Agent Communication Languages

▼ ACL definition

ACLs are standard formats for agents to exchange knowledge. They are protocols for the exchange of represented knowledge between autonomous information systems.

▼ KQML

It's a Knowledge Query and Manipulation Language. Within them there is something called a Knowledge Interchange Format (KIF).

KQML works as the envelope of a message. It holds the message inside of it. KQML has a set of performatives and a set of parameters. These are like the class of the object. In the same way as every object class has a different purpose or function, each performative in KQML is going to perform something different. A performative is a type of verb or action, which is done by simply saying it. For example, imagine promising doing something makes that something being done. The parameters hold all the information or member data. They are essentially the member variables of the class.

KIF represents the actual message content and domain knowledge. It can be used to write down ontologies. It is based on first-order logic (and, or, not...).

▼ Example

This is a message representing wanting to ask one question and getting one answer.

```
(ask-one
        :content (PRICE IBM?price)
        :receiver stock-server
        :language LPROLOG
        :ontology NYSE-TICKS
)
```

It's parameters would be content, receiver, language and ontology. These are written in KIF. We are sending a message to a stock-server, saying what language it's in (therefore assuming it knows Prolog) and saying we are using the ontology NYSE-TICKS. We are asking for IBM's price, and so we need to specify an identifier for it in that particular server (it could be different for each server).

▼ Examples of KQML parameters

- **:content** content of the message in KIF

- **:force** wether the sender will ever deny the content of the message

- **:reply-with** is a reply expected? If so, an identifier for the reply

- **:in-reply-to** reference to :reply-with

- **:sender** the sender

- **:receiver** the intended recipient

▼ Problems with KQML

- The basic KQML performative set was fluid, never tightly constrained, making it quite large and difficult to use

- It has interoperability problems between different versions, languages...

- The transport mechanisms for KQML messages were not precisely defined

- Semantics of KQML were not rigorously defined, their meaning was instead defined with informal descriptions

- There were no commissive performatives in KQML, so agents can't coordinate their behaviour without the ability to commit to a task

- The performative set was arguably ad-hoc and overly large

Hence, KQML was superseded (replaced) by FIPA.

▼ FIPA

Foundation for Intelligent Physical Agents. It superseded KQML.

- Basic structure similar to KQML

- Smaller performative set (20)

- Formal semantics defined in SL(which draw heavily from Speech Acts)

- It has pre and post conditions, like a feasibility precondition (is it feasible to do something) and a rational effect (what will happen in a rational world, essentially a post condition)

- All performatives are built upon inform and request.

- It is a protocol. Provides a language for writing messages down.

- It says nothing about how messages are passed between agents

- Several software platforms have been developed to support FIPA-ACL-based communication

- One of the most widely used is JADE

- Provides transparent transport of ACL messages (from the perspective of the agent designer)

performative	passing info	requesting info	negotiation	performing actions	error handling
accept-proposal			x		
agree				x	
cancel		x		x	
cfp			x		
confirm	x				
disconfirm	x				
failure					x
inform	x				
inform-if	x				
inform-ref	x				
not-understood					x
propose			x		
query-if		x			
query-ref		x			
refuse				x	
reject-proposal			x		
request				x	
request-when				x	
request-whenever				x	
subscribe		x			

▼ Notation

Agent "i" believes "psi"

$$B_i\psi$$

Agent "i" believes either "psi" is true or "psi" is false (the agent has some sort of certainty about "psi"

$$Bif_i\psi$$

Agent "i" is uncertain of "psi" (it believes "psi" is more likely than "not psi"

$$U_i\psi$$

Agent "i" is uncertain of "psi" and certain of "not psi" (they think either "psi" is more likely or "not psi" is more likely. They know something but are not for telling).

$$Uif_i\psi$$

▼ Examples

▼ The "inform" performative

Agent i wishes to inform agent j of statement psi.

It requires agent i believing in statement psi, and that agent j has some knowledge of the truth of statement psi.

The rational effect is that agent j will believe statement psi.

$\langle i, inform(j, \phi) \rangle$
feasibility precondition: $B_i\phi \wedge \neg B_i(Bif_j\phi \vee Uif_j\phi)$
rational effect: $B_j\phi$

▼ The "request" performative

Agent i requests that agent j does an action alpha.

First, agent i has to believe that agent j can actually do the action alpha. Also, agent i needs to believe agent j does not intend to do alpha already.

The rational effect will be that the action alpha will get done.

$\langle i, request(j, \alpha) \rangle$
feasibility precondition: $B_i Agent(\alpha, j) \wedge \neg B_i I_j Done(\alpha)$
rational effect: $Done(\alpha)$