

MAS meta-model in PASSI

Version: 28-Aug-2003

Document Author: Massimo Cossentino

In this document I will describe the (meta-)model of MAS that is behind a design process with PASSI. A schema of this meta-model is represented in the following figure.

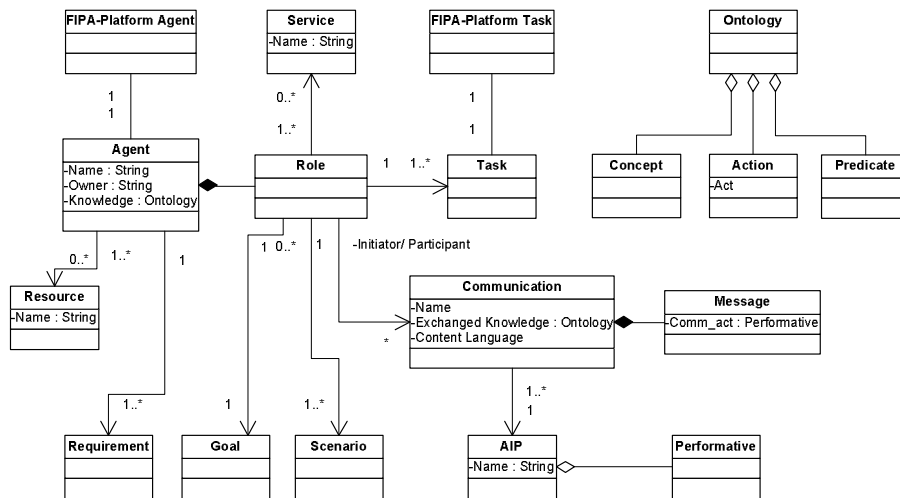


Figure 1: the MAS meta-model in PASSI

In PASSI, agents (*Agent* class in Figure 1) are supposed to be FIPA agents. Therefore they are implemented within a FIPA-compliant platform (relationship with the *FIPA-Platform Agent*); their tasks/behaviours are implemented as tasks/behaviours of the selected platform.

Each agent is supposed to be devoted to the fulfilment of one or more functional requirements (relationship with the *Requirement* class). Requirements in PASSI are expressed as use cases in the Domain Description Phase and assigned to agents in the Agent Identification Phase. An agent can have at its disposal some resources that it can use to achieve its goals, provide the functionalities associated with the requirement associated to it and/or can share it with other agents.

Agents are composed of roles that aims at reaching a specific goal (relationship among *Role* and *Goal*), and/or providing a service to the agent society. Roles activity is described in scenarios (sequence diagrams of the PASSI Role Identification Phase).

An agent role can be the initiator or participant in a communication. Each communication is composed by messages (each one referring to a specific communicative act or performative) and is characterized by a name (used to distinguish the communication from the others at design time), an exchanged knowledge (a piece of the ontology of the whole domain) and a content language.

Ontology is composed of concepts, predicates and actions.

Glossary of PASSI MAS components

In the following table a list of the entities used to compose the previous described MAS meta-model is presented with a description of the specific meaning that it has within PASSI and the list of PASSI diagrams that concur to define it and/or report it.

MAS component	Description	PASSI diagram(s) dealing with it	Relation MAS component/ PASSI diagram
Agent	An agent (FIPA agent) is a software entity: - which is capable of action in an environment - which can communicate directly with other agents typically using an Agent Communication Language - which is driven by a set of tendencies (in the form of individual objective or of a satisfaction/survival function which it tries to optimise) - which possesses resources of its own - which is capable of perceiving its environment (but to a limited extent) - which has only a partial representation of this environment (and perhaps none at all) - which possesses skills and can offer services - which may be able to reproduce itself - whose behaviour tends towards satisfying its objective, taking account of the resources and skills available to it and depending of its perception, its representation and the communication it receives.	A.ID., R.Id., T.Sp., C.O.D., R.D., (M) A.S.D, (M) A.B.D	Identified (Name and Requirement) in A.ID., Structure (tasks) introduced in T.Sp. and sometimes in (M).A.S.D. Behavior (and control flow) in (M)ABD Communications in COD (most of them deduced from R.Id.)
Role	A portion of the social behavior of an agent that is characterized by a goal and/or provides a functionality/service.	R.Id., R.D., C.O.D.	Identified in R.Id. Described in R.D. Shown in COD
Task	A task specifies the computation that generates the effects of the behavioural feature. Its granularity address the significance of atomic part of the overall agent behaviour.	T.Sp., C.O.D., R.D., (M)ASD, (M)ABD	Identified in T.Sp. and sometimes in MASD Used in COD, RD, MABD
Requirement	In this document we will only refer to functional requirements. A requirement represents a feature that the system to be must exhibit, it can be a functional requirement such as service or a non-functional requirement such as a constraint on the system (or a specific part of it) performance.	D.D., A.Id.	Introduced in DD. Reported in AID and assigned to agents
Goal	A goal is a set of states of the world that an agent is committed to achieve/maintain. Therefore a goal is a situation, but not all situations are goals. A set of states of the world is generally not a goal unless there is an agent committed to achieve/maintain this set of states.	Not explicitly designed	
Scenario	A concrete, informal description of a single feature of the system.	R.Id.	from textual scenarios
Communication	An interaction among two agents, referring an Agent Interaction Protocol and a piece of the domain ontology (knowledge exchanged during the interaction). Usually it is composed of more	R.Id., C.O.D., R.D.	Identified in R.Id. (sometimes also in C.O.D.) Described (details) in

	than one messages, each one associated with one performative.		C.O.D. Reported in R.D.
Message	An individual unit of communication between two or more agents. A message corresponds to a communicative act, in the sense that a message encodes the communicative act for reliable transmission between agents. Note that communicative acts can be recursively composed, so while the outermost act is directly encoded by the message, taken as a whole a given message may represent multiple individual communicative acts.	(M)A.B.D.	Messages are described in MABD with the syntax: <i>message (AIP, ontology, performative)</i> or the simplified one: <i>message (ontology, performative)</i>
Agent Interaction Protocol	A common pattern of conversations used to perform some generally useful task. The protocol is often used to facilitate a simplification of the computational machinery needed to support a given dialogue task between two agents. Throughout this document, we reserve protocol to refer to dialogue patterns between agents, and networking protocol to refer to underlying transport mechanisms such as TCP/IP.	C.O.D., P.D., (M)A.B.D.	Assigned to communications in COD, Non FIPA standard protocols are designed in P.D. Reported in (M)ABD
Performative	It represents a verb that describes the action associated to a content sent to recipients. It carries the meaning of the message and what is the intention of the sender.	P.D., (M)ABD	Non FIPA standard protocols are designed in P.D. Performatives are assigned to messages in (M)ABD
Ontology, concept, action, predicate	An ontology is an explicit specification of the structure of a certain domain (e.g. e-commerce, sport, ...). For the practical goals of FIPA (that is enabling development and deployment of inter-operable agent-based applications), this includes a vocabulary (i.e. a list of logical constants and predicate symbols) for referring to the subject area, and a set of logical statements expressing the constraints existing in the domain and restricting the interpretation of the vocabulary. Ontologies therefore provide a vocabulary for representing and communicating knowledge about some topic and a set of relationships and properties that hold for the entities denoted by that vocabulary.	D.O.D., C.O.D., (M)A.B.D.	Designed in D.O.D., Referred in COD and (M)ABD
FIPA-Platform Agent	The software implementation of the Agent in the selected (FIPA) platform	(S) A.S.D, (M) A.B.D, (S) A.B.D, C.R., C.C., D.C.	Listed in (M)ABD (list of Agents = list of FIPA-Platform Agents) Structure described in (S)ASD, Behaviour in (M)ABD and (at method level) in (S) A.B.D Pattern reused in C.R. Code completed in C.C. Deployment configuration designed in D.C.
FIPA-Platform	The software implementation of the Task in the	(M) A.S.D,	Listed in (M) A.S.D

Task	selected (FIPA) platform	(S) A.S.D, (M) A.B.D, (S) A.B.D, C.R., C.C.	(as generic tasks) Designed in (S)ASD. Behavior (high level) designed in (M)ABD, detailed in (S) ABD. Pattern applied in CR Code completed in C.C.
Service	A service is a single, coherent block of activity in which an agent will engage. A set of services is associated with each agent role.	R.D.	Introduced in R.D.
Resource	An entity that can be acquired (also using sensors), shared, or produced by agents.	R.D.	Introduced in R.D.