The JADE Semantic Agent:

Towards Agent Communication Oriented Midleware

Vincent Louis & Thierry Martinez, France Telecom Research & Development, France, {vincent.louis; thierry.martinez}@francetelecom.com

Recently, much effort has been expended towards making agents interoperate in open environments. Despite this effort, this goal has still not yet been achieved. As an illustration, the Agentcities European project [1] that resulted in the deployment of a worldwide open testbed environment, underlined the lack of "spontaneous" exchanges between agents running in this environment. In almost all cases, agents can only interact with agents they have been designed to interact with. One reason is that implementing agents with finite state machines in order to conform to a limited set of interaction protocols generally results in rigid agents. In fact, they cannot handle messages that are not specified by the protocol, even if they were able to process these messages according to their capabilities and the meaning of these messages. On the contrary, an agent which could interpret the sense of messages, as defined by its semantics rather than its syntax, could intrinsically support more flexible interaction.

The FIPA standard

The Foundation for Intelligent Physical Agents (FIPA) [2] has defined a standard agent communication language, namely FIPA-ACL. This language has the advantage that it relies not only on a syntactic definition, but also on a semantic one. In other words, the FIPA-ACL standard formally specifies a precise meaning for each communication primitive in the language. Agents should benefit from this meaning to soundly understand messages and react properly. For instance, a *Request* message from an agent means s/he intends the receiver to perform some action. Similarly, an *Inform* message about her/his intention to perform some action has the same meaning. Therefore it should be interpreted in the same way as the receiving agent.

To date, numerous platforms have implemented the FIPA specifications, the most popular ones being FIPA-OS, Zeus and JADE. Even if they claim that they comply with the standard, unfortunately none of them provides the proper support to handle the semantic dimension of the FIPA-ACL language. At best, they provide support for handling the interaction protocols specified by FIPA. However, this set of protocols only provides a limited account for the semantics of FIPA-ACL. For example, no protocol currently handles the above case of an *Inform* message about an agent's intention to perform an action.

The JADE Semantic Agent framework

In order to fill this gap, France Telecom Research & Development has developed the JADE Semantic Agent framework (JSA) [3] upon the JADE platform [4]. This framework facilitates the programming of agents which naturally conform to the semantics of FIPA-ACL. Not only do the developed agents automatically interpret the incoming messages according to their formal meaning but they also automatically send proper messages in response to the interpretation.

To make these ideas more concrete, let us consider the example illustrated by Figure 1. In this scenario, the "sensor" agent provides some information about the temperature, the "display" agent displays the temperature, the "son" agent adapts its clothing depending on the temperature, and the "mother" and "daughter" agents interact with these agents, for example to get the temperature value or to request the son to change his clothing, etc. More precisely, the display sends a subscribe message (callout #1) meaning it wants to be informed of temperature changes. Whilst monitoring the temperature, the sensor additionally sends an inform message (#2) to the display each time the temperature changes. Subsequently, as the display knows the temperature, it can in turn inform other agents about the temperature values. For instance, it can answer queries from the son about whether the temperature is greater than a given value (#3a) as well as subscription requests (#3b). The son then puts on or takes off clothing items with respect to the received information (#4): for example, if he becomes aware of a temperature greater than 0°C, he takes off his cap. Additionally, he also puts on or takes off clothing items when the mother asks him to do so, either by requesting him to perform such an action (#5a) or by informing him that she intends him to wear a given piece of clothing (#5b). In the first case, he replies to the mother using an inform message (#6) stating that the requested action has been done. At the contrary, if the same kind of request comes from the daughter (#7), the son refuses to perform the requested action and answers using a *Refuse* message (#8).

In this scenario, using the JSA framework to implement the agents results in automatic handling of all the described interactions. Moreover, many other interactions between agents are naturally handled without the need for any additional code. For example, the son can directly subscribe to the sensor, the mother can request the son to query the display or query him about his clothing, etc. Consequently, this framework makes it possible to implement with little additional effort, much more flexible agents than those usually implemented on FIPA compliant platforms.

The JSA architecture

The main activity of a JADE Semantic Agent consists of interpreting the received FIPA-ACL messages. This interpretation activity can be refined into two main functions (see Figure 2): the first one produces some sense about the input message, while the second one consumes this sense and updates the agent's respective activities

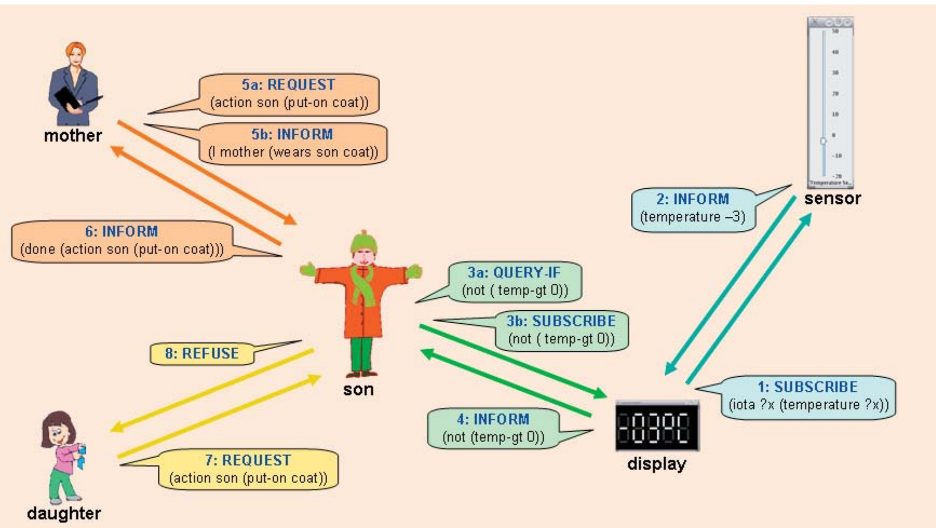

Figure 1: A scenario illustrating the interaction of several agents using FIPA-ACL.

and beliefs. Beliefs represent the state of the world perceived by the agent. These are expressed using logical formulae. Activities consist of performing some arbitrarily complex courses of action. These actions include the communicative actions defined by FIPA-ACL, such as inform or request, as well as domain-specific actions.

One of the core concepts used by the interpretation activity is the *Semantic Representation* (SR), which represents a produced or consumed sense. Actually, this concept is implemented as a formula expressed in the same logical language as the one used to formally specify the FIPA-ACL semantics. For instance, the following SR states that the display agent believes that the sensor intends him to believe the current temperature value is -3:

(B display (I sensor (B display (temperature -3))))

This SR is generated by the production function of the display's interpretation activity when s/he receives an inform message from the sensor stating that the temperature is -3°C. Afterwards, the consumption function may add the following belief to the display's belief base:

(B display (temperature -3))

The other core concept for interpreting incoming messages is the *Semantic Interpretation Principle* (SIP). The production and consumption functions are implemented by a set of SIPs that reify the theoretical principles underlying the semantics of the FIPA-ACL language. The application of each SIP also results in querying or updating the agent's belief base and adding to, or removing activities from the agent. Possibly, a SIP may dynamically alter the interpretation activity itself by updating the set of SIPs. In the previous example, both SRs are produced by standard SIPs (see below), some of them being customizable. In particular, it is possible to specify the beliefs or intentions that can be adopted by an agent depending on various criteria. For instance, in the scenario described above, the son obeys his mother whereas he rejects the requests from his sister. This kind of customization makes it possible to accurately set up the degree of cooperation the agents develop using the JSA framework.

Standard SIPs for FIPA-ACL

As shown in Figure 2, the JSA framework provides a set of standard SIPs implementing the formal semantics of the FIPA-ACL communicative actions.

First of all, the "ActionFeature" SIP produces several SRs from a received message that represent the semantic features of the corresponding communicative action. The FIPA-ACL standard defines two semantic features for each communicative action: the feasibility precondition and the rational effect. The former mainly gives rise to a SR stating the precondition was necessarily true before the communicative action was performed. The latter gives rise to an SR (usually called "intentional effect") stating the sending agent intends the rational effect to become true. For example, the SR #1 in Figure 3 represents the intentional effect of an inform action, the two middle lines denoting the corresponding rational effect. The SRs resulting from the application of the "ActionFeature" SIP convey the primary meaning of received communicative actions. These SRs feed the other SIPs, leading to the agent's reaction.

The "BeliefTransfer" SIP manages the adoption of beliefs suggested by other agents. It applies to any SR stating an external agent intends the interpreting agent to believe a fact, and produces a new SR stating the interpreting agent actually believes this fact. This application may be customized to specify which facts are believable with respect to the external agent and the fact itself. For example, the SR #2 in Figure 3 results from applying the "BeliefTransfer" SIP to the SR #1. It states the display now believes the son's intention, which was originally conveyed by the inform message. Similarly, the "IntentionTransfer" SIP manages the adoption of other agents' intentions. It applies to any SR stating an external agent has an intention, and produces a new SR stating the interpreting agent actually has the same intention. This application may be customized to specify the expected cooperative attitude for the interpreting agent in terms of the intentions to adopt and the external agents to cooperate with. For example, the SR #3 in Figure 3 results from applying the "IntentionTransfer" SIP to SR #2. It states that the display now has the intention that the son knows the temperature value.

Finally, the last three SIPs manage the planning capabilities of the interpreting agent by creating proper activities in order to satisfy her/his intentions. The "ActionPerformance" SIP tries to directly perform an intended action (for example, see step #4 in Figure 3). The "RationalityPrinciple" SIP searches the agent's base of actions for an action whose rational effect matches a given intention (for example, see step #5 in Figure 3, where the rational effect of the selected inform-ref communicative action matches the intention conveyed by the SR #3). Lastly, the "Planning" SIP uses an external planner to find an (arbitrarily complex) action plan whose performance brings about the input intention.

This set of SIPs, in addition to the provided "Subscribe" SIP, which is not described here, makes it possible to handle complex and flexible interactions, such as those for the scenario introduced above. For instance, Figure 3 shows how the interpretation of two different formulations for the same request actually results in the same reaction. Act1 is an inform message telling the display that the son intends it to know the temperature value, whereas Act2 is a query-ref message that directly requests the display to tell the son the temperature value.

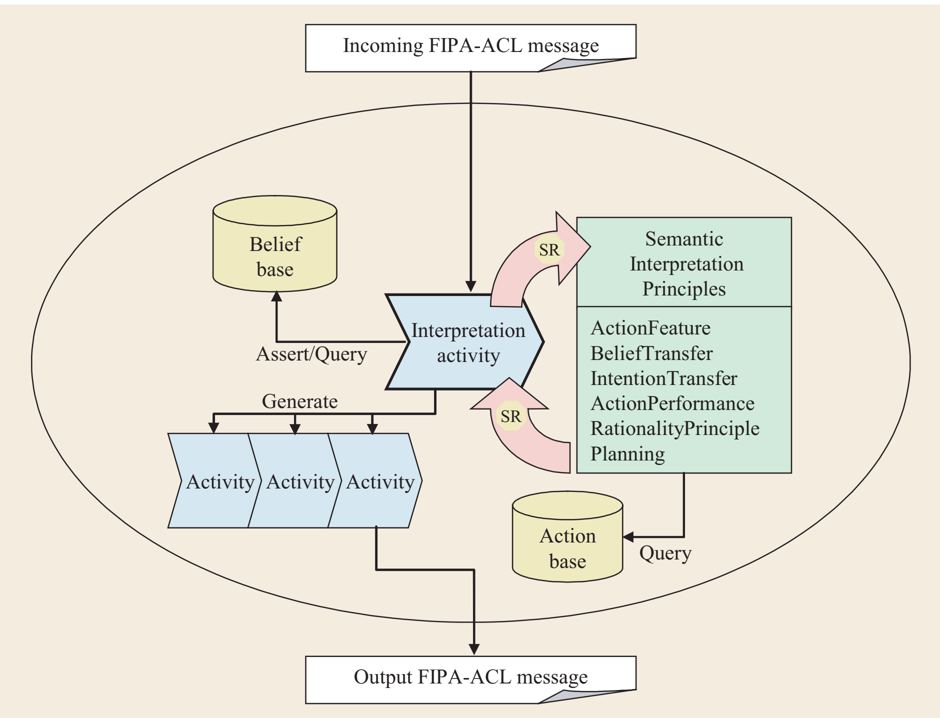

Figure 2: Interpreting FIPA-ACL messages.

Implementing JADE Semantic Agents

Implementing a JADE Semantic Agent is quite simple. It actually consists of implementing the cooperative and domain-specific agent features rather than analyzing in detail and coding all possible messages and interaction protocols. More precisely, the agent programmer has to carry out three main tasks. The first task consists of implementing the domain-specific actions – these are part of the JSA framework. In the scenario of Figure 1, the only two domain-specific actions that have been implemented are "put-on" and "take-off". The second task consists in coding the agent's belief base management to handle domain-specific facts. In the scenario, such a code ensures, for example, that if the agent believes the temperature is 20°C, then s/he also believes it is greater than 10°C. It also ensures that the agent always believes the temperature has one single value at the same time. Finally, the last task consists of customizing the cooperation principles of the agent, including the "BeliefTransfer" and "IntentionTransfer" SIPs. In the scenario, the son's "IntentionTransfer" SIP has been customized such that he only obeys requests from the mother. That is why he refuses to put on his coat when asked by the daughter. Interestingly, the son's "BeliefTransfer" SIP is not blocked with respect to the daughter. Consequently, if she tells the son that the mother intends him to put on his coat (instead of directly requesting him to do so), then he will believe that the mother has actually this intention, and will finally put on his coat.

In most cases, a significant part of the code developed to implement a JADE Semantic Agent is domain-specific. Hence, it can be reused by other agents in this domain without any additional development. For example, in the scenario of Figure 1, all the agents share exactly the same code to manage temperature-related facts.

Perspectives

The JADE Semantic Agent framework, developed by France Telecom, is an add-on to the JADE platform that can be downloaded from http://jade.tilab.com/. To our knowledge, it is the first public attempt to implement the semantics of FIPA-ACL. Such an implementation makes it possible to build more flexible and dynamic multiagent systems. Actually, JADE Semantic Agents are able to understand each other without an explicit need for rigid protocols. As a result, we think that this add-on makes the JADE platform take a new step towards a real agent communication oriented middleware. Such a middleware should include both the usual message-oriented functionalities and semantic interpretation capabilities. We envision that these capabilities are essential ingredients to support flexible and unpredictable interactions required by future wide open multiagent systems where each agent will be either software or human.

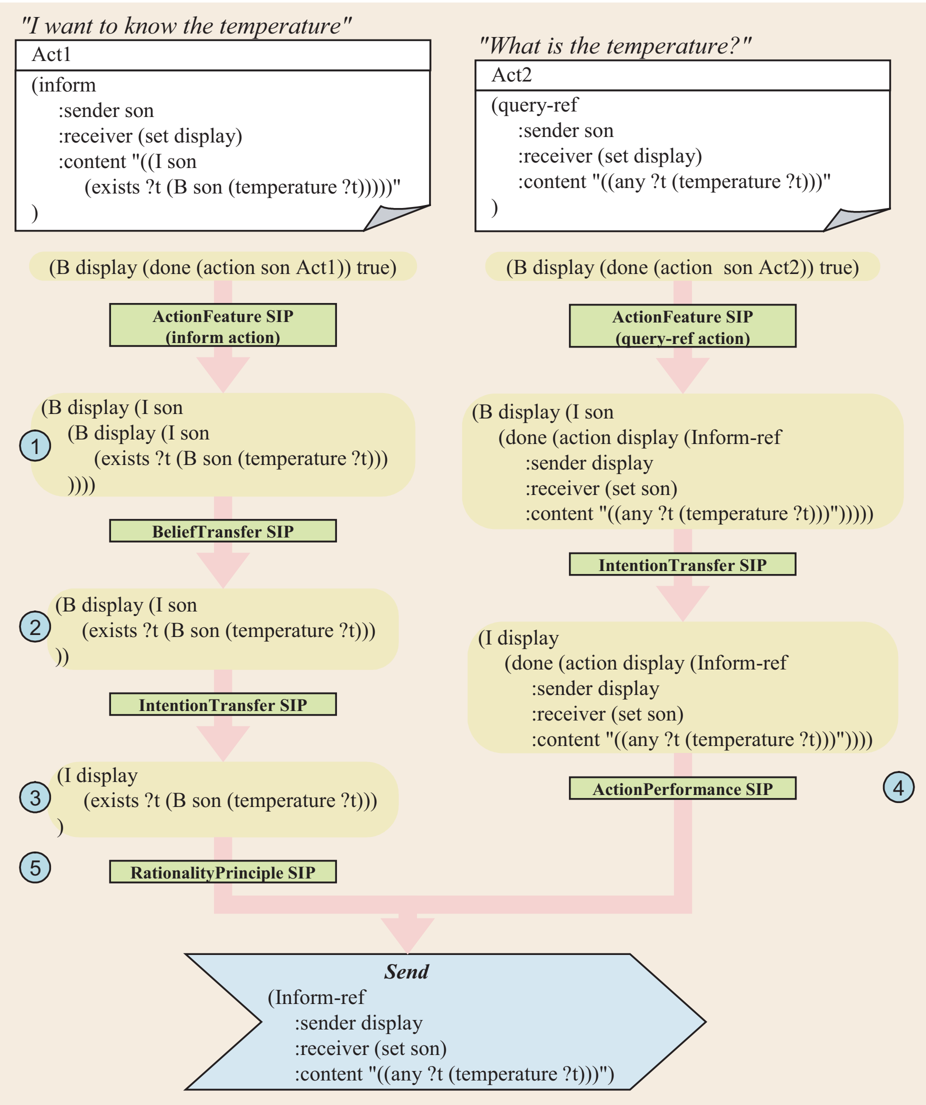

Figure 3: An illustration of how the interpretation of two different formulations for the same request can result in the same message.

References

1. Willmott, S. (ed.): "Technical Input and Feedback to FIPA from Agentcities.RTD and the Agentcities Initiative". Agentcities Task Force Technical Note 00003, http://www.agentcities.org/note/00003/ (2003)
2. Foundation for Physical Intelligent Agents (FIPA), Geneva, Switzerland, http://www.fipa.org
3. Louis V., Martinez T.: "An operational model for the FIPA-ACL semantics". Proceedings of the AAMAS'05 Workshop on Agent Communication (AC2005), Utrech, The Netherlands, 2005.
4. Bellifemine, F., Caire, G., Poggi, A., Rimassa, G.: "JADE A White Paper. exp in search of innovation", Volume 3(3), Telecom Italia Lab (2003) 6–19. See also the Java Agent DEvelopment Framework at http://jade.tilab.com/
5. Sadek D.: "Attitudes mentales et interaction rationnelle : vers une théorie formelle de la communication". PhD Thesis, Rennes I University, France, 1991.

FIPA
Agent Communication Language

FIPA-ACL is the agent communication language standardized by the FIPA foundation since 1997. It defines a set of 4 primitive and 18 composite communicative acts, together with a formal meaning for each of them. This meaning results from the interpretation of the communicative actions as particular actions within the general BDI-style theory of agency proposed by Sadek [5]. More precisely, each communicative action is semantically defined by its feasibility precondition, which states the condition that must necessarily hold for this action to be performed, and its rational effect, which states the result expected by agents performing this action.

For example, the Inform communicative action is used to tell an *agent* a fact. Its precondition states the author believes this fact and also believes the receiver does not already know about it. Its rational effect states that the receiver comes to believe this fact.

Futher details of the complete FIPA-ACL communicative action library can be found at ***http://www.fipa.org/specs/fipa00037/.***