

# Conceptual Modelling as a New Entry in the Bazaar: The Open Model Approach

Stefan Koch, Stefan Strecker, and Ulrich  
Frank

# Introduction

- Collaborative development processes
  - Open source software
  - Open content
  - Open hardware
  - ...
- Fundamental principles
  - named credit, anti-forking,...
  - possible boundaries and preconditions?

# Introduction

- Conceptual modelling as an open process
  - develop reference models for everyone to copy, distribute, use, and refine with the collaboration of a large number of participants in a public process
- Reasons
  - advantages for (open source) software development
  - research on transferability of principles
  - freedom of information

# Conceptual Modeling

- Model
  - abstraction of real-world phenomena relevant to a certain modelling task
- Conceptual models
  - representations of software systems accessible not only to modellers and software developers, but also to domain experts and prospective end users
  - contribute to reduction of complexity and risk, help to overcome cultural chasm between developers and end users

# Conceptual Modeling

- Reference model
  - conceptual model not just suiting one system, but a whole range of systems
  - provide generalisations of domains, blueprints for good system design
  - only small number available
  - deficiencies: remain in prototypical state, fail to be deployed in practice
- Ideal subjects for open process
  - high complexity, diverse participants, exchange between academia and practice

# Open Model Process

- Appropriate licence
- Roles and stakeholders
  - Maintainer, Modeller, Contributor, Reviewer, End user
- Motivations and incentives (intrinsic motivation, academic reputation)
- Parallelisation of work
- Modularity
- Collaboration tools

# Open Model Process

- Implementation
  - Choosing an appropriate licence
  - Choosing suitable domain
  - Choosing appropriate abstractions
  - Choosing modelling languages and tools
  - Design the appropriate processes
  - Preparing the necessary infrastructure
  - Delivering plausible promise (prototype)
  - Continuously evaluating processes, products and community

# Discussion

- Evaluation of conceptual models
  - challenging task
  - discursive evaluation
  - participation of researchers, domain experts, prospective users
  - open culture
- Model should be accompanied with reasonings about the model, changes to the model and discussions about these



# Discussion

- Reference models as subject for teaching
- Benefits for open source software development
- Main challenge
  - reach a critical mass of participants to start a sustainable open process
  - necessary infrastructure to reduce transaction costs
  - motivation of potential participants

# Summary and Future Work

- Adopt principles of open source software development for conceptual modelling
- Goal: to develop reference models for everyone to copy, use, refine and later implement with the collaboration of a large number of participants in a public process.

# Summary and Future Work

- Next step: verify viability in real-world example
  - create a technological infrastructure
  - initial funding (open “model versioning system” and developing prototype)
- Intention to fund such a project
  - substantial benefits, both in itself, and as an academic field study