



Towards Ontology-driven Requirements Engineering (ODRE)

Katja Siegemund¹, Edward J. Thomas², **Uwe
Aßmann**¹, Jeff Pan², Yuting Zhao²

1 Technische Universität Dresden, Germany

2 University of Aberdeen, UK

SWESE

Oct 24, 2011

Deficiencies of Current RE Methods



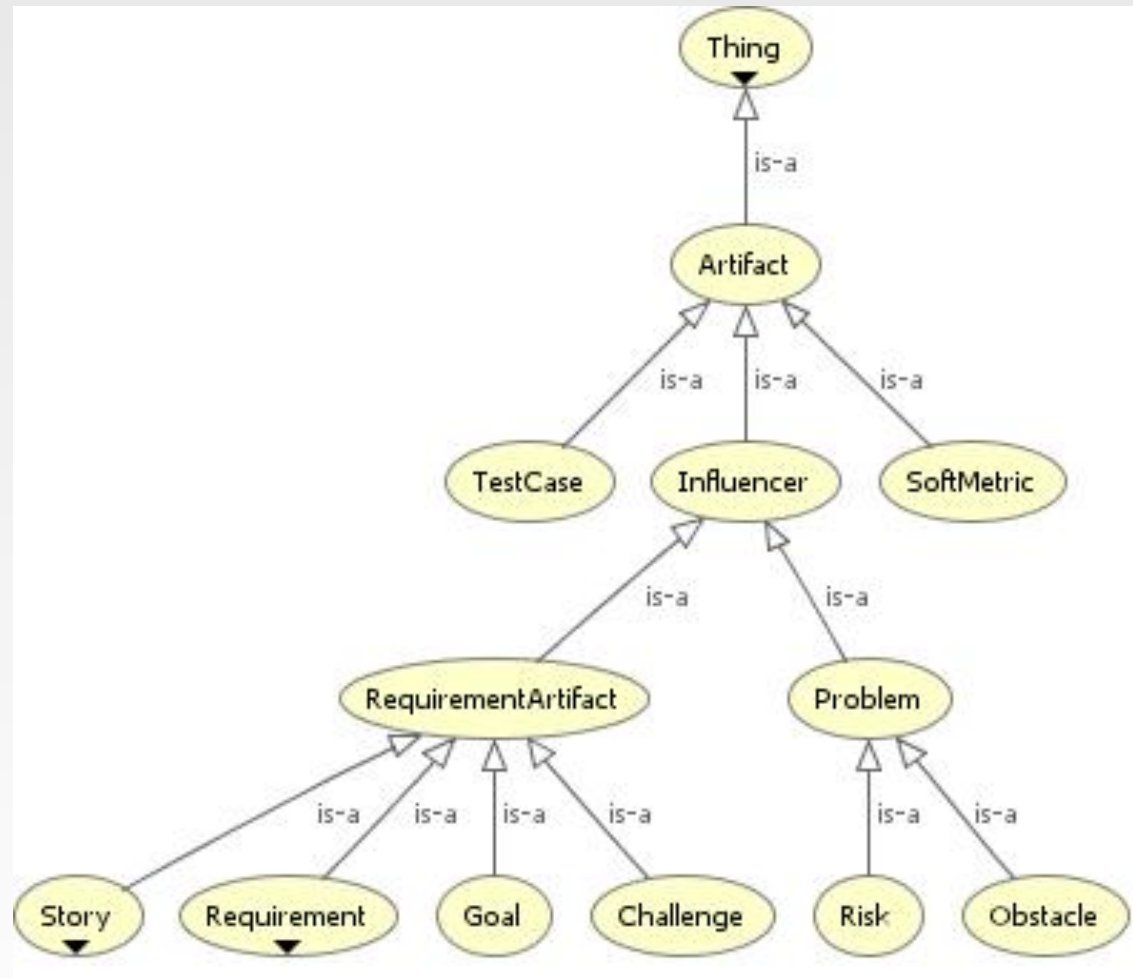
- Relationships among requirements are inadequately captured
- Requirement problems (e.g. conflicts, incompleteness) are detected too late or not all
 - Causal relationship between consistency, completeness and correctness [Zowghi2002]
 - Completeness and consistency are not verified
- Models for RE need richer and higher-level abstractions (goals, problems) [Mylopoulos1999]

Use of Reasoning for RE

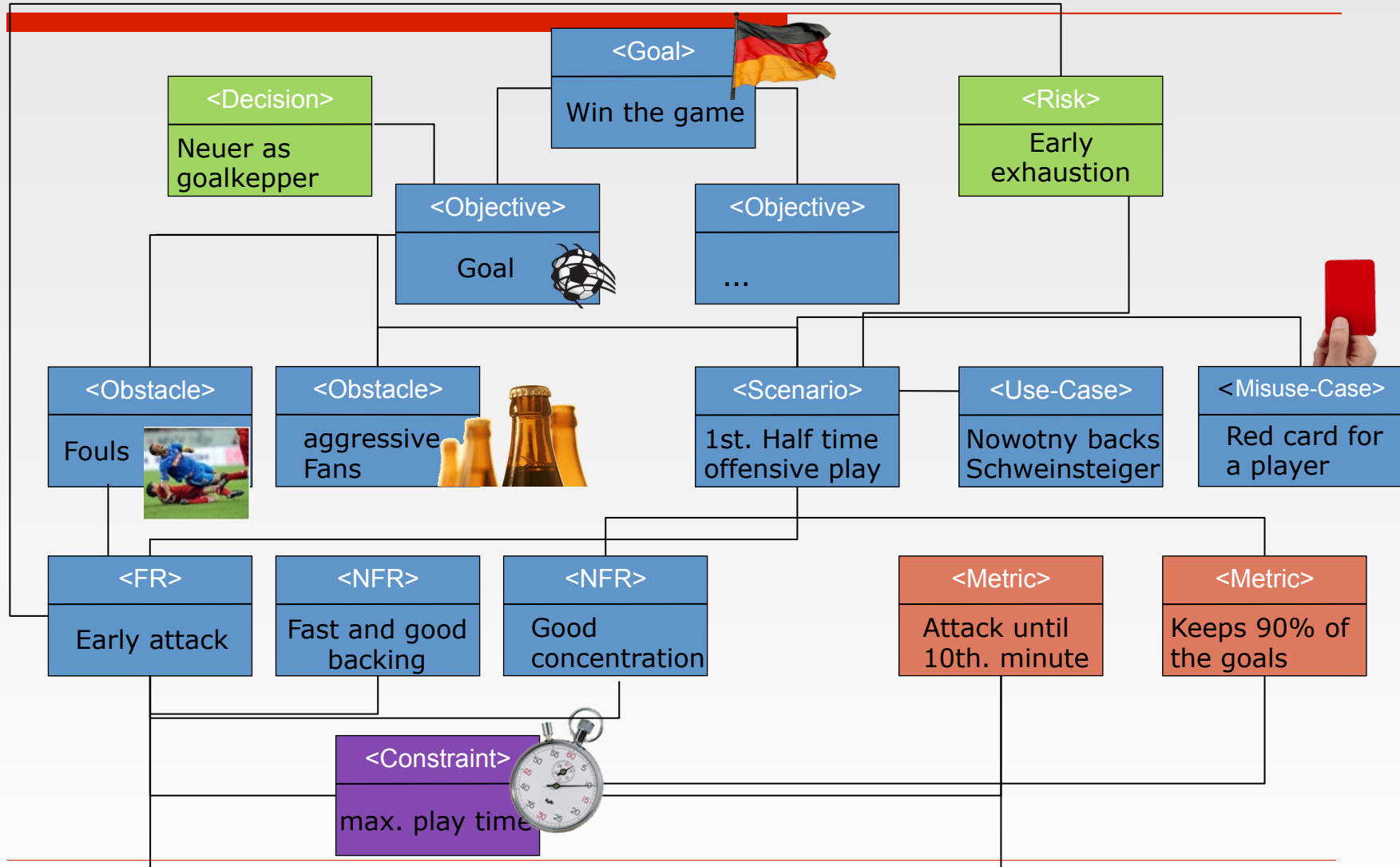


- Support Goal-oriented RE (GORE)
 - Provide metamodel with a huge set of relevant metadata and requirement relationships
- Provide meaningful checks for completeness and consistency
- Specific suggestions to repair inconsistencies and incompleteness

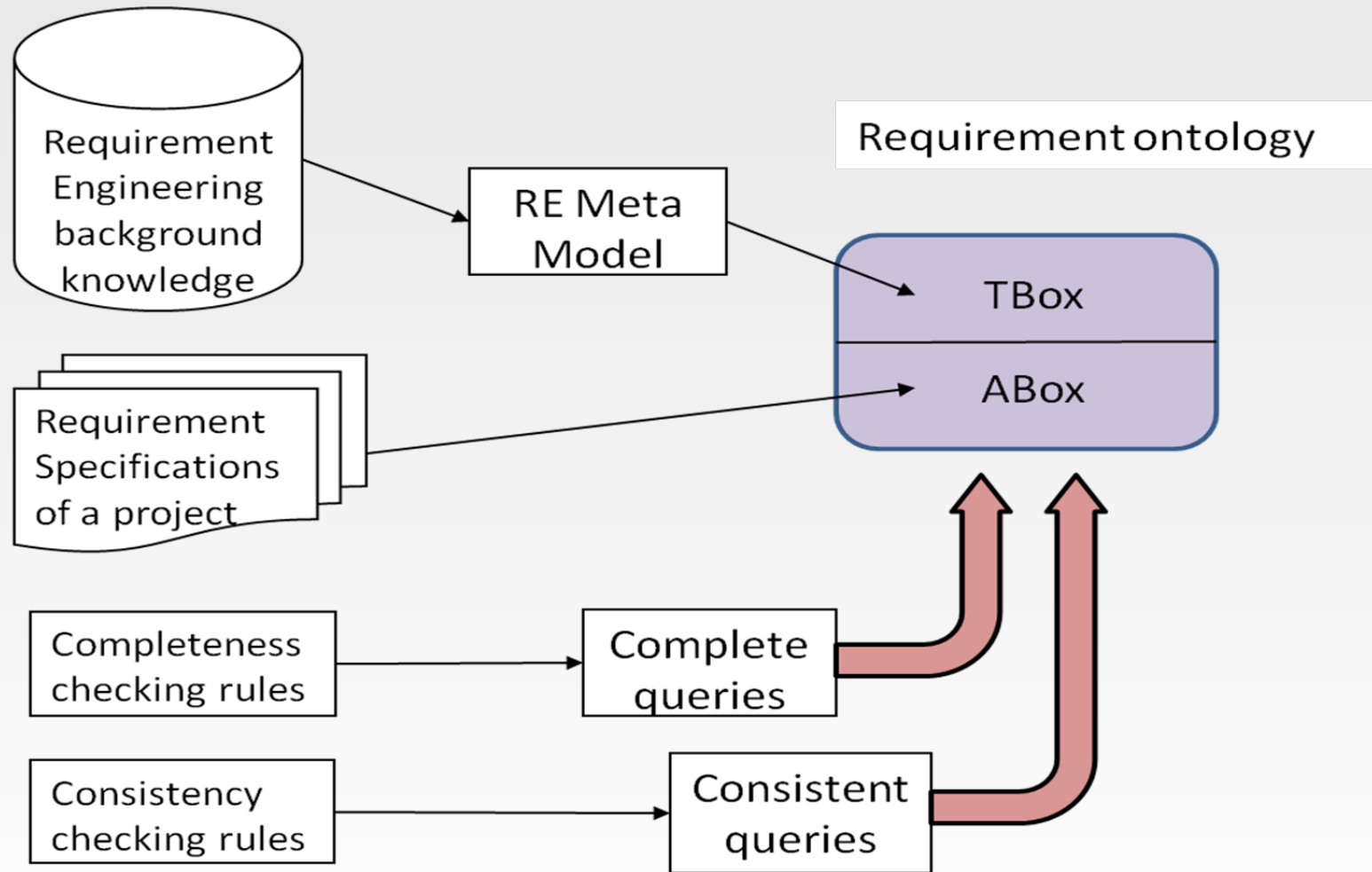
GORE – Tbox



Goal-Oriented RE (Motivation Example)



Reasoning for RE - Architecture



Reasoning for RE – Completeness Check



- E.g. “Every Functional Requirement (FR) must define whether it is mandatory or optional.”
- GORE needs
 - 46 rules
 - Implemented as SPARQL queries
 - Requirements Model deemed incomplete if specific rule fails
 - Closed World for negation as failure supported by SPARQL 1.1 and TrOWL reasoner

Reasoning for RE – Completeness Check (Example)



Metadata
(requirements, goals,
relationships, ...)

Every FR must define whether it is mandatory or optional

IF FR is NOT mandatory AND NOT optional

THEN

print *error*: "You did not specify whether the following FRs are mandatory or optional: [FR_n]."

"Please specify whether these FRs are mandatory or optional."

Reasoning for RE – Completeness Check (Example)



Extract of individuals and relationships:

isRelatedTo(Goal2;UseCase7)

NonFunctionalRequirement (NonFunctionalRequirement1)

IsOptional(NonFunctionalRequirement1; true)

FunctionalRequirement(FunctionalRequirement1)

Error.

You did not specify whether the following FR are mandatory or optional:

FunctionalRequirement1. Please specify this attribute for the FR:

FunctionalRequirement1. Every FR must specify AT LEAST ONE requirement relationship.

Reasoning for RE – Consistency Check



- GORE needs 6 consistency rules
 - among requirement artefacts (valid relations between requirement artefacts)
 - Based on a chosen subset of requirement artefacts
 - Consistency rules encoded as DL axioms
- Instance specific error messages resulting from validation displayed by Guidance Engine

Reasoning for RE – Consistency Check (Example)



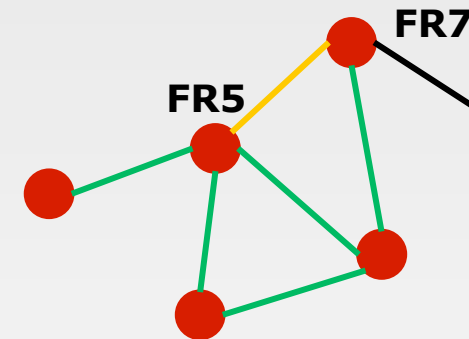
isExclusionOf (Functional Requirement5; Functional Requirement7)
ChosenRequirement(Functional Requirement5)
ChosenRequirement(Functional Requirement7)

Error.

The following requirements exclude others:

FunctionalRequirement5.

Please choose one of the following options:



Suggestion.

Exclude the following requirements from the chosen requirement set: FunctionalRequirement5. **OR**

Find alternatives for: FunctionalRequirement5 or

Revise the requirement relationships of(FunctionalRequirement5, FunctionalRequirement7).

- Minor evaluations with available Use Cases
 - Problem: available requirement specifications do not provide sufficient information (much less than could be captured by ODRE)
- Primary evaluation within MOST Project
 - Capture all requirement artefacts
 - Detect all inconsistencies and incomplete metadata
- Main evaluation planned

Conclusion



- All Requirement artefacts and meaningful relationships can be captured within an Ontology Metamodel
- ODRE Approach detects **inconsistent** and **incomplete** requirements
- Specification of requirements uses OWA
 - Verification needs CWA
- First evaluation proves applicability for medium requirement specifications

Outlook



- Further work concentrates on:
 - Guidance
 - Traceability
 - Integration into Eclipse

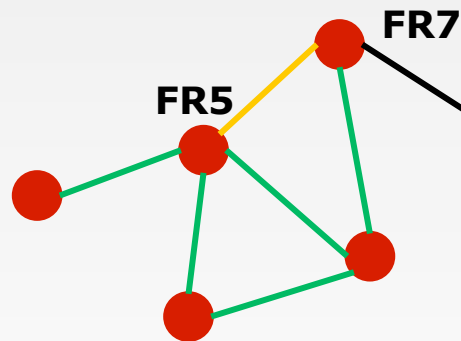
References



- [Mylopoulos1999] John Mylopoulos, Lawrence Chung, and Eric Yu. From Object-oriented to Goal-oriented Requirements Analysis. *Communications of the ACM*, 42(1):31-37, 1999.
- [Zowghi2002] Didar Zowghi and Vincenzo Gervasi. The Three Cs of Requirements: Consistency, Completeness, and Correctness. In *Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality, (REFSQ'02)*, 2002.
- [Lamsweerde2000] Axel van Lamsweerde. Requirements Engineering in the year 00: A Research Perspective. In *International Conference on Software Engineering*, pages 5-19, 2000.



Metadata
(requirements, goals,
relationships, ...)





□ **Consistency check of requirement selection (6 rules)**

Excluding requirements must not be included in one set.

IF excluding requirements are included in one set
THEN print *error*: "The following requirements exclude Others: [R_n]."

"Please choose one of the following options:

Exclude the following requirements: [R_n],

Find alternatives for [R_n] or

~~Revise the requirement relationships of [[R_x, R_y],...]."~~

- Lamsweerde defines goals as "declarative statements of intent to be achieved by the system under consideration" [Lamsweerde2000]

- Benefits of GORE:
 - Goals provide a meaningful criterion for sufficient completeness of a requirement specification
 - Specification of pertinent requirements
 - relationships between goals and requirements can help to choose the best one
 - Concrete requirements may change over time whereas goals pertain stable
 - Goals drive the identification of requirements