Multi-Level Modelling for Interoperability

Andreas Jordan, Wolfgang Mayer and Markus Stumptner

University of South Australia

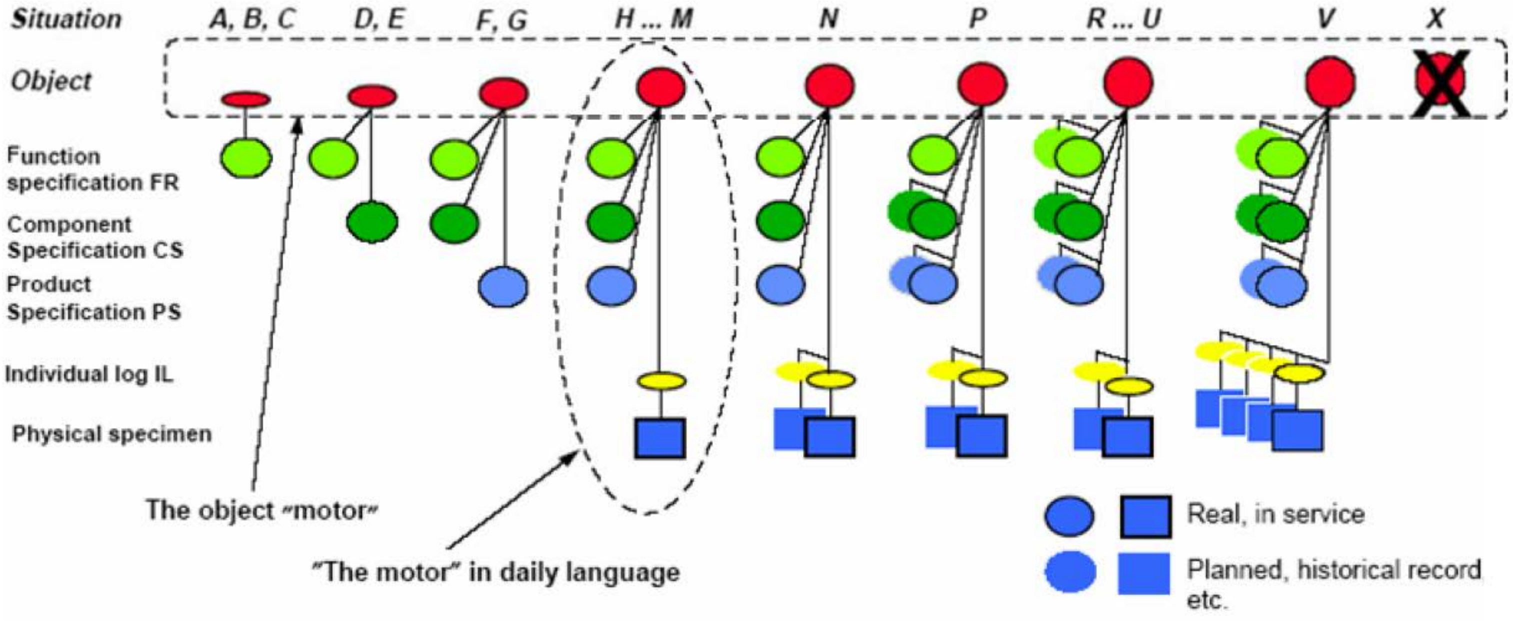
1st International Workshop on
Multi-Level Modelling (MULTI 2014)

Overview

- Motivation: Modelling complex domains (OGI Pilot)
- Modelling extensions
- Example
- Evaluation

Engineering Lifecycle

- A key concern: same component in a real system can be subject to multiple classifications.

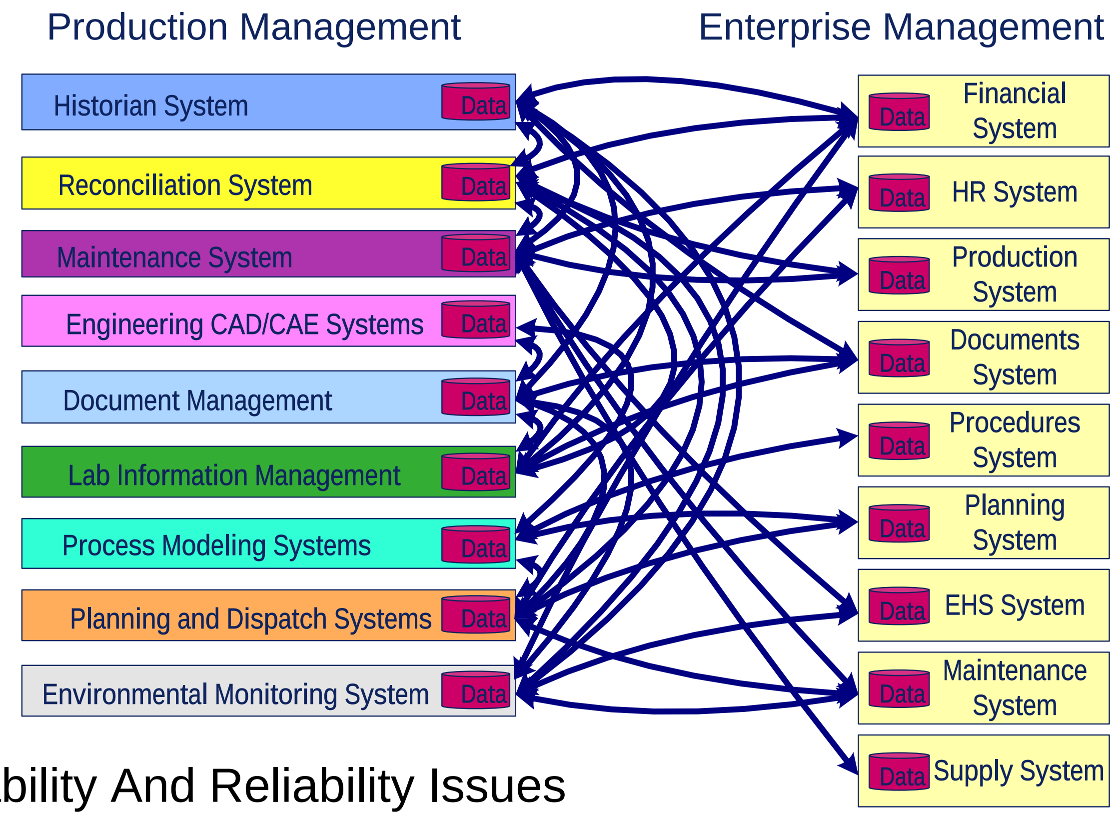


- Contribution: set of modelling extensions to overcome limitations of existing approaches (heterogeneous level)

Oil & Gas Interoperability Pilot

- Goal: Interoperability in the Oil & Gas Industry
- Automated Data Translation between different “software ecosystems”
- Based on the standards landscape:
  - “Reference Environment” (EPC): ISO 15926
  - “Execution Environment” (O&M): MIMOSA
- Multi-standards-organisation/multi-partner effort under ISO TC184/WG6
- Incrementally covering multiple use cases
- Youtube live demo video

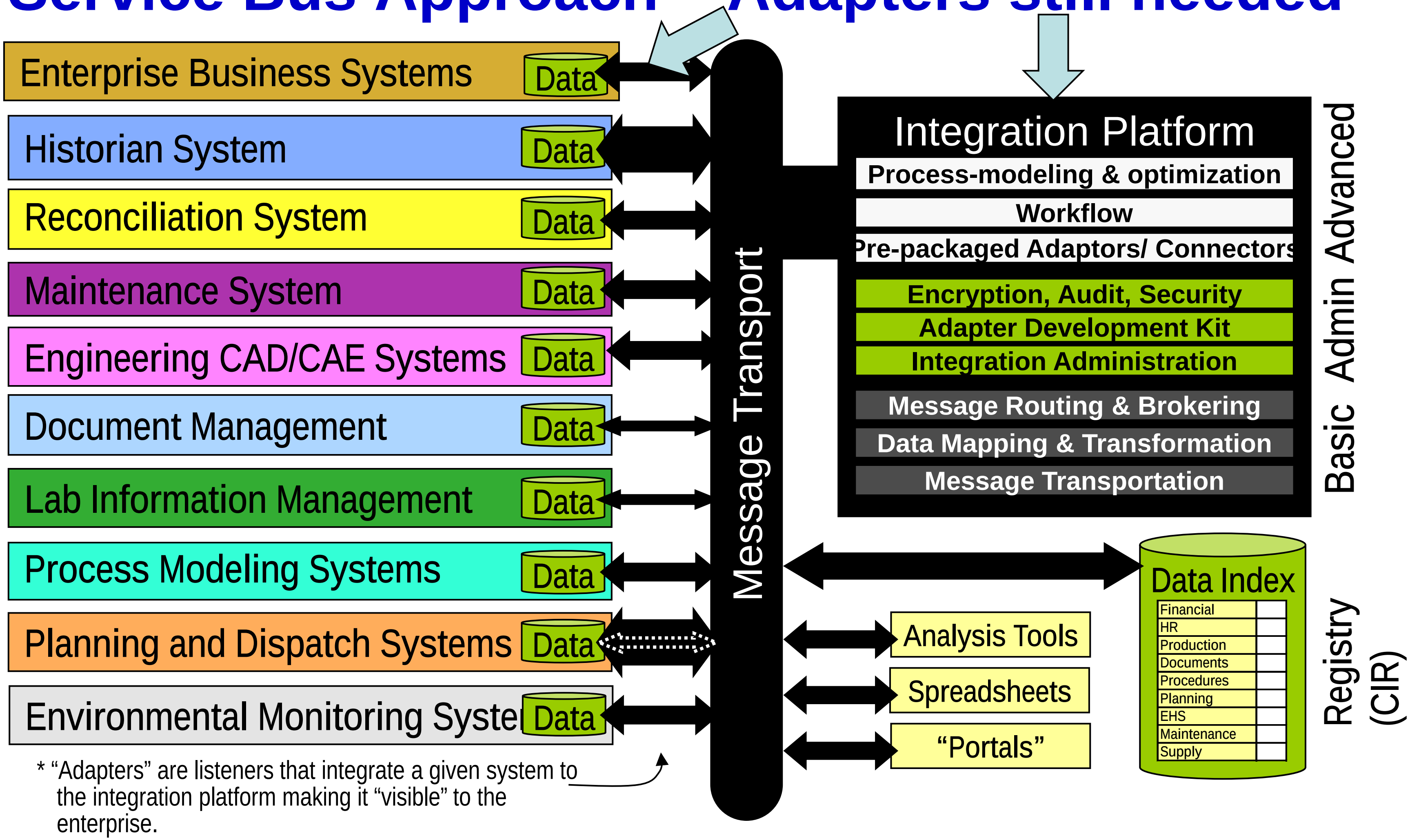
Data Exchange in a Production Enterprise



- System Stability And Reliability Issues
- No Single Version Of The Truth
- System Maintenance Issues
- Difficult Access To Multiple Systems/Applications

(courtesy of Emerson Process Management)

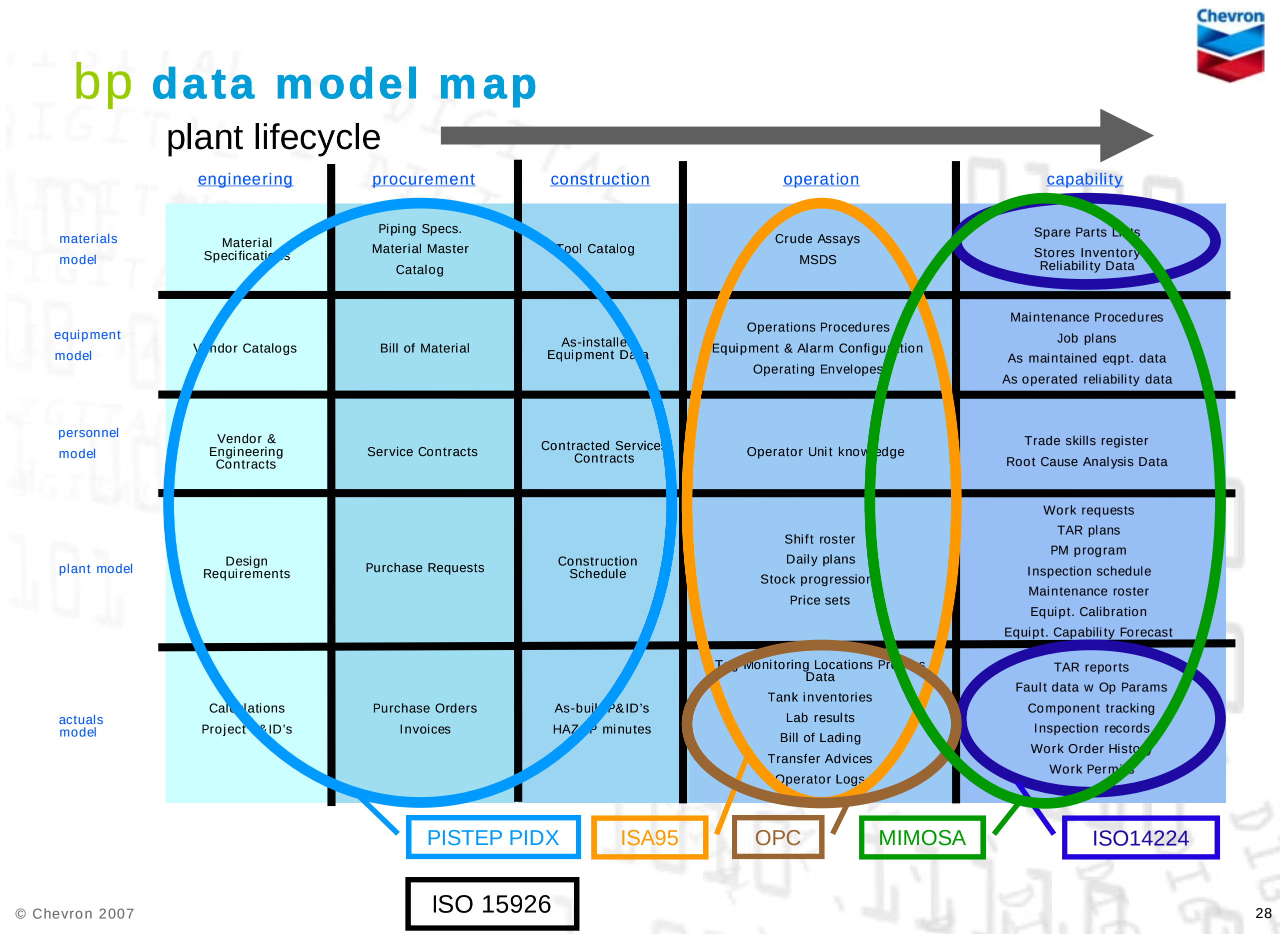
Service Bus Approach – Adapters still needed



Federation *NOT* Replication

Challenges

- Provision of a set of modelling mechanisms to allow other standards to be mapped to ISO 159261



Challenges

- Representing multiple levels of classification:

1. **Business level** which consists of complex taxonomies relevant from the business/ERP perspective
2. **Specification level** which provides the specifications of the physical entities.
3. **Physical entity level** where both designs and the physical entities of a product catalogue must be represented and have their own life-cycles which forms the physical entity level.

Modelling Extensions for Specialisation

- **Specialisation by extension:**
  - Adopts standard monotonic specialisation semantics
  - Extends a class by adding attributes, associations or behaviour
  - Introduces a new model level
- **Specification by refinement:**
  - Allows the introduction of subtypes that restrict the domain of the specialised class (e.g., by restricting the domains of properties and associations, or adding domain constraints on properties)
  - Does not introduce additional model levels
  - Allows for an arbitrary number of subtypes that simply refine the level of granularity

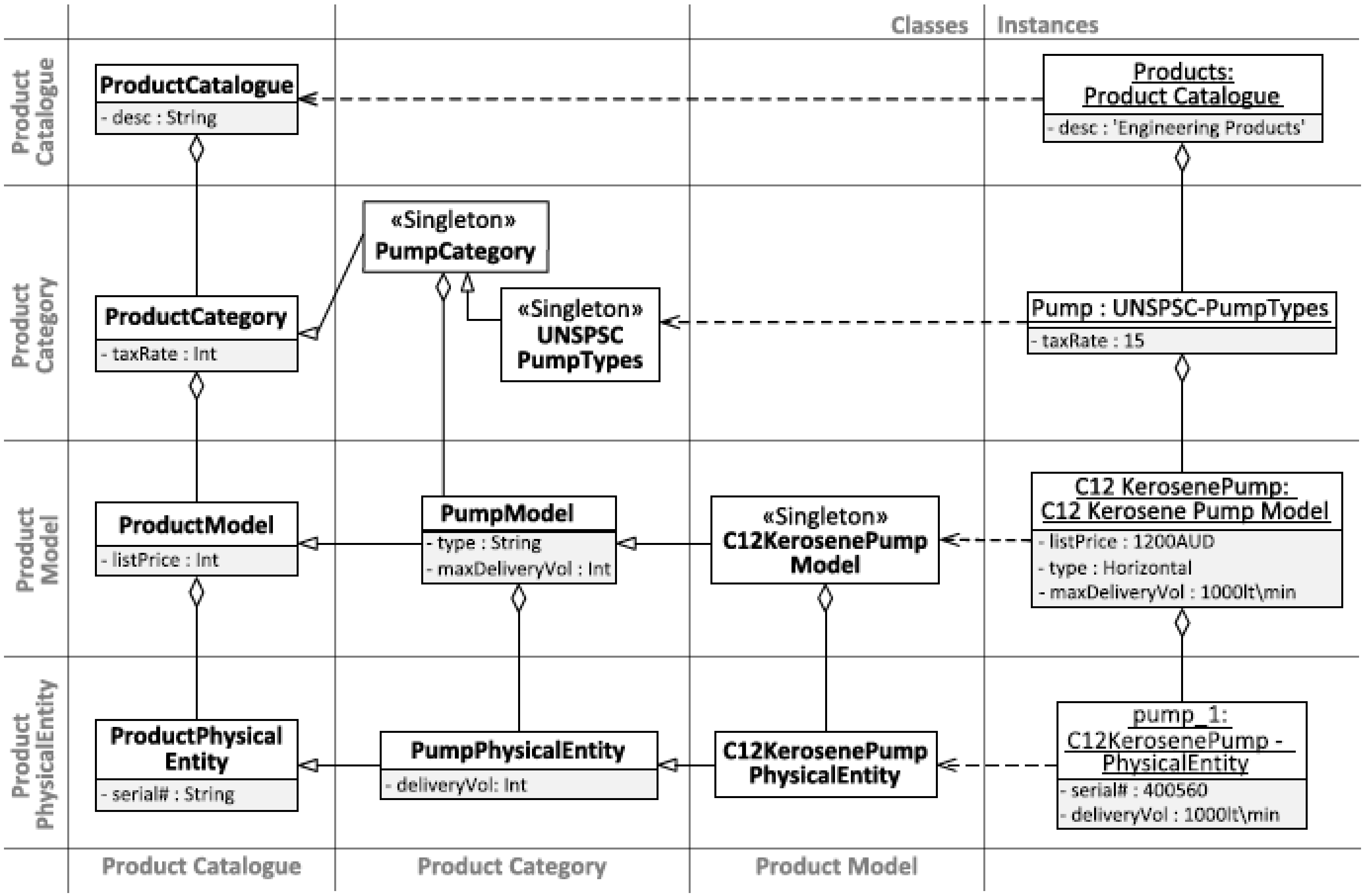
Modelling Extensions for Instantiation and Subset by Specification

- **Instantiation with extension:**
  - Allows for additional attributes, behaviour, etc. to be added to the concept that can then be instantiated or inherited further to lower model levels
- **Subset by specification:**
  - Represents the existence of a class of specification construct that identifies particular subtypes of another type
  - The specification exists at the same level as the type it refers to
  - Combined with *instantiation with extension*, this relationship can be used to construct the powertype pattern

Modelling Extensions to Associations

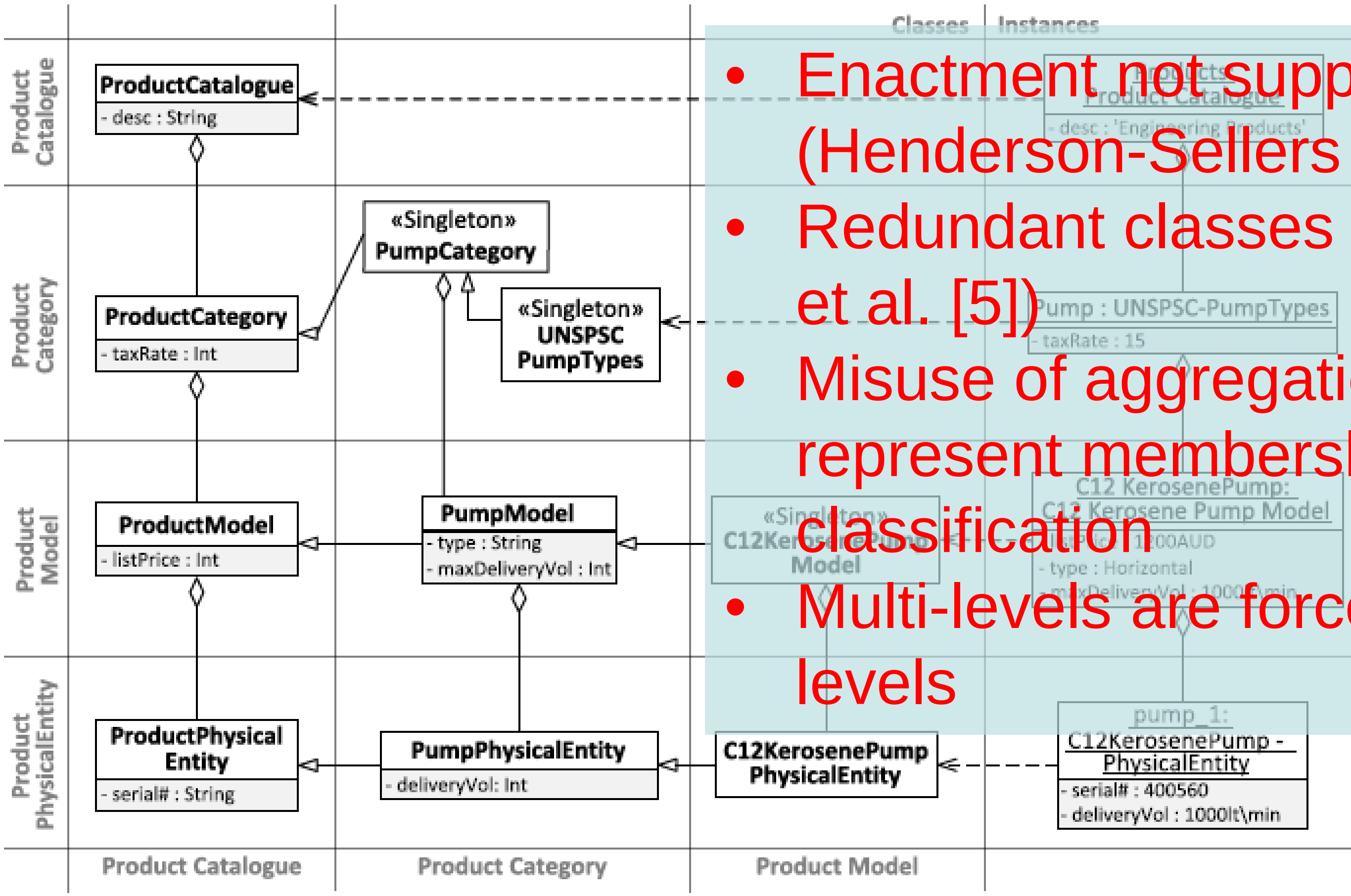
- **Member:**
  - Member associations can cross level boundaries
  - Requires the existence of a “primary” instantiation relation
  - Basic set membership relation that allows us to deal with multiple inheritance
- **Specification by enumeration:**
  - Represents a relationship between concepts A and B that describes how the extensions of the sets of entities that they represent are related i.e. the members of A are instances of B

Example



- Product catalogue modelled in plain UML, using generalisation, instantiation and aggregation (adapted from Neymayr et al. [5])

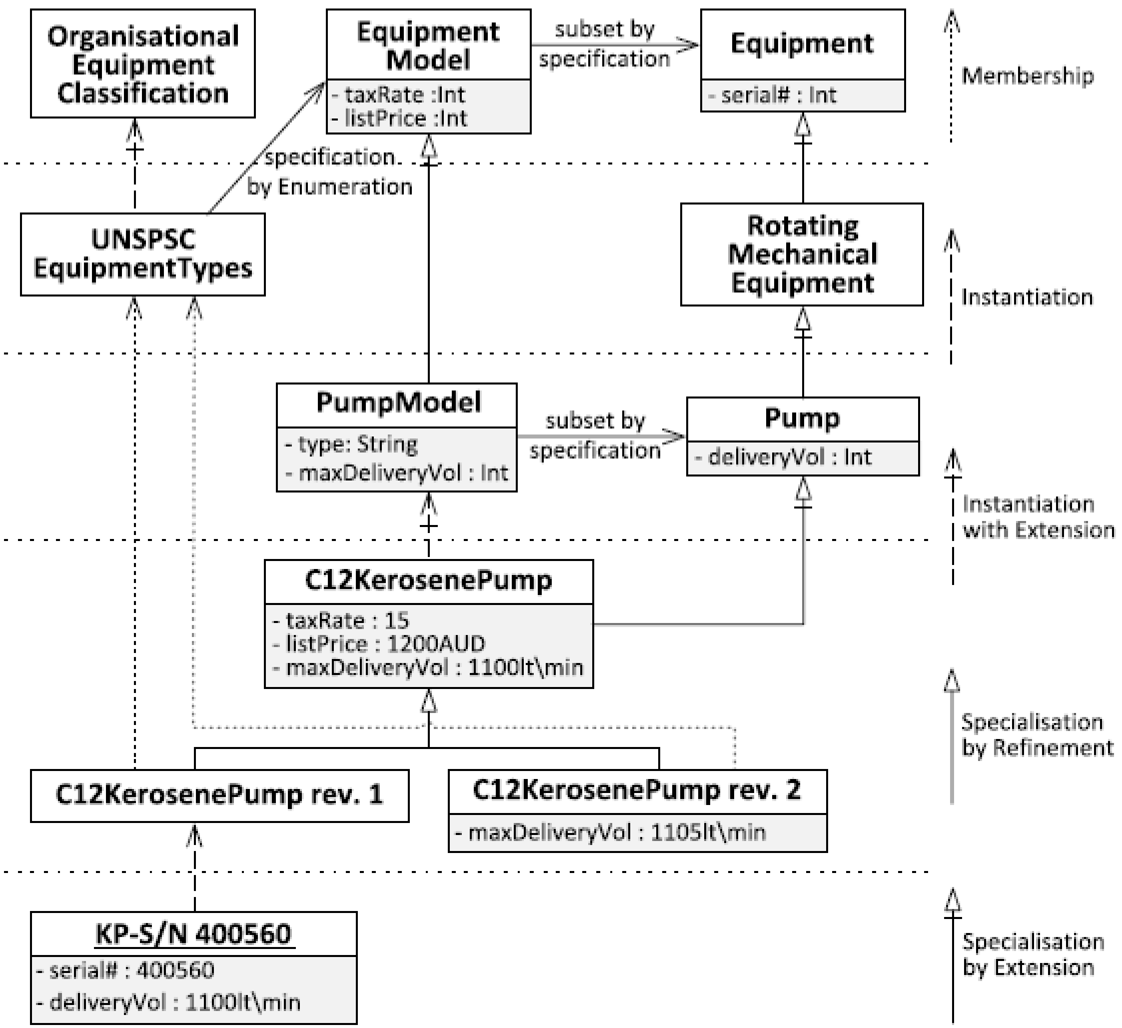
Example: Problems



- Enactment not supported (Henderson-Sellers et al. [6])
- Redundant classes (Neumayr et al. [5])
- Misuse of aggregation to represent membership and classification
- Multi-levels are forced into two levels

- Product catalogue modelled in plain UML, using generalisation, instantiation and aggregation (adapted from Neymayr et al. [5])

Example Modelled in our Framework



Evaluation

- Criteria introduced by Neumayr et al. [5]:
  - *Compactness*: encompasses modularity and redundancy-free
  - *Query Flexibility*: queries can be performed to access the model elements at the different levels of abstraction.
  - *Heterogeneous Level-Hierarchies*: Introducing new levels without causing changes to other levels.
  - *Multiple Relationship Abstractions*: Whether an approach supports multi-level abstraction of relationships.
- Additional criteria:
  - *Locality of Attributes & Relationships*: Attributes/relationships are defined locally if they are defined on the model elements closest to where they are used.
  - *Clarity of Relations' Semantics*: Relations have clearly delineated semantics from other relations.

Ongoing Work

- An Ontological Core for Conformance Checking
- Currently the ontology is not “live” in the transformation
  - Different computational environment
  - Map to ontology from the transformation model
- Extension to process ontology for complex use cases

Conclusion

- Effective exchange of information about processes and industrial plants, their design, construction, operation, and maintenance requires sophisticated information modelling and exchange mechanisms.
- Need increases with the growing tendency for direct interaction of information systems from the sensor level to corporate boardroom level.
- Introduction of modelling primitives that support the multilevel level modelling paradigm for information integration in heterogeneous information systems

Questions?