Information Systems Interoperability and its Implications for Enterprise Modelling

Guttorm Sindre
NTNU and (visiting) U. Auckland
Contents

- Background on INTEROP and Interoperability
- Background on enterprise modelling
- INTEROP and UEML
- Conclusions and Future Work
Background on INTEROP and Interoperability
INTEROP

- EU NoE project (Network of Excellence)
  - http://interop-noe.org/
- Around 50 partners from 13 EU states + Norway (4) and Switzerland
- Coordination: U. Bordeaux (F)
- Duration: 3 years from Nov 2003
- Budget: 12 M€, whereof 6.5 M€ from EU
- NoE: no funding for new research
  - but for the integration of research
Rationale (slide taken from http://interop-noe.org/)

Interoperability, key to increase competitiveness of enterprises

- The cost of non-interoperability are estimated to 40% of enterprises IT budget.

(Source: the Yankee Group 2001)
Interoperability: Definition (http://interop-noe.org/)

Capability of a system or a product to work with other systems or products without specific effort from the user.

Capacity of an enterprise software or application to interact with others.
Interoperability, many meanings

- 6 different levels of concern:
  - Data, Object, Component, Application, System, Enterprise, …Community

- 3 kinds of integration (heterogeneities):
  - Syntactic (e.g., different data formats)
  - Structural (e.g., different database schema)
  - Semantic (different meanings)

- INTEROP’s main interest:
  - Semantic interoperability on the application, system, and enterprise level
Approach (slide taken from http://interop-noe.org/)

- The originality of the project is to take a multidisciplinary approach by merging three research areas supporting the development of Interoperability of Enterprise Applications and Software:
  - **Architecture & Platforms**: to provide implementation frameworks,
  - **Enterprise Modelling**: to define Interoperability requirements and to support solution implementation,
  - **Ontology**: to identify Interoperability semantics in the enterprise.

Knowledge integration for Interoperability research
Background on Enterprise Modeling
What is enterprise modeling?

□ Describing the enterprise
  ▾ E.g., function, behaviour, information, resources, org.units, ...

□ Using models
  ▾ often (semi-)formal and graphic
  ▾ Many different kinds of perspectives, e.g.
    ▪ Information, process, org.structure, economy, resources, business rules, ...
Why enterprise modeling?

- **Main goals:**
  - Understand / explain
  - Experiment
  - Learn and decide
  - Operate and control

- **Major drivers:**
  - Diagnosis of the organization
  - Restructuring
  - Business process reengineering
  - Large scale systems integration
  - Implementation of huge packages (e.g., ERP)
  - Tuning the org. to face business change
  - Alignment to norms (e.g., ISO9000)
  - Supporting management decisions
Challenges for enterprise modelling

- Enterprises need to cooperate
  - …but have different business processes
  - …modelled with different languages
  - …and implemented in different, proprietary tools

- Goal: bridge this gap
  - enterprise models be understood across businesses
  - Need interoperability between enterprise modelling languages (EML’s)
Starting point for INTEROP WP5: UEML1.0

- Unified Enterprise Modeling Language
  - Another EU project (ended June 2003)
- Goal: a unified and expandable modeling language
  - Build European consensus on core constructs
  - Facilitate interoperability within the frame of ongoing standardisation efforts
- Further pursued in the project Athena
Main goals of UEML

1. Capture, represent, and structure the enterprise
2. Create, modify, and exchange enterprise models
3. Create, modify, and exchange enterprise modelling languages
4. Utilise models to solve different categories of problems
5. Ensure end-user involvement

– UEML Deliverable 2.3, March 2003
INTEROP and UEML
Modifying the goals

- Abandoned idea of a new modelling language
  - Just provide an exchange language
  - Existing EML’s can be translated back and forth
- Instead increased ambitions in
  - Not just providing a language
  - But also methods and tools for analyzing and comparing existing and emerging EML’s
- Challenge:
  - Huge number of EML’s
  - With many different perspectives
Approaches taken

- **Top-down**
  - Distributed requirements collection for UEML 2.0
  - From various partners
  - Using a requirements template for uniform input

- **Bottom-up**
  - Look at many different EML’s
  - Categorize and compare their constructs
## Requirements template (taken from D5.1)

<table>
<thead>
<tr>
<th>Detail level I</th>
<th>Detail level II</th>
<th>Detail level III</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Better refine your first idea by formulating a new and complete sentence</td>
<td>1. Formulate a first new concept of (1.) And / Or 1.2. Formulate a second new concept (1.)</td>
<td>1.1.1 Reformulate 1.1 (no replication) and / or 1.1.2 Reformulate 1.1 (no replication) and / or 1.1.3 Reformulate 1.1 (no replication) 1.2.1 Reformulate 1.2 (no replication) and / or 1.2.2 Reformulate 1.2 (no replication) and / or 1.2.3 Reformulate 1.2 (no replication)</td>
</tr>
</tbody>
</table>

- One requirement, e.g., one new construct wanted in the language
### Example of filled in template (taken from D5.1)

<table>
<thead>
<tr>
<th>Name:</th>
<th>Capture enterprise goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>UEML should be able to capture and show the enterprise goals in the models</td>
<td>1. The relation between enterprise goal and the model should be always evident</td>
</tr>
<tr>
<td></td>
<td>1.1. Each time a scenario simulation is made or a modelling decision is taken it has to be related to the enterprise goals</td>
</tr>
<tr>
<td></td>
<td>1.1.1 Enterprise goals means to take into account the context in which enterprise is operating</td>
</tr>
<tr>
<td></td>
<td>1.1.2 A number of such methodologies have been developed eg. Enterprise Knowledge Development (EKD), the i* methodology, Business Modeling with UML.</td>
</tr>
<tr>
<td></td>
<td>1.1.3 It is necessary to make explicit what the enterprise goals are and how they influence the enterprise analysis</td>
</tr>
<tr>
<td></td>
<td>1.2.1 A methodology for goal modelling must be included into the enterprise modelling component</td>
</tr>
<tr>
<td></td>
<td>1.2.2 UEML should use goals as constraints to the modelling and to decision making</td>
</tr>
<tr>
<td></td>
<td>1.2.3 Similar goal oriented methodologies have been developed and used in Sweden for many years.</td>
</tr>
</tbody>
</table>
A huge number of requirements captured

Core category requirements (identifying name):
- Modelling the complexity
- Goal-structures
- Robustness
- Capture different kinds of activities
- Capture agents
- Capture more control flow constructs
- Capture events
- Capture knowledge and operational levels
- Capture roles and positions
- Capture social effects
- Actor-relationships
- Capability modelling
- Openness
- Usability
- Integration support
- User friendliness

- Representation effectiveness
- Uncertainty handing
- Managerial support
- Timing aspects
- Modelling of flows
- Modelling the complexity

Basic category requirements (identifying name):
- Conceptual foundations
- Flexibility
- Continuous Improvement
- Dynamic modelling

Extended category requirements (identifying name):
- Communication capabilities
- Model and language integration
- Simulation capabilities
Problems encountered

- Huge diversity of requirements
  - Inspired by various modelling languages and tools
- Incorporating all would make UEML 2.0 very complex, cf. example meta-model next page
  - Partly due to weaknesses in starting point?
  - Lacking ontological foundation of UEML1.0?
Example meta-model (Bergholtz et al., 2005)
The bottom-up approach

- **Which languages to look at?**
  - Developed criteria for language selection
    - Quality of languages
    - Industrial impact

- **How to analyze languages?**
  - Developed a template for describing languages
  - The Bunge-Wand-Weber ontology used for comparison
## Initial list of proposed languages

- IEM
- EEML/MEML
- GEM
- ITM
- PetriNets
- CIMOSA language(s)
- GRAI (many sub-languages)
- IDEF (0,1,3)
- PSL
- WPDL/XPDL
- BPMN/BPML
- UML profiles for EAI
- ebXML/UBL
- Business Diagrams RUP profile
- ODP - ISO 15414
- I*/GRL
- Albert
- Aris (many sub-languages)
- Business Process Definition Metamodel (OMG)
- UML 2.0
- UEML 1.0
- EKD
- AUML
- E-R diagram
- Organizational charts
- ISO/DIS 19440
Languages analyzed at so far

- UML 2.0 (Class and Activity Diagrams)  
  - www.uml.org
- ISO/DIS 19440  
- IDEF 3  
- BPMN  
  - http://www.bpmn.org/
- GRL  
  - http://www.cs.toronto.edu/km/GRL/
- Coloured Petri Nets  
- XPDL  
Template for analyzing languages

☐ A textual template to be filled in
  ▪ For each modelling construct
  ▪ Of each language looked at

☐ Based on underlying meta-model by Opdahl & Henderson-Sellers (2004)
  ▪ Which is again based on the BWW ontology
  ▪ Establish a common ontology of enterprise modelling constructs

☐ Separate meta-models for presentation, representation, and internal structure
Fields of the textual template

- Document title
- File name
- Preamble
- Construct name
- Alternative construct names
- Related, but distinct construct names
- Related terms
- Language
- Diagram type
- Presentation
- Icon, line style, text
- Builds on
- Built on by
- User-definable attributes
- Relationships to other constructs
- Layout conventions
- Semantics
- Instantiation level
- Classes of things
- Properties (and relationships)
- Behaviour
- Modality (permission, recommendation etc)
- Open Issues
Underlying meta-model
Taken from Deliverable DEM.1: UEML 2.1 (Berio et al., 2005)
Example: related constructs in two EML’s

IEM.Action: Construct Definition

ISO.Activity: Construct Definition

Activity System1: Represented Class

Activity System2: Represented Class

ComponentSystem: Class

GoalseekingComponentSystem: Class

Taken from Deliverable DEM.1: UEML 2.1 (Berio et al., 2005)
Example of interoperability (in theory)

Taken from Deliverable DEM.1: UEML 2.1 (Berio et al., 2005)
Conclusions and Future Work
Conclusions

- The best thing for interoperability (wrt enterprise modelling):
  - All use the same language, in a consistent manner
  - Probably not going to happen
- Making a new modelling language considered infeasible
- Even an exchange language difficult to achieve
  - Diversity, evolution
  - Many different requirements
  - But good progress made on this within INTEROP
- The UEML template approach promising
  - But demanding
Future directions

☐ Looking at more languages
☐ Detail-level of the common ontology?
  ■ Coarse-grained, fine-grained, or both?
☐ Ontological foundation
  ■ Is BWW sufficient?
☐ Presentation: not much dealt with so far
☐ Formal definition of UEML template
☐ Tool support
☐ Empirical validation
☐ Standardization
References