Multi-agent Semantic Web Systems: Agent Communication

Michael Rovatsos

School of Informatics

15 March 2012
1 Overview

2 Speech Acts

3 KQML/KIF

4 FIPA

5 Agent Protocols

6 Conclusion
What are Agents?

- Situated in an environment;
- can sense the environment, and
- perform actions.
What are Agents?

- ‘Intelligent’ characteristics of agents:
  - autonomy
  - reasoning ability
  - learning ability
  - mobility
  - sociability
  - cooperation
  - negotiation

**Autonomy**

- Can initiate tasks without external intervention.
- Behaviour is based on experience as well as in-built knowledge.
Multi-Agents Systems (MAS)

- Distributed system which incorporates independent agents.
- The collection of agents interact, and solve problems that are outside their individual capacities.
- Focus on properties that emerge from cooperation (vs. capabilities of individual agents)
- ‘standard’ distributed system: coordination must be specified in advance
- Multi-Agent system: (some aspects of) coordination achieved dynamically at run-time

Possible advantages:
- fault-tolerance
- scalability
- concurrency
Why communicate?

- **Information Exchange**: share knowledge;
- **Collaboration**: optimising use of resources and distribution of tasks, coordinating execution;
- **Negotiation**: reaching agreement in the presence of conflict
Communication involves sharing at various levels, including:
- how to describe states of affairs (vocabulary, grammar),
- how to carry out speech acts (performative verbs),
- how to engage in dialogue.

Main goal of communication: influence other agents; e.g., to make them perform actions or to make them believe certain propositions.

Other agent decides whether to perform action or believe proposition.
Speech Acts

- Most treatments of communication in (multi-)agent systems borrow their inspiration from speech act theory.
- Speech act theories are pragmatic theories of language, i.e., theories of language use;
- attempt to account for how natural language is used to achieve goals and intentions.
- Original theory by J.L. Austin, *How to do things with words*.

**Speech Act Theory:** Communication as action

**Speech Acts:** Performatives “change the state of the world like a physical action”; everything we utter is uttered with the intention of satisfying some goal or intention
Kinds of Speech Act

- **Locutionary act**: the act of generating sounds that are linked together by grammatical conventions so as to say something meaningful.

- **Illocutionary act**: the speech act of doing something else in the process of uttering meaningful language.
  - *I will gladly pay you £1.00 next week*
  - illocutionary act of making a promise

- **Perlocutionary act**: the contingent effect on those who hear a meaningful utterance.
  - By telling a ghost story late at night, one may accomplish the perlocutionary act of frightening someone.
Performatives

Speech acts often divided into two components:

- performative verb: request, inform, promise, …
- propositional content: the door is closed

\( \phi = \text{the door is closed} \)

Example Performatives

REQUEST + \( \phi \) \( \Rightarrow \) Please close the door!
INFORM + \( \phi \) \( \Rightarrow \) The door is closed.
INQUIRE + \( \phi \) \( \Rightarrow \) Is the door closed?
More Speech Act Theory

Searle (1972) categorization of performatives:

- **Assertives:** informing, claiming something is true
- **Directives:** requesting, commanding
- **Commissives:** promising, refusing
- **Declaratives:** making change to state of world (e.g. marrying)
- **Expressives:** expressing mental states

Like all taxonomies, this is open to debate.
Conditions on Speech Acts

- Austin & Searle described conditions under which speech acts are successful.

- Austin’s felicity conditions:
  - Has to be conventional procedure for carrying out speech act;
  - procedure has to executed correctly;
  - act has to be sincere, uptake must be completed as far as possible.

- Searle:
  - Preparatory conditions must hold (e.g., requested action can be performed, speaker must believe this, hearer not already intending to perform action).
  - Sincerity conditions (speaker wants action to be performed).
DARPA Knowledge Sharing Effort (KSE)


- Project from early ’90s aimed at developing techniques, tools and re-usable resources to support building large-scale knowledge-based systems and knowledge bases.

- Results in the area of knowledge representation:
  - **Syntax**  KIF (propositional content)
  - **Semantics**  Ontolingua (language for defining sharable ontologies)
  - **Pragmatics**  KQML (high-level interaction language)
KIF and KQML are both Agent Communication Languages (ACLs)

**KQML** (Knowledge Query and Manipulation Language). ‘Outer’ language; defines message envelope format.

**KIF** (Knowledge Interchange Format). ‘Inner’ language; defines message content.

- Essentially a LISP-based notation for First Order Logic.
- Intended as interlingua for encoding declarative knowledge.
- Features: semantics independent of implementation; non-decidable, ‘human readable’
- Current specification at [http://logic.stanford.edu/kif](http://logic.stanford.edu/kif)
Possible to express:

- properties of things in a domain (*John has a mobile phone, Mary has short hair*)
- relationships between things in a domain (*Luke is married to Jane*)
- general properties of a domain (*every student has a matriculation number*)
- Standard vocabulary for data structures (numbers, strings etc., complex objects, lists etc.) and methods on them.
KIF Examples

The temperature of m1 is 83 Celsius

\[ (= \text{(temperature m1)} \text{(scalar 83 celsius)}) \]

X is a bachelor iff X is a man and is not married

\[ \text{(defrelation bachelor (?x) :=} \]
\[ \text{(and (man ?x) (not (married ?x)))))} \]

If X is a person then X is a mammal

\[ \text{(defrelation person (?x) :=} \text{(mammal ?x))} \]
Knowledge Query and Manipulation Language (KQML).

High-level, message-oriented language for information exchange.

Makes no assumptions about:

- transport mechanism;
- content language;
- ontology used in message content.

Message types particularly oriented towards multi-agent communication.
Example KQML Message

(tell :sender amazon.com
    :receiver info-agent0011
    :in-reply-to msid-7.24.97.45391
    :ontology http://amazon.com/Books
    :language prolog
    :content price(item567, gbp, 24.95)
)
Example KQML Message

```
(tell    :sender    amazon.com
 :receiver info-agent0011
 :in-reply-to msid=7.24.97.45391
 :ontology http://amazon.com/Books
 :language prolog
 :content  price(item567, gbp, 24.95)
)
```

- performative
Example KQML Message

(tell :sender amazon.com
   :receiver info-agent0011
   :in-reply-to msid-7.24.97.45391
   :ontology http://amazon.com/Books
   :language prolog
   :content price(item567, gbp, 24.95)
)

- performative
- parameter (an attribute/value pair)
Example KQML Message

KQML Message

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>tell</td>
<td></td>
</tr>
<tr>
<td>:sender</td>
<td>amazon.com</td>
</tr>
<tr>
<td>:receiver</td>
<td>info-agent0011</td>
</tr>
<tr>
<td>:in-reply-to</td>
<td>msid-7.24.97.45391</td>
</tr>
<tr>
<td>:ontology</td>
<td><a href="http://amazon.com/Books">http://amazon.com/Books</a></td>
</tr>
<tr>
<td>:language</td>
<td>prolog</td>
</tr>
<tr>
<td>:content</td>
<td>price(item567, gbp, 24.95)</td>
</tr>
</tbody>
</table>

- performative
- parameter (an attribute/value pair)
- declarative message content
### KQML Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>:content</td>
<td>content of message</td>
</tr>
<tr>
<td>:language</td>
<td>formal language of message content</td>
</tr>
<tr>
<td>:ontology</td>
<td>vocabulary employed by message</td>
</tr>
<tr>
<td>:force</td>
<td>will sender ever deny content of message?</td>
</tr>
<tr>
<td>:reply-with &lt;label&gt;</td>
<td>whether the sender expects a reply, and if so, a label for the reply</td>
</tr>
<tr>
<td>:in-reply-to &lt;label&gt;</td>
<td>the expected label in a reply</td>
</tr>
<tr>
<td>:sender</td>
<td>sender</td>
</tr>
<tr>
<td>:receiver</td>
<td>receiver</td>
</tr>
<tr>
<td>Performative</td>
<td>Interpretation</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>ask-if</td>
<td><em>Is it true that …?</em></td>
</tr>
<tr>
<td>achieve</td>
<td><em>Make it true that …</em></td>
</tr>
<tr>
<td>tell</td>
<td><em>It is true that …</em></td>
</tr>
<tr>
<td>reply</td>
<td><em>The answer is …</em></td>
</tr>
<tr>
<td>evaluate</td>
<td><em>Simplify the sentence …</em></td>
</tr>
</tbody>
</table>
Agent A sends the following performative to agent B:

\[
\text{(evaluate :language KIF}
\begin{align*}
&\text{:ontology motors} \\
&\text{:reply-with q1} \\
&\text{:content (val (torque motor1) (sim-time 5)))}
\end{align*}
\]

Agent B replies with:

\[
\text{(reply :language KIF}
\begin{align*}
&\text{:ontology motors} \\
&\text{:in-reply-to q1} \\
&\text{:content (scalar 12 kgf))}
\end{align*}
\]
KQML/KIF Evaluation

- KQML/KIF widely adopted, but
- list of performatives (∼ 41) not fixed, and lacks clear rationale;
- no formal semantics for performatives;
- lacks commissives (e.g., promises).
FIPA ACL (Agent Communication Language): a specification for inter-agent communication via message passing.

- Assumes that agents have Beliefs, Desires and Intentions (BDI model)
- FIPA ACL is similar in design to KQML, but gives an explicit BDI-based semantics to the performatives, using a Semantic language called SL.

Example FIPA ACL Message

```
(inform :sender agent1 :receiver agent2 :ontology hpl-auction :language fipa-sl :content (price item567 24.95))
```
- Basic performatives: `inform`, `request`
- All others (~20) defined in terms of these two.
- Semantics divided up into:
  - feasibility precondition, and
  - rational effect
- Semantics is formalized using following modal operators:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_i \phi$</td>
<td>$i$ believes that $\phi$</td>
</tr>
<tr>
<td>$B if_i \phi$</td>
<td>$B_i \phi \lor B_i \neg \phi$, i.e., $i$ has a definite opinion about the truth of $\phi$</td>
</tr>
<tr>
<td>$I_i \phi$</td>
<td>$i$ has the intention of bringing about $\phi$</td>
</tr>
<tr>
<td>Agent($\alpha$, $i$)</td>
<td>$i$ is capable of carrying out action $\alpha$</td>
</tr>
<tr>
<td>Done($\alpha$)</td>
<td>action $\alpha$ has been carried out</td>
</tr>
<tr>
<td>Uif$_i \phi$</td>
<td>$i$ is uncertain whether or not $\phi$ [ignored here]</td>
</tr>
</tbody>
</table>
\( \langle i, \text{inform}(j, \phi) \rangle \)

<table>
<thead>
<tr>
<th>feasibility precondition</th>
<th>( B_i \phi \land \neg B_i B j \phi )</th>
</tr>
</thead>
<tbody>
<tr>
<td>rational effect:</td>
<td>( B_j \phi )</td>
</tr>
</tbody>
</table>

**inform that** \( \phi \) indicates that the sending agent:

- holds that \( \phi \) is true,
- does not already believe that the receiver \( j \) has any knowledge of the truth of the \( \phi \), and
- intends that \( j \) also comes to believe that \( \phi \) is true.
### Request

<table>
<thead>
<tr>
<th>$\langle i, \text{request}(j, \alpha) \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>feasibility precondition</strong> $B_i Agent(\alpha, j) \land \neg B_i l_j Done(\alpha)$</td>
</tr>
<tr>
<td><strong>rational effect:</strong> $Done(\alpha)$</td>
</tr>
</tbody>
</table>

**request that** $\phi$ **indicates** that the sending agent:

- holds that $j$ is capable of carrying out action $\alpha$, and
- does not believe that $j$ is intending to bring it about that $\alpha$ is carried out.

For more details, and information about other performatives, see [http://www.fipa.org/specs/fipa00037/SC00037J.pdf](http://www.fipa.org/specs/fipa00037/SC00037J.pdf)
Agents can have divergent interests (e.g., seller and buyer) but still cooperate.

**Use protocols:**
- Defines ‘rules of engagement’ between agents.
- Example: an QUERY message is followed by an INFORM message.
- Often modelled as a finite-state machine.

Protocols will also contain more detailed specifications depending on the type of interaction.

Example: fixed price sale vs. an auction.
Protocols as FSMs

- Protocols are often implemented as Finite State Machines.
- Each state represents a stage in the conversation — sometimes thought of as an information state.
- Arcs represent the exchanged messages that allow transition to a new state.
- Valid messages depend on the current state of the conversation — represented by outgoing arcs.
- Final states represent completed conversations.
Example FSM for a Conversation
Notion of agent ‘protocol’ often refers to stereotyped pattern of conversation between agents.

Available protocols are usually pre-specified by the agent designer.

Agents somehow need to discover which protocol to follow.

Choice of protocol to follow could be negotiated,

but in FIPA ACL, convention is to place name of the protocol in the :protocol parameter of the message.
FIPA-Query-Protocol

Initiator

Participant

query-if

query-ref

refuse

agree

[agreed]

failure

inform-t/f

[query-if]

inform-result

[query-ref]
Explanation of FIPA-Query-Protocol

UML Sequence Diagram. The diamond symbol indicates a decision that can result in zero or more communications being sent.

- Initiator requests Participant to perform a `inform` action:
  - `query-if`: whether proposition $P$ is true or false
  - `query-ref`: query about specified objects

- Participant processes request and decides whether to accept or refuse.

- If decides to accept, “agreed” becomes true.

- Participant uses `inform-t/f` to assert whether $P$ is true or false;

- uses `inform-result` to refer to object that was queried about.
Problems with Mentalistic Approach

- How can the sender rely on the receiver to adopt certain beliefs?

- More generally, the behaviour of an agent does not give unambiguous information about its mental state.

- Alternative approach in terms of social commitments:
  - agent commits / promises to carry out certain actions;
  - e.g., buyer in an auction commits to paying for goods;
  - fulfilment of commitments can be verified.
Summary

- Focus on agent-agent communication;
- Speech act theory provides theoretical framework for ACLs;
- Two examples:
  - KQML/KIF
  - FIPA-ACL
Reading: Wooldridge *An Introduction to MultiAgent Systems*, Chapter 8

See also Wooldridge’s web site:
http://www.csc.liv.ac.uk/~mjw/pubs/imas