Achieving Enterprise Model Interoperability through the Model-Based Architecture Framework for Enterprises

Håvard D. Jørgensen, Oddrun P. Ohren

SINTEF ICT, PO Box 124, Blindern, N-0314 Oslo, Norway
{hdj,odo}@sintef.no

Abstract. This paper describes an ontology for enterprise modelling, The ontology has enabled conceptual integration of two different modelling methodologies, one based on UEML (Unified Enterprise Modelling Language), the other a UML profile for enterprise modelling for software development. The ontology, called Model-based Architecture Framework for Enterprises (MAFE), extends a general framework for model based architectures (MAF). It contains a minimal number of core concepts, which were found to be designated by numerous different terms in the two pre-existing methodologies.

1 Background and Motivation - The MAF Framework

MAF [1] is a framework for describing enterprise and ICT architectures, based on the IEEE 1471 standard [2]. IEEE defines 'architecture' as the collection of the *essential components* of the enterprise, their interrelations and evolution over time [2]. MAF frameworks consist of:

- a *conceptual model* (ontology) of architectural descriptions depicting the core concepts to be used for describing architectures;
- a set of *structuring rules*, specifying which *views* and *models* should be included in an architecture description;
- a *terminology* covering the subject domain;
- a set of *language mappings* between the *ontology* and *modelling languages* (terminologies) to be used to create architecture descriptions;
- a *methodology*, specifying how to go about developing architecture descriptions;
- *principles* of conformance, consistency, specialisation and realisation;
- a framework for an *experience well* repository for reusing knowledge.

While the concepts in the ontology should be defined independently of the viewpoints of particular stakeholders, terminologies and modelling languages may reflect the concerns and viewpoints of particular groups of stakeholders.

An ontology is a specification of a conceptualisation. *Concepts* are units of thought. Concepts can have a linguistic representation as *terms* in a language, and refer to objects and properties in the physical world. The meaning of a concept is determined by its *reference* set (the physical and ideational objects it refers to), and its *intension*. The intension specifies the properties that apply to the concept [3]. Two concepts are synonyms if both their reference set and their intension are equal.

Concepts can be categorised according to *logical strength* (ability to express facts precisely) into indefinite individuals, definite individuals, classes, relations and quantitative properties. Over the lifecycle of a model, all of these concept categories may appear, typically evolving from vague concepts towards increased logical strength.

Conceptual models must be able to cater for all of these degrees of vagueness and precision. *Perspective integration* [4] requires that different degrees of specificity may be expressed with the same concepts. We thus choose to include in the conceptual model of MAFE only the core concepts, and allow local terminologies to represent these concepts in different ways, at different degrees of specificity and logical strength. The core principle is that concepts directly refer to ideas or physical things, properties and phenomenon, while terms capture the current state of a description of these concepts. Well-formedness rules define what is a correct model, not what may occur in the world, and should thus be defined for terms, rather than concepts.

2 Conceptual Models of Enterprises

MAFE is a refinement of MAF. Whereas MAF covers any kind of system, MAFE focuses on the specific kind of systems called ‘enterprise’, defined as a project, undertaking or business activity.   The conceptual model of MAFE is a refines that of MAF. All MAFE models contain terms that designate MAFE concepts, which are depicted in Figure 1.

- **Enterprise** is an undertaking. It is the system of analysis for an enterprise model. It contains other enterprise elements.
- **Task** is a unit of work at any level of granularity or specificity. Terms like action, process, activity, work item etc. typically designate different kinds of tasks. Such term reflect different points of view, and are thus terms, not concepts.
- **Resource** is anything needed to perform work. Resource is actually a point of view on objects that primarily exist as something else, e.g. persons, documents, and things. Resources are specialised depending on their role
  - **Actor**, the active subjects that perform the work, (humans or computers).
  - **Information**, objects that bring knowledge to the process.
  - **Tool**, objects applied by the actor to perform the work.
  - **Object**, material things that are manipulated or applied in the work.

  In addition to these resources, **Knowledge**, **Time** and **Money** are also important for an enterprise. These aspects are intertwined with the resources, e.g. determining availability.
- **Decision**, a choice among alternatives. A decision may also include e.g. the identification and exploration of alternatives, and the timing of selections.
- **Dependency** is the general kind of association between enterprise elements. Among tasks, dependencies capture sequencing and coordination needs, while dependencies between tasks and resources entail that the resources are applied for performing the task. Two important dependencies are defined. **Fills** relates an element and a resource, signifying that one takes the role that the other specifies,

while **Flow** signifies that something should occur before the other something else.

- **Goal** (from MAF), representing visions, objectives, goals of actors, tasks, enterprises etc.

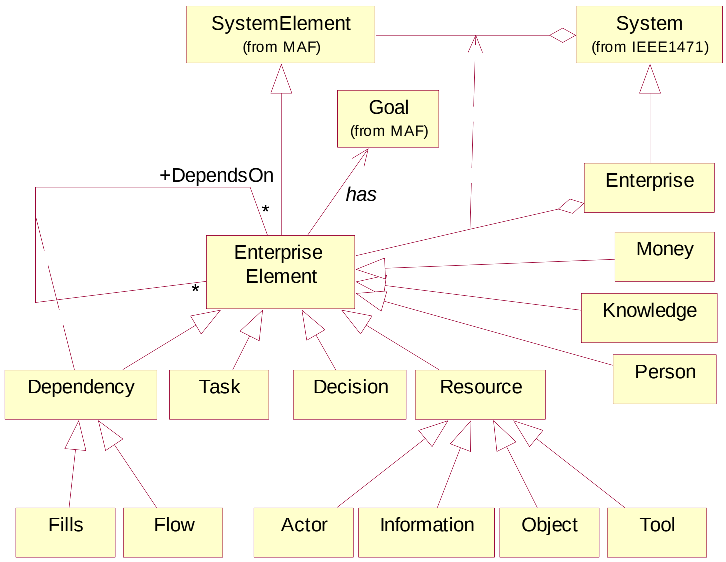

Figure 1. MAFE enterprise ontology.

All of these objects may be decomposed into objects of any kind. For instance, a dependency may be decomposed into a process of tasks that manage this dependency, and a resource may be decomposed into a decision tree that articulates which actors may fill the role.

2.1 Application of MAFE to UEML and UML

MEML is a modelling notation for the concepts described above. MEML has roots in data flow diagrams, which was extended with resource modelling and control flow semantics in APM [5]. Later the language was refined for the EXTERNAL model driven infrastructure (as EEML [6]) and integrated with other languages in the UEML (Unified EML) project [7]. Finally, we simplified and cleansed the language in the Monesa project [8] (MEML = Monesa EML).

Like MAFE is a specialisation of MAF for enterprise architectures, so is MAFIS for information systems. The environment viewpoint in MAFIS describes what surrounds the information system, including a number of enterprise elements. MAFIS was developed for a particular purpose in a government agency. We were able to define all the enterprise terms in MAFIS with MAFE concepts. An example (from organisation modelling) is shown in the table below (with MAF and MAFE concepts in *italic*). By defining MAFIS terms (model element classes) in this way, we were thus able to align the UML profile with a standard enterprise modelling language. In a similar manner, we were also able to show that the enterprise viewpoint terminology of ISO RM-ODP (Reference model for open distributed processing [9]) also designates MAFE concepts.

MAFIS Term	**MAFE Concept**
Organisation unit	*Composite Actor*
Parent_organisation	*Decomposition* of *Actor*
Structural relationship	*Dependency* between *Actors*

3 Directions for Further Work

We are now completing the operationalisation of the MAFE methodology. The *metametamodel* structure and services is of particular concern in this work, since it controls how users may define new concepts, terms, modelling elements and views needed in their local domain. In general, we are concerned about how modelling can be made more user-friendly and less computer-oriented, allowing different degrees of formalisation. Towards this aim, we have found *reflective instances* (containing their own definition) to be a promising approach, removing much of the complexity of class-oriented approaches. To make the framework comprehensible, we seek to minimise the number of meta-layers, offering the users templates that they may copy rather than classes to instantiate. Finally, computer-oriented encapsulation, where the behavioural semantics of each class is defined inside the class and hidden from the rest of the model, is replaced by *semantic holism*, where the meaning of each model element may depend on the whole rest of the model. This approach better mimics the flexibility, contextuality, simplicity and expressive efficiency of natural language [8].

References

1. Aagedal, J. Ø., et al., *Model-Based Architecture Framework, v. 3.0*. 2003, SINTEF, Norway.
2. IEEE, *Standard 1471-2000 - IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*. 2000.
3. Bunge, M., *Philosophy of Science - Vol.1 From Problem to Theory*. Revised ed. Science and Technology Studies, ed. M. Bunge. 1998, New Brunswick, USA: Transaction Publishers.
4. Opdahl, A.L. and G. Sindre, *Facet Modelling: An Approach to Flexible and Integrated Conceptual Modelling.* Information Systems, 1997. **22**(5): p. 291-323.
5. Carlsen, S., *Conceptual Modeling and Composition of Flexible Workflow Models*, PhD thesis 1997, Norwegian University of Science and Technology: Trondheim, Norway.
6. Krogstie, J., Jørgensen, H.D., *Interactive models for supporting networked organisations*, CAiSE 2004.
7. Zelm, M. *Towards User Oriented Enterprise Modelling - Comparison of Modelling Languages*. in *Concurrent Engineering (CE) Conference*. 2003. Madeira, Portugal: A.A. Balkema Publishers, Rotterdam, Netherlands.
8. Jørgensen, H.D., *Interactive Process Models*, PhD thesis 2004, Norwegian University of Science and Technology: Trondheim, Norway.

9. ISO, *Information Technology - Open distributed Processing - Reference Model - Enterprise Language*. 2002, ISO/IEC 15414: Geneva, Switzerland.