

Requirements Engineering II

KAOS

Knowledge Acquisition in au**t**omated **S**pecification

"Inconsistencies are the rule, not the exception"

Daniel Pfeifer, Stefan Hutter, Sebastian Müller

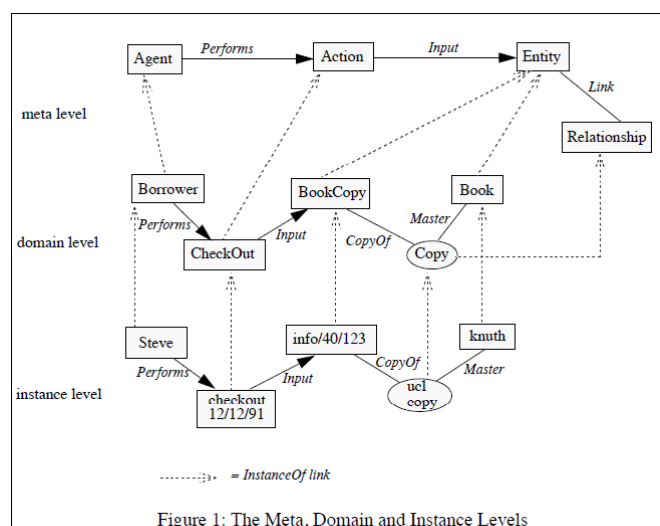
16.11.2009

Agenda

- KAOS – Main Ideas and Language Constructs
- Goal-Driven Requirements Elaboration
- Conflict Detection and Resolving

The KAOS Layers

- External Semantic Net Layer
 - Elaborate Goals, Constraints, Agents, Objects and Actions and the Linking inbetween them
 - AND/OR Graphs to show Dependencies of Requirements /Goals and Alternative Solutions



*Domain
independent
abstractions
(KAOS)*

*Application
domain*

*Specific
instance of
domain-level
concept*

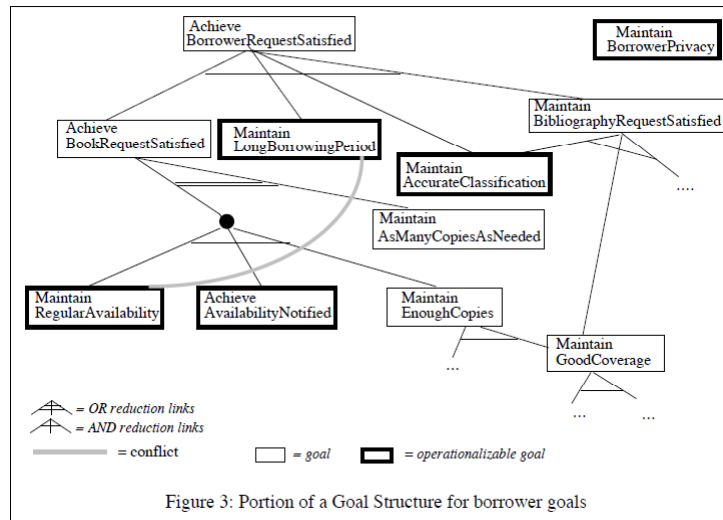
- Inner Formal Assertion Layer
 - Formal Specification Language for Goals and Constraints using Real Time Temporal Logic
 - Views

```
Goal Achieve[ParticipantsConstraintsKnown]
Concerns Meeting, Participant, Scheduler
Refines MeetingPlanned
RefinedTo ConstraintsRequested, ConstraintsProvided
InformalDef A meeting scheduler should know the
constraints of the various participants invited
to the meeting within some deadline d after
invitation.
FormalDef   $\forall m: \text{Meeting}, p: \text{Participant}, s: \text{Scheduler}$ 
Invited(p, m)  $\wedge$  Scheduling(s, m)
 $\Rightarrow \Diamond_{\leq d} \text{Knows}(s, p.\text{Constraints})$ 
```

*Concept**Parent Goal**Sub Goals**Description**Formalization*

Hide Complexity, Use Formal Patterns!

- Using Domain independent Meta-Level Knowledge
- Refining Goals into Subgoals (AND/OR)
 - Refinement Tactics
- Automation by using Formal Language
 - Checks and Heuristics



Conflict Detection

- Regressing Negated Assertions
- Divergence Patterns
- Divergence Identification Heuristics

Conflict Resolving

- Assertion Transformation
 - Avoiding Boundary Conditions
 - Alternative Refinement, Restoration or Weakening Goals
 - Divergence Resolution Heuristics
 - Conflict Anticipation
- Object Transformation
 - Object Refinement
 - Agent Refinement