

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Methodology: Glossary

Rel. 1.0

Document title	FIPA Methodology: Glossary of Terms		
Document number	TBA	Document source	FIPA TC Methodology
Document status	TC Internal	Date of this status	2003/11/18
Supersedes			
Contact	cossentino@pa.icar.cnr.it		
Change history			
2003/11/18	Approved		

© 2000 Foundation for Intelligent Physical Agents - <http://www.fipa.org/>

Geneva, Switzerland

Notice

Use of the technologies described in this specification may infringe patents, copyrights or other intellectual property rights of FIPA Members and non-members. Nothing in this specification should be construed as granting permission to use any of the technologies described. Anyone planning to make use of technology covered by the intellectual property rights of others should first obtain permission from the holder(s) of the rights. FIPA strongly encourages anyone implementing any part of this specification to determine first whether part(s) sought to be implemented are covered by the intellectual property of others, and, if so, to obtain appropriate licenses or other permission from the holder(s) of such intellectual property prior to implementation. This specification is subject to change without notice. Neither FIPA nor any of its Members accept any responsibility whatsoever for damages or liability, direct or consequential, which may result from the use of this specification.

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Procedures for Technical Work. A complete overview of the FIPA specifications and their current status may be found in the FIPA List of Specifications. A list of terms and abbreviations used in the FIPA specifications may be found in the FIPA Glossary.

FIPA is a non-profit association registered in Geneva, Switzerland. As of January 2000, the 56 members of FIPA represented 17 countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found at <http://www.fipa.org/>.

Index

1	Scope	1
2	List of terms	2
3	List of recommended terms.....	6

1 Scope

This document contains a glossary of (development) methodology and agent-oriented systems related terms. Its aim is to create a common reference base for researchers and developers designing MAS with the meta-methodology approach. It is one of the outcomes of the FIPA Methodology TC and will be referred in the activity of identifying an unifying approach for the development of FIPA-based systems.

A section dealing with common software engineering and agent systems terms (section 3) has been included in order to report the specific meaning that the specific term assumes throughout all the FIPA Methodology TC work. Terms definitions have been taken from FIPA existing specifications, literature, or their most diffused and agreed interpretation.

2 List of terms

Term	Definition	Sources
Action	A fundamental unit of behavior specification that represents some activity which an agent may perform. A special class of actions is the communicative acts.	FIPA97 specs, UML 2 specification version 03-04-01
Adaptive multi-agent system (AMAS)	An adaptive multi-agent system is a multi-agent system which is able to change its behaviour while running. It does this to adjust its behaviour to its dynamic environment in order to achieve the task it is designed for or to improve its function or its performances. Such a system is characterised by the following points: (1) it is plunged into an environment, (2) it has a function to achieve and (3) it is composed of interacting agents. The AMAS theory says that for all functionally adequate system (realising the desired function) there is at least a system having a co-operative internal medium which realises an equivalent function. In other words, to design a system realising the desired function, having a system formed by co-operative agents is sufficient; this co-operation directs the social attitude of these agents.	
Adaptive system	An adaptive system is a system that is able to evolve and adapt its behavior when modifications occur in the environment or in the goals defined for this system.	
Agent Characteristics	A characteristic is an intrinsic or physical property of an agent.	
Agent Interaction Protocol (AIP)	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.	
Autonomous agent	An agent which has the autonomy property (see Autonomy).	
Autonomy	The autonomy of an agent can be expressed as following: 1) An agent has its own life, independently of the existence of other agents, 2) An agent is able to survive in dynamic environments without an external control, 3) An agent takes internal decisions about its behaviour only considering the perceptions, knowledge and representations it possesses.	
Behaviour	A behavior is the observable effects of an operation or an event, including its results. It specifies the computation that generates the effects of the behavioral feature. See also Task	
Collaboration	Collaboration is concerned with the interactions between agents in a multiagent system when the whole system is also considered as an agent with certain structure of system's global state. Particularly, it is concerned with the relationships between individual agents' mental structures and internal states and the system's collective mental structure and state. For example, a collaborative model of multiagent systems may contain a model of system's global intention and individual agent's intention, and we can talk about congruence (that is the consistency between	

Term	Definition	Sources
	an agent's behaviour and the whole system's global goal or intention) and coherence (that is the consistency between an agent's internal state, such as intention, and the system's goal or intention).	
Communicative act	A special class of actions that correspond to the basic building blocks of dialogue between agents. A communicative act (CA) has a well-defined, declarative meaning independent of the content of any given act. CAs are modelled on speech act theory. Pragmatically, CAs are performed by an agent sending a message to another agent, using the standard FIPA message format.	FIPA 97 specs
Content language	The content of a FIPA message refers to whatever the communicative act applies to. If, in general terms, the communicative act is considered as a sentence, the content is the grammatical object of the sentence. This content can be encoded in any language, the content language, denoted by the language parameter of the communicative act.	FIPA 97 specs
Conversation	An ongoing sequence of communicative acts exchanged between two (or more) agents relating to some ongoing topic of discourse. A conversation may (perhaps implicitly) accumulate context that is used to determine the meaning of later messages in the conversation.	FIPA 97 specs
Cooperation	Cooperation is concerned with those aspects of the interactions between agents that are related to the internal structures and mechanisms of individual agents and describes how an agent participate in the interactions. For example, in a cooperative model of multiagent systems, we would expect that the model of agents' mental state of willingness allows agents to participate in an interaction with other agents. Studying cooperation, agents are, essentially, considered as white-boxes.	For the Zhu's definition: http://jamesodell.com/AAMAS-2002-Parunak.pdf
Cooperation failure	A cooperation failure corresponds to the detection of a Non Cooperative Situation. Such a failure can be viewed as a cooperation protocol which is not obeyed or "bad" (wrong) interactions that may occur between the system and its environment.	
Cooperative Agent	When working with co-operative agents to build Adaptive Multi-Agent Systems, the autonomy is the main property of an agent. An agent is able to decide of its own behaviour. Furthermore, agents in AMAS have a special social attitude: they must be cooperative. So, an agent must detect and process Non Cooperative Situations to always act to come back in a state it judges being cooperative from its own point of view. For instance, an agent that does not possess an information requested by another agent will do all it can to find another agent able to answer this request.	
Coordination	Coordination is concerned with the interactions between agents that can be modelled, described and observed without refer to the internal structures of the agents. It is about the information flows between the agents and the temporal ordering of actions and events, in particular, the synchronisation issues, related to the behaviours of agents in a multi-agent system. Studying coordination, agents are essentially considered as black-boxes.	http://jamesodell.com/AAMAS-2002-Parunak.pdf

Term	Definition	Sources
Emergence	A computer system aims at realising a function which must be adequate with what we can expect of it. This function may evolve with time and we want it to emerge. This function is emerging if the system coding does not depend on the knowledge of the function. This coding must contain mechanisms enabling the system adaptation during its exchanges with the environment in order to always lead to the adequate function. In the AMAS theory, changing the function is realised by changing the organisation of the components of the system. The mechanisms are specified by rules governing the self-organisation between components. These rules do not depend on the knowledge of the collective function.	
Environment	The environment of an agent refers to all that is external to the agent, this includes the totality of circumstances surrounding an agent or group of agents. We can consider the physical and social environment. Physical Environment: The combination of external physical conditions that affect and influence the growth, development, actions and survival of agents. Social Environment: The complex of social and cultural conditions affecting the nature of an agent or a community.	[Lind 01] - Jürgen LIND - Iterative Software Engineering for Multiagent Systems - Volume 1994 of Lecture Notes in Artificial Intelligence, Springer Verlag, Heidelberg, 2001. [Russell 95] - S. RUSSEL & P. NORVIG - Artificial Intelligence: a Modern Approach - Prentice-Hall. [Woolridge 00] - Michael WOOLRIDGE - On the Sources of Complexity in Agent Design - In Applied Artificial Intelligence. 14(7):623-644. 2000.
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.	FIPA 97 specs
Method	A repeatable technique for solving a specific problem	B. Bruegge and A.H. Dutoit. Object-Oriented Software Engineering. Prentice Hall eds. 2000
Method Engineering	It is a discipline where methods themselves are the engineering target and its aim is to design, construct and adapt methods, technique and tools for the development of software systems. It is based on reuse and integration of contributions coming from existing methodologies in order to obtain the best process for own specific problem.	Brinkkemper
Method fragment	See Method Fragment Definition document produce by the Methodology TC	
Non cooperative situations	When the environment is unpredictable, or when the system is open, classical algorithms fail because the designer is unable to find an algorithm that is able to list all the existing possibilities. The aim of the AMAS technology is to design systems that do their best when a difficulty is encountered. In classical programs, these unexpected events can be processed as exceptions. In the	

Term	Definition	Sources
	AMAS theory context, these "exceptions" - expressing unusual situations that an agent may be faced with - are called "Non Cooperative Situations" (NCS). Different kinds of NCS exist, such as: - Incomprehension, - Ambiguity, - Incompetence, - Concurrency, - Conflict, - Uselessness.	
Performative	See Communicative Act	
Task	Often used as synonymous of Behaviour (see Behaviour) but with the significance of atomic part of the overall agent behaviour.	
Work definition	A WorkDefinition describes the work performed in the process. Instances of the WorkDefinition class can be created to represent composite pieces of work that are further decomposed. It has explicit inputs and outputs.	SPEM
Work product	A work product or artifact is anything produced, consumed, or modified by a process.	SPEM

3 List of recommended terms

Term	Definition	Sources
Agent (FIPA 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.	FIPA97 specs,FIPA abstract architecture document (http://www.fipa.org/specs/fipa00003/OC00003.pdf) FIPA Abstract Arch. Spec. : http://www.fipa.org/specs/fipa00001 [Ferber 99] - Jacques FERBER - Multi-Agent System: An Introduction to Distributed Artificial Intelligence - Addison Wesley Longman, 1999.
Analysis	Analysis also called requirements analysis or requirements engineering. The term has been invented to cover all of the activities involved in discovering, documenting and maintaining a set of requirements for a computer-based system. It corresponds to the very first phase in life cycle.	based on Sommerville and Sawyer definition
AUML	Agent UML (AUML) is a graphical language based on the modeling language UML. Its aim is to provide a notation for the design of multiagent systems. It takes the elements (diagrams, stereotypes) from UML and add new diagrams and stereotypes to cope agent characteristics such as autonomy.	
Design	Design is both the process of defining the architecture, components, interfaces, and other characteristics of a system or component and the result of that process The software design objectives: a) to produce various models that can be analyzed and evaluated to determine if they will allow the various requirements to be fulfilled, b) to examine and evaluate various alternative solutions and trade-offs, and c) to plan the subsequent development activities.	IEEE specs, http://cs.wvc.edu/~aabyan/435/
Event	The specification of a signification occurrence that has a location in time and space and can cause the execution of an associated behavior.	UML 2 specification version 03-04-01
Functional adequacy	A system is functionally adequate when it realises the right function, the function for which it has been designed.	
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.	[Eurescom 00] Eurescom - Project P907-GI - MESSAGE: Methodology for Engineering Systems of Software Agents, Deliverable 1- Initial Methodology - 2000. http://www.eurescom.de/~pub-deliverables/P900-series/P907/D1/P907D1
Implementation	the phase of the software life cycle where code is written. Platform independent models from the design phase are now	

Term	Definition	Sources
	instantiated for specific platforms.	
Interaction	A specification of how stimuli are sent between instances to perform a specific task. The interaction is defined in the context of collaboration.	UML 2 specification version 03-04-01
Interaction language	The interaction language is a set of tools needed by the agent to directly or indirectly communicate towards other agents or towards its environment.	
Methodology	A collection of methods for solving a class of problems	B. Bruegge and A.H. Dutoit. Object-Oriented Software Engineering. Prentice Hall eds. 2000
Multi-Agent System	A Multi-Agent System is a system composed of a great number of autonomous entities, named agents, having a collective behaviour that allows to obtain the desired function/service.	FOLDOC (Free On-Line Dictionary Of Computing, http://wombat.doc.ic.ac.uk/fol/doc/)
Notation	A graphical or textual set of rules for representing a model	1) Webster dictionary 2) B. Bruegge and A.H. Dutoit. Object-Oriented Software Engineering. Prentice Hall eds. 2000
Ontology	A set of symbols together with an associated interpretation that may be shared by a community of agents or software. An ontology includes a vocabulary of symbols referring to objects in the subject domain, as well as symbols referring to relationships that may be evident in the domain.	FIPA97 specs, FIPA abstract architecture document (http://www.fipa.org/specs/fipa00003/OC00003.pdf)
Open systems	An open system is a multiagent system where agents can enter and leave at any moment during the execution.	
Openess	openess is a feature of an open system	
Perception	Perception is a mean to receive information from the physical or social (other agents) environment. Thus, the designer has to give some perceptive capabilities to the agent.	
Process	A set of activities and associated results that is performed towards a specific purpose	UML 2 specification version 03-04-01, OMG SPEM (Software Process Engineering Metamodel) Specification
Requirement (functional and non functional)	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.	B. Bruegge and A.H. Dutoit. Object-Oriented Software Engineering. Prentice Hall eds. 2000
Role	A portion of the social behavior of an agent that is characterized by some specificity such as a goal, a set of attributes (for example responsibilities, permissions, activities, and protocols) or providing a functionality/service.	[Woold00]: M. Wooldrige, N. R. Jennings and D. Kinny, The Gaia Methodology for Agent-Oriented Analysis and Design, Autonomous Agents and Multi-Agent Systems, volume 3, pp 285-312, Kluwer Academic Publishers, The Netherlands, 2000.
Self-organisation	System that adapts itself using self-organisation. The function of the system emerges, at the global level, from the interaction between its components at the local level. A cooperative agent,	[Heylighen 01] - F. HEYLIGHEN - The Science of Self-organization and Adaptivity - In

Term	Definition	Sources
	in our case, does not know this global function and only knows its own goal.	The Encyclopedia of Life Support Systems, (EOLSS Publishers Co. Ltd), 2001.
Self-organising systems	System that adapts itself using self-organisation. The function of the system emerges, at the global level, from the interaction between its components at the local level.	
Service (Software service)	A service is a single, coherent block of activity in which an agent will engage. A set of services can be associated with each agent role. For each service that may be performed by an agent, it is necessary to specify its properties. Specifically, we must identify the inputs, outputs, pre-conditions, and post-conditions of each service.	
Speech act	The notion of a speech act is derived from the linguistic analysis of human communication. It is based on the idea that with language the speaker not only makes statements, but also performs actions, e.g. a request or an assertion. In this context, a verb denoting a speech act, is called a performative, since saying it makes it so.	FIPA 97 specs
Speech Act theory	A theory of communications which is used as the basis for ACL. Speech act theory is derived from the linguistic analysis of human communication. It is based on the idea that with language the speaker not only makes statements, but also performs actions. A speech act can be put in a stylised form that begins "I hereby request ..." or "I hereby declare ...". In this form the verb is called the performative, since saying it makes it so. Verbs that cannot be put into this form are not speech acts, for example "I hereby solve this equation" does not actually solve the equation.	
Testing	Testing finds cases where a program does not meet its specification (given specifications F and program P, find as many of (f1, f2, ...,fp) of F not satisfied by P).	[Hailpern] B. Hailpern, P. Santhanam. Software Debugging, testing and verification. IBM Systems Journal. Vol 41, 1, 2002.
Validation	The process of evaluation of software at the end of the software development process to ensure compliance with requirements	[Hailpern] B. Hailpern, P. Santhanam. Software Debugging, testing and verification. IBM Systems Journal. Vol 41, 1, 2002.
Verification	Given a program and a set of specifications, verification shows that the program satisfies those specifications.	[Hailpern] B. Hailpern, P. Santhanam. Software Debugging, testing and verification. IBM Systems Journal. Vol 41, 1, 2002.