

A Modeling Language for Communicating Architectural Solutions in the Domain of Robot Control

CAR 07

Robin Passama, groupe *OBASCO*



Introduction

- Domaine de recherche : **Conception** des architectures de contrôle de robots
- Question fondamentale : Comment **décrire**, **communiquer** et **comparer** les solutions architecturales proposées dans le domaine ?
- Proposition : un **langage de modélisation** d'architectures de contrôle / de solutions architecturales



Introduction

- Pourquoi un langage de modélisation dédié ?
 - Un **cadre terminologique et conceptuel** commun pour se comprendre facilement
 - Adequation des **représentation graphiques** avec la façon de se représenter une architecture

- Obstacles
 - Trouver les **bonnes abstractions** et la bonne **séparation des préoccupations**
 - Mettre en relations les **pratiques** existantes

- Bénéfices potentiels
 - Des patrons de conception ?

Plan de la présentation

- Introduction

- Réflexion sur l'état de l'art
 - › Défi, problème & direction
 - › Constat sur les pratiques courantes
 - › Points communs

- Langage de modélisation
 - › Concepts et terminologie
 - › Meta-modèle

- Exemple

- Conclusion

Réflexion sur l'état de l'art

Défi, limitation & direction

- **Défi** : exprimer et intégrer des expertises diverses dans une architecture de contrôle
 - contrôle, navigation, vision, IA, etc.
 - médecine, défense, etc.
- **Limitations** :
 - Les paradigmes utilisés sont centrés sur les **besoins et pratiques des ingénieurs logiciels** par sur ceux des experts du domaine
 - Pas de paradigme du tout ...
- **Direction**: Les DSL (Domain specific languages) permettent aux experts de domaine d'exprimer leur solutions avec une terminologie métier
- ✓ **Un DSL pour les architectes robotiques**

Réflexion sur l'état de l'art

Constat sur les pratiques courantes

- Différents niveaux d'abstraction et différents niveaux de détails techniques
 - Framework
 - Aspects matériels
 - langages/paradigmes (composants, objets, etc.)

- Nouvelles définitions
 - **Architecture de contrôle de robot**
 - « is a model of a robot controller that captures the **decomposition of robot decisional process**, its perception and action capabilities into interacting pieces of different levels of decisional complexity and of different responsibilities within each level. »
 - Architecture logicielle de robot

Réflexion sur l'état de l'art

Constat sur les pratiques courantes

- Variété des méthodologies de conception d'architectures de contrôle [Mataric, 2002]
 - Délibérative, Réactive, Comportementale, Hybride
 - Différences essentiellement basées sur l'approche de décomposition choisie (fonctionnelle, agent, mixte).

- Différentes interprétations (ex: hybrides)
 - Décomposition fonctionnelle et agent (ex : Aura)
 - Interaction entre couches non-adjacentes (ex : LIRMM, CLARATy)
 - Décomposition en systèmes hiérarchisés collaborant (ex : IDEA, CLARATy)

- ✓ Identifier des concepts génériques utilisables quelle que soit la méthodologie



Réflexion sur l'état de l'art

Points communs

- **Concepts récurrents**
 - **Couche** : niveau d'abstraction/ de complexité de la prise de décision
 - **Activité** : entité possédant la responsabilité d'un ensemble de décisions (planification, contrôle, navigation, etc.)

- **Concepts « exceptionnels » mais toujours applicables**
 - **Système** : partie du processus décisionnel responsable du contrôle d'une partie du système physique
 - **Connaissance** : entité contenant les informations sur un sujet donné (partie matérielle, environnement, mission, etc.)

- **Variation récurrente**
 - **Coordination** : interactions entre activités participant au processus décisionnel

Langage de modélisation

Concepts et terminologie

- Focus : Decomposition du processus decisionel

- Réutilisation des concepts précédents
 - **Couche** : organisation hiérarchique
 - **Systeme** : relation explicite entre le processus décisionel et le système physique contrôlé
 - **Activité et Coordination** : décomposition « atomique » du processus décisionel
 - **Connaissance** : qualification précise du « sujet » sur lequel travaille chaque partie du processus décisionel

- Contraintes :
 - Pas de considération des détails d'implémentation

Langage de modélisation

Concepts et terminologie

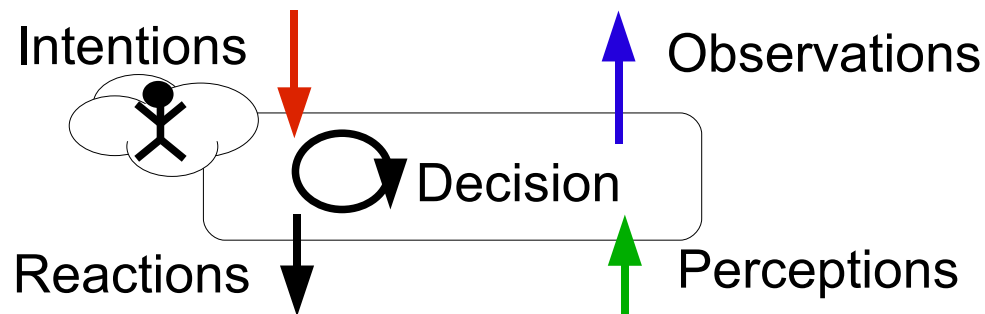
- **Connaissance (Knowledge)**
 - « A Knowledge entity identifies a structured piece of information about the world within which the robot controller evolves »
- **Catégorisation des aspect stable en robotique [Brugali and Salvaneschi, 2006]**
 - **Embodiement** : refers to the consciousness of having a body that allows the robot to experience the world directly.
Types : *Body, Infrastructure*
 - **Situatedness** : refers to evolving in a complex, dynamic and unstructured environment that strongly affects the robot behavior.
Types : *Environment, Interaction, Phenomenon*
 - **Intelligence** : refers to the ability of the robot to adopt adequate and useful behaviors while interacting with the dynamic environment.
Types : *Behavior, Action, Strategy, Objective*

Langage de modélisation

Concepts et terminologie

■ Activité (Activity)

- « An Activity entity is responsible for the achievement of a task that plays a role in the robot decisional process, using a set of Knowledge entities »



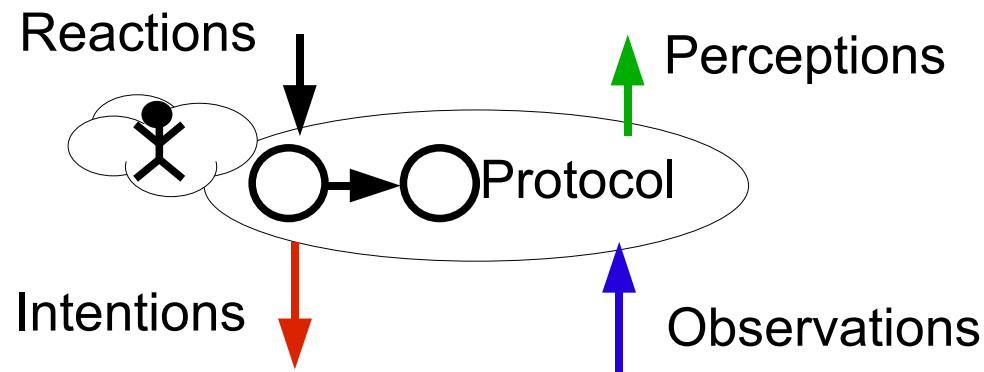
- **Decision** : « mechanism that computes coherent Reactions and Observations according to Intentions and Perceptions of any level of abstraction »
- **Perceptions** : « required pieces of information, or significant phenomena notification »
- **Reactions** : « results of decision that define command of any level of abstraction »
- **Intentions** : « goals that an activity intends to accomplish »
- **Observations** : « interesting observed states of environment or robot »

Langage de modélisation

Concepts et terminologie

▪ Coordination

- « A coordination entity is responsible of the way a set of Activity entities interact by exchanging or sharing Knowledge entities »



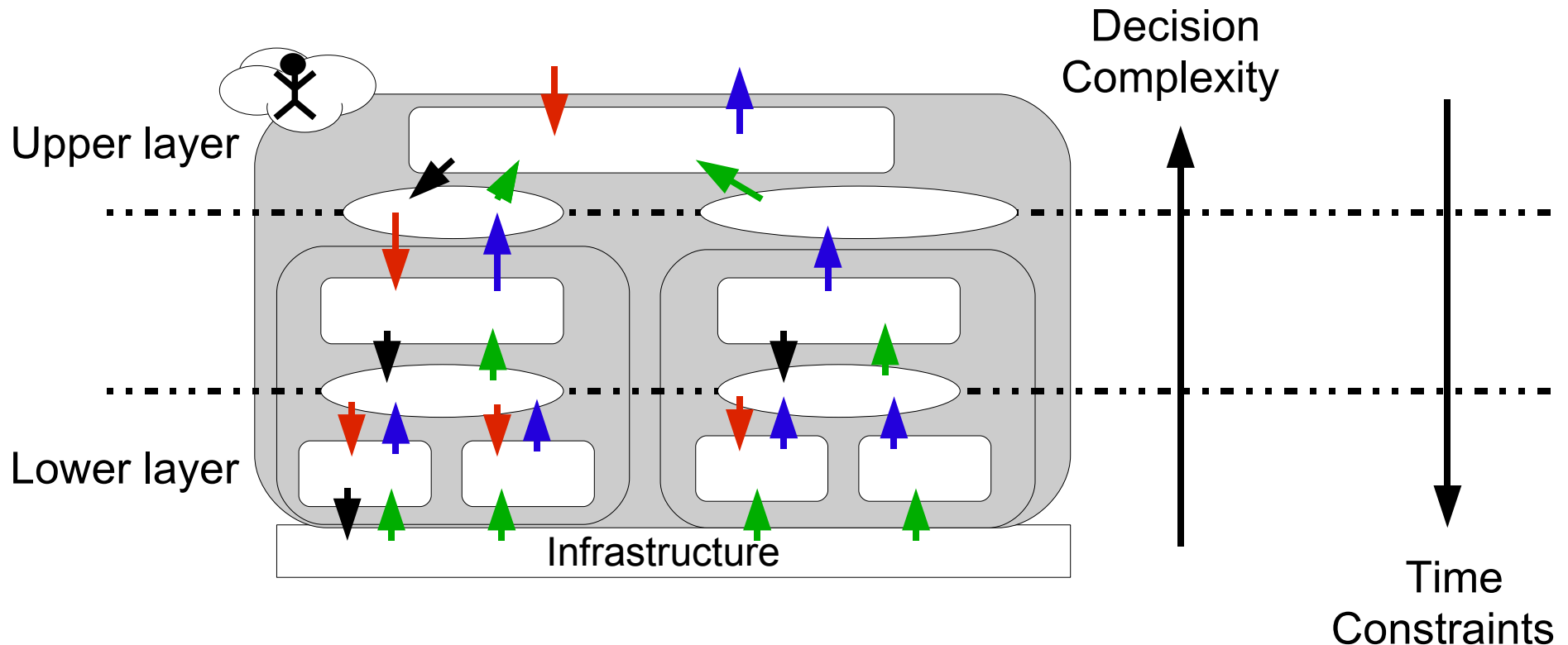
- **Protocol** : « mechanism that defines the way Activity entities interact and which Knowledge entities they exchange »

Langage de modélisation

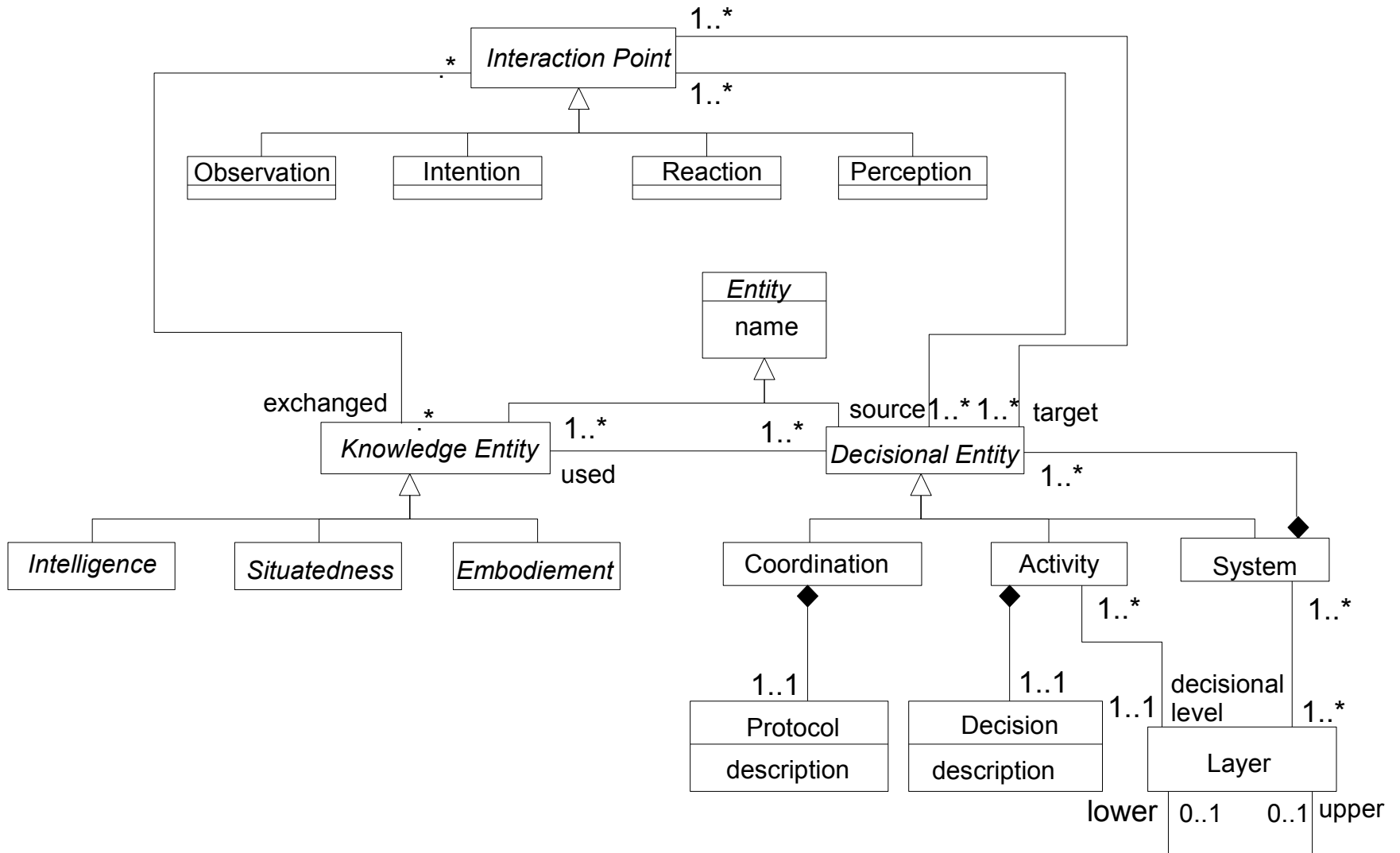
Concepts et terminologie

▪ Système (System)

- « A System entity is an abstraction of the control of a physical entity (represented by a Body entity). It is responsible of the way a set of Activity and Coordination entities, concerning this physical entity, are organized in order to achieve a set of tasks.»

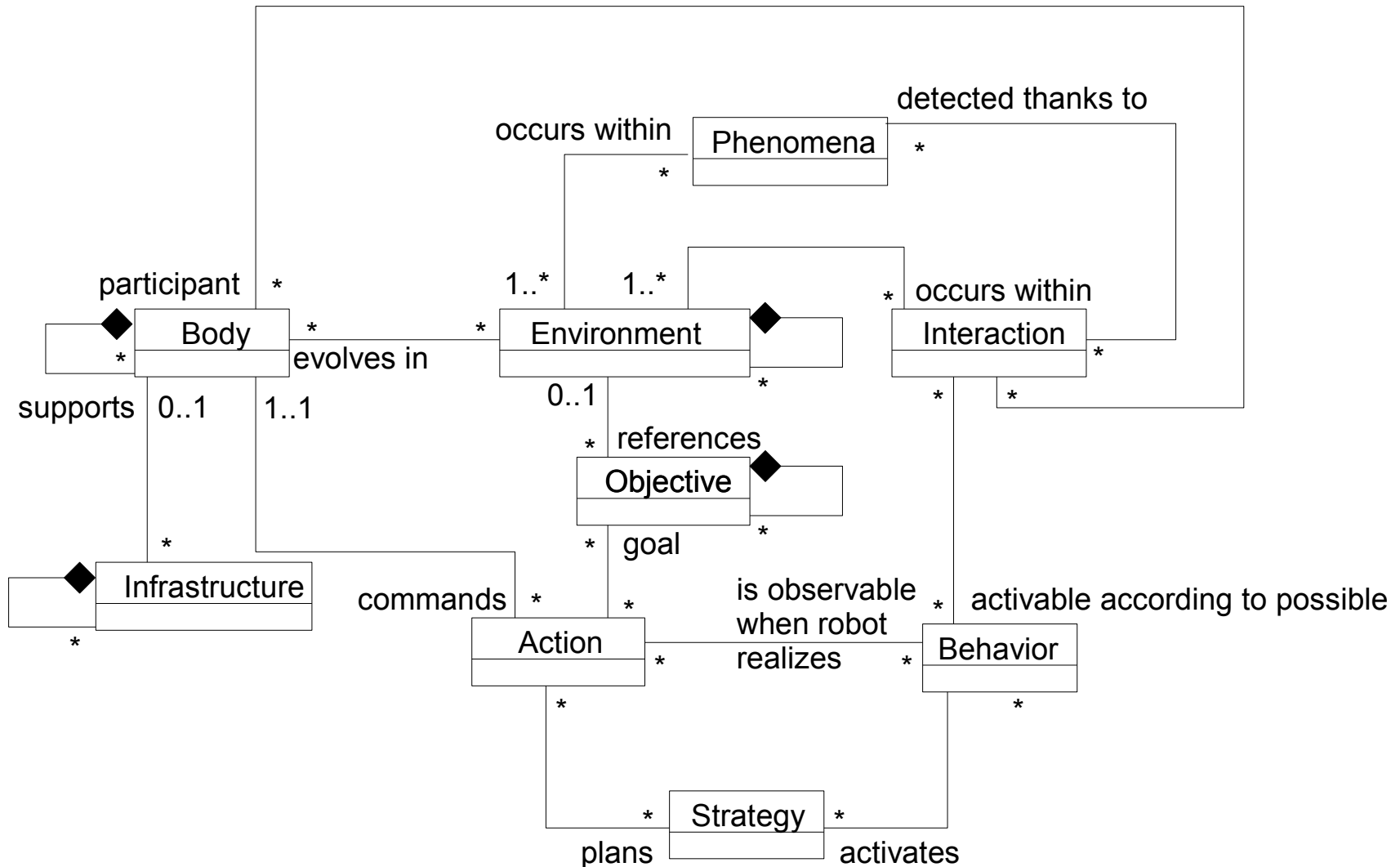


Langage de modélisation *Meta-modèle*

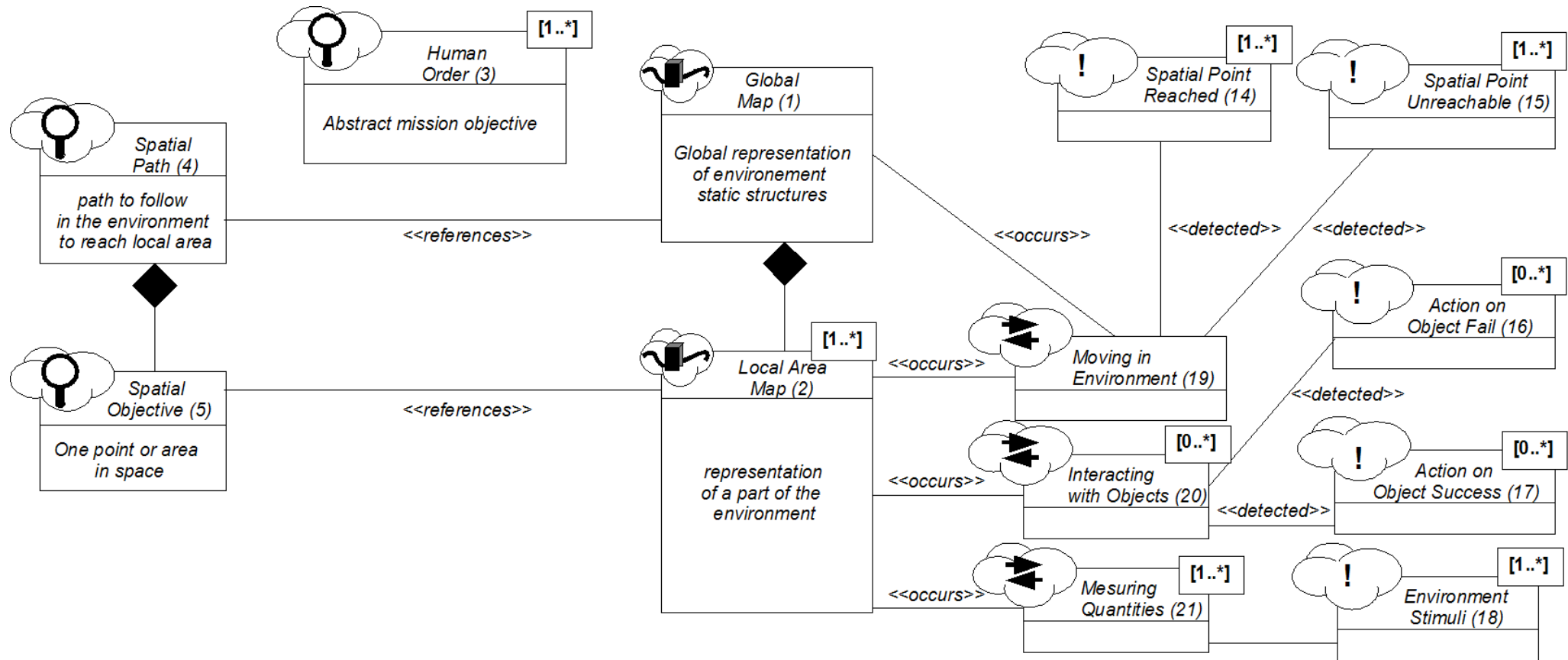




Langage de modélisation *Meta-modèle*

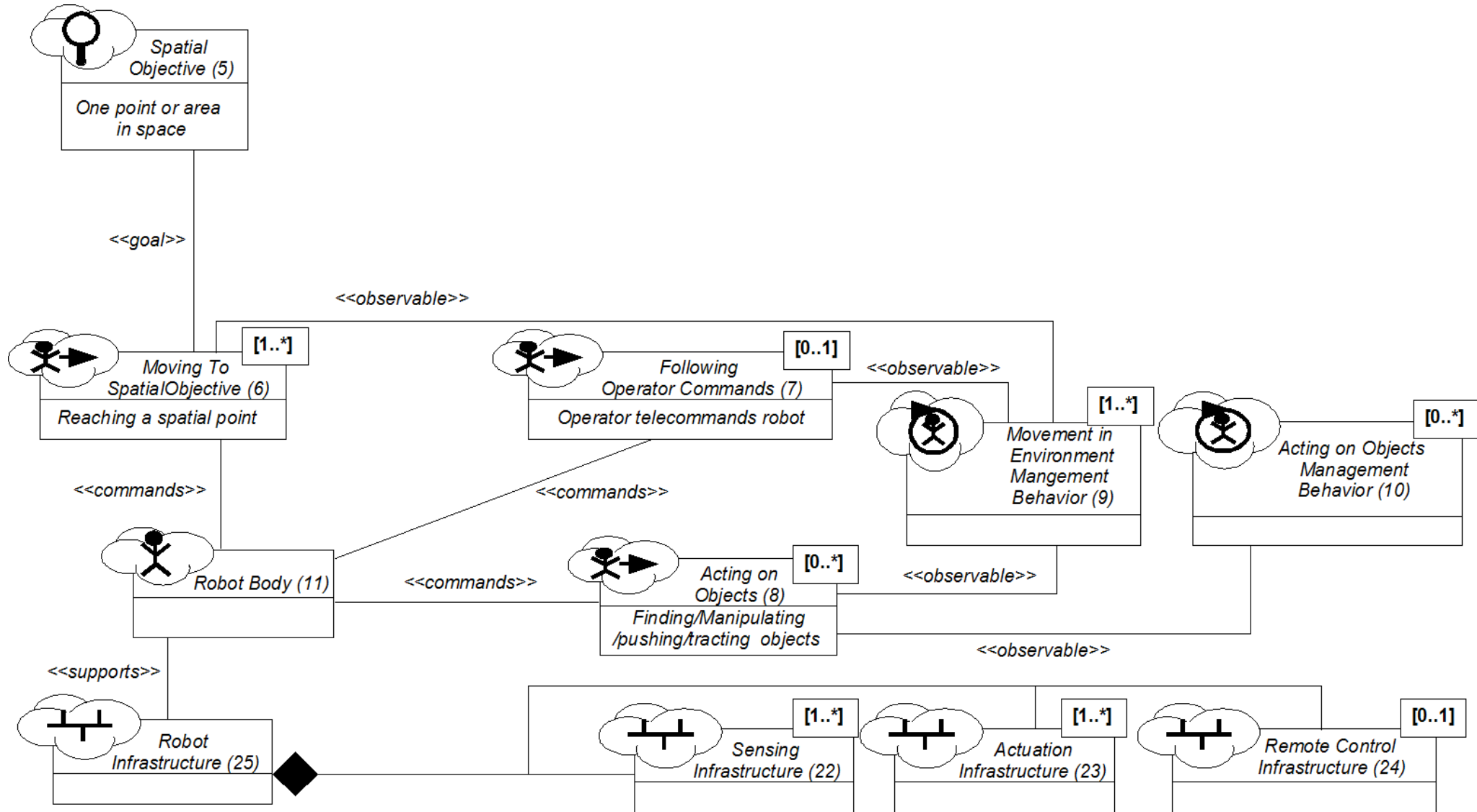


Exemple Aura



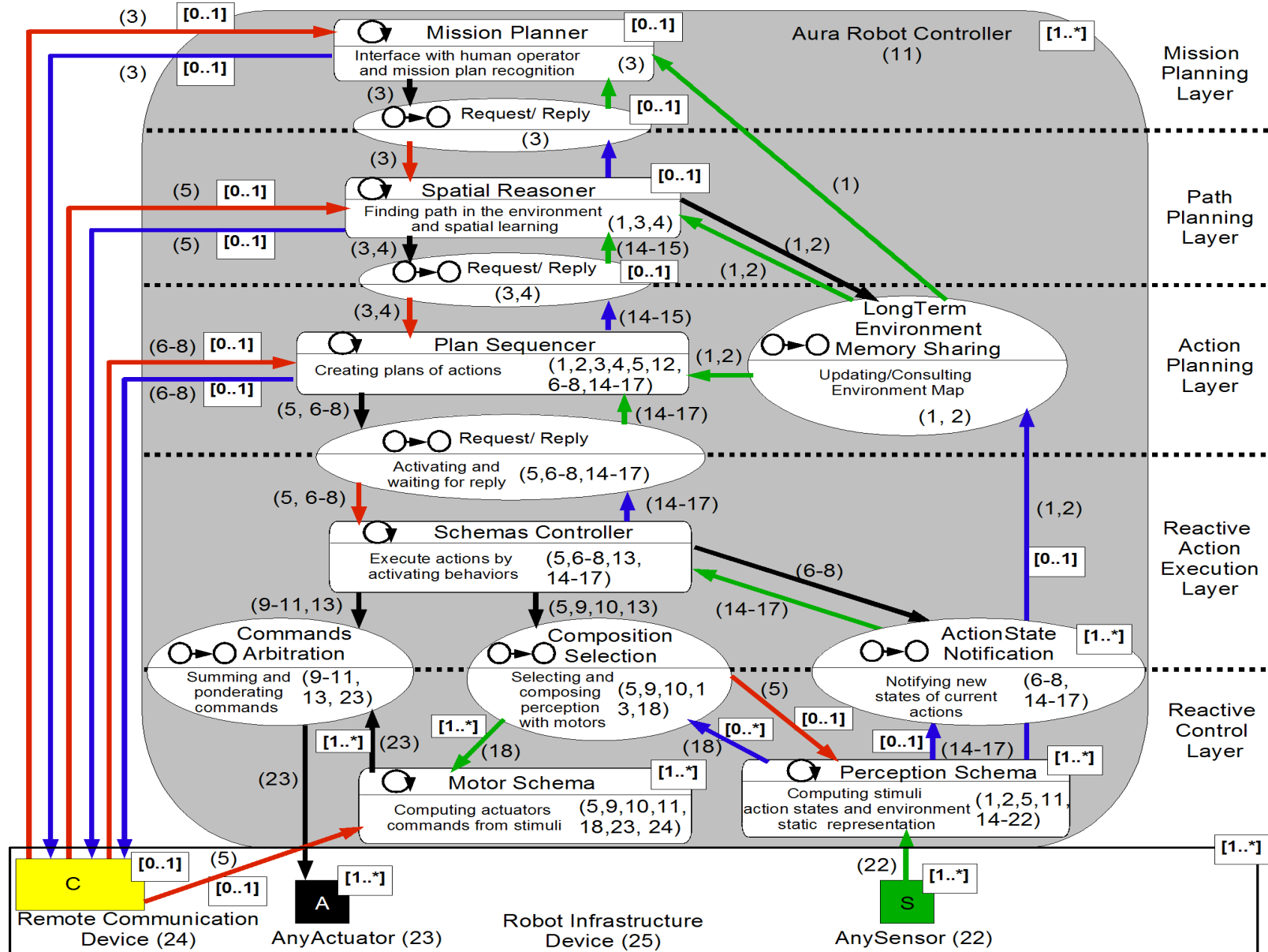


Exemple Aura





Exemple Aura





Conclusion

- **Langage de modélisation**
 - Basé sur un constat général : difficile de communiquer et comparer les propositions d'architectures de contrôle
 - Concepts fondateurs basés sur une analyse du domaine
 - Qui a pour focus la **décomposition du processus décisionnel**
 - Qui permet primer des solutions architecturales (généricité)

- **Perspectives**
 - Comparaison des travaux existants : vers une définition des points communs et variations
 - Evolution : vers une description plus détaillée (formelle ?) des propriétés des entités (Activité, Coordination, Connaissance) ?
 - Evolution : un langage exécutable ? génération de code ?