Multi-Agent Systems
Lecture 9

Computer Science WPI

Spring 2002

Adina Magda Florea
adina@wpi.edu

Agent languages
Lecture outline

1 Overview of agent languages
2 The Actor model
3 IBM Aglets
4 Agent-oriented programming
1 Overview of agent languages

- **Agent language**
  - a language that allows to program software or hardware components in terms of some concepts of agents’ theories

- **Agent languages versus agent platforms**

**Concurrent object languages**

- self-contained concurrently executing objects
- have an internal state
- respond to messages from other such objects
  
  **Actors** (Hewitt, 1977, Agha, 1986)
  
  **ABCL** (Yonezawa, 1990)

---

**Agent-oriented programming**

**AGENT0** (Shoham, 1990, 1993)

- a specialization of object-oriented programming
- a new computational paradigm
- program agents in terms of mental states and high-level communication primitives

**PLACA** (Thomas, 1993)

- Planning Communication Agents, an extension of **AGENT0**
- Includes operators for planning to do actions and achieve goals
  
  - In both **AGENT0** and **PLACA**, relationship between logic and PL is loosely defined
Formal concurrent agent languages

Concurrent METATEM (Fisher, 1994)
- Based on temporal logic
- Agents are executed concurrently
- Communicate via asynchronous message passing
- Agents are programmed by giving the temporal logic specification of their behavior

Mobile agents / Scripting languages

TELESRIPT (General Magic Inc., 1995)
- 2 key concepts: agents and places
- Places = virtual locations occupied by the agents
- Agents =
  - software processes
  - mobile (from one place to another)
  - providers or consumers of goods in an electronic marketplace

Agent-Tcl (Gray, 1995)
- Based on Tcl (Tool Communication Language)
- Tcl - a script language with an interpreter
- Tk (toolkit)
- Agent-Tcl - Development of Tcl that permits agents to migrate from one computer to another
- Provision of security mechanisms
- Tcl primitives for communication
- Script language as the central agent language

IBM Aglets (IBM)
- Mobile agents
Reactive agent languages

ABLE (Wavish, 1992)
- Agent Behavior Language
- Subsumption architecture
- Agents are programmed in terms of simple, rule-like licences

SWORM (Hiebeler, 1994, Swarm Dev. Gr.)
- Swarm’s inspiration comes from the field of Artificial Life.
- Swarm is a set of libraries that facilitate implementation of agent-based models.
- A Swarm is a temporal container and physical home for an agent or a set of agents. Swarms can contain other Swarms.

2 The Actor Models

Mike Ciaraldi’s presentation
3 IBM Aglets

Alex Weiner’s presentation

4 Agent-oriented programming

- promotes a societal view of computing
- uses mental states to design agents; mental categories appear in the programming language
- the semantics of the PL is related to the semantics of mental states
- Mental states: beliefs, commitments, abilities, decisions
4.1 OOP versus AOP

<table>
<thead>
<tr>
<th></th>
<th>OOP</th>
<th>AOP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic unit</td>
<td>object</td>
<td>agent</td>
</tr>
<tr>
<td>Parameters defining basic unit</td>
<td>unconstrained beliefs, commitments, abilities, decisions (goals)</td>
<td></td>
</tr>
<tr>
<td>Process of computation</td>
<td>message passing and response to methods</td>
<td>message passing and response to methods</td>
</tr>
<tr>
<td>Types of messages</td>
<td>unconstrained based on speech act theory</td>
<td>honest, consistency, etc.</td>
</tr>
<tr>
<td>Constraints on methods</td>
<td>none</td>
<td></td>
</tr>
</tbody>
</table>

- Other differences?

A complete AOP system should have 3 components:

- A **formal** (logical) **language** for defining the mental state of the agents
- An (interpreted) **programming language** in which to define and program agents, with communication primitives based on speech-act theory; the semantics of the PL must match the semantics of the mental states
- An "**agentification**" process converting neutral devices into programmable agents
4.2 Mental categories in AGENT0

- Linear discrete time
- Beliefs: \( B_a^t \phi \)
- Actions: \( \text{raise\_arm(\text{robot})}^t \)
- Commitments (obligations): \( \text{CMT}_{ab}^t \phi \)
- Decisions (choices): \( \text{DEC}_{a}^t \phi = \text{CMT}_{a,a}^t \phi \)
- Abilities (capabilities): \( \text{CAN}_{a}^t \phi \)

**Properties of mental categories**

- Internal consistency of beliefs and commitments
- Good faith
- Introspection - limited
- Persistence of beliefs
- Persistence of commitments

4.3 Agent language

- Fact statements: \( t (\text{employee smith ibm}) \)
- Private action statements: \( \text{DO} \ t \ p\text{-action} \)
- Communicative action statements
  - \( \text{INFORM} \ t \ a \ \text{fact} \)
  - \( \text{REQUEST} \ t \ a \ \text{action} \)
  - \( \text{UNREQUEST} \ t \ a \ \text{action} \)
  - \( \text{REFRAIN} \ \text{action} \)
- Conditional action statements
  Test mental conditions before acting - by a particular agent at a time moment

  **Mental conditions examples:** \( (B \ \text{fact}) \ (\text{CMT} \ b \ \text{action}) \)

  General form \( (\text{IF} \ \text{mental-condition action}) \)

  **Ex:** \( (\text{IF} \ (B \ (t1 \ (\text{employee smith ibm})))) \)

  \( (\text{INFORM} \ t \ a \ (t1 \ (\text{employee smith ibm})))) \)
**Variables**

agent names, fact statements, action statements  ?x  ?!x

Ex:  
(IF (NOT ((CMT ?x) (REFRAIN sing)))  sing)  
(IF (B (t (emp ?!x ibm)))  (INFORM a (t (emp ?!x ibm))))

**Commitment rules** - key to AOP

Conditions for entering a commitment

- mental condition
- message condition

**Message condition** = a logical combination of message patterns

**Message pattern**  
(sender type content)

- type: INFORM, REQUEST, UNREQUEST, REFRAIN
- content: a fact statement or an action statement

**Commitment rule**

(COMMIT message-cond mental-cond (agent action))

Ex:  
(COMMIT (?a REQUEST ?action)  
(B (now (myfriend ?a)))  
(?a ?action))

---

A commitment rule adds a commitment to "action" if:

- the message condition holds for the incoming message
- the mental condition holds for the current mental state
- the agent is capable of doing action
- the agent is not committed to REFRAIN action
- the agent is not committed to action1 if action=REFRAIN

**Agent program**

- define agent’s abilities for all agents
- define agents’ initial beliefs
- fix the time grain
- state sequence of commitment rules

- Hypotheses - an agent loop fits into a time grain
  - if agents on different machines, the clocks are synchronized
Behavior of the agents - basic loop
1. Read current incoming messages
2. Update beliefs
3. Update commitments: add commitment, verify consistency
4. Execute commitments – see details bellow

Execute commitments: depends on the type of actions
- INFORM, REQUEST, UNREQUEST - send the appropriate message
- REFRAIN - no effect, just record
- DO - consult beliefs and commitments + check mental conditions associated to abilities
  - if it holds then perform private actions
- IF - consult beliefs and commitments + test mental conditions
  - if it holds then execute the action

References