

i* and Tropos

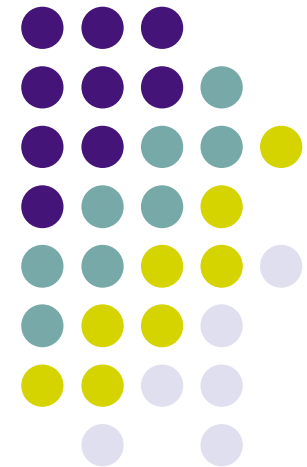
within: Requirements Engineering II

by

Reto Zenger (05-709-597)

Michael Küchler (06-712-681)

16.11.2009





Agenda

- Motivation
- The main ideas
- Quality of early-phase requirements
- Translation of goal-oriented requirements into software solutions



Motivation

- *The objective, in these “late-phase” requirements engineering tasks, is to produce a requirements document to pass on to the developers. (Yu 1997)*
- *Less attention has been given to supporting the activities that precede the formulation of the initial requirements. [...] The emphasis here is on understanding the “**whys**” that underlie system requirements. (Yu 1997)*



The main ideas

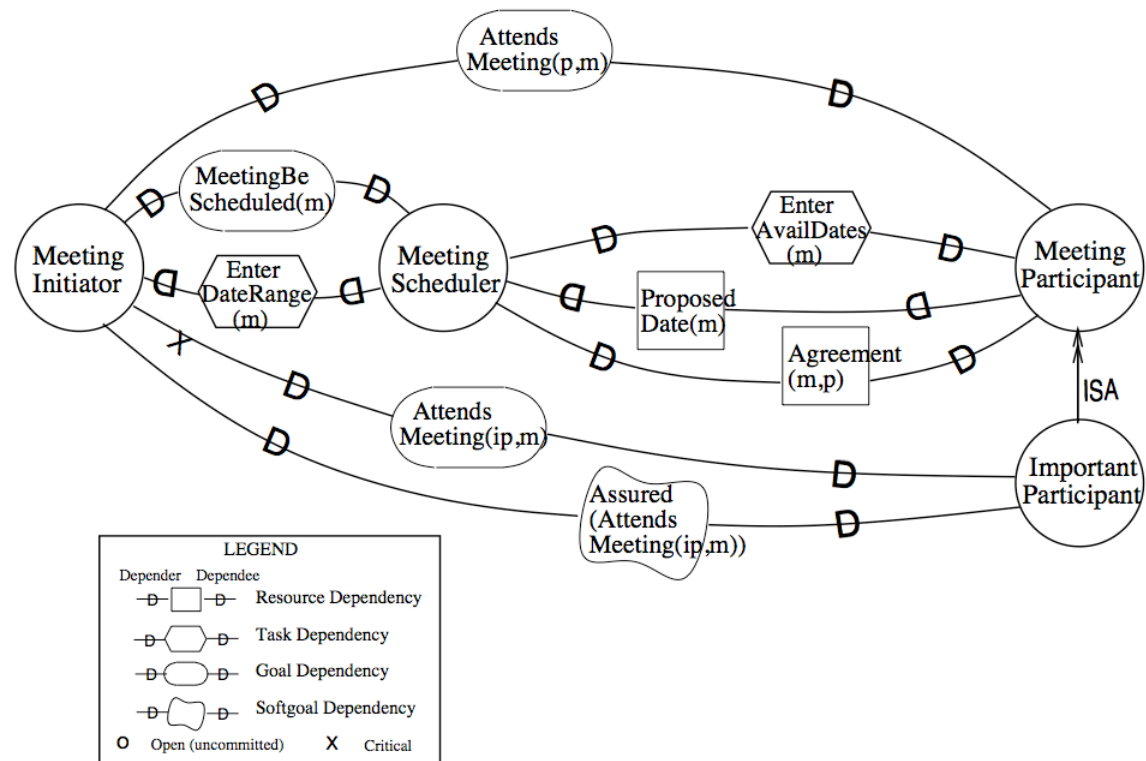
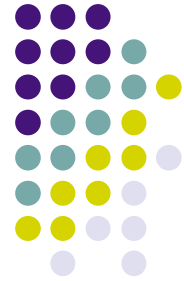
- Emphasize the “**early phase**” of requirements engineering
 - Receives few attention traditionally
- Identify the **goals/agents** from which the requirements arise
 - What is the needs of the user
 - How can the needs be addressed
- What are the rationales behind the goals
 - Ask “**Why**”



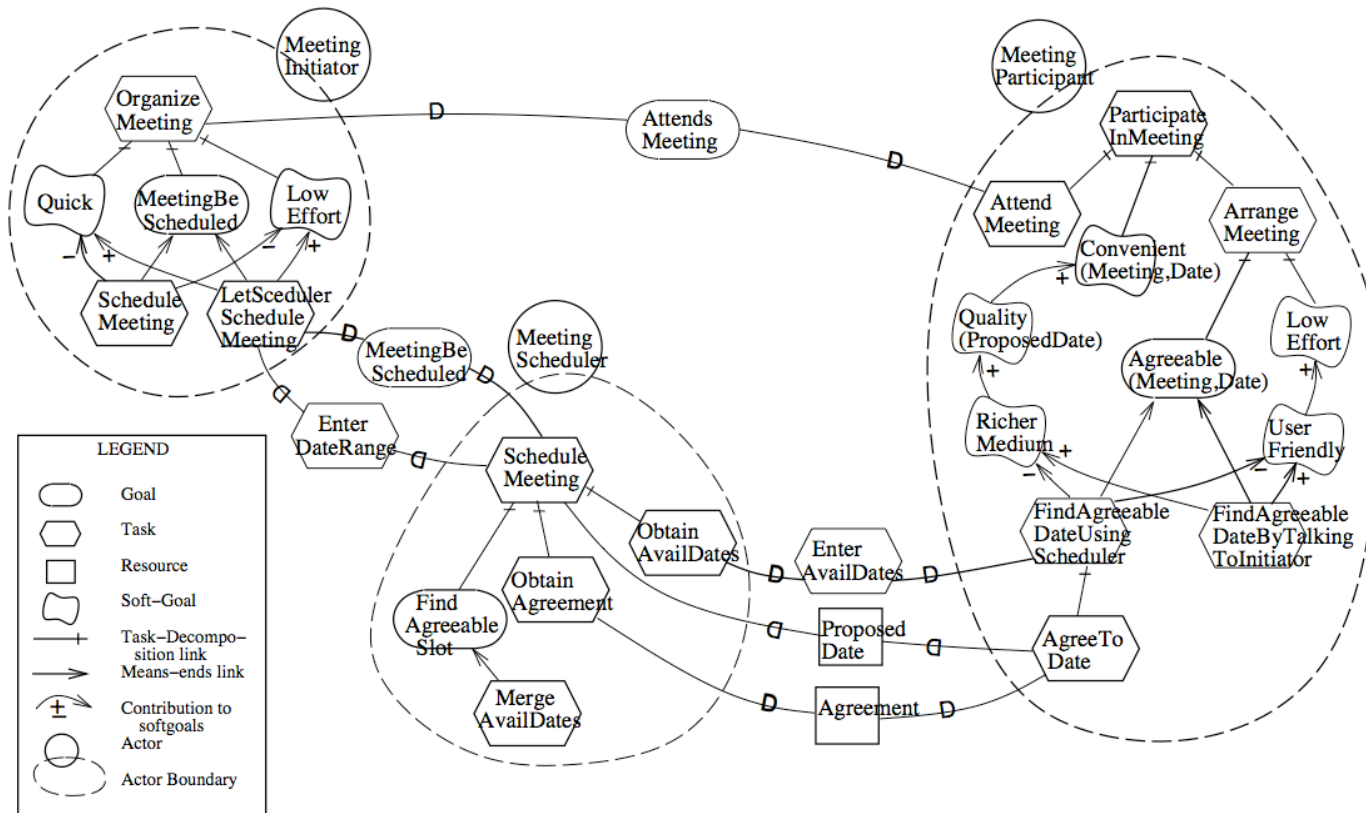
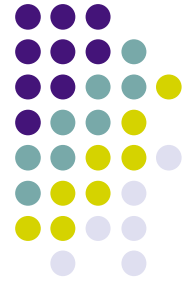
i* provides two models

- Strategic dependency model
 - Describes the dependency relationships among various actors
 - Black box view on agents
- Strategic rationale model
 - Describes the stakeholder interests and concerns, and how they might be addressed
 - White box view on agents

Strategic Dependency Model



Strategic Rationale Model



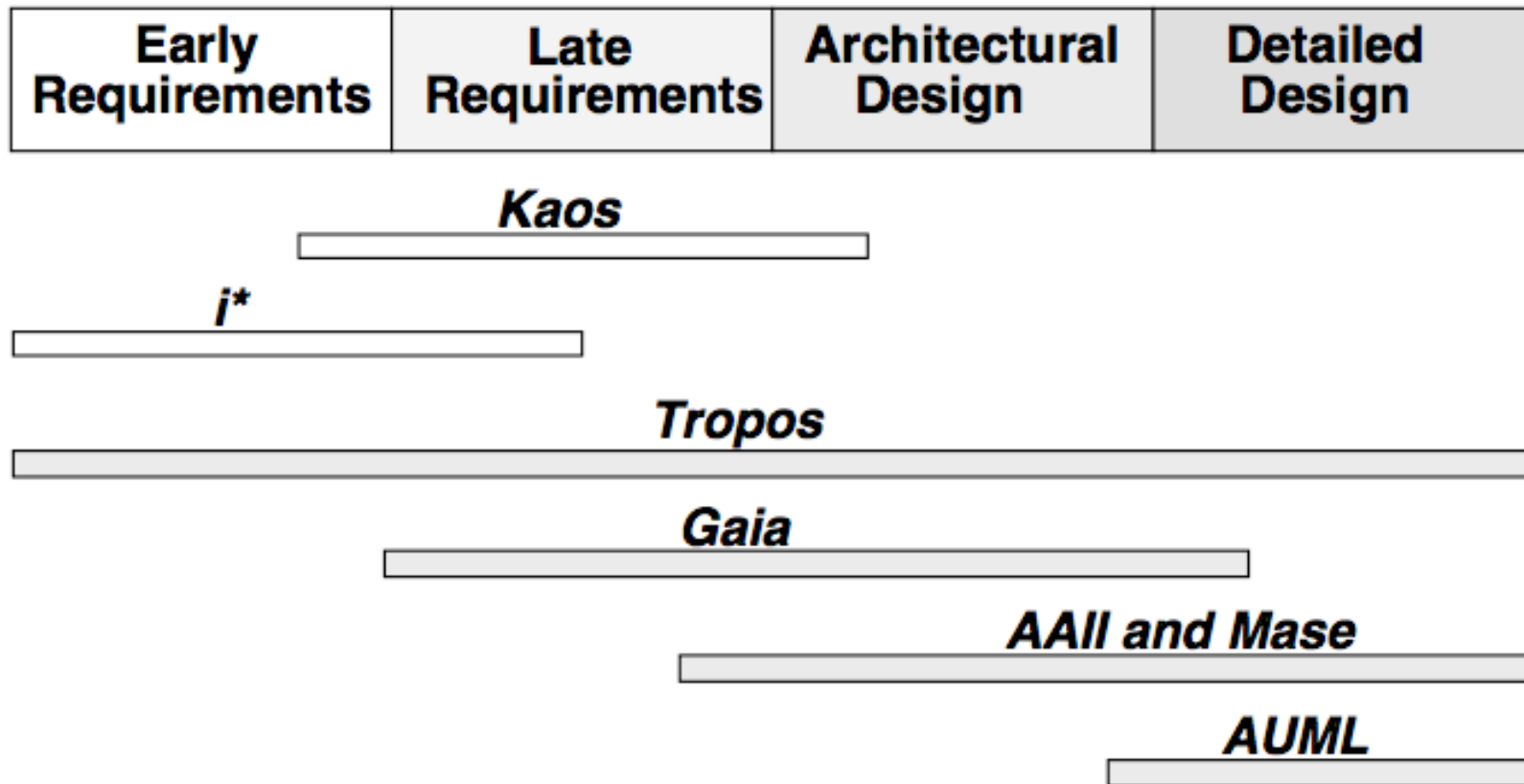


Tropos: a bigger scope

- Emphasizes the early-phase RE like i^*
- Agent oriented
- Intended to support **all** analysis and design activities in the software development process
 - From application domain analysis down to the system implementation
 - Bigger scope than i^* (and others)



Scope of different methods



Verification is still a problem



- Methods like i^* and Tropos do not solve the problem of verification
- Most formal methods are designed to work in later phases of development
- **Formal Tropos**, an extension of Tropos, attends to this gap



Formal Tropos

- Includes all concepts of i^*
 - Enables the description of structural aspects of the model
- Includes temporal specification
 - Enables the description of dynamic aspects of the model
- Supports the application of formal analysis tools



Formal analysis

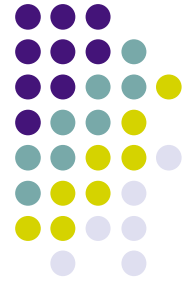
- Extension of existing technique **model checking**
 - Including extension of tool support (T-Tool)
- Automated checking enabled
 - Translate Formal Tropos into an intermediate language (IL)
 - IL can be transformed in order to be formally analyzed by tools

Questions?



... thanks for your attention!





References

- Bresciani, P., A. Perini, P. Giorgini, F. Giunchiglia, J. Mylopoulos (2004). Tropos: An Agent-Oriented Software Development Methodology. *Autonomous Agents and Multi-Agent Systems* **8**, 3 (May 2004). 203-236.
- Fuxman, A., L. Liu, J. Mylopoulos, M. Pistore, M. Roveri, P. Traverso (2004). Specifying and Analyzing Early Requirements in Tropos. *Requirements Engineering* **9**, 2. 132-150.
- van Lamsweerde, A. (2009). Goal Orientation in Requirements Engineering. In van Lamsweerde, A. *Requirements Engineering: From System Goals to UML Models to Software Specifications*. Wiley. 259-283.
- Yu, E. (1997). Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. *3rd IEEE International Symposium on Requirements Engineering (RE'97)*, Annapolis MD, USA. 226-235.