Motivation

- Agents aren’t objects – therefore they must communicate in order to influence other agents (as opposed to invoking methods on other agents)

- We treat such communications just like other actions, and call it "speech act theory"\(^1,2\)


Example

- The state I might want to bring about is that “you believe it's raining in London”:

\[ B_{\text{you}} \text{Weather(london, raining)} \]

- but just because I send you “Weather(london, raining)” in a message doesn't necessarily mean that you will come to believe it (I may be a compulsive liar, or very bad at weather forecasts)

Speech Acts

- Performative verbs
  - Eg: request, inform, promise

- Aspects:
  - Locutionary act: making an utterance
  - Illocutionary act: action performed in saying something
  - Perlocution: the effect of the act

- Felicity conditions:
  - There exist an accepted procedure, and circumstances and persons must be specified
  - Must be executed correctly and completely
  - Must be sincere, and must have uptake

Example

- Request
  - Normal I/O conditions:
    - Hearer not deaf
    - Not uttered in a play or film
  - Preparatory conditions:
    - Speaker must correctly choose the act
    - Hearer must be able
    - Speaker must believe the hearer is able
    - Hearer wouldn't have done the act anyway
  - Sincerity conditions:
    - Not sarcasm, etc.

- Representatives: commits the speaker to the truth of a proposition
  - Eg: Informing

- Directives: attempt to get the hearer to do something
  - Eg: Requesting

- Commissives: commit to a course of action
  - Eg: Promising

- Expressives: express some psychological state
  - Eg: Thanking

- Declarations: effects some change in an institutional state of affairs
  - Eg: Declaring war
Rational Action

- *Speech act theory* can be considered as a specialization of a more general theory of rational action.
- Eg: "A request is an attempt on the part of spkr, by doing r, to bring about a state where, ideally, (i) addr intends α (relative to the spkr still having that goal, and addr still being helpfully inclined to spkr), and (ii) addr actually eventually does α, or at least brings about a state where addr believes it is mutually believed that it wants the ideal situation." 1, p. 241

FIPA Messages

- **performative**: the type of the communicative act of the message
- **sender**: the sender of the message
- **receiver**: the intended recipients of the message
- **reply-to**: subsequent messages in this conversation thread are to be directed to the agent named
- **content**: the content of the message, equivalently denotes the object of the action
- **language**: the language in which the content parameter is expressed
- **encoding**: the specific encoding of the content field
- **ontology**: the ontology(s) used to give a meaning to the symbols in the content expression
- **protocol**: the interaction protocol that the sending agent is employing with this ACL message
- **conversation-id**: an expression used to identify the sequence of communicative acts that form a conversation
- **Reply-with**: an expression that will be used by the responding agent to identify this message
- **in-reply-to**: an expression that references an earlier action to which this message is a reply
- **in-reply-by**: a time and/or date expression indicating the latest time the sending agent wants a reply

FIPA Performatives (informally)

<table>
<thead>
<tr>
<th>Performative</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept/proposal</td>
<td>accepting a previous proposal</td>
</tr>
<tr>
<td>agree</td>
<td>agreeing to perform some action</td>
</tr>
<tr>
<td>cancel</td>
<td>canceling proposals to perform an action</td>
</tr>
<tr>
<td>confirm</td>
<td>informing the object of a given proposition being true</td>
</tr>
<tr>
<td>description</td>
<td>informing the object that a given proposition is true</td>
</tr>
<tr>
<td>failure</td>
<td>an action was attempted but failed</td>
</tr>
<tr>
<td>inform</td>
<td>informing the object of some true proposition</td>
</tr>
<tr>
<td>not-understood</td>
<td>informing the object that the content is not understood</td>
</tr>
<tr>
<td>propagate</td>
<td>passing on an embedded message</td>
</tr>
<tr>
<td>promise</td>
<td>making a promise to perform an action</td>
</tr>
<tr>
<td>query</td>
<td>asking whether or not a proposition is true</td>
</tr>
<tr>
<td>reject</td>
<td>refusing to perform an action</td>
</tr>
<tr>
<td>reject-proposal</td>
<td>rejecting a proposal during negotiation</td>
</tr>
<tr>
<td>request</td>
<td>requesting to perform some action</td>
</tr>
<tr>
<td>request-whenever</td>
<td>requesting to perform some action whenever the proposition becomes true</td>
</tr>
<tr>
<td>subscribe</td>
<td>subscribing to notify of the value of a reference whenever the object changes</td>
</tr>
</tbody>
</table>

FIPA Semantics

- **Agent (j, φ)**
- **Proposition Bφ**: j believes φ
- **Feasibility Precondition**: FP: Bφ ∧ ¬B(Bifφ v Uifφ)
- **Rational Effect**: RE: Bφ
- **Uncertainty**: Uφ v U_1 φ (where U_1 means ‘_1’ is uncertain about φ)

FIPA inform semantics

- **Formally**: `<i, inform (j, φ)>
  FP: Bφ ∧ ¬B(Bifφ v Uifφ)
  RE: Bφ`

- **Example**: (inform
  :sender (agent-identifier :name i)
  :receiver (set (agent-identifier :name j))
  :content "weather (today, raining)"
  :language Prolog)

FIPA request semantics

- **Formally**: `<i, request (j, a)>
  FP: FP(a) [i]j ∧ B Agent (j, a) ∧ ¬B Ij Done (a)
  RE: Done (a)
FP(a) [i]j` denotes the part of the FPs of a which are mental attitudes of i.

- **Example**: (request
  :sender (agent-identifier :name i)
  :receiver (set (agent-identifier :name j))
  :content "open "db.txt" for input"
  :language vb)