑ Definition of Semantic Web by Tim Berners-Lee, James Hendler and Ora Lassila):  
„The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling **computers** and people to work in **cooperation**.“
- **Next step to make information available on the web also to machines!**
- ❑ The Semantic Web interoperability: machines working on behalf of humans

## 1.5 Ontologies

Partly based on  
[LEKS]



- ❑ Machines talking to machines – semantics need to be unambiguously represented: **Ontologies**
- ❑ **Interoperability** – information exchange and integration, from diverse sources, different applications: **mapping and reconciliation**
- ❑ Why Ontologies?
  - Terms not necessarily clarify the meaning.
  - Terms are misleading, concepts are unambiguous.
  - Agreement on terms is easier to reach, but is tricky.
  - Agreement on **concepts** is difficult to reach, but is stable



### Teaching objectives:

- Understand how ontologies can help to support interoperability
- Show how to model ontologies
- Show different ontology languages and tools
- Give insight into Semantic Web technology

## General References



- [Leser2006] Ulf Leser, Felix Naumann: Informationsintegration. dpunkt.verlag (2006)
- [SPD92] S. Spaccapietra, C. Parent, Y. Dupont: Model Independent Assertions for Integration of Heterogeneous Schemas. VLDB Journal 1:81 – 123 (1992)
- [Erl1] Thomas Erl: SOA:Principles of Design (auch in Deutsch verfügbar) (Pearson (Prentice Hall); ISBN-10: 0132344823)
- [Erl2] Thomas Erl: Service-Oriented Architecture (Pearson (Prentice Hall) , ISBN-10: 0131858580)
- [Erl3] Thomas Erl: SOA Design Patterns (Pearson (Prentice Hall); ISBN-10: 0136135161)
- [Erl4] Michael Papazoglou: Web Services: Principles and Technology (Pearson (Prentice Hall) , ISBN-10: 0321155556)
- [Alonso] Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju: Web Services. Concepts, Architectures and Applications (Springer-Verlag, ISBN-10: 3540440089)
- [Josuttis] Nicolai Josuttis: SOA in der Praxis – System-Design für verteilte Geschäftsprozesse
- [LEKS] F. D'Antonio, A. De Nicola, F. Taglino, N. Osimi, M. Missikoff: The INTEROP Projects ([www.interop.noie.org](http://www.interop.noie.org)) (2006)

## Interesting links



- ❑ [www.computerwoche.de/soa-expertenrat/](http://www.computerwoche.de/soa-expertenrat/)
- ❑ [www.306.ibm.com/software/solutions/soa/](http://www.306.ibm.com/software/solutions/soa/)
- ❑ [www.service-architecture.com/](http://www.service-architecture.com/)
- ❑ <http://www.ws-standards.com/glossary.asp>
- ❑ <http://www.thomaserl.com/>

MORE REFERENCES AND LINKS AT THE END OF EACH CHAPTER