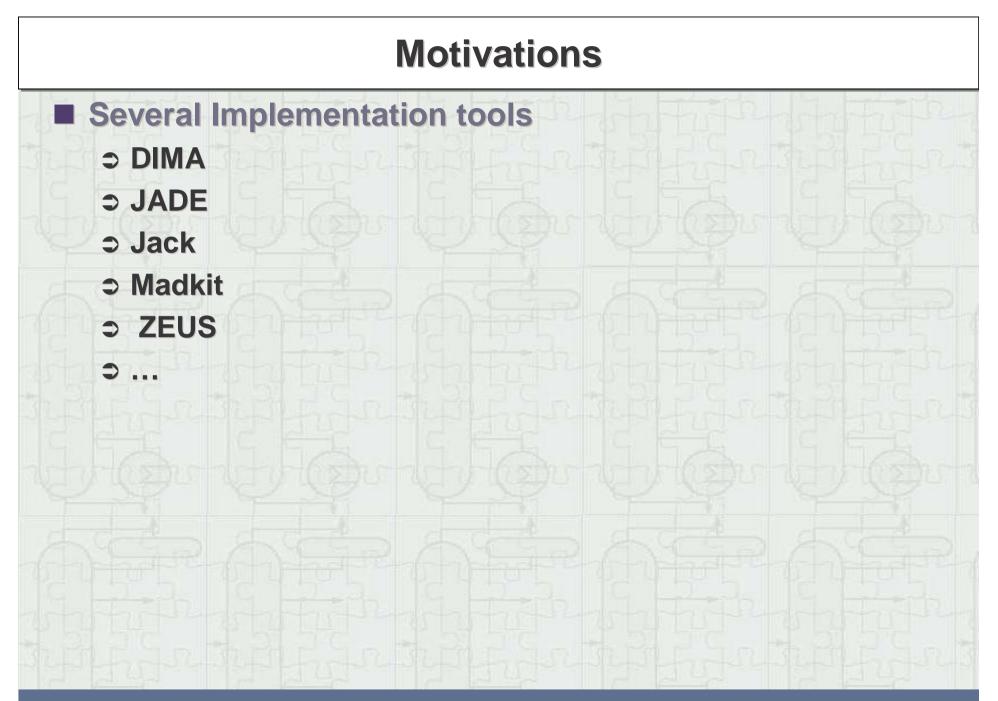# Meta-Models &
Model-Driven Architectures

Zahia Guessoum and Tarek Jarraya

OASIS (Objects and Agents for Simulation and Information Systems)
LIP6 (Laboratoire d'Informatique de Paris 6)

Zahia.Guessoum@lip6.fr
http://www-poleia.lip6/~guessoum

Motivations

- **Several Implementation tools**
  - DIMA
  - JADE
  - Jack
  - Madkit
  - ZEUS
  - …

Motivations

- **Several methodologies/Meta-models**
  - ⮕ **Aalaadin**
  - ⮕ **Adelfe**
  - ⮕ **GAIA**
  - ⮕ **INGENIAS**
  - ⮕ **PASSI**
  - ⮕ **RICA**
  - ⮕ **Tropos**
  - ⮕ **…**

Motivations

- **Problems**
  - ➦ **Methodologies do often not rely on existing agent-based development tools**
  - ➦ **MAS implementation requires deep knowledge of technical details of agent architectures and agent development tools, multi-agent concepts ...**
  - ➦ **Technical migration (new version of the platform ...) may evolve several hard problems**

- **Question: how to bring the gap between existing multi-agent tools and methodologies ?**

A Solution: MDA approach

- **Use a *Model-Driven Architecture* (OMG) approach to develop MAS**

- **Separate:**
    - ➲ **Application logic (e.g. PIM = Platform Independent Model)**
    - ➲ **From the underlying technologies (e.g. PSM = Platform Specific Model)**
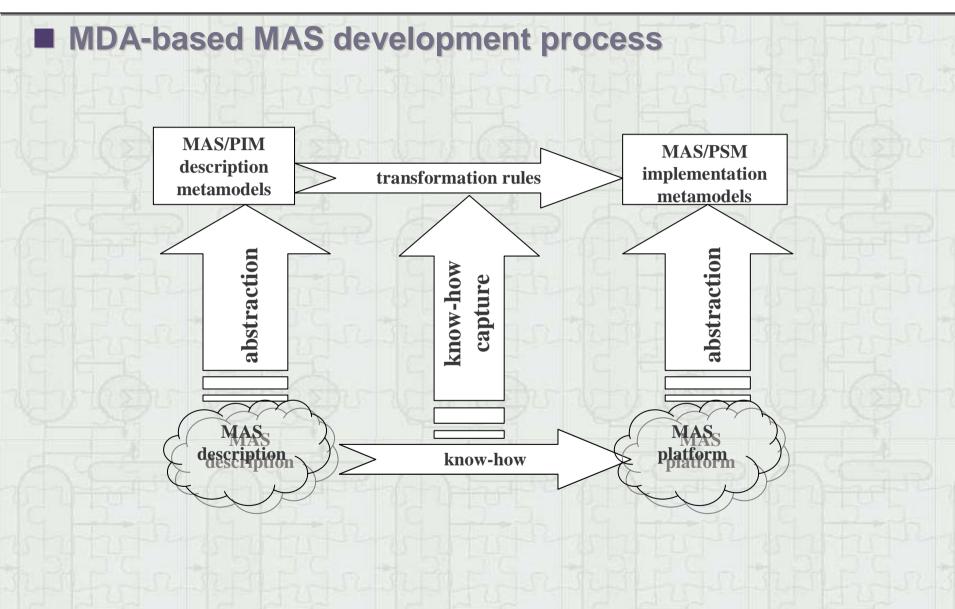
- **Mechanisms/tools:**
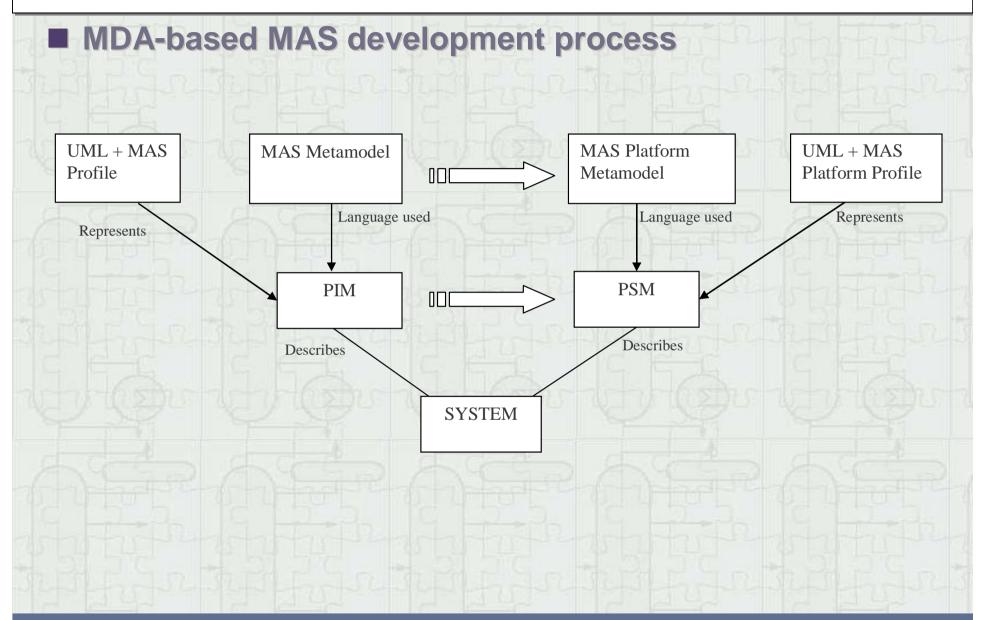   - ➲ **Meta-modeling**
    - ➲ **Transformations of models**

- **Aim:**
    - ➲ **Improve reusability**
    - ➲ **Facilitate the development process**

Overview of Meta-DIMA

■ **MDA-based MAS development process**

Overview of Meta-DIMA

■ **MDA-based MAS development process**


```
┌──────────────┐    ┌──────────────┐         ┌──────────────┐    ┌──────────────┐
│ UML + MAS    │    │ MAS Metamodel│  ▭▭▭▭▷  │ MAS Platform │    │ UML + MAS    │
│ Profile      │    │              │         │ Metamodel    │    │ Platform     │
│              │    │              │         │              │    │ Profile      │
└──────────────┘    └──────────────┘         └──────────────┘    └──────────────┘
```

Represents   Language used   Language used   Represents

```
              ┌──────────┐               ┌──────────┐
              │   PIM    │  ▭▭▭▭▷        │   PSM    │
              └──────────┘               └──────────┘
```

Describes   Describes

```
                    ┌──────────────┐
                    │   SYSTEM     │
                    └──────────────┘
```
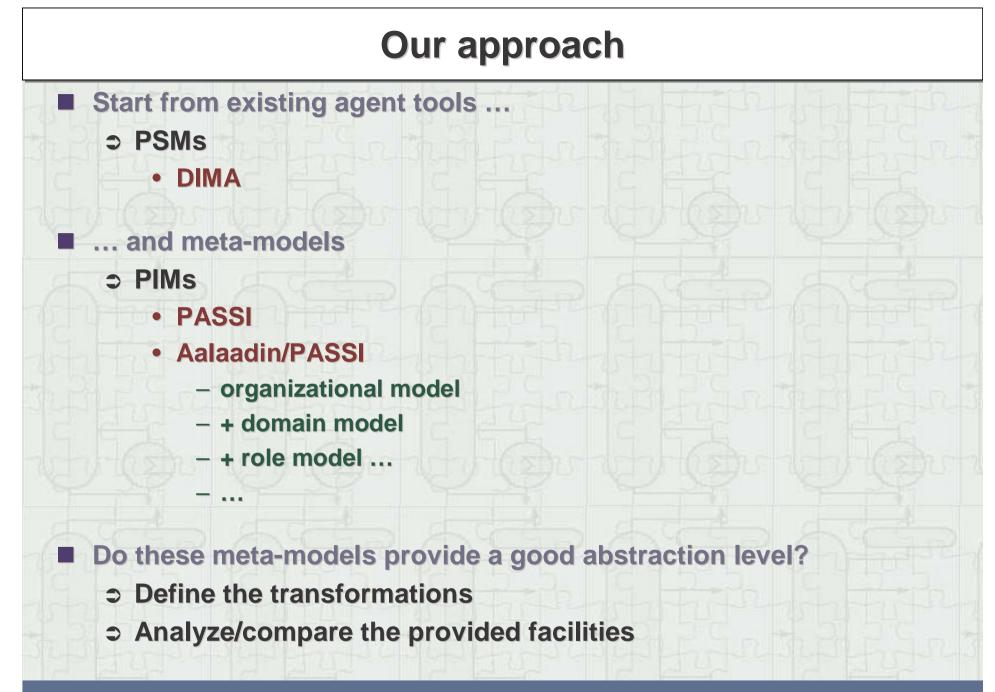
Overview of Meta-DIMA

- **The elaboration of Meta-DIMA is based on the following steps:**

  1. **Identification of the different abstraction levels: analysis of multi-agent applications, theories and tools to define the PIMs and the PSMs.**

  2. **Definition of a library of meta-models: identification of the concepts of each abstraction level and determination of the appropriate meta-models.**

  3. **Design of transformation rules: Analysis of the know-how involved in a multi-agent system development to define the transformation rules.**

Our approach

- **Start from existing agent tools …**
  - ➲ **PSMs**
    - **DIMA**

- **… and meta-models**
  - ➲ **PIMs**
    - **PASSI**
    - **Aalaadin/PASSI**
      - – **organizational model**
      - – **+ domain model**
      - – **+ role model …**
      - – **…**

- **Do these meta-models provide a good abstraction level?**
  - ➲ **Define the transformations**
  - ➲ **Analyze/compare the provided facilities**

DIMA-based PSM

- **DIMA**
  - ⮥ **Modular agent architecture**
  - ⮥ **Reuse of existing paradigms**
    - • rule-based systems, finite state machines, classifiers, CBR …

- **DIMA provides several frameworks**
  - ⮥ **ATN-based Framework**
  - ⮥ **Rule-Based Framework**
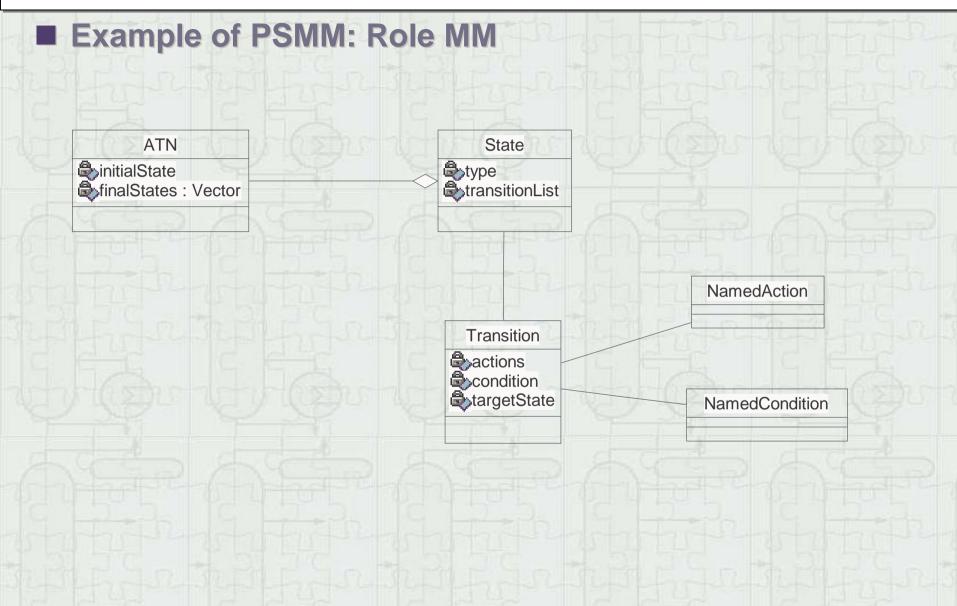 - ⮥ **Case-Based Framework**
  - ⮥ **Classifier-Based Framework**
  - ⮥ **…**

- **DIMA provides several libraries**
  - ⮥ **Agent models**
  - ⮥ **Interaction protocols**
  - ⮥ **Communication components**
  - ⮥ **…**

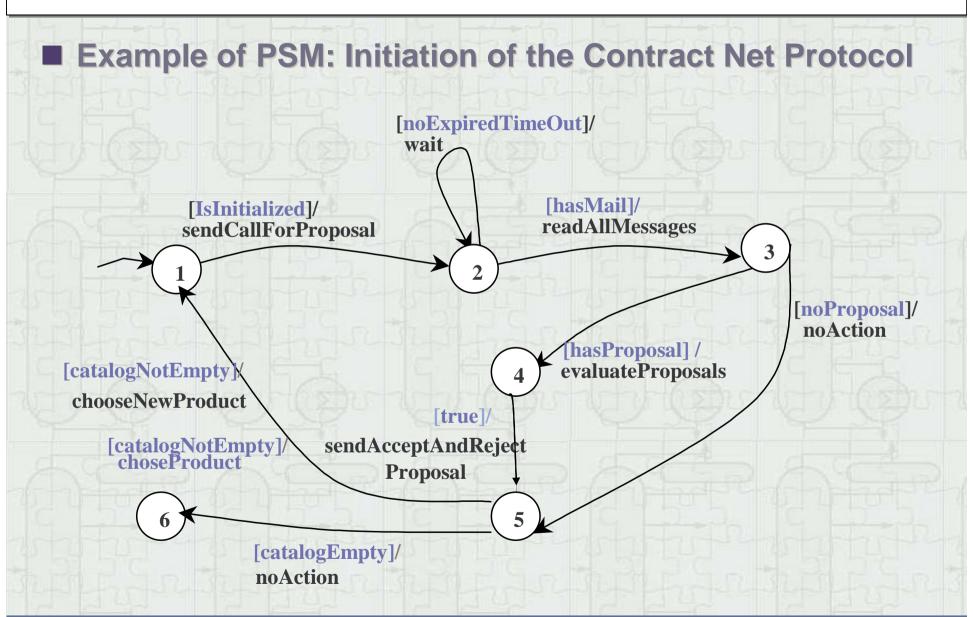# DIMA-based PSM

- **Several libraries**
  - ➲ **Provide facilities to build MAS**

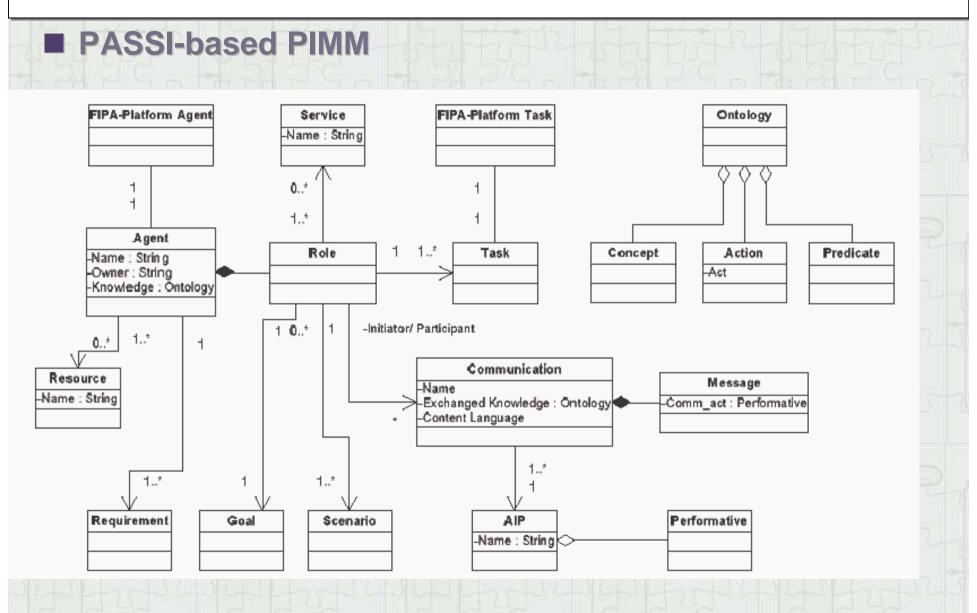 - ➲ **It is difficult to use all these libraries and their components**

  - ➲ **PSMs**
    - **Provide abstractions**
    - **Facilitate the use of DIMA**

DIMA-based PSM

■ **Example of PSMM: Interactive Agent MM**

■ **Example of PSMM: Role MM**

ATN
- initialState
- finalStates : Vector

State
- type
- transitionList

Transition
- actions
- condition
- targetState

NamedAction

NamedCondition

DIMA-based PSM

■ **Example of PSM: Initiation of the Contract Net Protocol**

PASSI
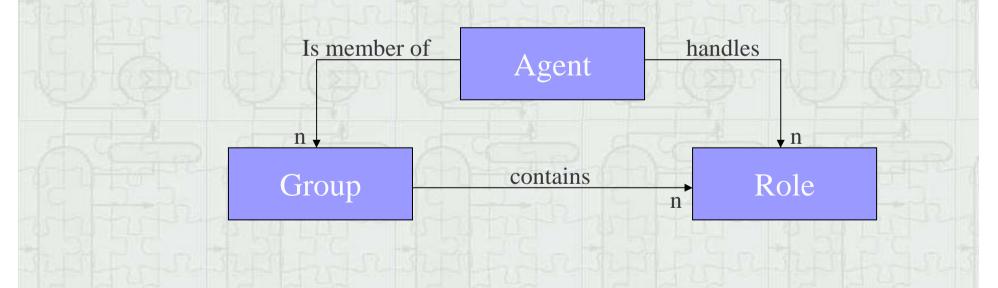
- **PASSI-based PIMM**

Aalaadin/PASSI-based PIM

- **Several kinds of MMs**
  - ⮑ **Domain MM**
    - Example: an ontology (see PASSI)
  - ⮑ **Organizational MM**
    - Example: Aalaadin
  - ⮑ **Role MM**
  - ⮑ **Agent MM (Simplified version of PASSI)**

Is member of | Agent | handles

n

Group | contains | Role

n | n

Transformations

- **PASSI -> DIMA-based PSM**
  - ⮑ **Transformation rules <=> matching rules**
  - ⮑ **Some concepts are not used**

- **Aalaadin/PASSI -> DIMA-based PSM**
  - ⮑ **Transformation rules represent the know-how**
    - • **Ex: deduce the agents from the graph of roles**
  - ⮑ **Few concepts**
    - • **Easy to use**

Comparison of the two PIMS

- **Three examples**
  - ➲ **BookStore**
  - ➲ **TimeTable**
  - ➲ **Preys/Predators**

- **Develop MAS:**
  - ➲ **Without MMs**
  - ➲ **With PASSI-based PIMM**
  - ➲ **With Aalaadin/PASSI-based PIMMs**

- **Conclusion**
  - ➲ **Developing MAS with MMs is easier**
  - ➲ **Developing MAS with Aalaadin/PASSI-based MMs is often easier than PASSI-based MMs**
    - **Ex: facilities to develop heterogeneous agents**

Questions

- **Several Meta-Models**
  - ➲ **Aalaadin, Adelfe, GAIA, INGENIAS, PASSI, RICA, Tropos …**

- **Meta-DIMA**
  - ➲ **Reuse these MMs**

- **The best solution?**
  - ➲ **Several MAS MMs**
    - **Characteristics of each MM**
    - **Target appplications**
  - ➲ **One MAS MM= Adelfe + GAIA + INGENIAS + PASSI + RICA +Tropos + …**
  - ➲ <u>**Several Meta-Models**</u>
    - **Agent MMs**
    - **Role MMs**
    - **Organization MMs**
    - **…**