An ontology-driven unifying metamodel for UML Class Diagrams, EER, and ORM2

C. Maria Keet

School of Mathematics, Statistics, and Computer Science,
University of KwaZulu-Natal, South Africa,
keet@ukzn.ac.za

joint work with Pablo Rúben Fillottrani
Universidad Nacional del Sur, Argentina

Dagstuhl Seminar "Automated Reasoning on Conceptual Schemas", Schloss Dagstuhl, Germany, May 20-24, 2013

Setting

- NRF/DST- and MINCyT-funded Project "Ontology-driven unification of conceptual modelling languages"

Setting

- NRF/DST- and MINCyT-funded Project "Ontology-driven unification of conceptual modelling languages"
- Aims of the project:
  - inter-model assertions among EER, UML v2.4.1, ORM2;
  - one formalization including *all* (structural, static) language features, where each of the languages is a fragment;
  - (converting among the representations, and reasoning across models);
  - some module management

Setting

- NRF/DST- and MINCyT-funded Project “Ontology-driven unification of conceptual modelling languages”
- Aims of the project:
  - inter-model assertions among EER, UML v2.4.1, ORM2;
  - one formalization including *all* (structural, static) language features, where each of the languages is a fragment;
  - (converting among the representations, and reasoning across models);
  - some module management

⇒ First step: identify commonalities and differences in terminology, syntax, semantics, and ontological commitments of the structural components of the three main languages (EER, UML Class Diagrams v2.4.1, ORM2)

⇒ Metamodel

A motivation why first metamodelling

$\mathcal{DLR}_{ifd}$	**OWL 2 DL**	**FOL**
– no implementation	+ several automated reasoners, relatively scalable	– few reasoners, not really scalable
– no interoperability	+ linking with ontologies doable	– no interoperability with existing infrastructures
– no integration	+ 'integration' with OntoIOP	+ 'integration' with OntoIOP
+ formalisation exist	– formalisation to do	± formalisation exist
+ little feature mismatch	– what to do with OWL 2 DL features not in the CM languages and vv.	+ little feature mismatch
– modularity infrastructure	+ modularity infrastructure	– modularity infrastructure

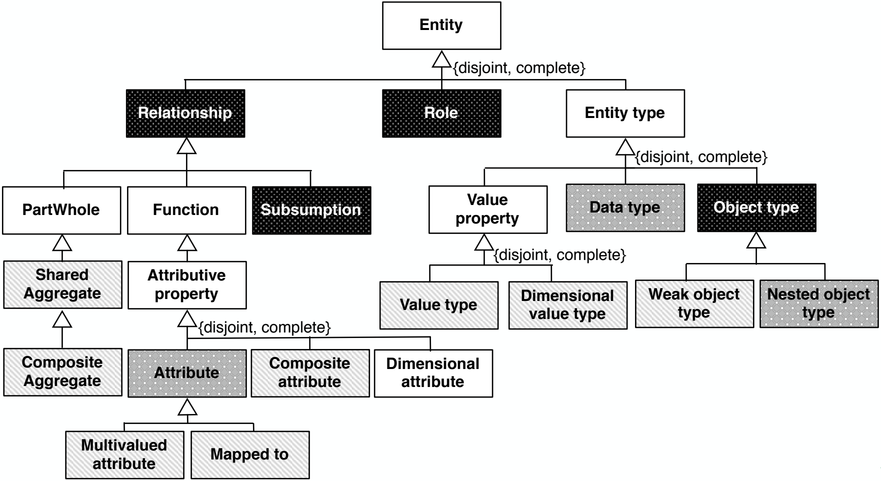
Ontology-driven

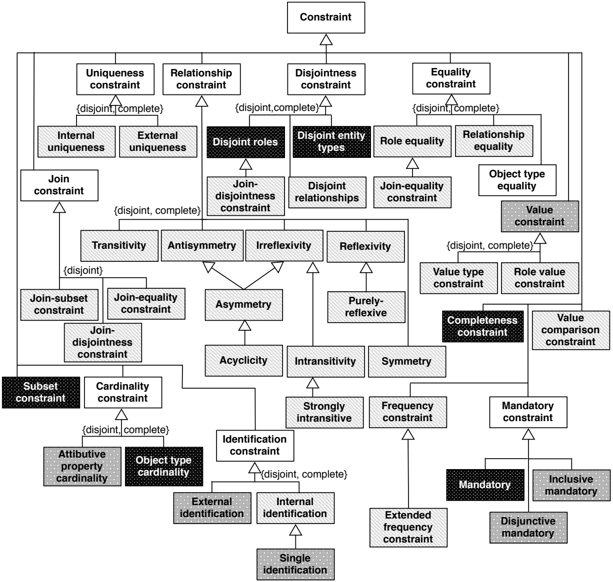
- Uncover ontological decisions embedded in the modelling language, among others:
  - Positionalism of relations (nature of relations)
  - Identification mechanisms (identity)
  - Attributes, ‘attribute-free’ or ‘attribute-hidden’ (attributions, quality properties)
  - Subrelations (meaning of a sub-relation)
  - Any differences/similarities for constraints (e.g., on when a relationship may be objectified)

Notes for the metamodel

- We use UML Class Diagram notation for the metamodel
- Not all constraints can be represented in that diagram, but added as textual constraint
- It has some redundancies (from a logic-based perspective), e.g., multivalued attributes
- Not all features may be 'good' features, but we do not judge about elegance (can be addressed in a formalization)
- Table with naming conventions for UML, EER, ORM2, and the metamodel terms

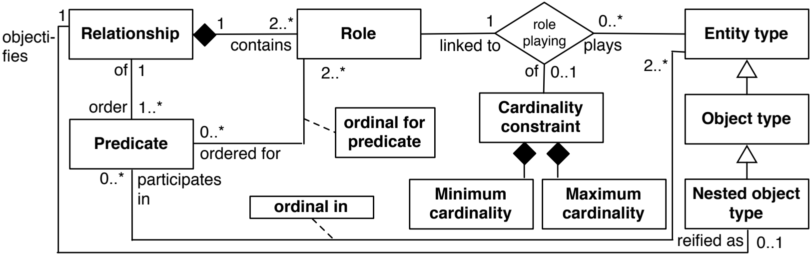
Principal static entities of the metamodel




Constraint
Uniqueness constraint
Relationship constraint
Disjointness constraint
Equality constraint
{disjoint, complete}
Internal uniqueness
External uniqueness
{disjoint,complete}
Disjoint roles
Disjoint entity types
{disjoint, complete}
Role equality
Relationship equality
Join constraint
Join-disjointness constraint
Disjoint relationships
Join-equality constraint
Object type equality
Value constraint
{disjoint, complete}
Transitivity
Antisymmetry
Irreflexivity
Reflexivity
{disjoint, complete}
Value type constraint
Role value constraint
{disjoint}
Join-subset constraint
Join-equality constraint
Asymmetry
Purely-reflexive
Completeness constraint
Value comparison constraint
Join-disjointness constraint
Acyclicity
Intransitivity
Symmetry
Subset constraint
Cardinality constraint
Strongly intransitive
Frequency constraint
Mandatory constraint
{disjoint, complete}
Identification constraint
Attibutive property cardinality
Object type cardinality
{disjoint, complete}
Mandatory
Inclusive mandatory
External identification
Internal identification
Extended frequency constraint
Disjunctive mandatory
Single identification

- UML, EER, and ORM are all **positionalist** [Keet(2009)]
  - *n*-ary relationship
  - Role that an entity plays in a relationship
  - No order on the roles (or: 'relationship' as a set of roles), but one can add an order
  - Relationship composed of roles
- Optional predicate, with order and no roles
- Cardinality
- Nested object type

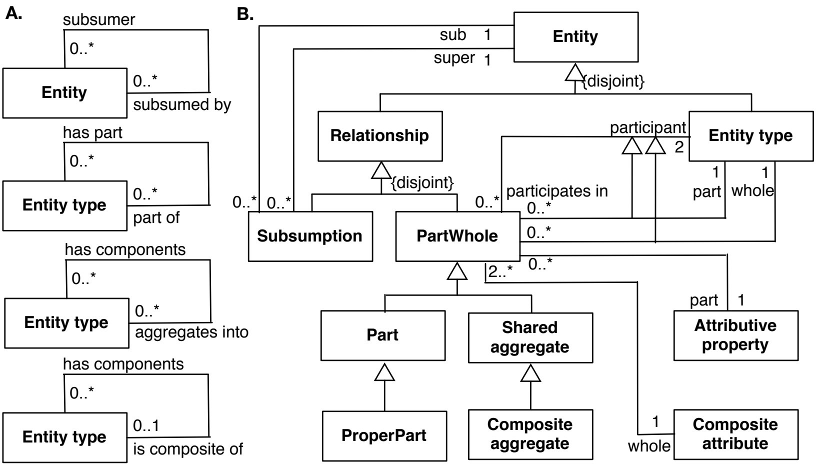
Principal relationships between Relationship, Role, and Entity type



Notes

- Relations between roles and a predicate can only exist if there is a relation between those roles and the relationship that that predicate is an ordering of (i.e., it is a join-subset)
- Entities that participate in the predicate must play those roles that compose the relationship of which that predicate is an ordered version of it

Subsumption and aggregation



What are attributes?

- An *attribute* (A) is a binary relationship between a relationship or entity type ($R \cup E$) and a data type (D), i.e., $A \mapsto R \cup E \times D$
- An attribute is no more, and no less
- For instance, one can have an attribute hasColour, that relates an object type to a string; e.g, hasColour $\mapsto$ Flower $\times$ `String`

Attribution in Ontology and ontologies

- Principally as *quality property*, formalised as unary predicate
- Separate relation to endurants or perdurants
- Separate relation to 'values' (qualia) of an attribution

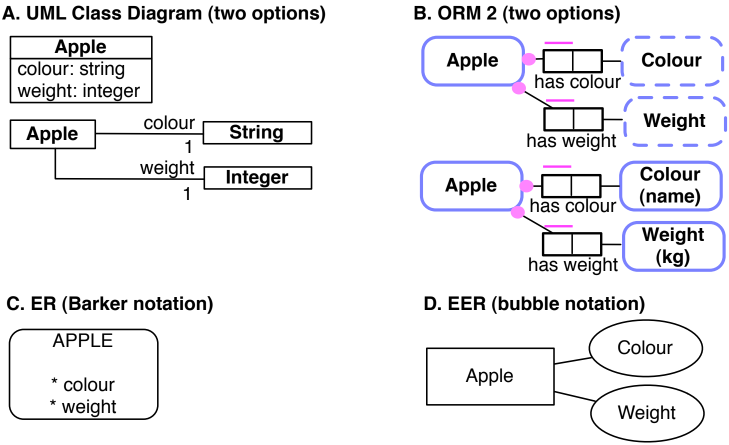
Attribution in Ontology and ontologies

- Principally as *quality property*, formalised as unary predicate
- Separate relation to endurants or perdurants
- Separate relation to 'values' (qualia) of an attribution
- Implemented as such in foundational ontologies, such as DOLCE and GFO

Attribution in Ontology and ontologies

- Principally as *quality property*, formalised as unary predicate
- Separate relation to endurants or perdurants
- Separate relation to 'values' (qualia) of an attribution
- Implemented as such in foundational ontologies, such as DOLCE and GFO
- Practically, the same quality property can be related to more than one entity

Examples of ‘attributes’ in UML, EER, ORM



Dimensional attributes and value types

- *dimension* for the value: implicit meaning in the values for some data types, which has to do with measurements
- e.g., hasHeight $\mapsto$ Flower $\times$ $\texttt{Integer}$ does not contain any of that information, but somehow has to be included

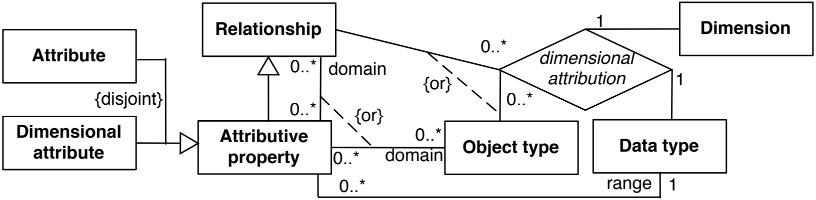
Dimensional attributes and value types

- *dimension* for the value: implicit meaning in the values for some data types, which has to do with measurements
- e.g., hasHeight $\mapsto$ Flower $\times$ `Integer` does not contain any of that information, but somehow has to be included
- How? e.g.:
  - hasHeight $\mapsto$ Flower $\times$ `Integer` $\times$ cm
    or perhaps with an approach along the line of:
  - hasHeight $\mapsto$ Flower $\times$ Height
  - mapped_to $\mapsto$ Height $\times$ `Integer`
  - hasDimension $\mapsto$ Integer $\times$ cm

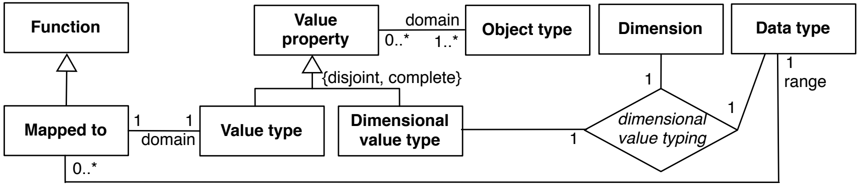
Dimensional attributes and value types

- *dimension* for the value: implicit meaning in the values for some data types, which has to do with measurements
- e.g., hasHeight $\mapsto$ Flower $\times$ `Integer` does not contain any of that information, but somehow has to be included
- How? e.g.:
  - hasHeight $\mapsto$ Flower $\times$ `Integer` $\times$ cm
    or perhaps with an approach along the line of:
  - hasHeight $\mapsto$ Flower $\times$ Height
  - mapped_to $\mapsto$ Height $\times$ `Integer`
  - hasDimension $\mapsto$ Integer $\times$ cm
- Within the scope of unification, add the notion of dimension (not a whole system of recording measurement data for a specific scenario)

Attributes in the metamodel



Value types in the metamodel



Related work: data and schema level

- Mapping and transformation algorithms using a common hypergraph, for small subsets of ER, UML, and ORM; set-based semantics vs. a model-theoretic semantics [Boyd and McBrien(2005)]
- Physical schema layer [Bowers and Delcambre(2006)]

Related work: conceptual data modelling and software engineering

- Compare the languages through their metamodels in ORM, highlight differences [Halpin(2004)]
- Metamodel for a part of ER and a part of NIAM in CoCoA and implemented in MViews, Pounamu [Venable and Grundy(1995), Grundy and Venable(1996), Zhu et al.(2004)]; omits, a.o., value types, composite attributes, and e.g., NIAM is forced to have the attributes as in ER
- (linking/integrating conceptual models represented in the same conceptual data modelling language [Atzeni et al.(2008), Fillottrani et al.(2012)])

Related work: Knowledge representation and reasoning

- Approach: mainly, choose a logic and show it fits neatly/sufficiently with one or more conceptual data modelling languages
- Separate partial formalisations various logics; a.o., [Berardi et al.(2005), Calvanese et al.(1998), Halpin(1989), Hofstede and Proper(1998), Queralt and Teniente(2008)]
- Partial unifications, using e.g., $\mathcal{ALUNI}$ [Calvanese et al.(1999)], several *DL-Lite* fragments [Artale et al.(2007)], $\mathcal{DLR}_{\text{ifd}}$ [Keet(2008)]
- They cannot simply be linked up and implemented

Related work: Knowledge representation and reasoning

- Approach: mainly, choose a logic and show it fits neatly/sufficiently with one or more conceptual data modelling languages
- Separate partial formalisations various logics; a.o., [Berardi et al.(2005), Calvanese et al.(1998), Halpin(1989), Hofstede and Proper(1998), Queralt and Teniente(2008)]
- Partial unifications, using e.g., $\mathcal{ALUNI}$ [Calvanese et al.(1999)], several *DL-Lite* fragments [Artale et al.(2007)], $\mathcal{DLR}_{\text{ifd}}$ [Keet(2008)]
- They cannot simply be linked up and implemented
- Distributed Ontology Language [Mossakowski et al.(2012)] and system that is currently being standardised by ISO (`http://ontoiop.org`).

Conclusions

- Unifying, ontology-driven metamodel capturing most of ORM/FBM, EER, and static UML v2.4.1 w.r.t their static, structural, entities, their relationships, and constraints
- The only intersection among all these conceptual data modelling languages are role, relationship, and object type
- Adhere to the positionalist commitment of the meaning of relationship
- Attributions are represented differently in each language, but, ontologically, they denote the same notions
- Several implicit aspects, such as dimensional attribute and its reusability and relationship versus predicate, have been made explicit
- Common constraints: disjointness, completeness, simple mandatory, object type cardinality

Current work

- Two papers submitted
- Near future: formalisation in FOL, then OWL 2 DL subset
- Extension of tool that will aid the process of complex systems design and information integration

References I

- Alessandro Artale, Diego Calvanese, Roman Kontchakov, Vladislav Ryzhikov, and Michael Zakharyaschev. Reasoning over extended ER models. In *Proceedings of the 26th International Conference on Conceptual Modeling (ER'07)*, volume 4801 of *LNCS*, pages 277–292. Springer, 2007.
- Paolo Atzeni, Paolo Cappellari, Riccardo Torlone, Philip A. Bernstein, and Giorgio Gianforme. Model-independent schema translation. *VLDB Journal*, 17(6):1347–1370, 2008.
- D. Berardi, D. Calvanese, and G. De Giacomo. Reasoning on UML class diagrams. *Artificial Intelligence*, 168(1-2):70–118, 2005.
- Shawn Bowers and Lois M. L. Delcambre. Using the uni-level description (ULD) to support data-model interoperability. *Data & Knowledge Engineering*, 59(3):511–533, 2006.
- Michael Boyd and Peter McBrien. Comparing and transforming between data models via an intermediate hypergraph data model. *Journal on Data Semantics*, IV:69–109, 2005.
- D. Calvanese, G. De Giacomo, and M. Lenzerini. On the decidability of query containment under constraints. In *Proc. of the 17th ACM SIGACT SIGMOD SIGART Sym. on Principles of Database Systems (PODS'98)*, pages 149–158, 1998.
- D. Calvanese, M. Lenzerini, and D. Nardi. Unifying class-based representation formalisms. *Journal of Artificial Intelligence Research*, 11:199–240, 1999.
- Pablo R. Fillottrani, Enrico Franconi, and Sergio Tessaris. The ICOM 3.0 intelligent conceptual modelling tool and methodology. *Semantic Web Journal*, 3(3):293–306, 2012.

References II

- J.C. Grundy and J.R. Venable. Towards an integrated environment for method engineering. In *Proceedings of the IFIP TC8, WG8.1/8.2 Method Engineering 1996 (ME'96)*, volume 1, pages 45–62, 1996.
- T. A. Halpin. *Advanced Topics in Database Research*, volume 3, chapter Comparing Metamodels for ER, ORM and UML Data Models, pages 23–44. Idea Publishing Group, Hershey PA, USA, 2004.
- T.A. Halpin. *A logical analysis of information systems: static aspects of the data-oriented perspective*. PhD thesis, University of Queensland, Australia, 1989.
- A. H. M. ter Hofstede and H. A. Proper. How to formalize it? formalization principles for information systems development methods. *Information and Software Technology*, 40(10):519–540, 1998.
- C. M. Keet. Ontology-driven formal conceptual data modeling for biological data analysis. In Mourad Elloumi and Albert Y. Zomaya, editors, *Biological Knowledge Discovery Handbook: Preprocessing, Mining and Postprocessing of Biological Data*, chapter 6, page in press. Wiley, 2013.
- C. Maria Keet. A formal comparison of conceptual data modeling languages. In *13th International Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD'08)*, volume 337 of *CEUR-WS*, pages 25–39, 2008. 16-17 June 2008, Montpellier, France.
- C. Maria Keet. Positionalism of relations and its consequences for fact-oriented modelling. In R. Meersman, P. Herrero, and Dillon T., editors, *OTM Workshops, International Workshop on Fact-Oriented Modeling (ORM'09)*, volume 5872 of *LNCS*, pages 735–744. Springer, 2009. Vilamoura, Portugal, November 4-6, 2009.

References III

C. Maria Keet. Enhancing identification mechanisms in UML class diagrams with meaningful keys. In *Proceeding of the SAICSIT Annual Research Conference 2011 (SAICSIT'11)*, pages 283–286. ACM Conference Proceedings, 2011. Cape Town, South Africa, October 3-5, 2011.

Till Mossakowski, Christoph Lange, and Oliver Kutz. Three semantics for the core of the Distributed Ontology Language. In Michael Grüninger, editor, *Proceedings of the 7th International Conference on Formal Ontology in Information Systems (FOIS'12)*. IOS Press, 2012. 24-27 July, Graz, Austria.

Anna Queralt and Ernest Teniente. Decidable reasoning in UML schemas with constraints. In Zohra Bellahsene and Michel Léonard, editors, *Proceedings of the 20th International Conference on Advanced Information Systems Engineering (CAiSE'08)*, volume 5074 of *LNCS*, pages 281–295. Springer, 2008. Montpellier, France, June 16-20, 2008.

J.R. Venable and J.C. Grundy. Integrating and supporting Entity Relationship and Object Role Models. In Mike P. Papazoglou, editor, *Proceedings of the 14th Object-Oriented and Entity Relationship Modelling Conferece (OO-ER'95)*, volume 1021 of *LNCS*, pages 318–328. Springer-Verlag, 1995. Gold Coast, Australia, Dec 13-16 1995.

N. Zhu, J.C. Grundy, and J.G. Hosking. Pounamu: a metatool for multi-view visual language environment construction. In *IEEE Conf. on Visual Languages and Human-Centric Computing 2004*, 2004. Rome, 25-29 Sept. 2004.