Software Engineering and Service-Oriented Systems: Summary and Further Topics

Martin Wirsing

LMU München

Ludwig—
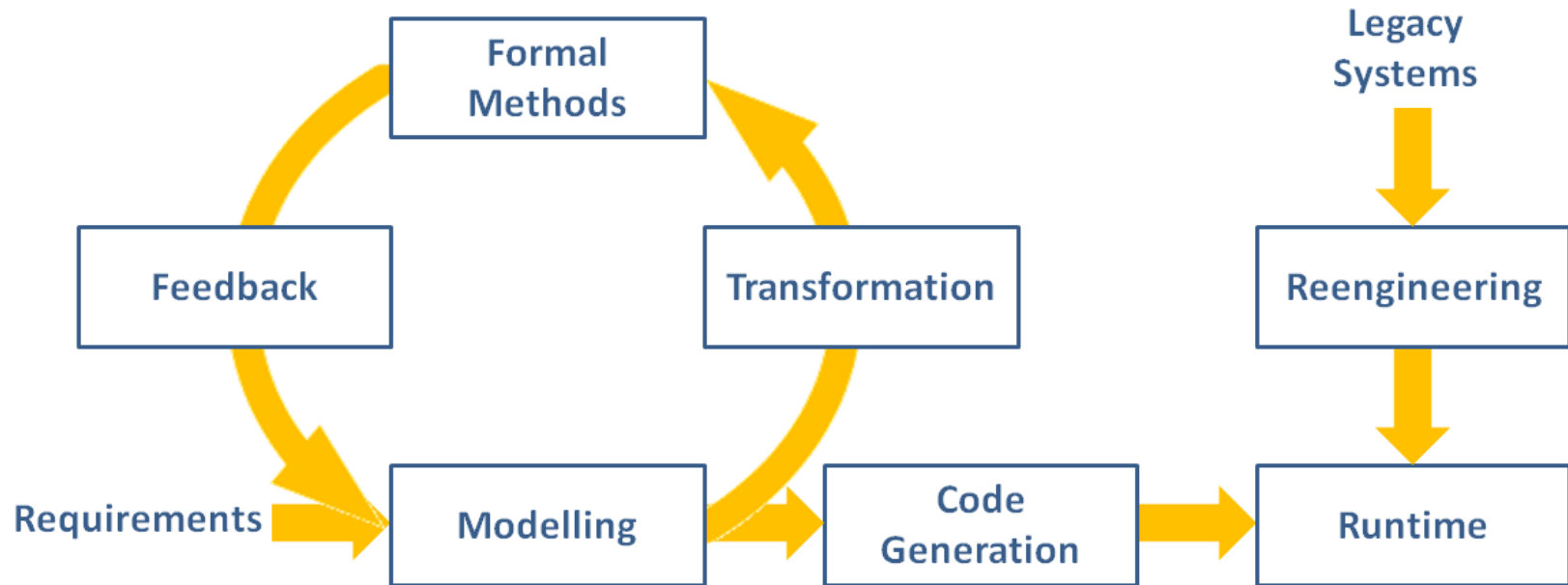Maximilians—
Universität—
München—

LMU

in co-operation with Francesco Tiezzi and
the SENSORIA team, in particular, Nora Koch, Philip Mayer, Rosario Pugliese, Stephen Gilmore
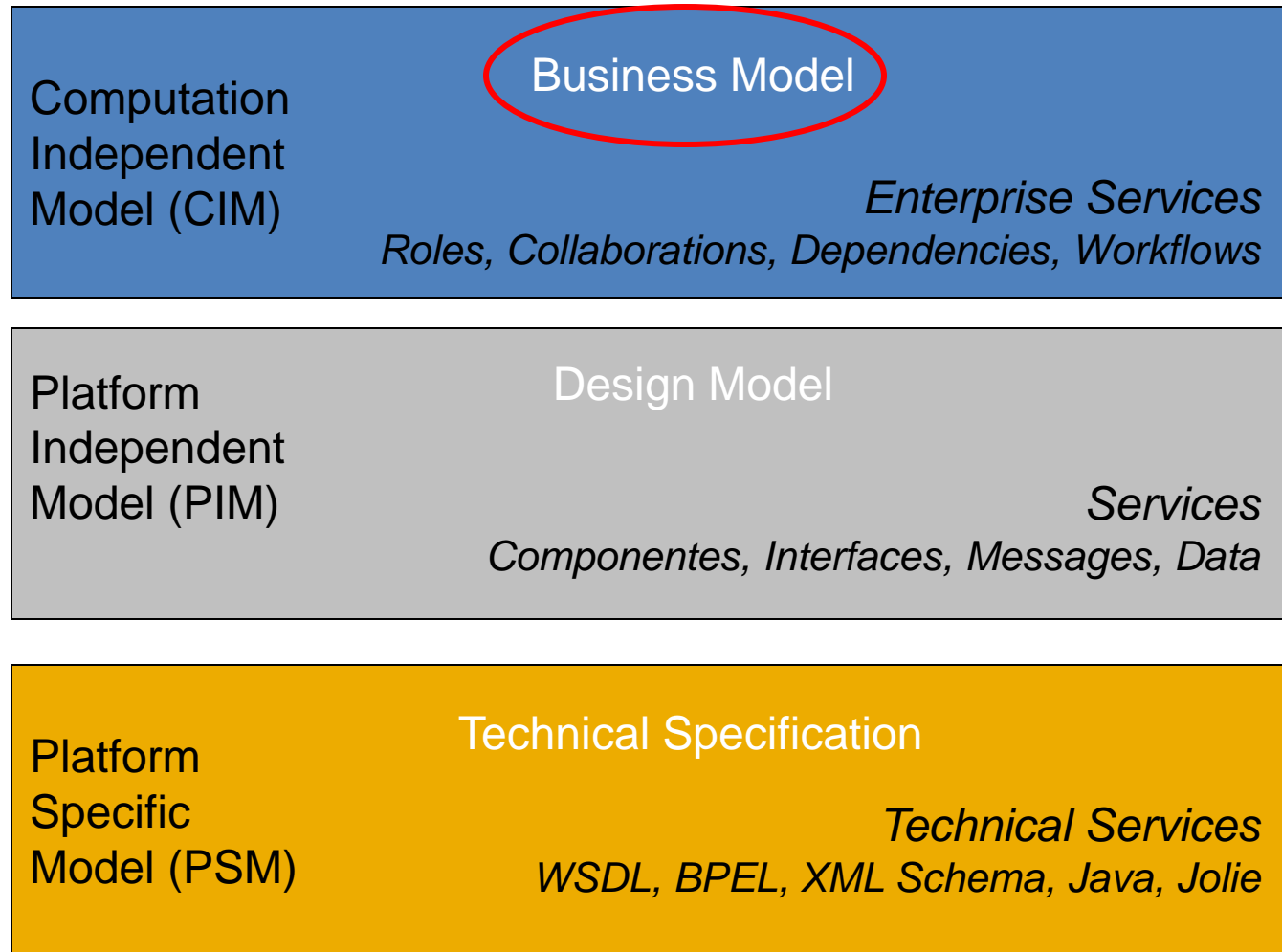and many other SENSORIA members

Contents of the Lectures

- ***SENSORIA*** Overview
- Model-Driven Development of Service-Oriented Systems
- Modal I/O Transition Systems as Semantics of UML4SOA
- Summary and Introduction to ASCENS

MDD4SOA:
SENSORIA model-driven development approach
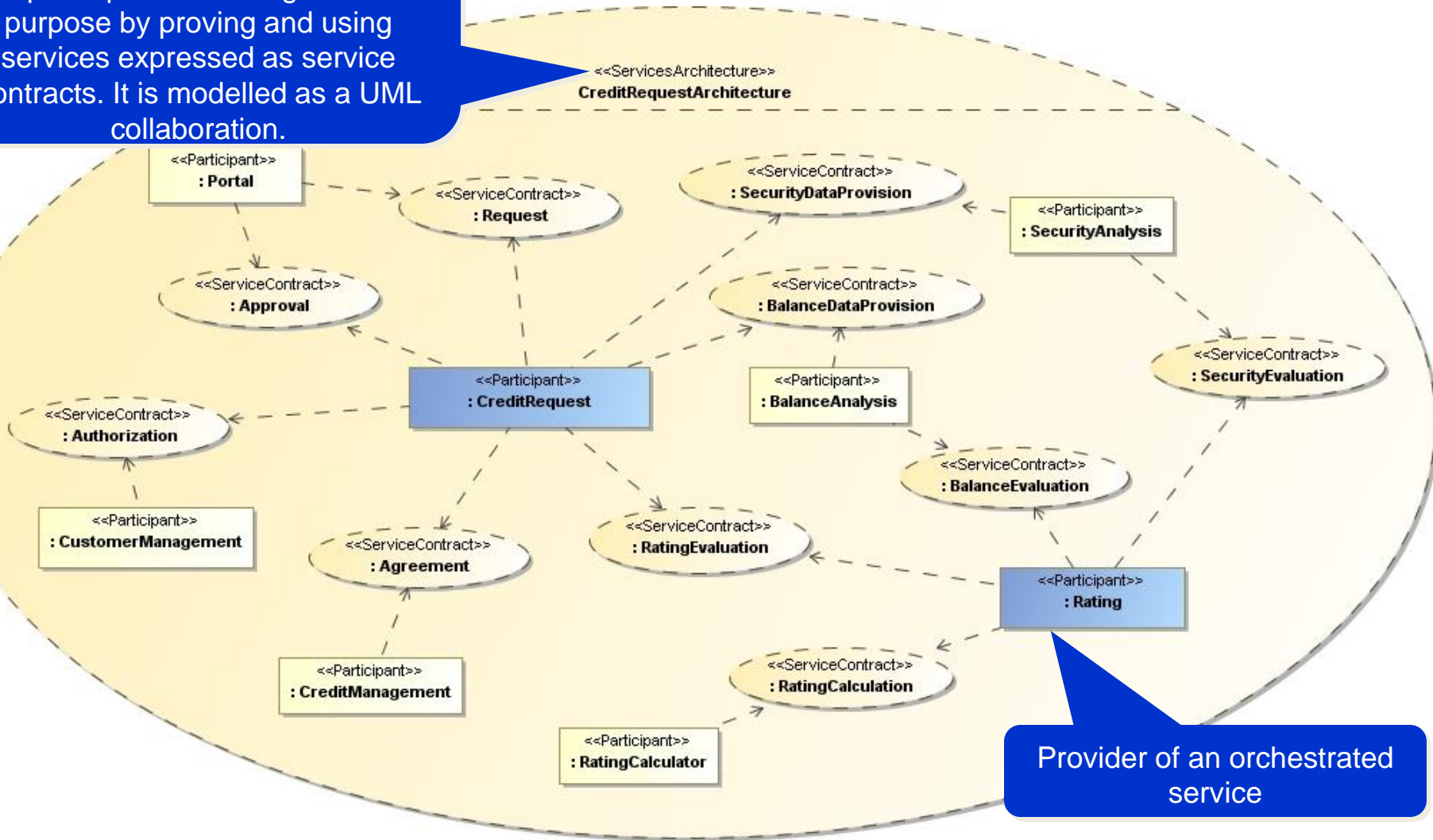
SOA models in the MDA context

Computation Independent Model (CIM)

Business Model

Enterprise Services
Roles, Collaborations, Dependencies, Workflows

Platform Independent Model (PIM)

Design Model

Services
Componentes, Interfaces, Messages, Data

Platform Specific Model (PSM)

Technical Specification

Technical Services
WSDL, BPEL, XML Schema, Java, Jolie

Refinement & Automation

*Source: Data Access Technologies, Inc*

Representing service architecture

A ***service architecture*** describes how participants work together for a purpose by proving and using services expressed as service contracts. It is modelled as a UML collaboration.

<<ServicesArchitecture>>
CreditRequestArchitecture

<<Participant>>
: Portal

<<ServiceContract>>
: Request

<<ServiceContract>>
: SecurityDataProvision

<<Participant>>
: SecurityAnalysis

<<ServiceContract>>
: Approval

<<ServiceContract>>
: BalanceDataProvision

<<ServiceContract>>
: SecurityEvaluation

<<Participant>>
: CreditRequest

<<Participant>>
: BalanceAnalysis

<<ServiceContract>>
: Authorization

<<ServiceContract>>
: BalanceEvaluation

<<Participant>>
: CustomerManagement

<<ServiceContract>>
: Agreement

<<ServiceContract>>
: RatingEvaluation

<<Participant>>
: Rating

<<Participant>>
: CreditManagement

<<ServiceContract>>
: RatingCalculation

<<Participant>>
: RatingCalculator

Provider of an orchestrated service

Orchestration of

- Service orchestration the process of combining existing services to form a service to be used any other service.

- Key distinguishing concepts

    - partner services
    - message passing among requester provider
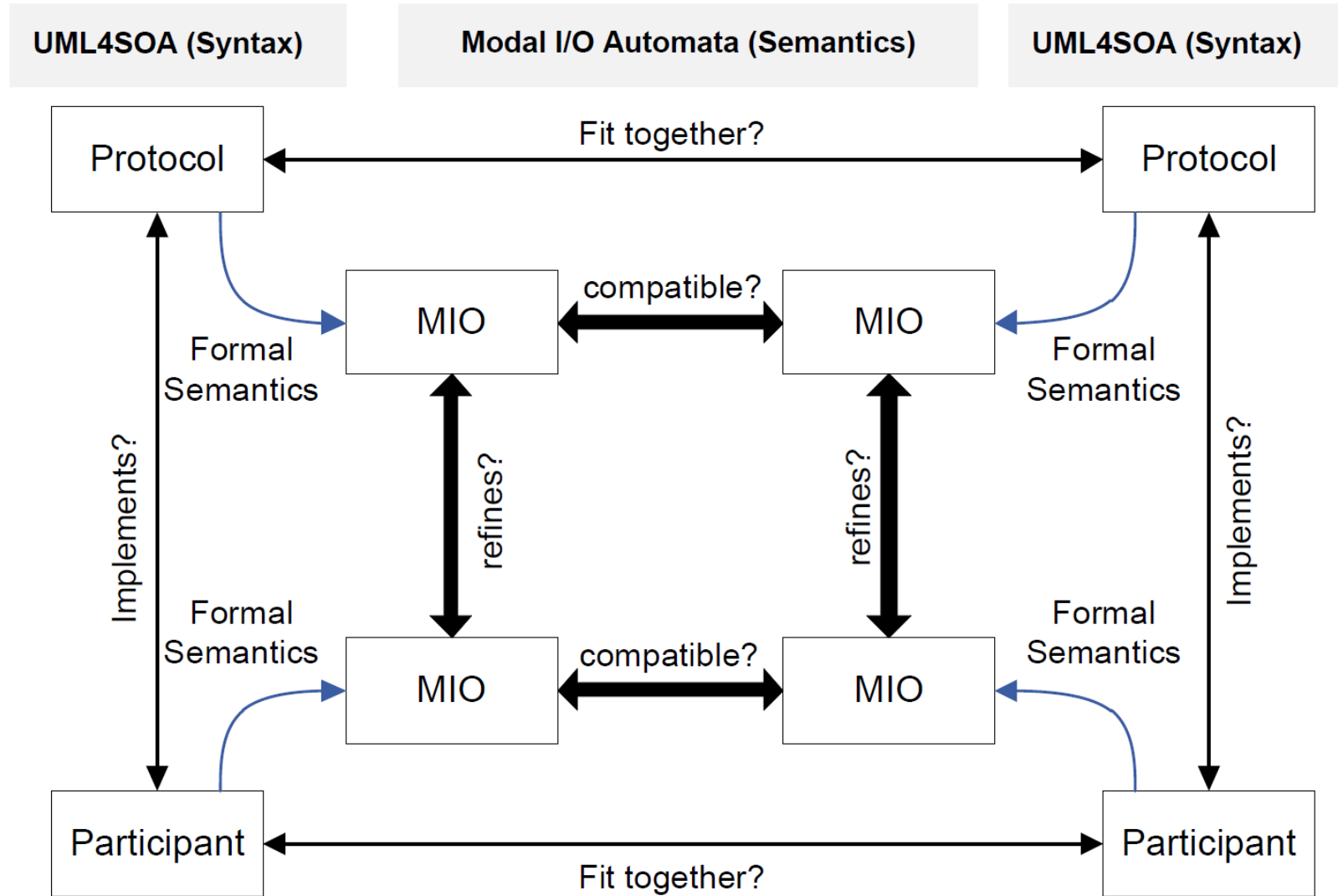   - long-running transactions
    - compensation
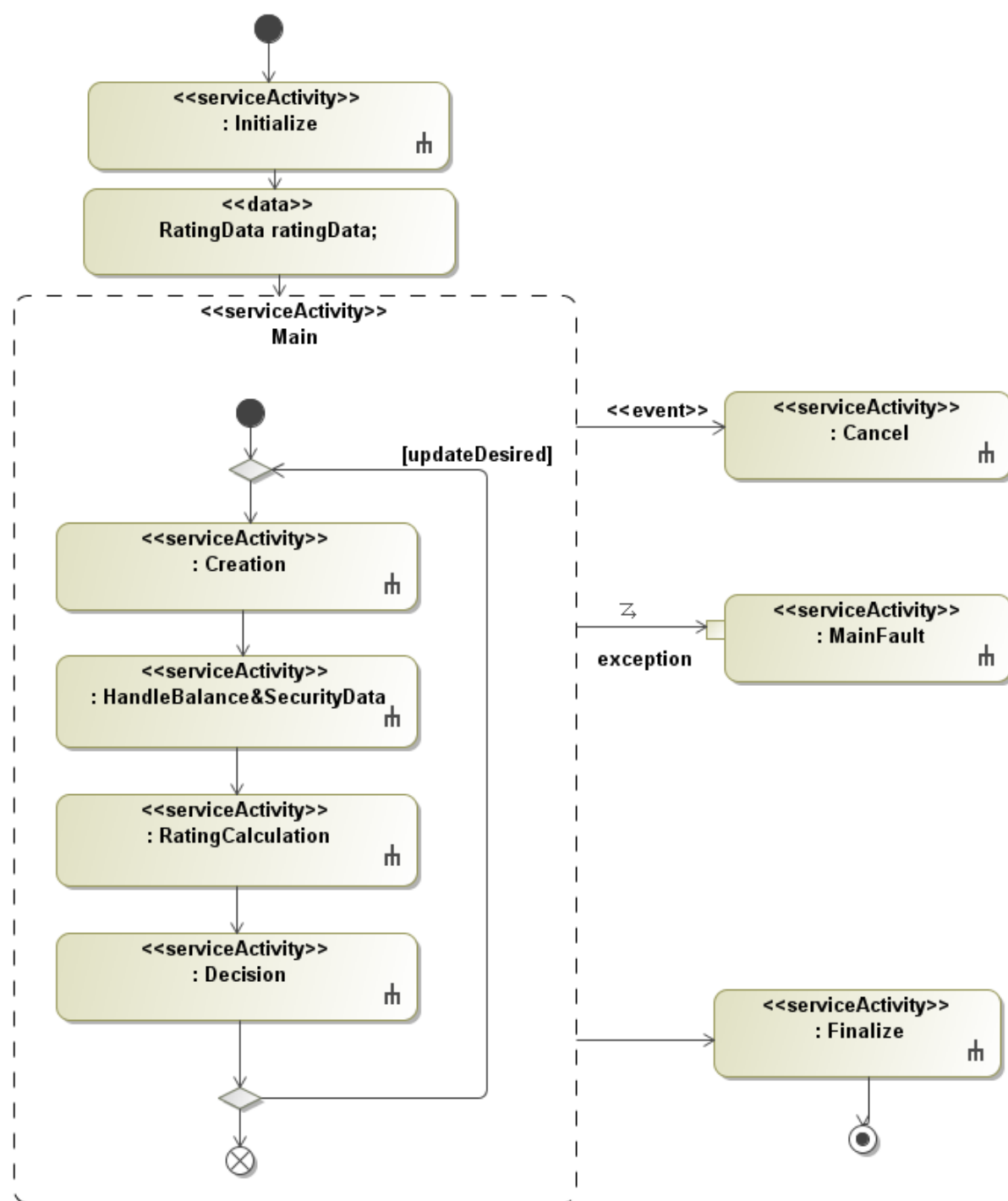
Modal I/O-Transition Systems (MIOs)

- Modalities ("may" and "must") for refinement (vertical relationship)
    - "must": what is required (~ bisimulation)
    - "may": what is optional (~ trace inclusion refinementt)
- Input/output for compatibility (horizontal relationship)
- Synchronous composition (shared actions are internalized)
- Output Compatibility (any outputs must be received)

Astrid Lindgren 1954
www.villa-galactica.de/

Overview of Qualitative Analysis Approach

UML4SOA (Syntax)	Modal I/O Automata (Semantics)	UML4SOA (Syntax)

Protocol — Fit together? — Protocol

MIO — compatible? — MIO

Formal Semantics

refines?   refines?

Implements?   Implements?

Formal Semantics

MIO — compatible? — MIO

Participant — Fit together? — Participant

Accident scenario of automotive case study
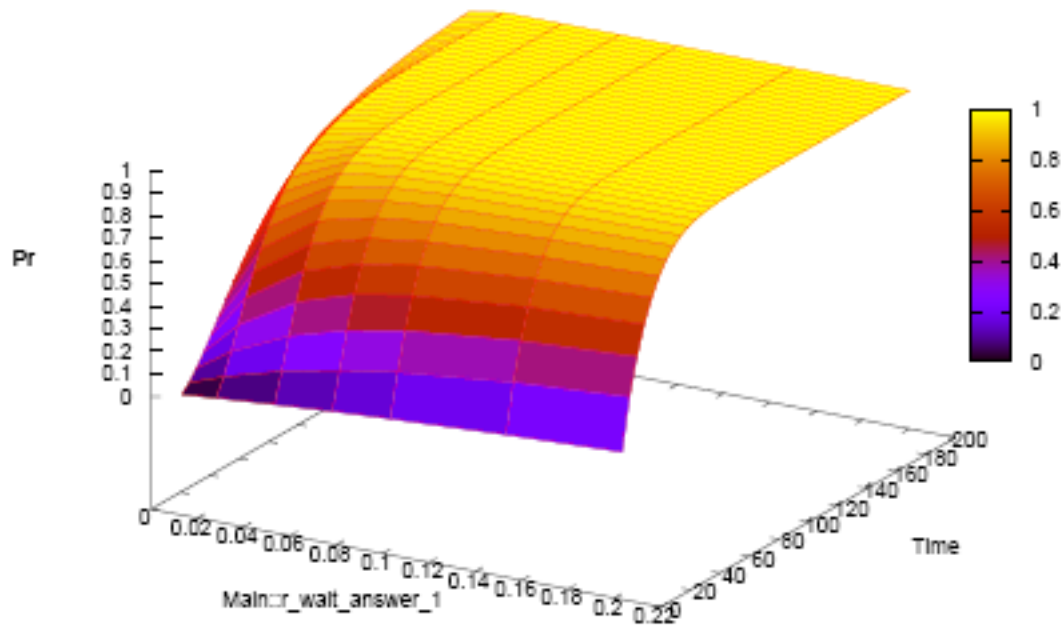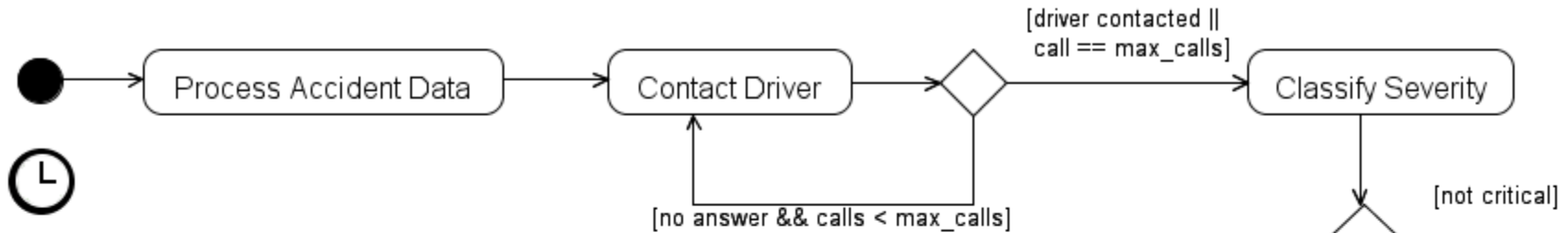
Example: SDE user interface
Graphical orchestration of tools

Orchestration
Defines data flow between tool functions

Pattern catalogue

- Relationships between patterns
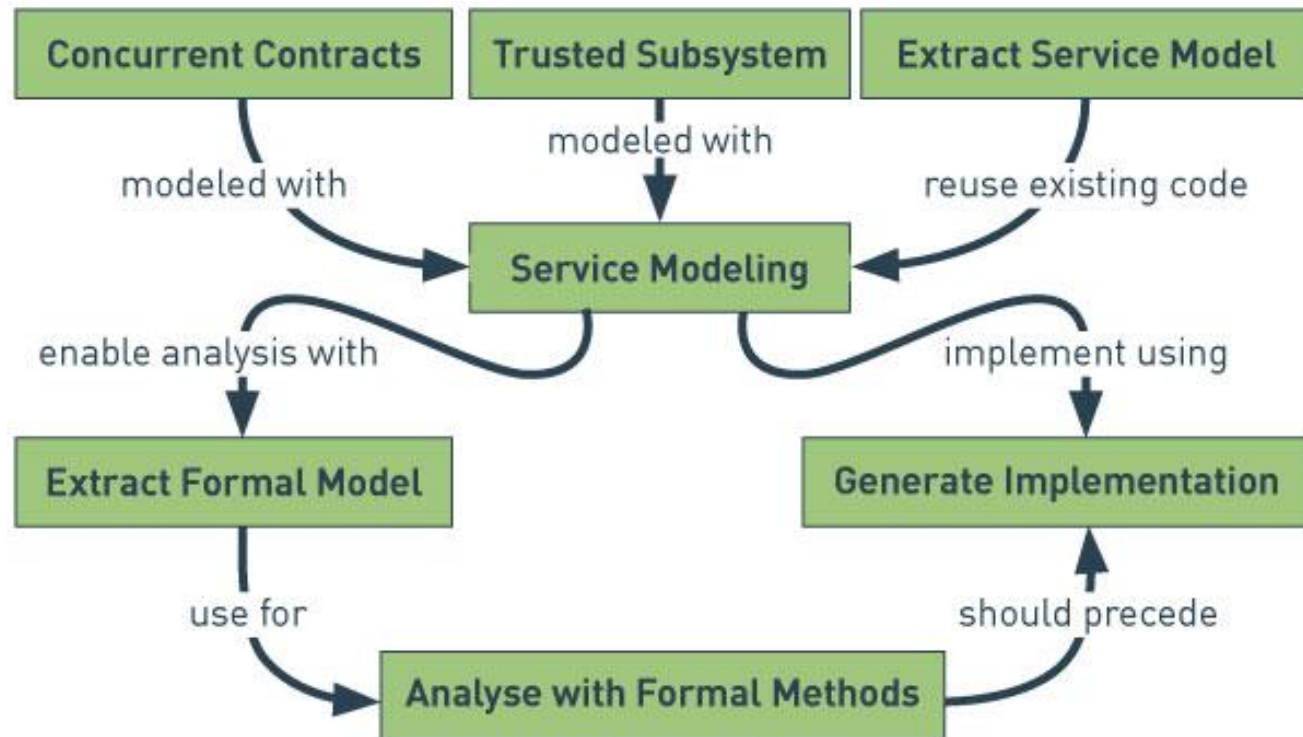
Bottom line: Ideas to take home

- Automated development approach
  - model-based and semantics driven
  - early qualitative and quantitative analysis based on formal techniques
  - model-driven (transformations)
  - pattern-based
- Relevance of domain specific modelling language
  - UML profile
  - must be simple, few constructs
- Importance of flexible tool support
  - easy (graphically) integration of diverse tools