

To those members of WG2 meeting in Wuhan,

I am unable to participate in your meeting. I am preparing a paper as input to the 11179-19763 Harmonisation workshop in November and was hoping to have that ready for the Wuhan meeting, but I have been too busy to complete it in time.

So, instead, here are some thoughts on 19763 that you might like to consider.

A big question for me, and I do not think it is documented anywhere, is whether the relevant metaclasses in Parts 3, 5, etc are subclasses or instances of the metaclasses in Part 2 that represent models and model components.

If they are subclasses, then Parts 3, 5, etc are providing the ability to register metamodels (M2 level) for ontologies, process models, etc. In other words, using Part 5 it would be possible to register the metamodel for IDEF0 process models, the metamodel for BPMN models, the metamodel for UML activity diagrams, the metamodel for SSADM data flow diagrams, the metamodel for Rummler-Brache style swim lane diagrams, etc. Having registered the metamodels it should then be possible to register the actual M1 models, for example the UML activity diagram for a particular system, using the model-mapping capabilities of Part 4. In this situation, Parts 3, 5, etc extend the registry defined in Part 2.

If they are instances then they are the M2 metamodels and are providing the ability to directly register the M1 models. In this situation, Parts 3, 5, etc do not extend the registry but are definitions of content that is to be registered.

Looking at what is available so far it looks as if they are the latter - instances of the metaclasses in the core model and population at the M2 level for the registry.

But this gives me a problem.

Taking Part 5 as an example, is it feasible, or even desirable, to create a metamodel that encompasses every possible flavour of process model? And, if we do decide to do that, how can we be sure that we have covered every possible flavour of process model?

Or should we be standardising separate metamodels for each known flavour of process model, for example a metamodel for IDEF0 process models, a metamodel for BPMN models, a metamodel for UML activity diagrams, a metamodel for SSADM data flow diagrams, a metamodel for Rummler-Brache style swim lane diagrams, etc, probably using separate clauses in Part 5 for each metamodel. But do most of these metamodels already exist elsewhere, either in ISO/IEC or with some other standards body, such as OMG?

I have personal experience of developing a metamodel (for entity-relationship models) that encompassed just two approaches to entity-relationship modelling (ORACLE and SSADM) and the result was not entirely successful. In 1995 the UK proposed the resulting combined metamodel as a 'content module' for the IRDS standards (Horiuchi-san may remember this), but it was not accepted. Had we tried to include, for example, IDEF1X, I think it would have been impossible to come up with

a single model because there are some fundamental discrepancies between the concepts used in these different notations.

And now I have started talking about entity-relationship models, that brings up another problem. 19763 is the Metamodel Framework for ***Interoperability***. In all the work I have done on interoperability (and I have been involved in multi-national interoperability project work) the thing that has been at the heart of interoperability is information, not processes or services. Interoperability is about exchanging data so that information can be shared. So, where are entity-relationship models (or UML class diagrams) and database structures in 19763? When challenged in Jeju, Okabe-san reluctantly agreed that the definition of an ontology could be stretched to include data structures, but I do not think that there is enough detail in Part 3 to cover the concepts in each flavour of entity-relationship modelling and each flavour of database structure comprehensively enough to be able to map elements in one model to elements in another model in a different notation. It looks as if we need a number of different 'data structure' metamodels. For data modelling we will need UML class models, Object Role Modelling, and as many entity-relationship modelling notations as we can identify. For database structures we will need SQL and ODMG ODL as a minimum. But do most of these metamodels already exist?

For any modelling notation, data modelling, process modelling, etc, for which CASE tool support already exists metamodels must exist to describe the CASE tool repository. But are those metamodels standardised, or are they the CASE tool vendor's perception of what the metamodel should be, or what the vendor had to do to get the tool to work?

Whether the Parts 3, 5, etc metaclasses are subclasses or instances of the Part 2 metaclasses and whether or not 19763 is extended to include data modelling and database structures, the viability of each subsequent part depends on Part 2, and the content of Part 2 is still not agreed. It looks to me as if the most important task for the 19763 'team' is to develop Part 2, since everything else should depend on Part 2.

Sorry, this is a bit rushed and a bit unstructured, but I hope you will find it helpful.

Keith Gordon
20 August 2009