On the unification of conceptual data modelling languages

C. Maria Keet

Department of Computer Science, University of Cape Town, South Africa,

mkeet@cs.uct.ac.za

Seminar at the Poznan University of Technology, Poland, June 16, 2015

---

Joint work with Pablo Rubén Fillotrani, Universidad Nacional del Sur,
Bahía Blanca, Argentina
Outline

1. Motivation

2. Unification approach
   - Metamodel
   - Transformations and intermodel assertions

3. Quantitative analysis

4. Conclusions
Outline

1. Motivation

2. Unification approach
   - Metamodel
   - Transformations and intermodel assertions

3. Quantitative analysis

4. Conclusions
Motivation

Unification approach

Quantitative analysis

Conclusions

Context

- Bilateral project “ontology-driven unification of conceptual data modelling languages” (mid 2012 - mid 2015)\(^2\), funded by SA Dept. of Sci & Tech and AR’s MINCyT

- Conceptual data modelling for complex system development and information integration

- Languages for conceptual modelling: UML Class Diagram, ER and EER, ORM and ORM2

- Develop formal basis for model linking and integration, tools and techniques

---

\(^2\)Project page: http://www.meteck.org/SAAR.html
Example: isiZulu termbank (simplified)
Example: ICOM (Franconi and others)
Previous work

- Inter-model assertions between models in the same language [Atzeni et al.(2008), Fillottrani et al.(2012)]
- Inter-model assertions between models in different languages, but subset only [Atzeni et al.(2012), Boyd and McBrien(2005), Venable and Grundy(1995), Zhu et al.(2004)]
- Limited model transformations [Atzeni et al.(2012), Boyd and McBrien(2005)]
- Limited or no automated reasoning, verification [Calvanese et al.(1999), Fillottrani et al.(2012), Keet(2009)]
Outline

1 Motivation

2 Unification approach
   - Metamodel
   - Transformations and intermodel assertions

3 Quantitative analysis

4 Conclusions
Overview

- All static, structural elements of main CDM languages
- First ontological, then logical, finally implement
  - Develop unifying and ontology-driven metamodel, then formalise it
  - Mechanism for inter-model assertions and transformations
  - Quantitative evaluation to prioritise rule specification
    - Language profile specification (tractable languages?)
  - Implement, and look at modularisation (ongoing)
Overview

- All static, structural elements of main CDM languages
- First ontological, then logical, finally implement
- Develop *unifying* and *ontology-driven* metamodel, then formalise it
- Mechanism for inter-model assertions and transformations
- Quantitative evaluation to prioritise rule specification
  - Language profile specification (tractable languages!)
- Implement, and look at modularisation (ongoing)
**Metamodel: overview**

- Captures all structural elements in the selected languages\(^3\)\(^4\)
- Captures also their relations and constraints
- Describes the rules in which they may be combined
- The metamodel is designed in UML Class Diagram, and formalized in FOL (precision) and OWL (practical usability)\(^5\)

---

\(^3\) Keet, C.M., Fillottrani, P.R. Toward an ontology-driven unifying metamodel for UML Class Diagrams, EER, and ORM2. ER’13. W. Ng, V.C. Storey, and J. Trujillo (Eds.). Springer LNCS vol. 8217, 313-326.


\(^5\) Fillottrani, P.R., Keet, C.M.. *KF metamodel formalization*. Technical Report, Arxiv.org

**Static entities**

{Disjointness axioms among the subclasses of Relationship are:
\{PartWhole, Attributive property, Subsumption\} and
\{Qualified relationship, Attributive property, Subsumption\} }
Constraints
Selection of constraints between them (1/2)
A Weak identification is a combination of one or more Attributive property of the Weak object type it identifies together with the Identification constraint of the Object type it has a Relationship with and this Object type is disjoint with the Weak object type.

The Single identification has a Mandatory constraint on the participating Role and the Relationship that Role is contained in has a 1:1 Cardinality constraint declared on it.

Qualified identification and External identification are declared on only Attributive property.

A Qualified relationship participates in a Qualified identification only if the Cardinality constraint is 1.
Transformation Rules and Inter-model assertions

- Process for linking and translating models
- Based on different kinds of rules: mappings, transformations, approximations
- Together with the (formalised) metamodel, it can be used to verify inter-model assertions

---

6 Fillottrani, P.R., Keet, C.M. Conceptual Model Interoperability: a Metamodel-driven Approach. RuleML’14,
A. Bikakis et al. (Eds.). Springer LNCS vol. 8620, 52-66.
Approach (inter-model assertions)

- classify entities of M1 and M2 into MM entities;
- process mapping assertions using the transformation algorithms and compare output with element in M2;

input model M1 and M2 in language X and Y, resp.

formalised metamodel

input inter-model assertion

output model M12 or NO

vocabulary with lists which entities should be mapped, transformed, approximated, non-mappable

1:1 mappings
UML class : ORM Entity Type
... : ...

Transformations
UML attribute : ORM Value type
... : ...

Approximation
... : ...

No mappings
ORM role equality : UML x
...

∀(x) Relationship(x) → Entity(x)
...

∀(x) (¬(Datatype(x) ∧ Qualifier(x)))
...

Inter-model assertions
UML class `Flower`: ORM Entity Type `Flower`
... : ...

name:string
colour:string
Flower

name:string
colour:string
Flower

name
colour
Flower (ID)

name
colour
Flower (ID)
1:1 mapping rules and the metamodel (selection)

(R1) Association $\xrightarrow{\text{UML to MM}}$ Relationship

in:
Association(AssociationEnd : Class, AssociationEnd : Class)

out: AssociationEnd $\rightarrow$ Role \hspace{1cm} // i.e., using (Ro1)
out: Association $\rightarrow$ Relationship
out: Class $\rightarrow$ Object Type \hspace{1cm} // i.e., using (O1)
out: Relationship(Role:Object type, Role:Object Type)

(1R) Relationship $\xrightarrow{\text{MM to UML}}$ Association

in: Relationship(Role:Object type, Role:Object Type)

out: Role $\rightarrow$ AssociationEnd \hspace{1cm} // i.e., using (1Ro)
out: Relationship $\rightarrow$ Association
out: Object Type $\rightarrow$ Class \hspace{1cm} // i.e., using (1O)
out:
Association(AssociationEnd : Class, AssociationEnd : Class)
Generating and mapping

**GenOT**  Class $\xrightarrow{\text{UML to ORM}}$ Entity type

- in: $C$
- out: (O1)
- out: (2O)
  // i.e., an ORM EntityType named $C$

**MapR**  Association $\xrightarrow{\text{UML to ER}}$ Relationship

- in: $A(ae_1 : C_1, ae_2 : C_2)$
- out: (R1)
- out: (3R)
- out: match pattern out(3R) with $R(rc_1 : E_1, rc_2 : E_2)$
∀(x, y)(Contains(x, y) → Relationship(x) ∧ Role(y))
∀(x)∃≥^2 y(Contains(x, y))
∀(x)(Role(x) → ∃(y)(Contains(y, x)))
∀(x, y, z)(Contains(x, y) ∧ Contains(z, y) → (x = z))
∀(x, y, z)(RolePlaying(x, y, z) → Role(x) ∧ CardinalityConstraint(y) ∧ EntityType(z))
∀(x)(Role(x) → ∃(y, z)(RolePlaying(x, y, z)))
∀(x, y, z, v, w)(RolePlaying(x, y, z) ∧ RolePlaying(x, v, w) → (y = v) ∧ (z = w))
∀(x, y, z, v, w)(RolePlaying(x, y, z) ∧ RolePlaying(v, y, w) → (x = v) ∧ (z = w))
∀(x)(CardinalityConstraint(x) → ∃(y)(MinimumCardinality(x, y) ∧ Integer(y))))
∀(x)(CardinalityConstraint(x) → ∃(y)(MaximumCardinality(x, y) ∧ Integer(y))))
∀(x, y)(Identifies(x, y) → (∃(y)(IdentificationConstraint(x) ∧ ObjectType(y)))))
∀(x)(IdentificationConstraint(x) → ∃(y)(Identifies(x, y))))
∀(x, y, z)((Identifies(x, y) ∧ Identifies(x, z)) → (y = z))
∀(x)(ObjectType(x) → ∃(y)(Identifies(y, x))))
∀(x, y, z)((DeclaredOn(x, y) ∧ DeclaredOn(x, z) ∧ IdentificationConstraint(x) ∧ (¬(y = ValueProperty(y) ↔ ¬AttributiveProperty(z)))))
∀(x)(IdentificationConstraint(x) → ∃(y)(DeclaredOn(x, y))))
∀(x, y)((DeclaredOn(x, y) ∧ SingleIdentification(x)) → (Attribute(y) ∨ ValueType(y))}
∀(x)(SingleIdentification(x) → ∃(y)(DeclaredOn(x, y)))
∀(x, y, z)((SingleIdentification(x) ∧ DeclaredOn(x, y) ∧ DeclaredOn(x, z)) → (y = z))
∀(x, y)(Contains(x, y) → Relationship(x) ∧ Role(y))
∀(x)∃≥2y(Contains(x, y))
∀(x)(Role(x) → ∃(y)(Contains(y, x)))
∀(x, y, z)(Contains(x, y) ∧ Contains(z, y) → (x = z))
∀(x, y, z)(RolePlaying(x, y, z) → Role(x) ∧ CardinalityConstraint(y) ∧ EntityType(z))
∀(x)(Role(x) → ∃(y, z)(RolePlaying(x, y, z)))
∀(x, y, z, v, w)(RolePlaying(x, y, z) ∧ RolePlaying(x, v, w) → (y = v) ∧ (z = w))
∀(x, y, z, v, w)(RolePlaying(x, y, z) ∧ RolePlaying(v, y, w) → (x = v) ∧ (z = w))
∀(x)(CardinalityConstraint(x) → ∃(y)(MinimumCardinality(x, y) ∧ Integer(y))))
∀(x)(CardinalityConstraint(x) → ∃(y)(MaximumCardinality(x, y) ∧ Integer(y))))
∀(x, y)(Identifies(x, y) → (IdentificationConstraint(x) ∧ ObjectType(y)))
∀(x)(IdentificationConstraint(x) → ∃(y)(Identifies(x, y)))
∀(x, y, z)((Identifies(x, y) ∧ Identifies(x, z)) → (y = z))
∀(x)(ObjectType(x) → ∃(y)(Identifies(y, x))))
∀(x, y, z)((DeclaredOn(x, y) ∧ DeclaredOn(x, z) ∧ IdentificationConstraint(x) ∧ (¬(y = z)) (ValueProperty(y) ↔ ¬AttributiveProperty(z))))
∀(x)(IdentificationConstraint(x) → ∃(y)(DeclaredOn(x, y))))
∀(x, y)((DeclaredOn(x, y) ∧ SingleIdentification(x)) → (Attribute(y) ∨ ValueType(y)))
∀(x)(SingleIdentification(x) → ∃(y)(DeclaredOn(x, y)))
∀(x, y, z)((SingleIdentification(x) ∧ DeclaredOn(x, y) ∧ DeclaredOn(x, z)) → (y = z))
Formalised metamodel (section), highlighted for step 5

\forall (x, y) (\text{Contains}(x, y) \rightarrow \text{Relationship}(x) \land \text{Role}(y))
\forall (x) \exists \geq 2 y (\text{Contains}(x, y))
\forall (x) (\text{Role}(x) \rightarrow \exists (y) (\text{Contains}(y, x)))
\forall (x, y, z) (\text{Contains}(x, y) \land \text{Contains}(z, y) \rightarrow (x = z))
\forall (x, y, z) (\text{RolePlaying}(x, y, z) \rightarrow \text{Role}(x) \land \text{CardinalityConstraint}(y) \land \text{EntityType}(z))
\forall (x) (\text{Role}(x) \rightarrow \exists (y, z) (\text{RolePlaying}(x, y, z)))
\forall (x, y, z, v, w) (\text{RolePlaying}(x, y, z) \land \text{RolePlaying}(x, v, w) \rightarrow (y = v) \land (z = w))
\forall (x, y, z, v, w) (\text{RolePlaying}(x, y, z) \land \text{RolePlaying}(v, y, w) \rightarrow (x = v) \land (z = w))
\forall (x) (\text{CardinalityConstraint}(x) \rightarrow \exists (y) (\text{MinimumCardinality}(x, y) \land \text{Integer}(y)))
\forall (x) (\text{CardinalityConstraint}(x) \rightarrow \exists (y) (\text{MaximumCardinality}(x, y) \land \text{Integer}(y)))
\forall (x, y) (\text{Identifies}(x, y) \rightarrow (\text{IdentificationConstraint}(x) \land \text{ObjectType}(y)))
\forall (x) (\text{IdentificationConstraint}(x) \rightarrow \exists (y) (\text{Identifies}(x, y)))
\forall (x, y, z) ((\text{Identifies}(x, y) \land \text{Identifies}(x, z)) \rightarrow (y = z))
\forall (x) (\text{ObjectType}(x) \rightarrow \exists (y) (\text{Identifies}(y, x)))
\forall (x, y, z) ((\text{DeclaredOn}(x, y) \land \text{DeclaredOn}(x, z) \land \text{IdentificationConstraint}(x) \land (\neg (y = z)) \rightarrow (\text{ValueProperty}(y) \leftrightarrow \neg \text{AttributiveProperty}(z)))
\forall (x) (\text{IdentificationConstraint}(x) \rightarrow \exists (y) (\text{DeclaredOn}(x, y)))
\forall (x, y) ((\text{DeclaredOn}(x, y) \land \text{SingleIdentification}(x)) \rightarrow (\text{Attribute}(y) \lor \text{ValueType}(y)))
\forall (x) (\text{SingleIdentification}(x) \rightarrow \exists (y) (\text{DeclaredOn}(x, y)))
\forall (x, y, z) ((\text{SingleIdentification}(x) \land \text{DeclaredOn}(x, y) \land \text{DeclaredOn}(x, z)) \rightarrow (y = z))
Outline

1. Motivation

2. Unification approach
   - Metamodel
   - Transformations and intermodel assertions

3. Quantitative analysis

4. Conclusions
Few elements belong to all three language families

⇒ Is it worth trying to link or integrate or translate their models?

- Collected available models on each language, and studied the usage of metamodel elements on them (approx. 35 on each language)
  - Only 64% of the entities are the kind of entities that appear in all three language families
  - When more features are available in a language, they are used in the models (though some very few times)
  - Specification of a feature-based 'characteristic profile' for each family

---

Keet, C.M., Fillottrani, P.R. An analysis and characterisation of publicly available conceptual models. ER'15. Springer LNCS. (accepted)
Few elements belong to all three language families

⇒ Is it worth trying to link or integrate or translate their models?

Collected available models on each language, and studied the usage of metamodel elements on them (approx. 35 on each language)

- Only 64% of the entities are the kind of entities that appear in all three language families
- When more features are available in a language, they are used in the models (though some very few times)
- Specification of a feature-based ‘characteristic profile’ for each family

---

7 Keet, C.M., Fillottrani, P.R. An analysis and characterisation of publicly available conceptual models. ER’15. Springer LNCS. (accepted)
**Table:** Prevalence of particular entity in the models, as percent of total number of entities for that family, aggregated by model family and rounded off to one decimal. OT: Object type; VT: Value type; Rel.: Relationship; Int. Unique.: Internal uniqueness constraint; ID: Identifier.

<table>
<thead>
<tr>
<th>Top-5</th>
<th>UML CD</th>
<th>ORM/2</th>
<th>(E)ER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute (31.2%)</td>
<td>OT cardinality (29.0%)</td>
<td>Attribute (39.5%)</td>
<td></td>
</tr>
<tr>
<td>OT (21.2%)</td>
<td>OT (14.5%)</td>
<td>OT cardinality (22.1%)</td>
<td></td>
</tr>
<tr>
<td>OT cardinality (17.5%)</td>
<td>2-ary Rel. (14.4%)</td>
<td>2-ary Rel. (11.6%)</td>
<td></td>
</tr>
<tr>
<td>2-ary Rel. (12.4%)</td>
<td>Int. unique. (13.1%)</td>
<td>OT (11.5%)</td>
<td></td>
</tr>
<tr>
<td>OT subsumption (9.6%)</td>
<td>VT (10.4%)</td>
<td>single ID (7.7%)</td>
<td></td>
</tr>
</tbody>
</table>
### Ratios of entities aggregated by family and combined

<table>
<thead>
<tr>
<th>Ratio</th>
<th>UML</th>
<th>ORM/2</th>
<th>(E)ER</th>
<th>comb.</th>
</tr>
</thead>
<tbody>
<tr>
<td>model size:total entities</td>
<td>0.8</td>
<td>0.5</td>
<td>0.7</td>
<td>0.6</td>
</tr>
<tr>
<td>Attribute or Value type:Object type</td>
<td>1.5</td>
<td>0.7</td>
<td>3.5</td>
<td>1.7</td>
</tr>
<tr>
<td>binaries:n-aries</td>
<td>180.5</td>
<td>12.4</td>
<td>20.9</td>
<td>20.4</td>
</tr>
<tr>
<td>Subsumption(class):Object type</td>
<td>0.5</td>
<td>0.3</td>
<td>0.2</td>
<td>0.3</td>
</tr>
<tr>
<td>Relationship (non isa):Object type</td>
<td>0.8</td>
<td>1.1</td>
<td>1.1</td>
<td>1.0</td>
</tr>
<tr>
<td>Object type cardinality: other constraint</td>
<td>7.4</td>
<td>1.2</td>
<td>2.2</td>
<td>1.8</td>
</tr>
<tr>
<td>Single identification:other ID</td>
<td>–</td>
<td>17.3</td>
<td>5.4</td>
<td>8.4</td>
</tr>
<tr>
<td>role:relationship naming</td>
<td>4.3</td>
<td>(readings, mostly)</td>
<td>0.1</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Common features: Object type, Relationship, Object type cardinality, Subsumption (object type), Single identification, Disjoint and Complete object types.

⇒ Seems to fit some tractable language; which one(s)?

- Avail of Description Logic languages to gain insight in language and computational complexity
- Common core that covers ±87%; language-specific profiles
- There is no DL that matches precisely, but a PTIME language is feasible—$ALN_\forall$ for the Core Profile
- Good match is $CFDI_{nc}^\forall^\neg$ (PTIME), with n-aries, identifiers

---

8 Fillottrani, P.R., Keet, C.M. Evidence-based Languages for Conceptual Data Modelling Profiles. ADBIS’15. Springer LNCS. Poitiers, France, Sept 8-11, 2015. (accepted)

9 Fillottrani, P.R., Keet, C.M., Toman, D. Polynomial encoding of ORM conceptual models in $CFDI_{nc}^\forall^\neg$. DL’15, CEUR-WS vol. 1350, 401-414.
Logic foundation for profiles

- Common features: Object type, Relationship, Object type cardinality, Subsumption (object type), Single identification, Disjoint and Complete object types.

⇒ Seems to fit some tractable language; which one(s)?

- Avail of Description Logic languages to gain insight in language and computational complexity
- Common core that covers ±87%; language-specific profiles
- There is no DL that matches precisely, but a PTIME language is feasible—\(ALN+\) for the Core Profile
- Good match is \(CFDInc^\forall\) (PTIME), with n-aries, identifiers

---

8 Fillottrani, P.R., Keet, C.M. Evidence-based Languages for Conceptual Data Modelling Profiles. ADBIS’15. Springer LNCS. Poitiers, France, Sept 8-11, 2015. (accepted)

9 Fillottrani, P.R., Keet, C.M., Toman, D. Polynomial encoding of ORM conceptual models in \(CFDInc^\forall\). DL’15, CEUR-WS vol. 1350, 401-414.
Outline

1. Motivation

2. Unification approach
   - Metamodel
   - Transformations and intermodel assertions

3. Quantitative analysis

4. Conclusions
Conclusions

- Unifying ontology-driven metamodel
- Inter-model assertions and model transformation approaches with basic set of rules (1:1, transformations, and approximations)
- Quantitative analysis on feature usages
- Profile characterisation
Ongoing and future work

- Integrate these results into design tools
- ‘Scalability’ of graphical representation and inferences?
- Modularisation
Example: ICOM
References I

Paolo Atzeni, Paolo Cappellari, Riccardo Torlone, Philip A. Bernstein, and Giorgio Gianforme.
Model-independent schema translation.

Paolo Atzeni, Giorgio Gianforme, and Paolo Cappellari.
Data model descriptions and translation signatures in a multi-model framework.

Michael Boyd and Peter McBrien.
*Comparing and transforming between data models via an intermediate hypergraph data model.*

D. Calvanese, M. Lenzerini, and D. Nardi.
*Unifying class-based representation formalisms.*

Pablo R. Fillottrani, Enrico Franconi, and Sergio Tessaris.
*The ICOM 3.0 intelligent conceptual modelling tool and methodology.*

C. Maria Keet.
*Positionalism of relations and its consequences for fact-oriented modelling.*

Vilamoura, Portugal, November 4-6, 2009.
J.R. Venable and J.C. Grundy.

Integrating and supporting Entity Relationship and Object Role Models.


Pounamu: a metatool for multi-view visual language environment construction.
In IEEE Conf. on Visual Languages and Human-Centric Computing 2004, 2004.
Thank you!