

INFS3101/7100 Ontology and the Semantic Web

Module 3 Semantic Heterogeneity

Last lecture: Key Terms



- ❖ **Institutional fact** is a record of a **speech act**. **Brute fact** X counts as institutional fact Y in **context C**.
- ❖ Context includes **framing rules** and **background**.
- ❖ Most information systems manage records of speech acts. Interoperating agents perform speech acts, both **performative** and **informative**.

Interoperating information systems

- ❖ Many entities interact in an industry as customers and suppliers for possibly complex goods and services
- ❖ Automated business interaction requires interoperation of their information systems
- ❖ Business-to-consumer (B2C) complex, business-to-business much more complex.
- ❖ How to build the ontology supporting an exchange?

Federated database view

- ❖ Sheth and Larsen (1990) five schema architecture
 - Local schema different for each system
 - Component schema in common data model
 - Export schema - view visible to federation
 - Federated schema - combines all export schemas
 - External schema - view visible to application
- ❖ Assumes autonomy among local systems. Ontology built bottom-up.
- ❖ Main technical challenge seen as structural heterogeneity

Differences in Data Representation

- ❖ Local DBs developed independently, and there are likely to be many different models, different representations for similar objects,
- ❖ There is a need for an integrated representation,
- ❖ Naming conflicts,
- ❖ Format differences (domain, scale, precision, etc),
 - local to global transformation can be simple but the inverse can be very complex,

....(cont)

Differences in Data Representation

- ❖ Structural differences
 - data value versus attribute
- ❖ Missing or conflicting data,
- ❖ Identification of semantically valid common reference points,
- ❖ Conflicts among constraints,

- ❖ Data integration is the main issue - ability to decide how to resolve the differences is an important skill.

A Simple Example

Consider two companies data models :
Company 1 records are stored in one table Emp;
Emp(Emp#, Fname, Lname, Bdate, Dept#, Rank, Salary)
Company 2 records are stored in many tables - one for each company department;
Dept XX(S-Id, Fname, Sname, Position, Phone#, e-mail, URL)

Build an integrated schema - number of changes must be made to at least one of component schemata and consequently to the local Application Programs.

Views offer a solution to the structural issues !

But views not always possible

- ❖ Consider employer E and taxation department T, two systems covering the same people
- ❖ E represents people in scheme
 - (Staff#, Name,DOB), with Name x DOB key
- ❖ T represents same people in scheme
 - (Tax#, Name, Address), with Name x Address key
- ❖ Impossible to identify the same person in both systems by any view. Problem is underdetermined. Not enough information.

In fact, views are almost never possible

- ❖ Is a law qualification the same in different jurisdictions?
- ❖ Is the parts list for a Honda the same as the parts list for a Mercedes?
- ❖ Macintosh computer the same as a WinTel?
- ❖ Is a FORTRAN program the same as a Java program?
- ❖ Is Advanced Database Systems at UQ the same course as Advanced Database Systems at National University of Singapore?

Updates are also rarely possible

- ❖ Consider using the American Express site to book a room at the New York Hilton
- ❖ Involves a room reservation at the Hilton
- ❖ And a debit to your AMEX account
- ❖ Could a multidatabase update do this? Of course.
- ❖ But what about the commission paid to AMEX Travel by Hilton?
- ❖ Actions better thought of as requests via messages
- ❖ Workflow technology more appropriate than multidatabase where updates are needed

Federated databases not viable

- ❖ By the mid 1990s people had discovered that the fundamental problem was not structural heterogeneity, but semantic heterogeneity
 - Similar words mean different things in different systems
 - Subtle differences in meaning can make large differences in effect
- ❖ Colomb (1997) Impact of semantic heterogeneity
Colomb (2004) Institutional Facts ... show why

Institutional Facts View

- ❖ An information system is a record of institutional facts
- ❖ To integrate two systems means to integrate two systems of speech acts
- ❖ Integrating systems of speech acts means integrating brute facts and context.
- ❖ Context includes both formal framing rules and background
- ❖ No reason to expect this to be possible

Airline Example

- ❖ Consider a trip involving two point-to-point airlines.
- ❖ Integration requires agreeing on intermediate airport, passenger etc.
- ❖ Also framing rules (payment, pre-boarding times, baggage allowance, etc.)
- ❖ Enforcement of baggage allowance is aspect of background.
- ❖ Suppose the first airline does not enforce the 18 kg baggage allowance but the second does, and your baggage is 20 kg?
- ❖ Integration fails!

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

13

Killer Example

- ❖ Team sports enable complex interoperations among many players in a system of institutional facts. The rules and practices of the game are a good example of an ontology.
- ❖ Some sports are similar. Cricket and baseball both have bats, balls, fielders, runs, innings, outs, umpires. Bowler and pitcher are similar concepts, as are wicket-keeper and catcher. It would be possible to build a common ontology with only a limited degree of semantic heterogeneity

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

14

Killer Example

- ❖ But interoperation would mean a cricket team playing cricket with a baseball team playing baseball!!! The concept is so weird as to be unthinkable.
- ❖ There are sports nearly identical
 - American and Canadian gridiron
 - American and Australian baseball
- ❖ But a game between an team playing American gridiron and a team playing Canadian gridiron is absurd. A tiny amount of semantic heterogeneity can make an enormous difference.

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

15

Autonomy must go

- ❖ To interoperate, the systems must agree on a common world of objects and actions
- ❖ Therefore must give up autonomy
- ❖ Agreed common world called an **ontology**
 - Representation of a conceptualization
- ❖ Autonomy sacrificed by **commitment** to the ontology. Process is top-down. Ontology comes first.
- ❖ Requires aligning systems of speech acts, including framing rules and background.
- ❖ Interoperating system now looks much like a single system
- ❖ Automated players called **agents** operating in a common system of speech acts.

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

16

Interoperability actually can work

- ❖ Where organisations conduct business together
 - Interbank funds transfer
 - B2B e-commerce using electronic data interchange (EDI)
 - Industry standard nomenclature, eg ISBN
- ❖ Common reporting requirements
 - stock exchange profit and loss statements
- ❖ Common culture
 - Apartment renting in a given area
- ❖ Organisations are not fully autonomous

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

17

To build interoperation

- ❖ Must give up some autonomy
- ❖ Parties must agree on a global schema
- ❖ More commonly called an ontology
- ❖ To play, each party must commit to the ontology
- ❖ Therefore must resolve view problems, changing business practices and database representations if necessary.
- ❖ Changing business practices, especially background is by far the most costly aspect.
- ❖ So complex view constructions less important

INFS3101/INFS7100 week 2, 21 November, 2005

Bob Colomb

18

Lots of ontologies exist already

- ❖ Koechel catalog of Mozart's music
- ❖ Periodic table of the elements
- ❖ SNOMED ontology of medical terms
- ❖ Gene Ontology
- ❖ Electronic Data Interchange (EDI)
- ❖ MPEG-21 multimedia framework
- ❖ Enterprise ontology
- ❖ Shipwreck ontology
- ❖ Z39.50 information retrieval

Used for all sorts of purposes

- ❖ Business applications
 - Run-time interoperation (e-commerce)
 - Common data models (global schemas)
- ❖ Analytic applications
 - Emergent property discovery (data mining)
 - Exchange of complex data sets (e-science)
- ❖ Engineering applications
 - Generating applications (model-driven architecture)

Summary: Key Terms



- ❖ **Bottom-up approach** to building an ontology fails, due to **semantic heterogeneity**. To interoperate, organisations need to align their business practices, implemented as systems of speech acts. Requires integrating contexts, including background.
- ❖ The ontology is a record of agreement created prior to the interoperation. Participating organisations must **commit** to the ontology. Approach is **top-down**.
- ❖ A system of interoperating agents operates within a single system of speech acts.

Resources

- ❖ **Essential**
 - Notes chapter 3 Heterogeneity
- ❖ **Further**
 - Colomb (1997) "Impact of semantic heterogeneity ..."
 - Colomb (2004) "Information Systems Technology Grounded on Institutional Facts"
 - Both on course web site.