GEFÖRDERT DURCH DIE DFG Deutsche Forschungsgemeinschaft – SFB-1608 – 501798263

CONVIDE CRC 1608

MULTI@MODELS 24 • September 22–27, 2024, Linz, Austria

TOWARDS DEEP REACTIONS IN MULTI-LEVEL, MULTI-VIEW MODELING

Thomas Weber, Arne Lange, Erik Burger, Lars König, Martin Armbruster

Karlsruher Institut für Technologie

Colin Atkinson, **Monalisha Ojha**, Mohammad Sadeghi

UNIVERSITY OF MANNHEIM

INTRODUCTION

CONVIDE CRC 1608

Overview

Integration of Multi-Level Modeling and SUM based Multi-View Modeling.

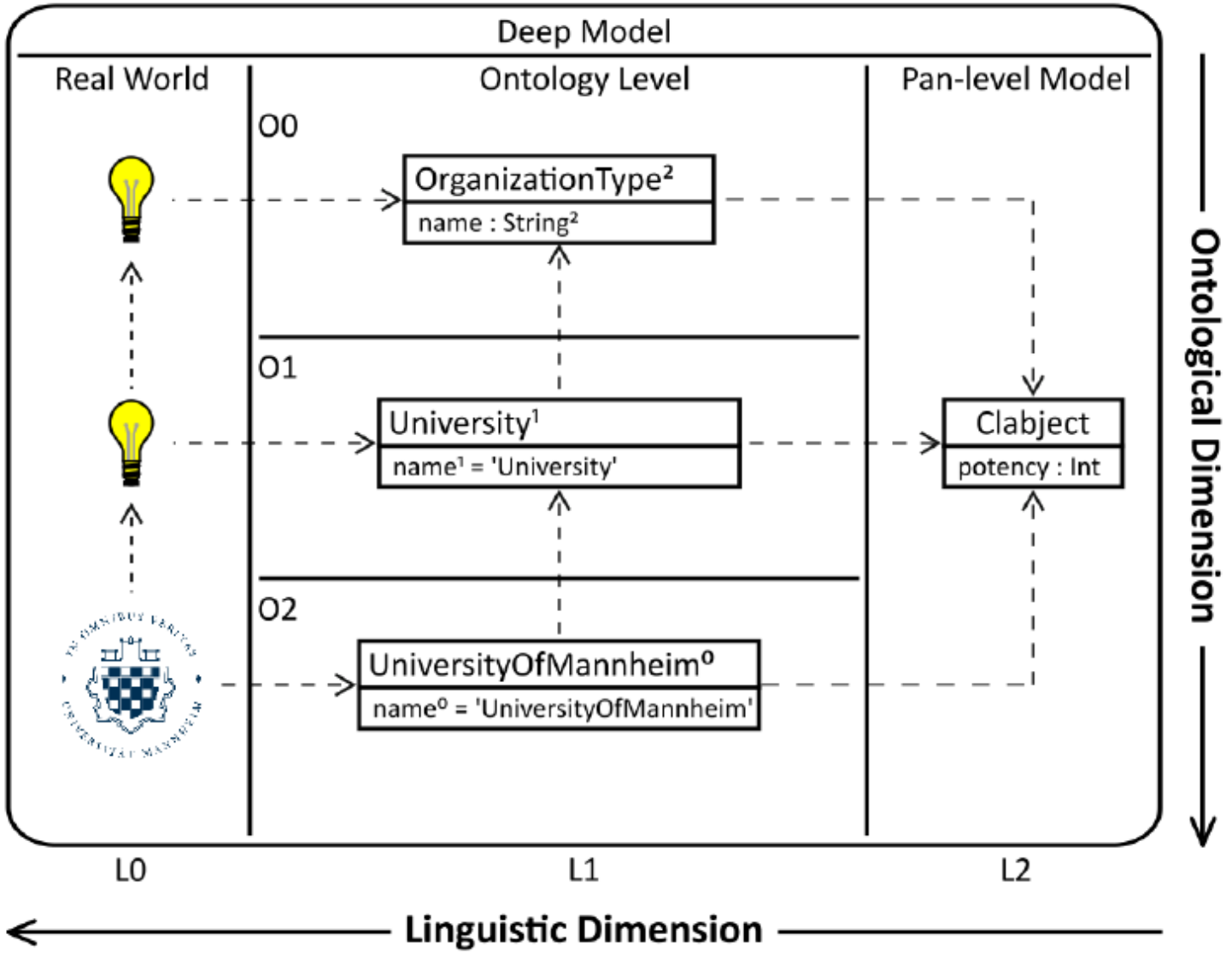
Key Problem

Few tools integrate these paradigms at a fundamental level.

Solution

Extending Vitruvius's Reactions language to support multi-level V-SUMs in Melanee’s multi-level modeling environment.

MULTI-LEVEL MODELING



SUM BASED MULTI-VIEW MODELING

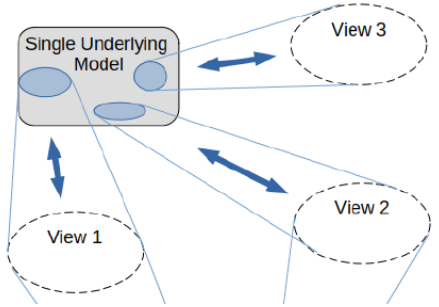
CONVIDE CRC 1608

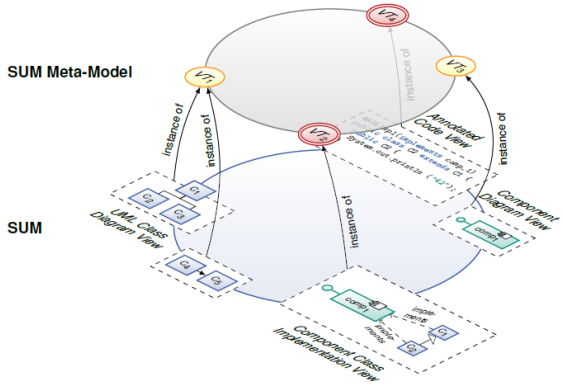
Projective

- Uses a Single Underlying Model (SUM) as the source, from which specific views are derived by projection.

Synthetic

- Multiple models are synthesized into a Single Underlying Model (SUM), integrating various aspects into one unified representation.





REACTIONS

CONVIDE CRC 1608

```
reactions: umlToJavaClass
in reaction to changes in uml
execute actions in java

reaction CreatedUmlClass {
    after element uml::Class
        created and inserted as root
    call {
        val umlClass = newValue
        createJavaClass(umlClass)
    }
}

routine createJavaClass(uml::Class umlClass) {
    match { /* retrieve_elements */ }
    create { /* create_elements */ }
    update { /* update_models */ }
}
```

Reactions Declaration

Reactions Definition

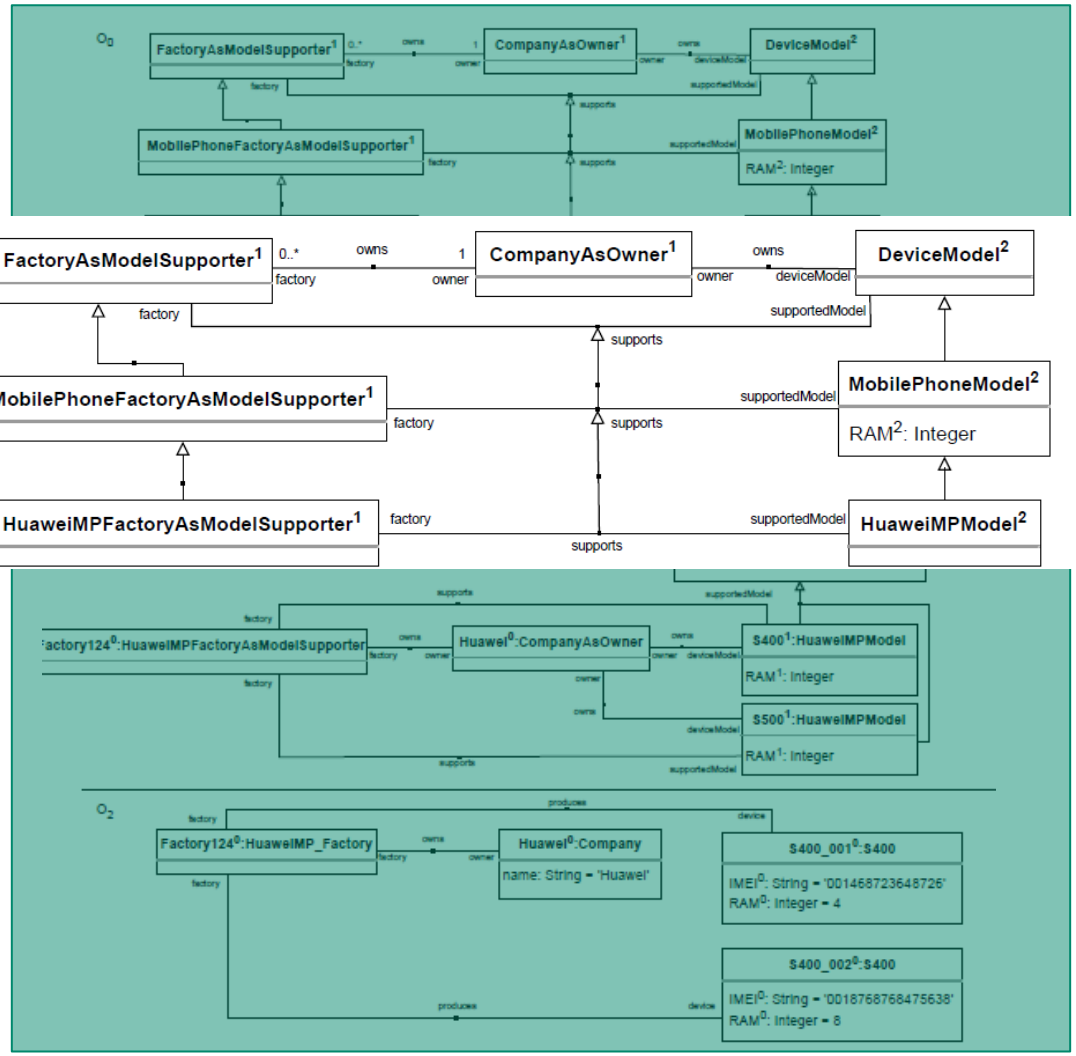
Routine Definition

Consistency specification rules

The Reactions Language

MOTIVATING EXAMPLE*

COLLABORATION COMPARISION CHALLENGE(MULTI 2022)



O0 Level

Multi-level Model

Melanee

*Thomas Kühne and Arne Lange. 2022. Melanee and DLM: a contribution to the MULTI collaborative comparison challenge. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22).

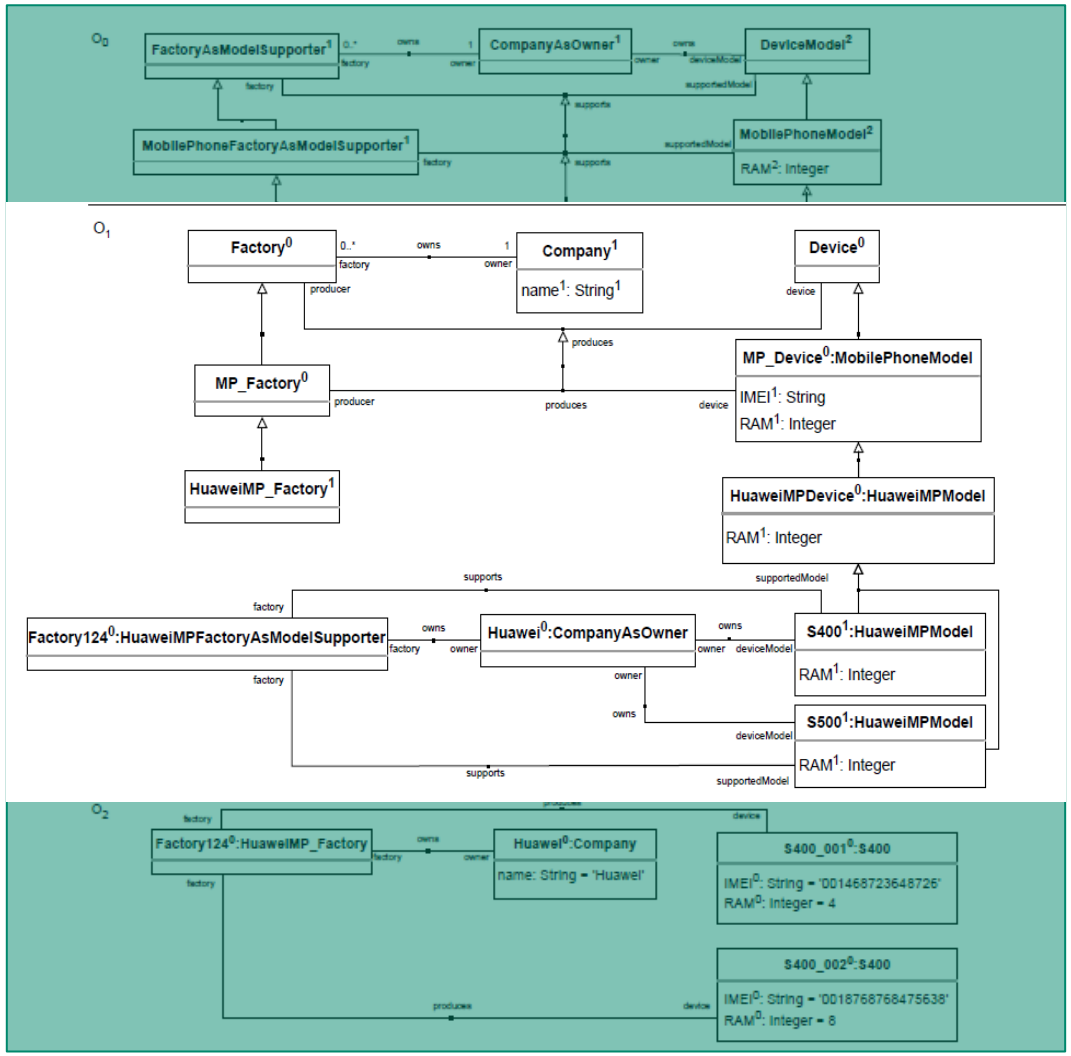
MOTIVATING EXAMPLE*

COLLABORATION COMPARISION CHALLENGE(MULTI 2022)

CONVIDE CRC 1608

Multi-level Model

Melanee



O1 Level

*Thomas Kühne and Arne Lange. 2022. Melanee and DLM: a contribution to the MULTI collaborative comparison challenge. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22).

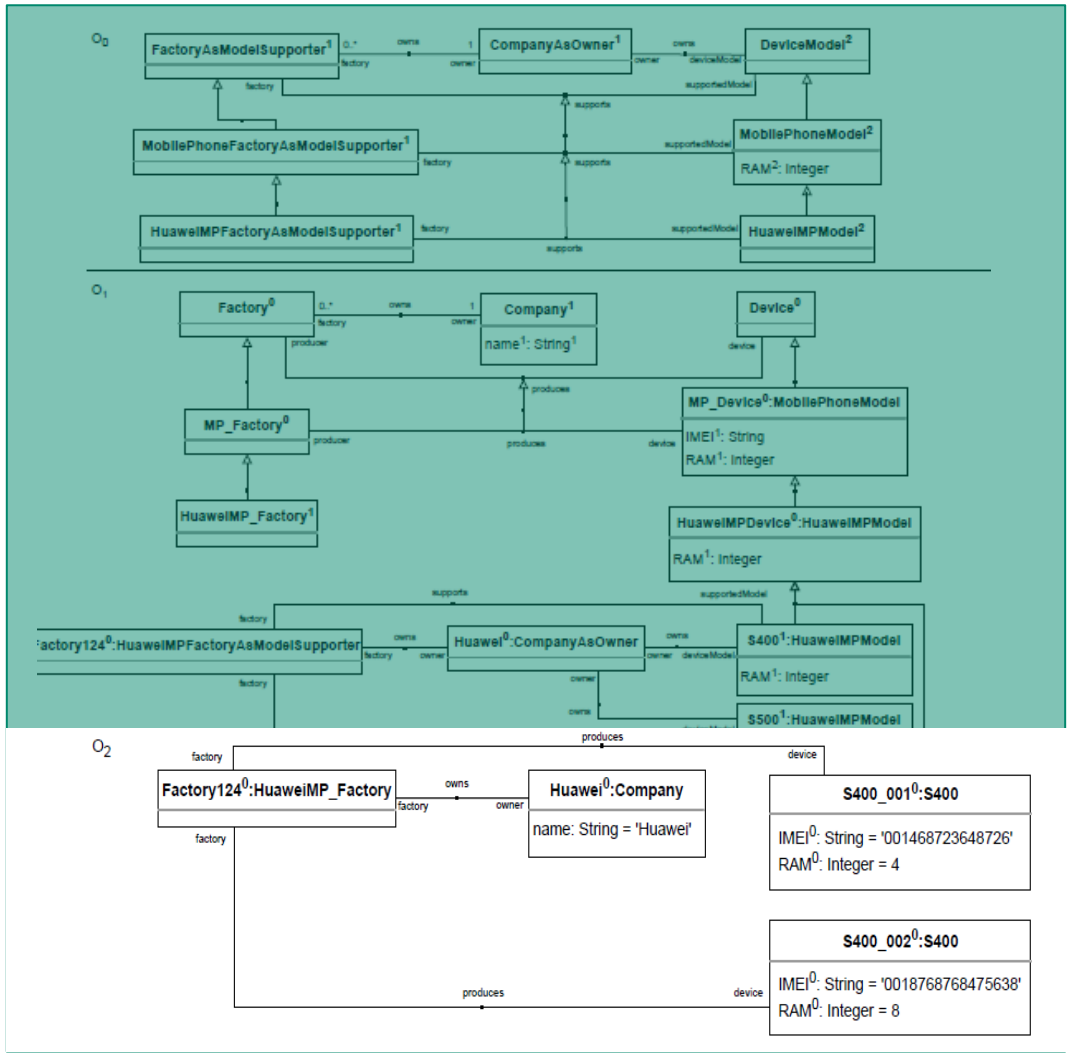
MOTIVATING EXAMPLE*

COLLABORATION COMPARISION CHALLENGE(MULTI 2022)

CONVIDE CRC 1608

Multi-level Model

Melanee



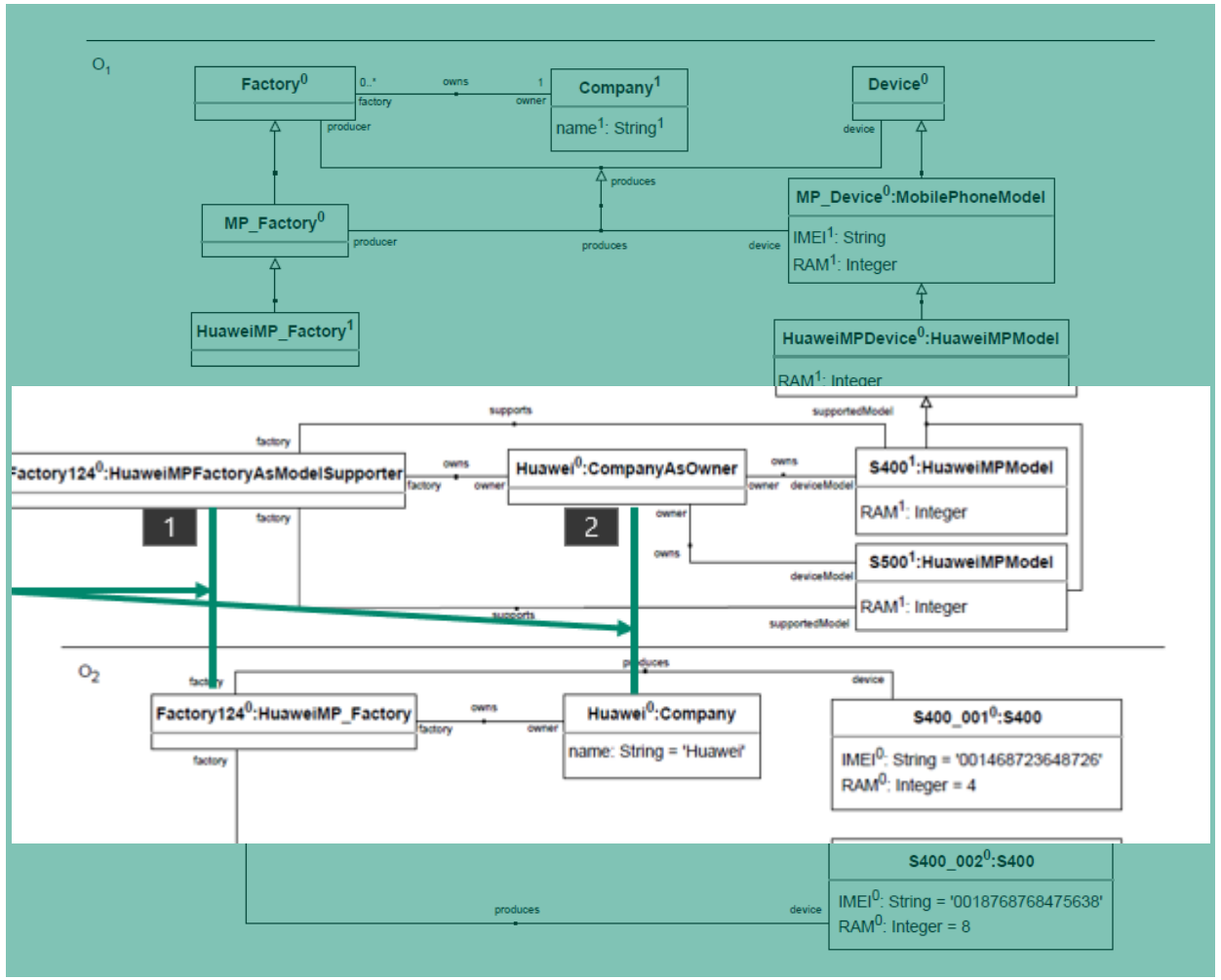
O2 Level

*Thomas Kühne and Arne Lange. 2022. Melanee and DLM: a contribution to the MULTI collaborative comparison challenge. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22).

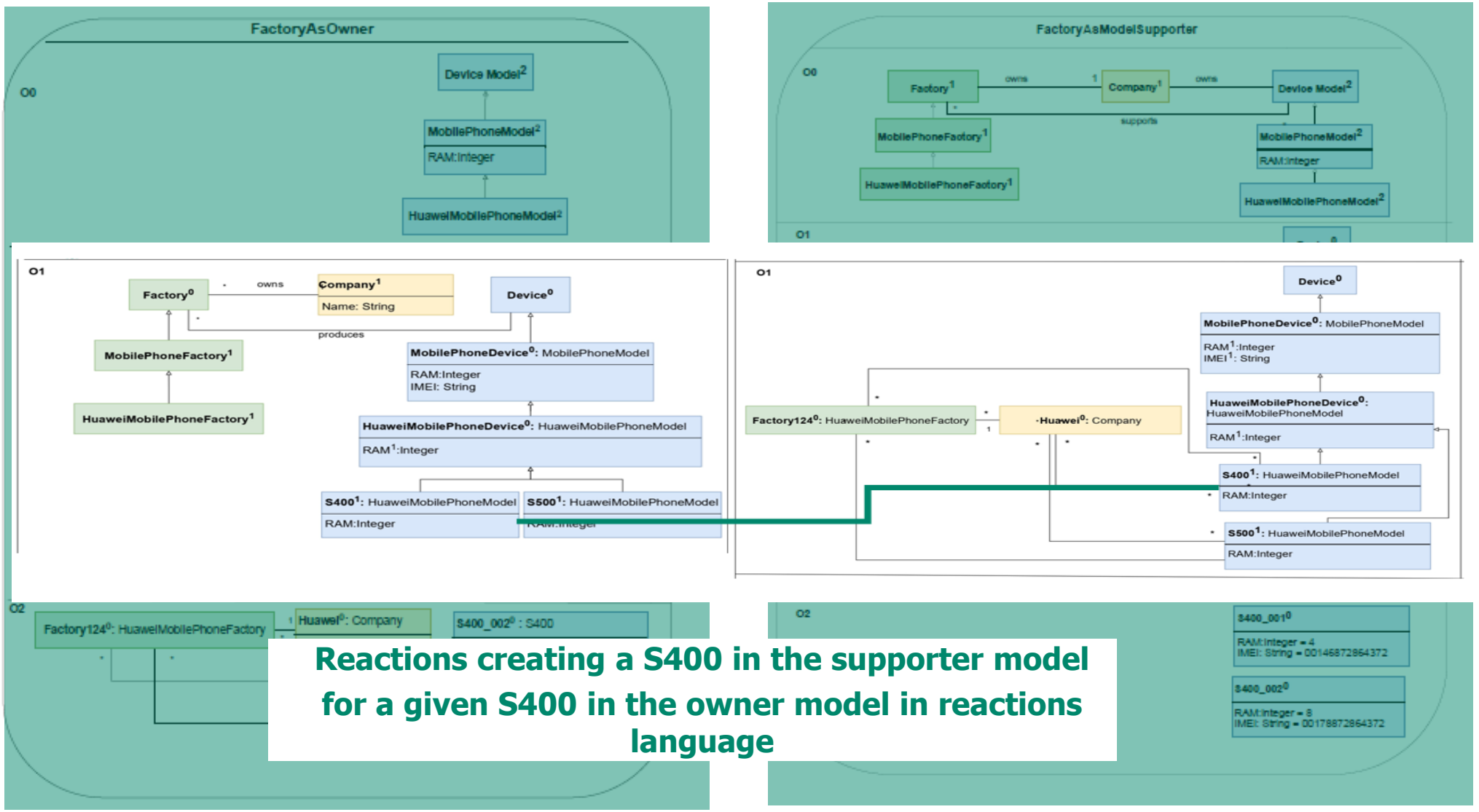
EXAMPLE

Previous Solution with Accidental Complexities

Duplication of elements had to be used as a "trick" to solve the challenge



PROPOSED SOLUTION



Reactions creating a S400 in the supporter model for a given S400 in the owner model in reactions language

CURRENT STATE

CONVIDE CRC 1608

1. Reactions language is primarily designed to support two-level models, i.e., models with only two classification levels: types and instances.
2. The language lacks the advanced query capabilities necessary to write precise responses for changes across multiple classification levels
3. The current version requires developers to manually specify and check the meta-levels of elements, leading to inefficiencies in handling deep models with multiple layers

```
1  reaction NewS400Inserted {
2   after element owner::OwnedElement inserted in owner::Level[
        content]
3   with {
4    affectedEObject.level === 2
5    && (newValue instanceof Entity)
6    && (newValue as Entity).directType.name == "S400"
7   }
8   call {
9    insertNewS400(newValue as Entity)
10  }
11 }
```

Reaction to describe the insertion of an OwnedElement into a Level

REQUIREMENTS OF DEEP REACTION

CONVIDE
CRC 1608

RQ1 Reactions language should be aware of the deepness of the models and meta-models.

RQ2 Reactions language should also be aware of levels.

RQ3 The deep Reactions language should have reflective capabilities

DEEP REACTION LANGUAGE FEATURES

CONVIDE CRC 1608

```
1  import deep "pathtolml_model" as owner
2  import deep "pathtolml_model" as supporter
3
4  reactions: owner2supporter
5  in reaction to changes in owner below level 1
6  execute actions in supporter below level 1
7
8  reaction NewS400Inserted {
9      after direct element owner::S400 inserted in owner at level 2
10
11 reactions: owner2supporter
12 in reaction to changes in owner below level 1
13 execute actions in supporter below level 1
14
15 routine addNewS400(owner::S400 oldS400) {
16     match {
17         val supDeviceLevel = retrieve level 2 in supporter
18     }
19     create {
20 reaction NewS400Inserted {
21     after direct element owner::S400 inserted in owner at level 2
22
23         supDevice.name = oldS400.name
24         addCorrespondenceBetween(oldS400, supDevice)
25         supDeviceLevel.content.add(supDevice)
26     }
27 }
```

Import of Deep Models using ‚deep‘

Support for Deep Types

Restriction of changes to certain levels

Making it Level Aware

CONCLUSION

CONVIDE
CRC 1608

1. Improved the challenge solution by separating it into two models, reducing complexity.
2. Identified improvements needed in the Reactions language during implementation.
3. Proposed a deep Reactions language for dynamic change handling in deep models.
4. Future work focuses on implementing these features in Vitruvius for multi-level modeling.

GEFÖRDERT DURCH DIE DFG Deutsche Forschungsgemeinschaft – SFB-1608 – 501798263

CONVIDE CRC 1608

THANK YOU!

TECHNISCHE UNIVERSITÄT DRESDEN

Technische Universität München

UNIVERSITY OF MANNHEIM

KIT Karlsruher Institut für Technologie