OCOPOMO – Supporting policy development through conceptual designs

WS 3 on Open Government @ EGOV 2011, Delft (The Netherlands) , 2nd September 2011

Maria A. Wimmer and Sabrina Scherer
wimmer | scherer @uni-koblenz.de

UNIVERSITÄT
KOBLENZ · LANDAU

Agenda

- ❖ Demands for ICT supported Governance and Policy Modelling and the OCOPOMO project
- ❖ Scenario building and analysis
- ❖ Integrating collaborative scenario texts with formal policy models
- ❖ Pilot cases, expected outcomes and innovation

Challenges in Policy Development

- ❖ Appropriate ICT support in policy planning not deployed widely

- ❖ Management of complexity in strategy and policy formation
- ❖ Development, visualisation and simulation of appropriate policy models usually done by experts
  - ➢ black-box approach

- ❖ Lack of open collaboration and therewith transparency in identifying the crucial features of complex social environments to inform policy models

- ❖ Online participation means not yet deployed widely in strategic decision making

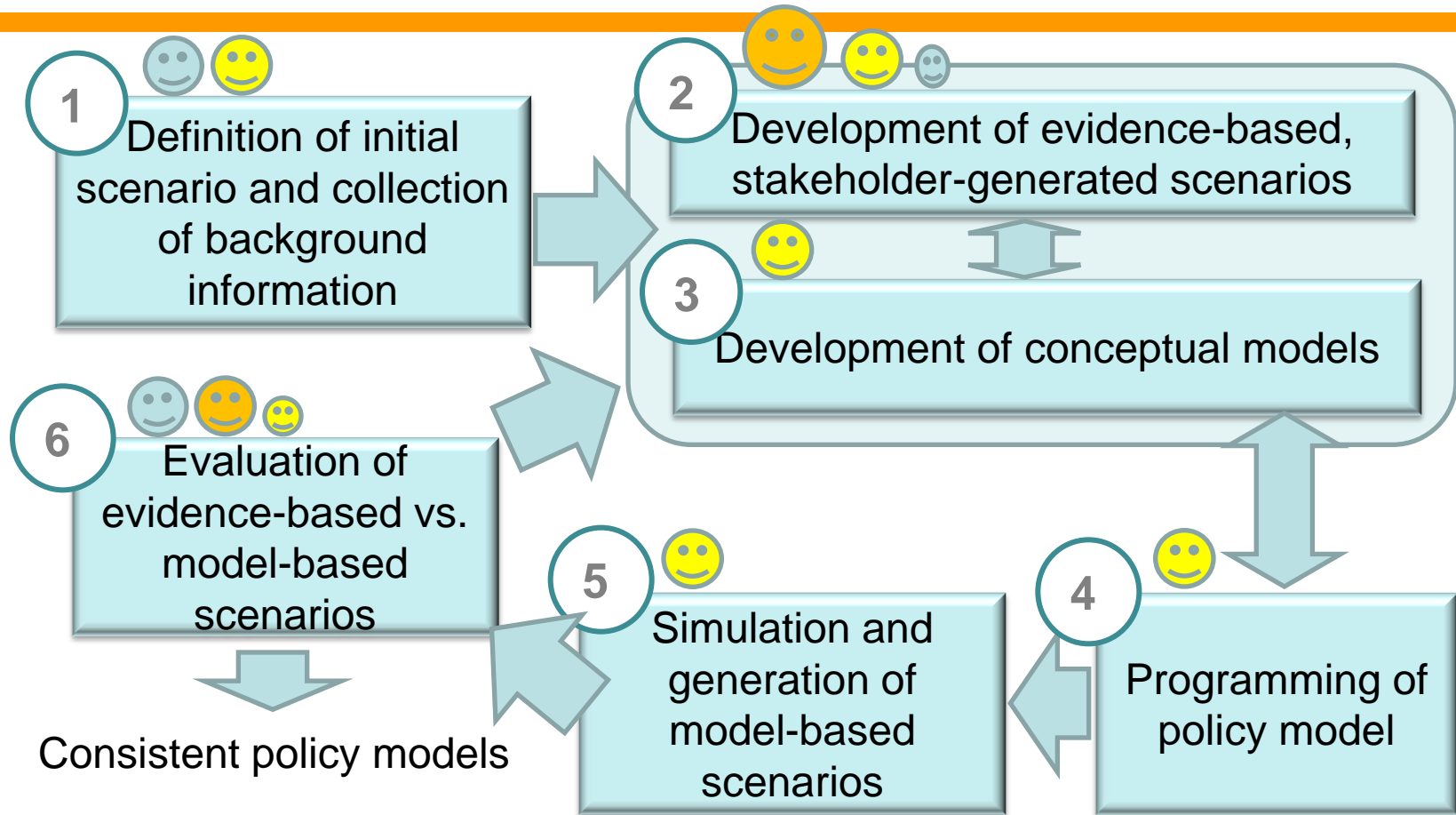# Need for ICT Support in Policy Development

- ❖ Comprehensive IT solutions to support
  - ➢ policy analysis, modelling and simulation
  - ➢ collaboration among
    - ▪ policy analysts and policy operators
    - ▪ wider stakeholder groups
    - ▪ general public
- ❖ ICT support along the whole policy development and stakeholder participation process
- ❖ Textual scenarios as a crucial but simple means
  - ➢ to actively engage constituencies in policy development
  - ➢ to enable transparency

© OCOPOMO consortium, www.ocopomo.eu

Aims of OCOPOMO Project

- ❖ Support key stakeholders to participate in the processes of policy formulation
- ❖ Integrate methods and tools of scenario-based policy formation with formal policy modelling
- ❖ Develop an integrated ICT platform for efficient policy making
    - ➢ Open collaboration along the policy process
    - ➢ Engagement of wide stakeholder groups

OCOPOMO's Integrated Policy Process and Involved Actors

**1** Definition of initial scenario and collection of background information

2 Development of evidence-based, stakeholder-generated scenarios

3 Development of conceptual models

6 Evaluation of evidence-based vs. model-based scenarios

Consistent policy models

5 Simulation and generation of model-based scenarios

4 Programming of policy model

Legend:
Prodess phase

Transition to next phase
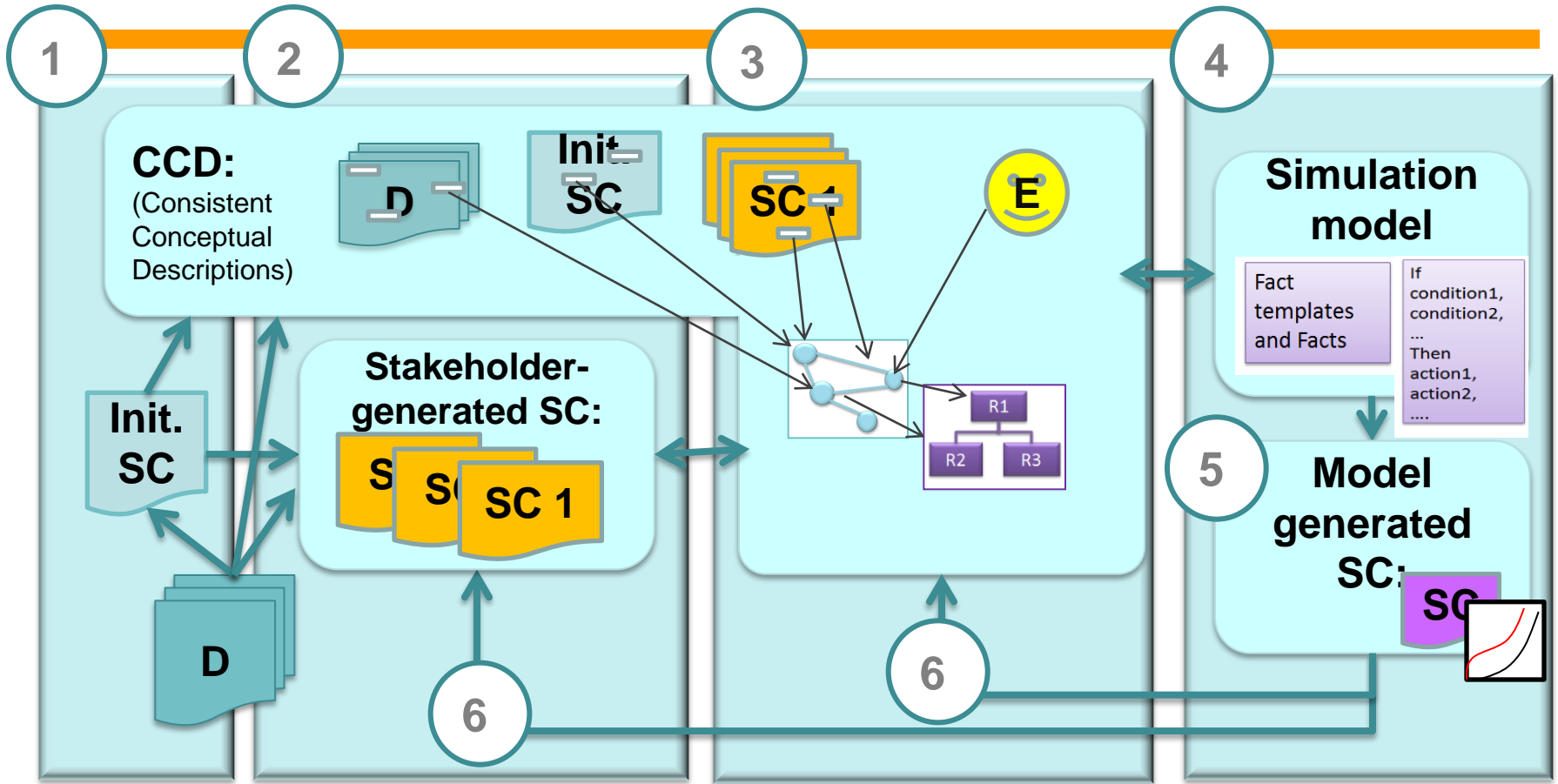
Actors:
Domain Experts (Policy Planner / Strategic Decision Maker)

Stakeholders involved

Experts for Policy Analysis / Policy Modelling

ICT Toolbox
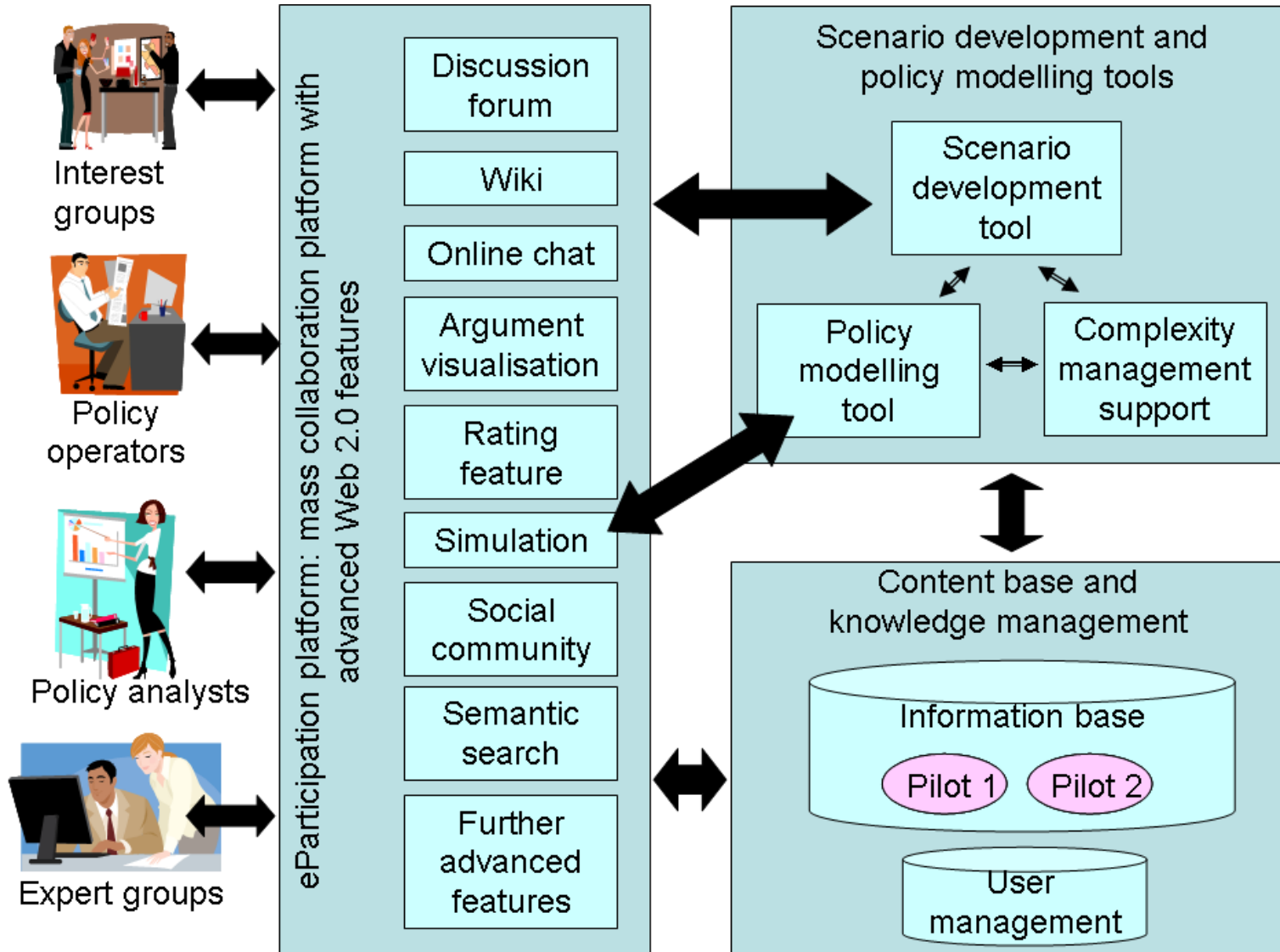
Artefacts along the Process Phases

Legend:

Symbol	Meaning
# Process phase	
Information flow	
Information flow detailed steps	
E Expert knowledge	
D Documents	
SC Scenarios	
Relevant aspect	
Network of social relationships	
Rule-Dependency-Graph	
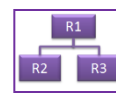
Transformation Needs
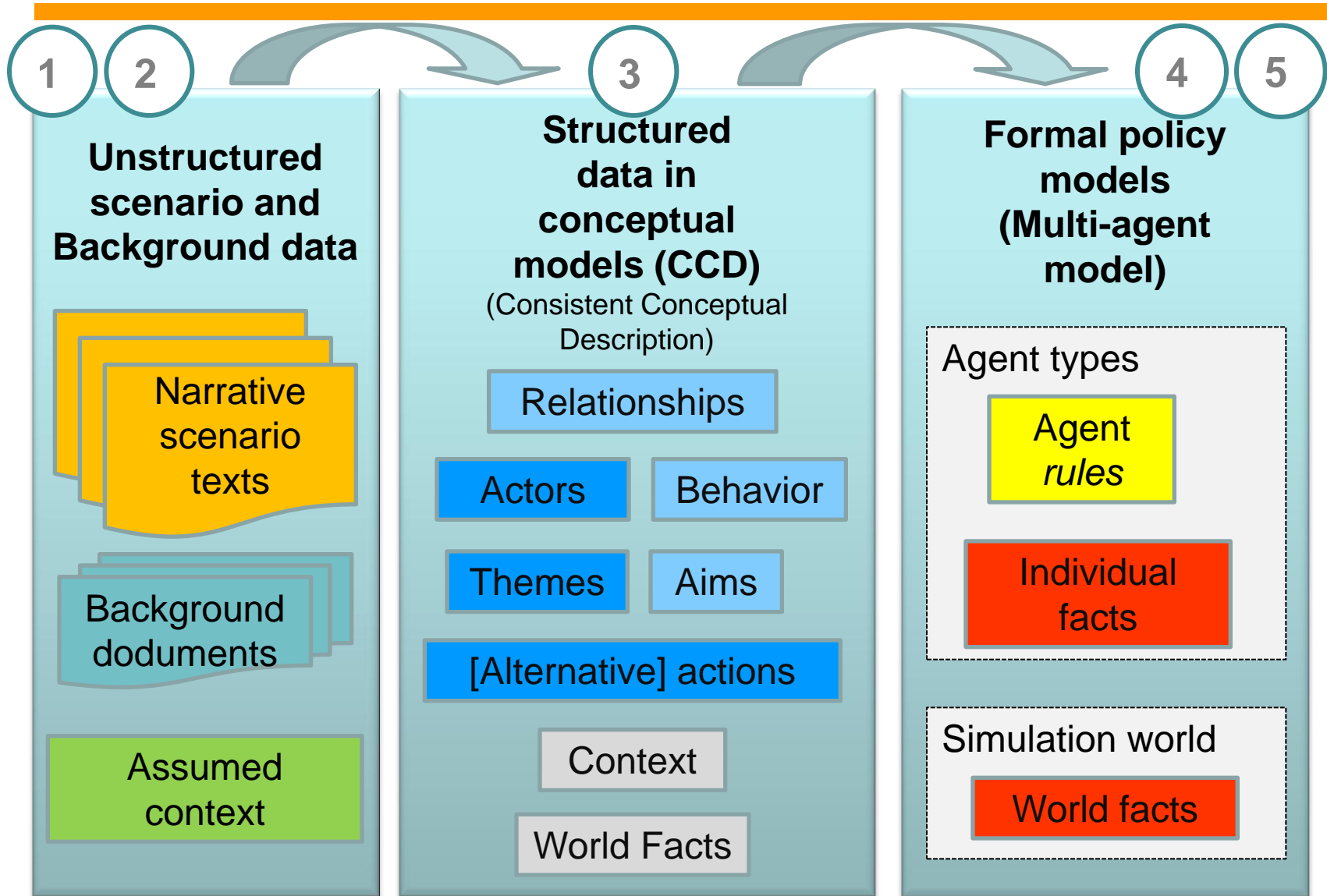
1 **2** **3** **4** **5**

Unstructured scenario and Background data

Narrative scenario texts

Background doduments

Assumed context

Structured data in conceptual models (CCD)
(Consistent Conceptual Description)

Relationships

Actors Behavior

Themes Aims

[Alternative] actions

Context

World Facts

Formal policy models (Multi-agent model)

Agent types

Agent *rules*

Individual facts

Simulation world

World facts

Agenda

❖ Demands for ICT supported Governance and Policy Modelling and the OCOPOMO project

❖ **Scenario building and analysis**

❖ Integrating collaborative scenario texts with formal policy models

❖ Pilot cases, expected outcomes and innovation

© OCOPOMO consortium, www.ocopomo.eu

What is scenario building

❖ Method for foresight according to Geschka (1978):

"*systematic,*

participatory,

future intelligence gathering and

medium-to-long-term vision building process

aimed at present-day decisions and

mobilising joint actions"

Collaborative Scenario Building in OCOPOMO

- ❖ Scenarios
  - ➢ Are textual description of a perceived view or understanding of a topic under discussion
  - ➢ Cover existing world status or mental model of stakeholders
- ❖ Alternative scenarios to describe different aspects and /or alternatives
- ❖ Different stakeholder groups develop different sets of scenarios independently
- ❖ Scenarios may be conflicting among stakeholder groups
- ❖ Scenarios may be extended and therewith advance an existing scenario (nesting scenarios)

Scenarios as instrument for stakeholder engagement

- ❖ Scenarios can be developed in a transparent and inter-subjective manner
- ❖ Scenarios used as common reference point for formal policy modelling and as communication instrument
- ❖ Relevant information and data can be included in scenarios in an unbiased manner by stakeholders
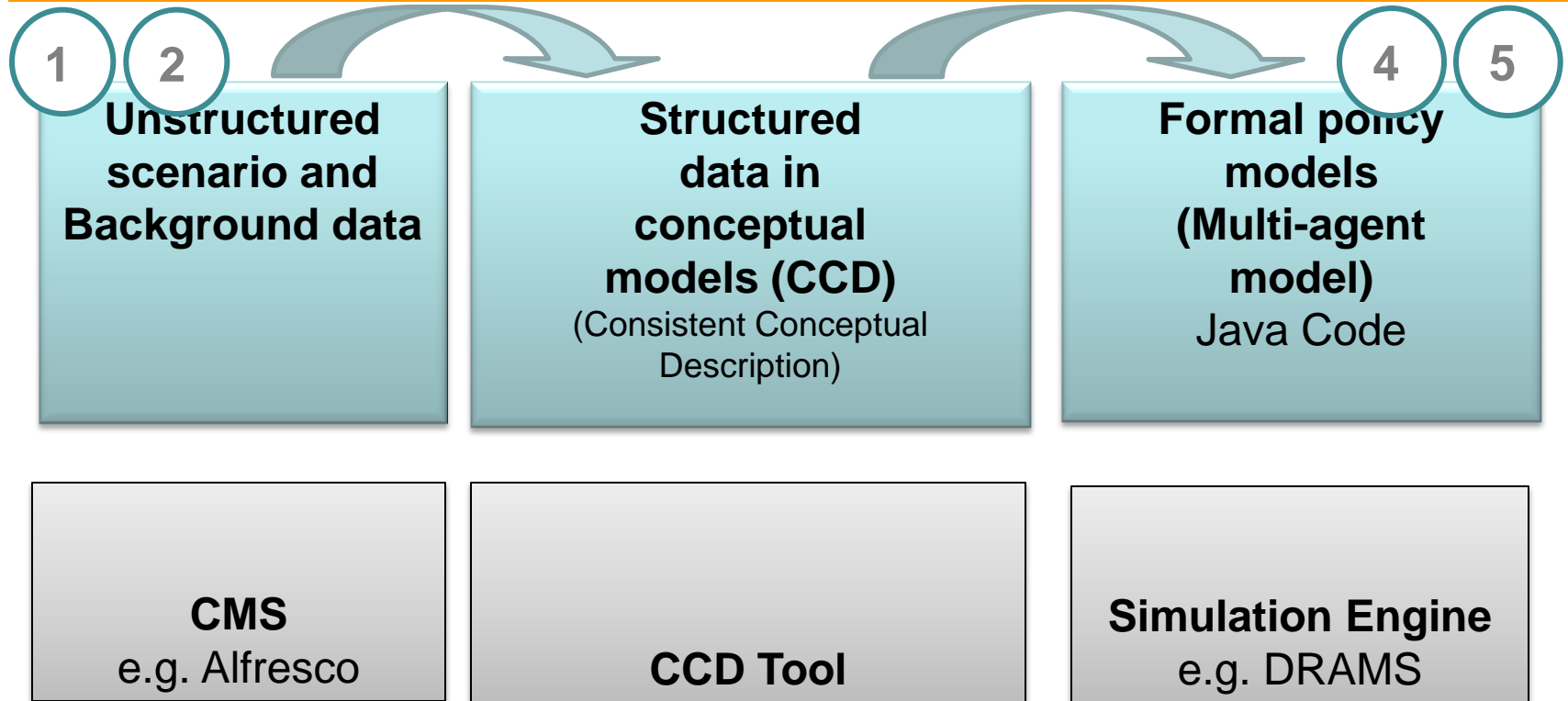- ❖ Assumptions on developments expressed through the scenarios are shared

© OCOPOMO consortium, www.ocopomo.eu

Integrating Scenario and Formal Model

- ❖ Goals, scope and social processes specified by participating stakeholders
- ❖ Stakeholder-generated scenarios inform formal policy model design
  - ➢ Key in model design: agent descriptions & if-then rules
  - ➢ Stakeholders see natural-language pseudo code
    - ▪ Enforces precision in use of language, expectations, goals
- ❖ Models produce simulations, which result in model-based scenarios
- ❖ Participating stakeholders evaluate model generated scenarios
  - ➢ Surprises involve further investigation of model & scenarios
  - ➢ Iterations in developing formal policy models

Agenda

- ❖ Demands for ICT supported Governance and Policy Modelling and the OCOPOMO project
- ❖ Scenario building and analysis
- ❖ Integrating collaborative scenario texts with formal policy models
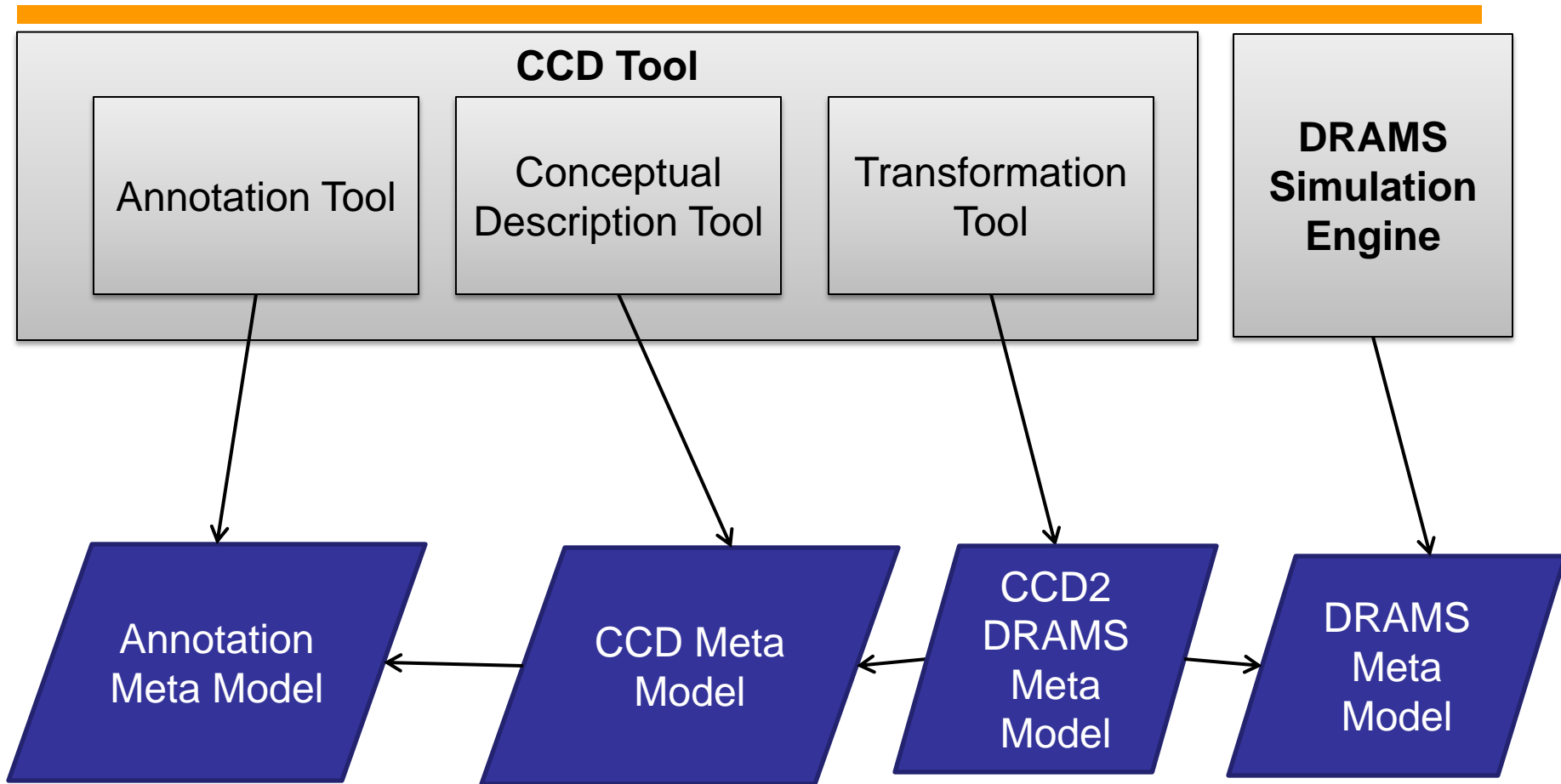- ❖ Pilot cases, expected outcomes and innovation

© OCOPOMO consortium, www.ocopomo.eu

Technical tool to support the OCOPOMO process

1 **2** **4** **5**

Unstructured scenario and Background data

Structured data in conceptual models (CCD)
(Consistent Conceptual Description)

Formal policy models (Multi-agent model)
Java Code

CMS
e.g. Alfresco

CCD Tool

Simulation Engine
e.g. DRAMS

© OCOPOMO consortium, www.ocopomo.eu

SEVENTH FRAMEWORK PROGRAMME

Technical tool to support the process

1 **2**

Unstructured scenario and Background data

Structured data in conceptual models (CCD)
(Consistent Conceptual Description)

4 **5**

Formal policy models (Multi-agent model)
Java Code

links

creates

generates

Annotation Tool

Conceptual Description Tool

Transformation Tool

CCD Tool

© OCOPOMO consortium, www.ocopomo.eu

Modules of the CCD Tool
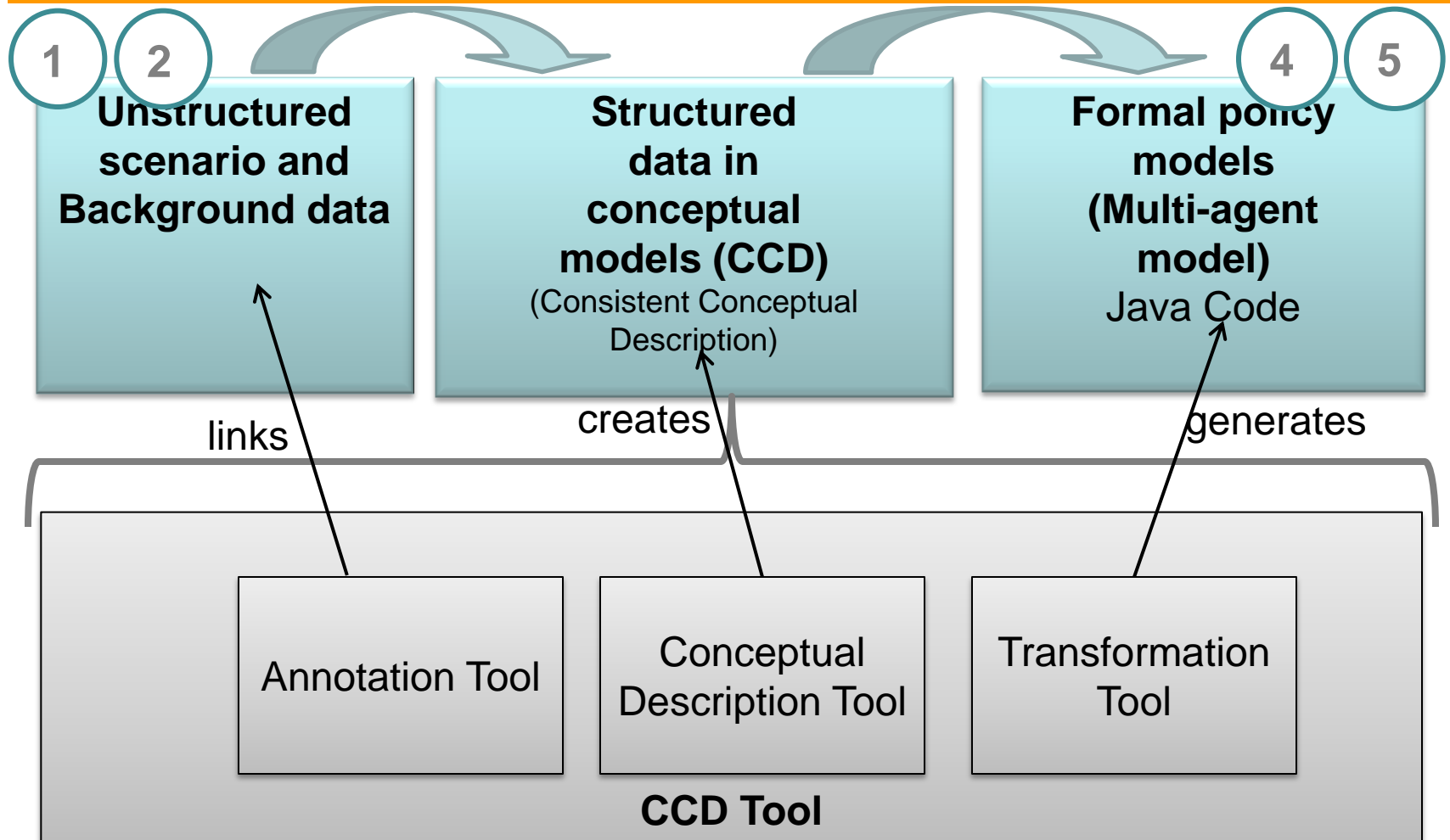
❖ *Conceptual Description Tool (CD Tool)*:

➢ supports the Facilitator and the Policy Modeller in describing a conceptual model of the policy case.

❖ *Annotation Tool (AnnoTool)*

➢ background information (e.g. statistics, stakeholder scenarios) are annotated and linked with relevant actors, objects and actions documented in a CCD file.

❖ *Transformation Tool (TransTool)*:

➢ supports the Policy Modeller in generating source code from the CCD

System Design

| CCD Tool | | | | DRAMS Simulation Engine |

Annotation Tool — Conceptual Description Tool — Transformation Tool — DRAMS Simulation Engine

Annotation Meta Model ← CCD Meta Model ← CCD2 DRAMS Meta Model → DRAMS Meta Model

Legend: uses

© OCOPOMO consortium, www.ocopomo.eu

SEVENTH FRAMEWORK PROGRAMME

System Design

CCD 1
(Domain Model)

CCD 2
(Domain Model)

CCD Tool

Annotation Tool

Conceptual
Description Tool

Transformation
Tool

**DRAMS
Simulation
Engine**

Annotation
Meta
Model

CCD
Meta
Model

CCD2
DRAMS
Meta
Model

DRAMS
Meta
Model

Legend: uses

SEVENTH FRAMEWORK
PROGRAMME

Meta Models

❖ What is meant with a *Meta Model*?

➢ In this context:

Model that defines the components of the concept.

❖ Annotation Model:

➢ Defines the components of the annotations

❖ CCD Meta Model:

➢ Defines the components of a CCD

❖ DRAMS Meta Model

➢ Defines the components of the DRAMS Simulation Engine

❖ CCD2DRAMS Meta Model

➢ Defines the matching between components in CCD und DRAMS

CCD Meta Model
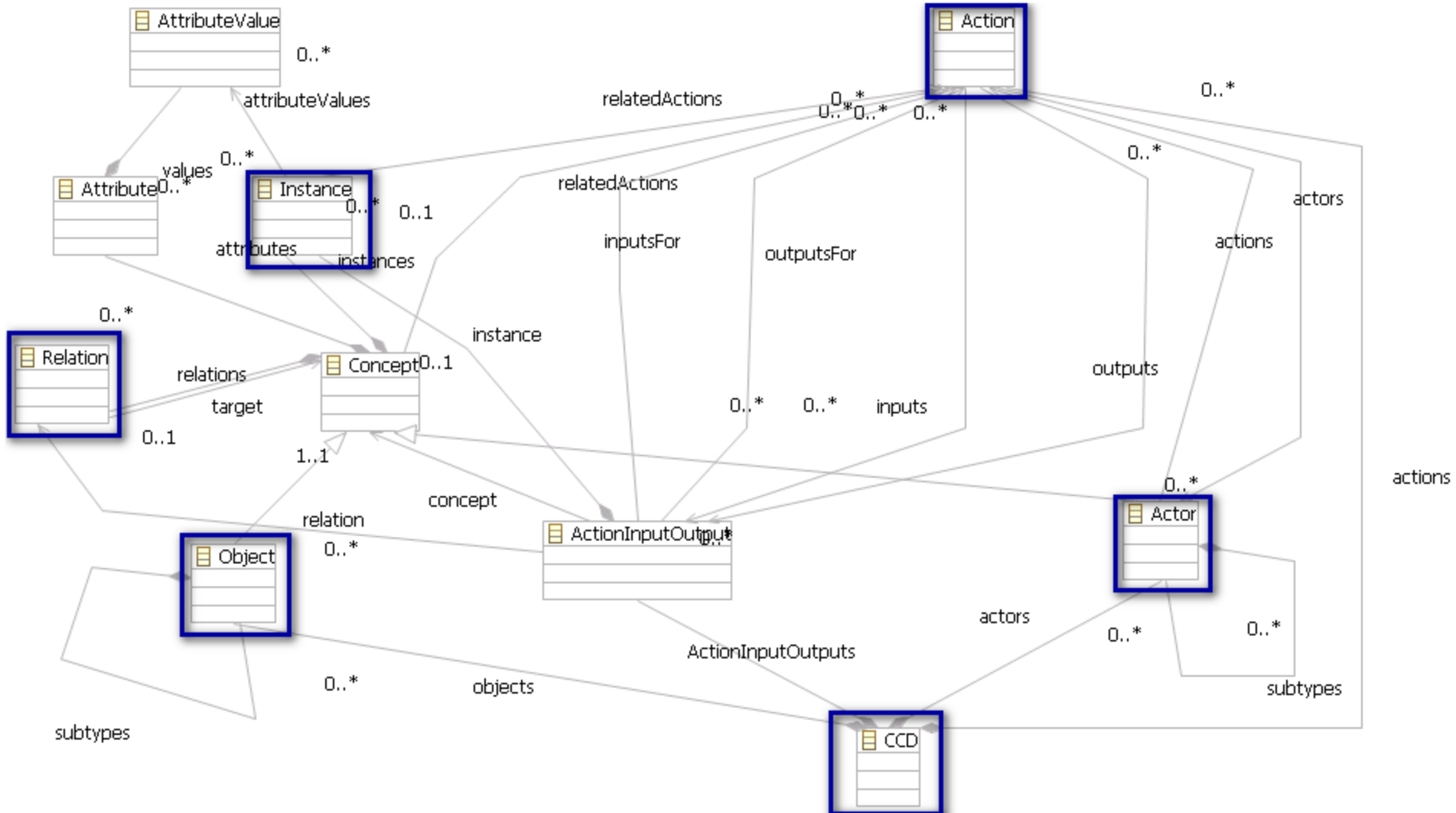
Example of a CCD File

```xml
<actors xmi:id="_OXPZjsPxEeCTKdubIHS_pw" name="Mayor" actions="_OXPZ5sPxEeCTKdubIHS_pw">
  <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZj8PxEeCTKdubIHS_pw" phrase="Mayor" start="5313" lengt
  <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZkMPxEeCTKdubIHS_pw" phrase="The&#xD;&#xA;Mayor has ne
  <relations xmi:id="_OXPZkcPxEeCTKdubIHS_pw" name="prepare and publish" target="_OXPZzMPxEeCTKdubIHS_pw">
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZksPxEeCTKdubIHS_pw" phrase="Mayor must prepare and
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZk8PxEeCTKdubIHS_pw" phrase=" Mayor responsible for
  </relations>
  <relations xmi:id="_OXPZlMPxEeCTKdubIHS_pw" name="work with" target="_OXPZisPxEeCTKdubIHS_pw">
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZlcPxEeCTKdubIHS_pw" phrase="The Mayor will work wit
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZlsPxEeCTKdubIHS_pw" phrase="These arrangements prov
  </relations>
  <relations xmi:id="_OXPZl8PxEeCTKdubIHS_pw" name="work with" target="_OXPZgsPxEeCTKdubIHS_pw">
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZmMPxEeCTKdubIHS_pw" phrase="The Mayor will work wit
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZmcPxEeCTKdubIHS_pw" phrase="These arrangements prov
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZmsPxEeCTKdubIHS_pw" phrase="HCA and its London boar
  </relations>
</actors>
<actors xmi:id="_OXPZm8PxEeCTKdubIHS_pw" name="Household">
  <relations xmi:id="_OXPZnMPxEeCTKdubIHS_pw" name="household income" target="_OXPZ38PxEeCTKdubIHS_pw"/>
</actors>
<objects xmi:id="_OXPZncPxEeCTKdubIHS_pw" name="Housing">
  <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZnsPxEeCTKdubIHS_pw" phrase="proportion of affordable
  <attributes xmi:id="_OXPZn8PxEeCTKdubIHS_pw" name="is affordable" description="A house is affordable if its price is in the lowe
    <annotations xmi:type="org.ocopomo.annotation:FileAnnotation" xmi:id="_OXPZoMPxEeCTKdubIHS_pw" phrase="The ratio of lower quar
    <values xmi:id="_OXPZocPxEeCTKdubIHS_pw" name="true"/>
    <values xmi:id="_OXPZosPxEeCTKdubIHS_pw" name="false"/>
  </attributes>
  <attributes xmi:id="_OXPZo8PxEeCTKdubIHS_pw" name="is market">
    <values xmi:id="_OXPZpMPxEeCTKdubIHS_pw" name="true"/>
    <values xmi:id="_OXPZpcPxEeCTKdubIHS_pw" name="false"/>
  </attributes>
  <attributes xmi:id="_OXPZpsPxEeCTKdubIHS_pw" name="tenure" description="If a private house is not rented it means that it is own
    <values xmi:id="_OXPZp8PxEeCTKdubIHS_pw" name="owned"/>
```

© OCOPOMO consortium, www.ocopomo.eu

Example for a CCD:
Actor – Network - Diagram

CCD Tool in Action

DRAMS

- ❖ Declarative agent modelling platform
  - ➢ Equip agents with expert system capabilities: describing agent behaviour by declarative rules
  - ➢ Individual rule sets for each agent type (or even instance)
  - ➢ Individual working memory for each agent instance
- ❖ Rule engine component for multi-agent simulation models
  - ➢ Distributed rule engine: agents behave in simulation runs autonomously according to their rule specifications
  - ➢ Simulation dynamics is generated by individual agent behaviour, together with interaction between agents (inter-agent communication)
- ❖ Java-based implementation
  - ➢ Full flexibility regarding interface definition for integration with OCOPOMO toolbox
  - ➢ Integrable with widely applied simulation tools (e.g. Repast)
  - ➢ Completely open source (including used external libraries)

DRAMS: core classes
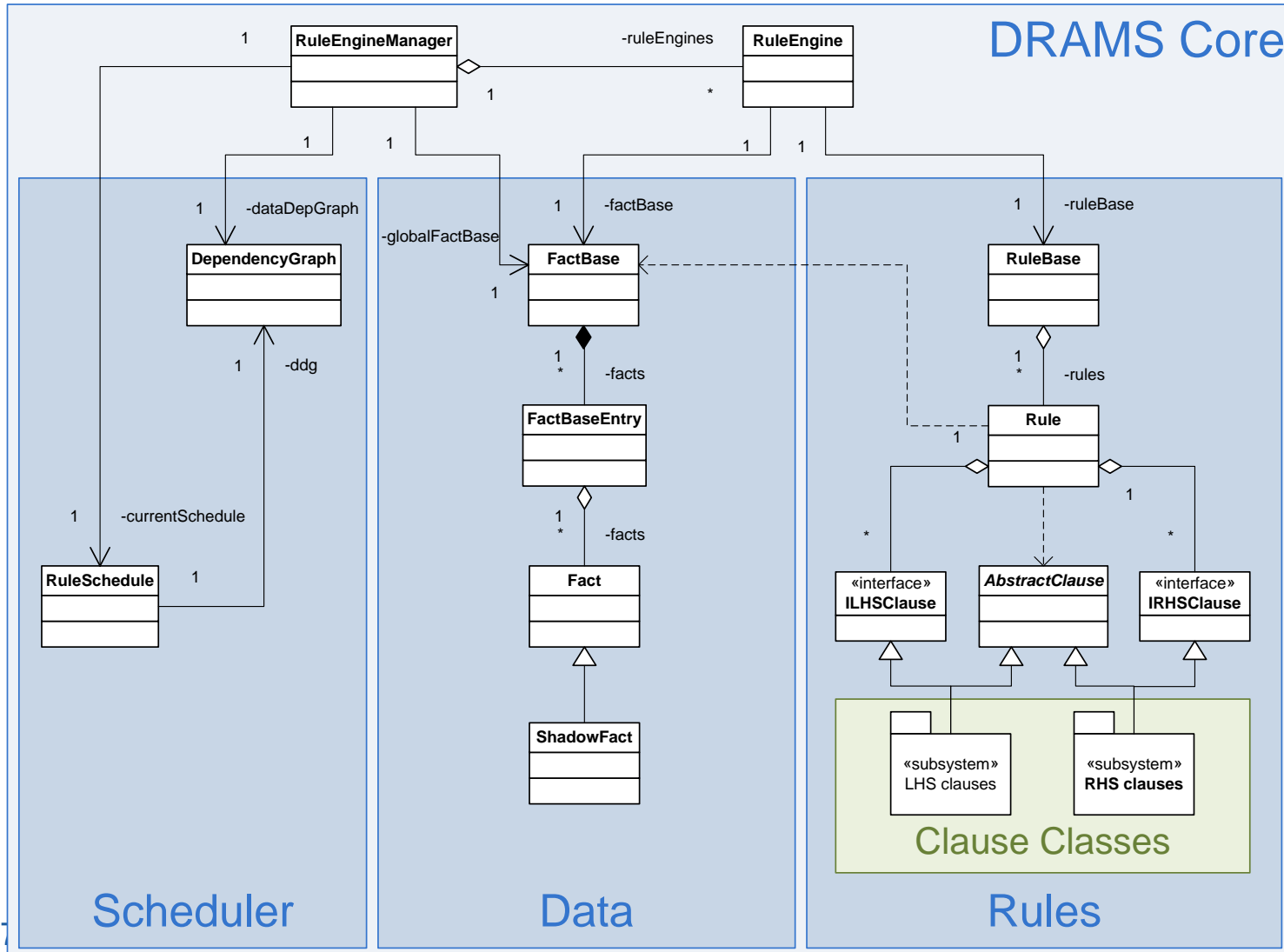
Some Implementation Details
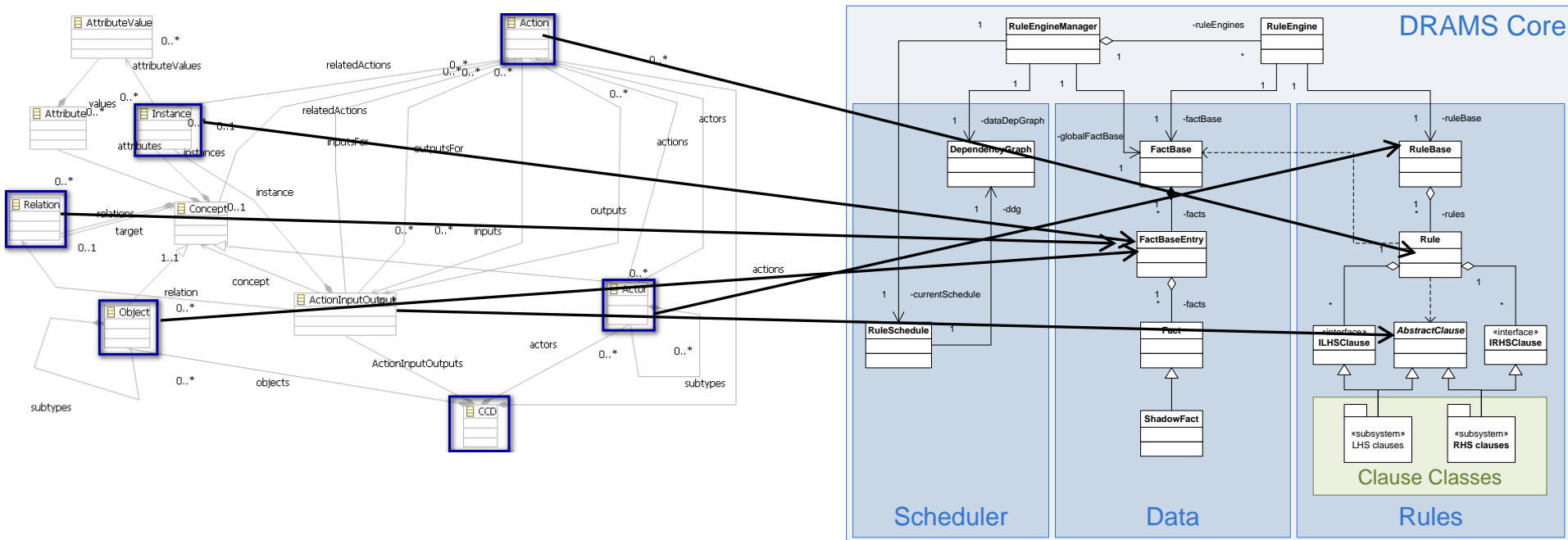
- ❖ CCD Tool:
  - ➤ Editor: Implementation as Eclipse-Plugin
  - ➤ Viewer: Will be implemented in RAP (Rich Ajax Applications)
  - ➤ Different visualisations for domain models (CCDs)
  - ➤ Modul based architecture => usable in other conceptual modeling approaches
- ❖ DRAMS
  - ➤ Declarative agent modelling platform
  - ➤ Implementation in Java based on Repast
  - ➤ Eclipse-Plugin available to support policy modellers

SEVENTH FRAMEWORK
PROGRAMME

Agenda

❖ Demands for ICT supported Governance and Policy Modelling and the OCOPOMO project

❖ Scenario building and analysis

❖ Integrating collaborative scenario texts with formal policy models

❖ **Pilot cases, expected outcomes and innovation**

Three cases

- ❖ Renewable energy policy in Kosice Self-governing Region (KSR)
- ❖ Knowledge transfer in tourism and cultural industries in Campania Region of Italy
- ❖ Housing policy in London (additional case)

Innovations in OCOPOMO

- ❖ OCOPOMO policy development process: integrated approach from narrative scenarios to formal policy models

  - ➢ Iterative process of identifying the parameters and features informing formal policy models

- ❖ Consistent conceptual description (CCD): Incorporating traceability in the iterative policy development process

- ❖ Open collaboration in policy development through integrated web 2.0 based e-participation toolbox

  - ➢ Enabling policy analysts, policy operators and wider stakeholder groups to work together collaboratively

Expected impact

- ❖ Contribution to strategic policies and to implement open government
- ❖ Contribution to transform government and administration to an open, effective and efficient participative governance (good governance principles)

- ❖ Provide new opportunities for open discourse among stakeholders of the policy domain and the policy experts
  - ➢ in stakeholder-oriented scenario generation
  - ➢ in evaluation of formal policy models

- ❖ Improve transparency and traceability in strategic decision making by involving different stakeholders in the participative process via the open collaboration platform

© OCOPOMO consortium, www.ocopomo.eu

www.ocopomo.org

Many thanks for your attention!

Project partners:

Agenda

- ❖ Demands for ICT supported Governance and Policy Modelling and the OCOPOMO project
- ❖ Scenario building and analysis
- ❖ Integrating collaborative scenario texts with formal policy models
- ❖ DRAMS
- ❖ Pilot cases, expected outcomes and innovation

© OCOPOMO consortium, www.ocopomo.eu

Agent design: overview

❖ Modelling framework is RePast 3.1

❖ DRAMS is implemented in Java and we have an abstract agent class independent of modelling framework and an abstract model class for RePast 3.1

❖ Agent design implemented as fact templates and rules

❖ Some facts can be shadow facts (using JavaBeans)

❖ Facts can be added to fact bases directly from Java to support updating and post-execution processing.

❖ Categories of rules for decision-making and for endorsements

© OCOPOMO consortium, www.ocopomo.eu

Agent design: DRAMS

❖ Each agent has a rule engine, rulebase and fact base

❖ Agents can read and write to their own fact bases and communicate with other agents by writing on the other agents' fact bases

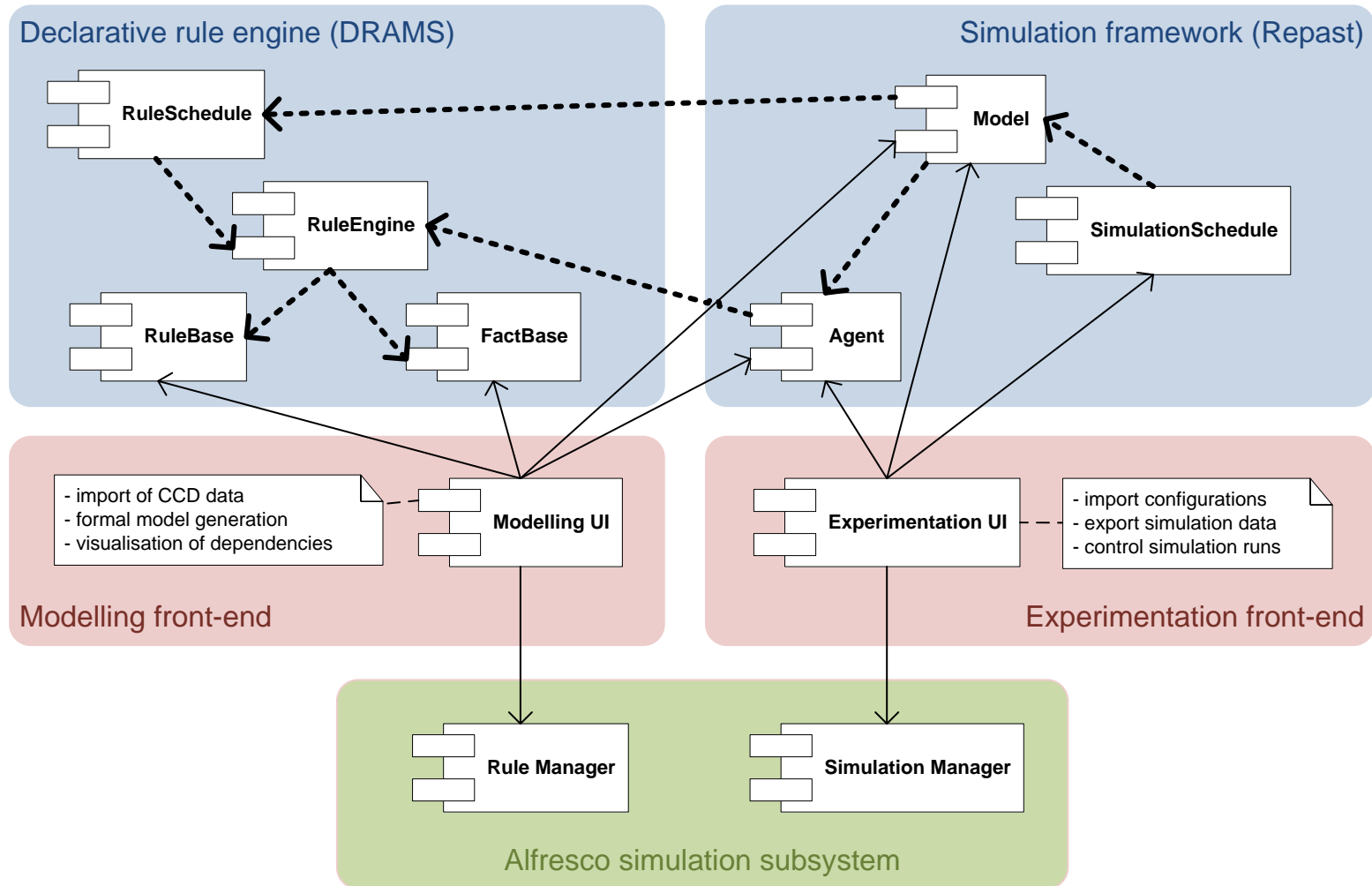❖ Public knowledge is represented by facts on a global fact base open to every agent for writing and reading
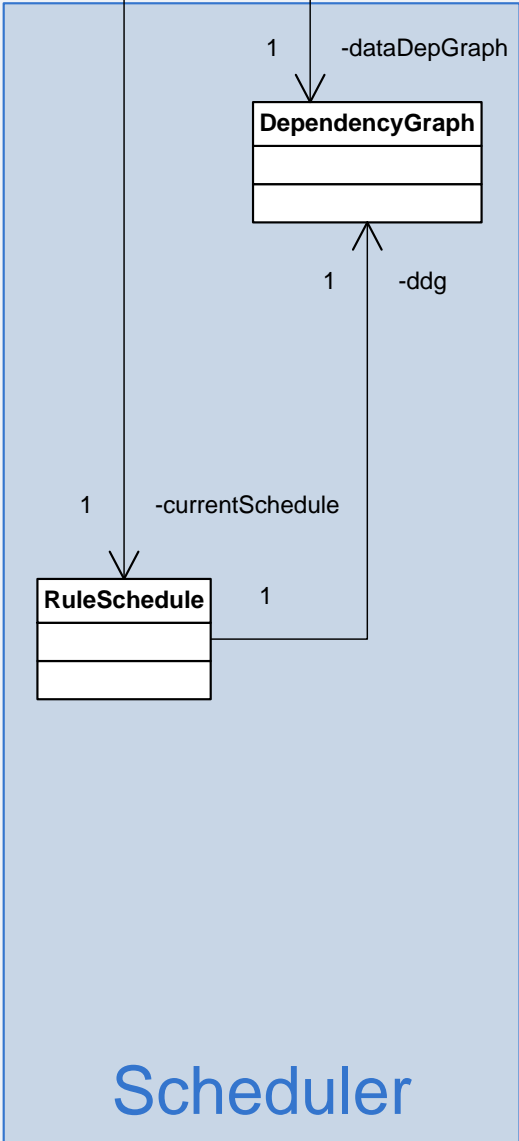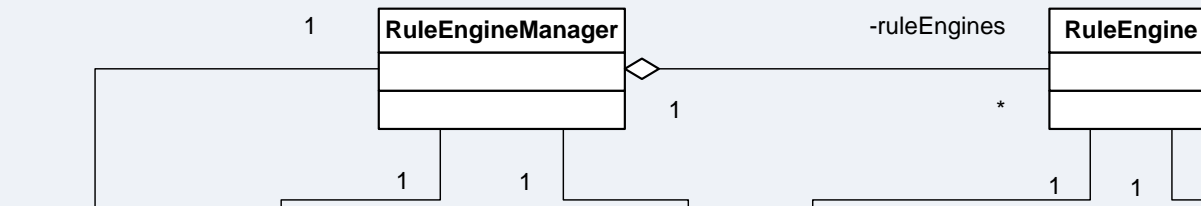
Agent design: endorsements

❖ Endorsements are mnemonic tokens attached by agents to other agents (reliable, trustworthy, etc.) or to other objects such as plans or mental models (successful, unsuccessful)

❖ Endorsements have ordinal values used to rank agents or other objects such as plans or technologies.

❖ For efficiency, endorsement values calculated in Java and asserted as facts to agent's fact base

DRAMS

- ❖ Declarative agent modelling platform
  - ➢ Equip agents with expert system capabilities: describing agent behaviour by declarative rules
  - ➢ Individual rule sets for each agent type (or even instance)
  - ➢ Individual working memory for each agent instance
- ❖ Rule engine component for multi-agent simulation models
  - ➢ Distributed rule engine: agents behave in simulation runs autonomously according to their rule specifications
  - ➢ Simulation dynamics is generated by individual agent behaviour, together with interaction between agents (inter-agent communication)
- ❖ Java-based implementation
  - ➢ Full flexibility regarding interface definition for integration with OCOPOMO toolbox
  - ➢ Integrable with widely applied simulation tools (e.g. Repast)
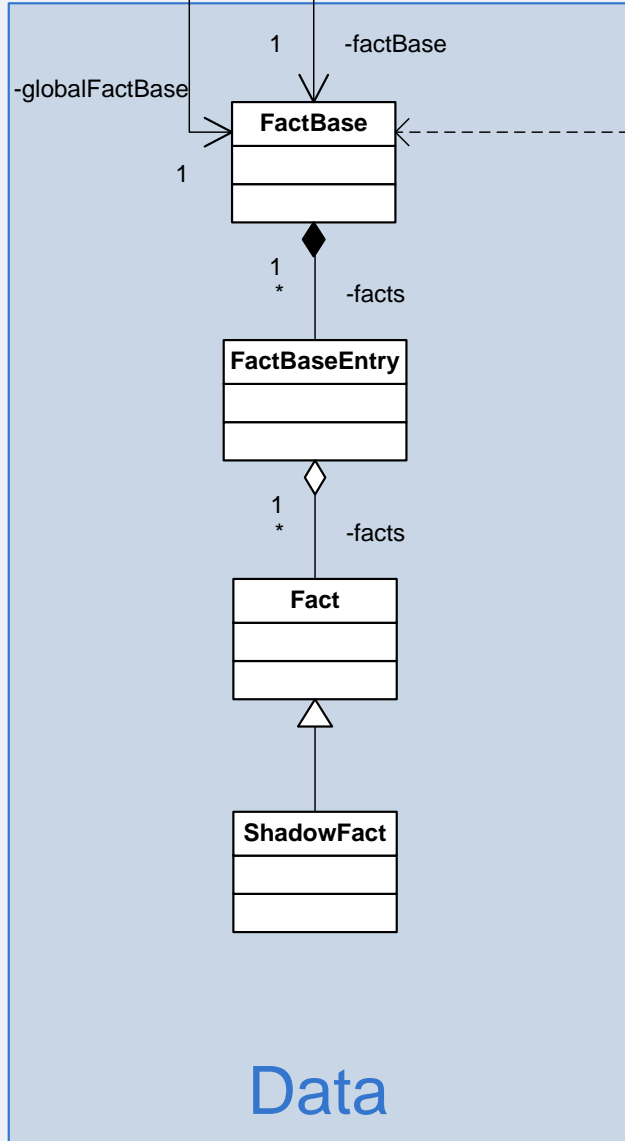 - ➢ Completely open source (including used external libraries)

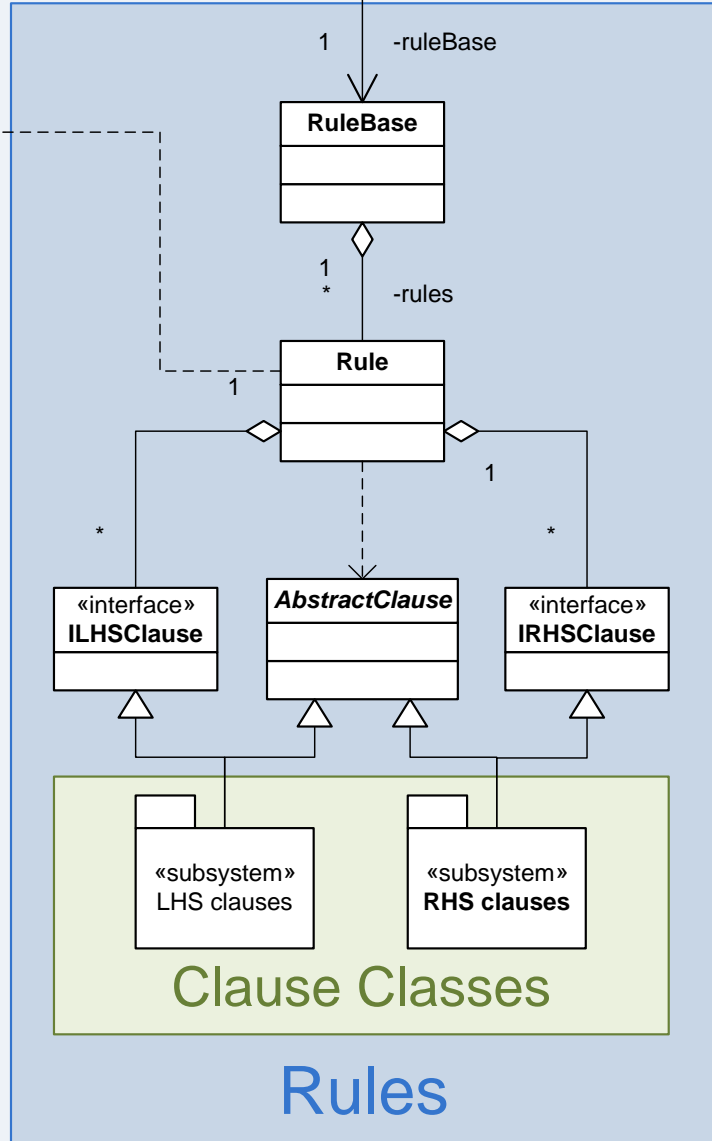© OCOPOMO consortium, www.ocopomo.eu

DRAMS: rule engine

❖ Software system, consisting of:
  ➢ A fact base, which stores information about the state of the world in the form of facts.
  ➢ A rule base, which stores rules describing how to process certain facts stored in fact bases. A rule consists of a condition part (called left-hand side, LHS) and an action part (called right-hand side, RHS).
  ➢ An inference engine, which controls the inference process by selecting and processing the rules which can fire on the basis of certain conditions.
❖ DRAMS is designed as a distributed, forward-chaining rule engine
  ➢ Incorporates a data-drive rule scheduling mechanism to efficiently cope with intensely dynamic fact base contents (which is typical for simulation applications)
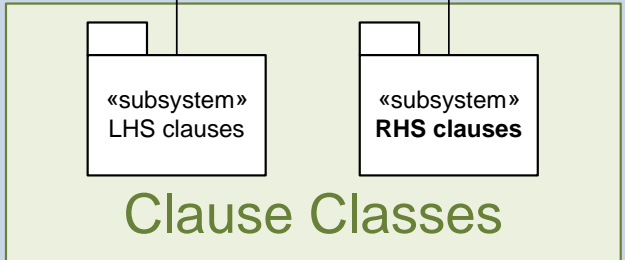 ➢ …