UML extensions for Agents

Jaelson Castro
April 2000

Overview

- UML
- Agent UML (AUML)
  - Agent Interaction Protocols
  - Richer Role Specification
  - Package Extension
  - Deployment Diagram Extension
- Conclusion
**UML**

- Static models: class, package diagrams
- Dynamic models:
  - Interaction diagrams (sequence and collaboration)
  - Statecharts diagrams
  - Activity diagrams
- Implementation models: component, deployment diagrams
- Object constraint language

**AUML: Agent UML**

- Both FIPA and OMG are exploring and recommending extensions to UML
- Applications:
  - Specification of Agent Interaction Protocols (AIP)
  - Richer role specification
  - Package extension
  - Deployment diagram extension
Agent Interaction Protocols

- AIP describes a communication pattern as an allowed sequence of messages between agents and the constraints of the content of those messages.
- FIPA has specified many protocols

FIPA Contract-net Protocol

- The manager solicits proposals from other agents by issuing a call for proposals, which specifies the task and any conditions the manager is placing upon the execution of the task.
- Agents receiving the call for proposals are viewed as potential contractors and are able to generate proposals to perform the task as propose acts.
- The contractor’s proposal includes the preconditions that the contractor is setting out for the task, which may be the price, time when the task will be done, etc.
- The contractor may refuse to propose.
- The manager receives back replies from all of the contractors, evaluates the proposals and makes its choice of which agents will perform the task. One, several, or no agents may be chosen. The agents of the selected proposal(s) will be sent an acceptance message, the others will receive a notice of rejection.
- Once the manager accepts the proposal the contractor acquires a commitment to perform the task. Once the contractor has completed the task, it sends a completion message to the manager.
- The protocol requires the manager to know when it has received all replies. In the case that a contractor fails to reply with either a propose or a refuse, the manager may potentially be left waiting indefinitely. To guard against this, the cfp includes a deadline by which replies should be received by the manager.
FIPA notation

AUML: A layered Approach to Protocols

- Level 1 - Representing the overall protocol (sequence diagrams, packages, templates)
- Level 2 - Representing interactions among agents (sequence, collaboration, activity, statechart diagrams)
- Level 3 - Representing internal Agent Processing (activity and statechart diagrams)
AUML: A layered Approach to Protocols

Level 1: Overall protocol
Level 1: Overall protocol (packages)

- Packages aggregate modeling elements into conceptual wholes
  - in UML 1.3 packages only group class diagrams
- Protocols can be codified as patterns of agent interaction
  - in AUML packages can group sequence diagrams (to model protocol patterns)

Level 1: Using packages to express nested protocols

- Purchasing protocol (Broker X Retailer)
- Supplying protocol (Retailer X Wholesaler)
**Level 1- Overall protocol: Templates**

- In order for a package to be a truly pattern (not simply a reusable component) customization must be supported
- A template is a parameterized model element whose parameters are bound at model time
- Represented by dotted box in upper right corner of the package

**L1- Scenario involving buyers and sellers (Contract Net Protocol)**
**Level 2: Representing interactions among agents**

- Extended sequence diagrams (concurrent threads of interaction)
- Collaboration diagrams
- Activity diagrams

**L2- Extended sequence diagrams: concurrent communication**

- The multiple vertical bars indicate that the receiving agent is processing the various communication threads concurrently
L2- Extended sequence diagrams: concurrent communication

- A decision box will decide which CAs (zero or more) will be sent.
- If more than one CA is sent, the communication is concurrent

L2- Extended sequence diagrams: concurrent communication

- Exclusive OR, so exactly one CA will be sent
L2- Extended sequence diagrams (Agent/Role)

Expressing the roles an agent play:
Agent/Role

L2- Collaboration diagrams

Another way of showing pattern of interaction among agents
**L2 - Activity diagrams**

- Provides an explicit thread of control
- Useful for complex interaction protocols that involve concurrent processing

![Activity Diagram](image)

- ECN: Electronic Commerce Network Agent

---

**Level 3: Representing Internal Agent Processing**

- Specifying the internal processing of agents that are not aggregates
  - Activities
  - Statecharts
L3- Internal Agent Processing: Activity Diagrams

- Processing within an Order Processor agent

L3- Internal Agent Processing: Statechart Diagram

- Payment Receiver and Invoice Sender Agents
Other AUML Considerations

- Richer role specification
- Package extension
- Deployment diagram extension

Richer role specification

- Often we need to express the role an agent may play in the course of its interaction with other agents
- If the number of agent and roles increases, UML diagrams become graphically too complex
Agent/Role

4 Agents playing 6 Roles:
- Customer, Negotiator, Contractor,
- Competitor Analyzer, Competitor, Debtor

Reducing the visual complexity

Each role with its own lifeline
Reducing the visual complexity

- Each agent with a single lifeline and each activation is labelled with the appropriate role name

Role and Collaboration Diagrams

- UML has no facility to represent agent roles on interaction lines
- Labelling messages with the role
Role and Activity Diagram

 Roles can be associated with activities

Role and Activity Diagram

 Roles can be represented by notes
**Package extension**

- The interface can be an agent itself

**Deployment diagram extensions**

- Indication of mobility paths and at-home declarations
Conclusion

- (A)UML provides tools for
  - specifying agent interaction protocols
  - representing the internal behavior of an agent
  - representing role specification, packages with agent interfaces, deployment diagrams indicating mobility, etc.

References
