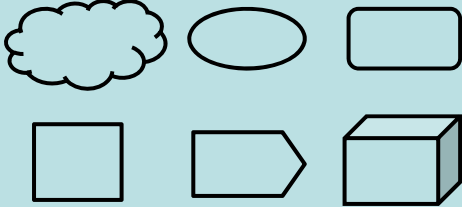


# Towards a Unified Requirements Modeling Language (URML)

Brian Berenbach, Jonas Helming, Maximilian Koegel, Florian Schneider, Michael Haeger, Christine Kaminski, Bernd Bruegge

What does this mean?



Domains for System Requirements Modeling

Goal Modeling	Feature Modeling	Requirements Modeling
Product Line Modeling	Requirements Modeling	Hazard/Threat Analysis
Use Case Modeling	Subsystem Decomposition	Stakeholder Modeling

## Problems in RE for Systems

Environment:

- Multiple Disciplines
- Multiple Experts
- Interdisciplinary Collaboration

Existing RE Modeling Solutions:

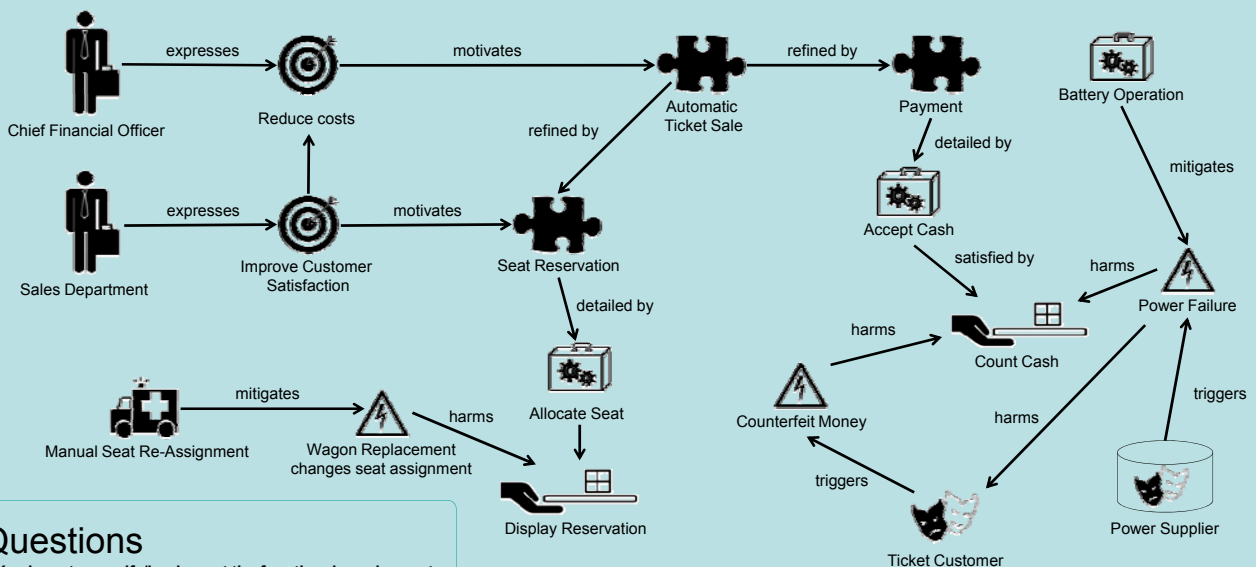
- **No Inter-domain Traceability**
  - Separation of concerns
  - Manual Traceability overhead
- **No Inter-domain Comprehensibility**
  - Low Usability, Poor visual Design
  - Tailored to specific domain experts

## Solutions

- **Inter-domain Traceability**
  - Unified Model
  - Explicit Meta-Model
  - Semantics allowing for Model Checking
  - Well-defined Traceability Links
  - Covering multiple Domains
- **Inter-domain Comprehensibility**
  - Visual Language
  - Interdisciplinary Usability
  - Design following visual paradigms
  - Evaluation based on focus groups

URML

## Example for URML: Ticketing System



### Questions

1. You have to specify/implement the functional requirement „Battery Operation“. Who should you talk to?
2. Manual seat reassignment is expensive. Do we really need it? Who should pay for it?
3. Which features have unresolved dangers?