

에이전트 지향의 소프트웨어 모델링 방법

김진태, 김민정, 박인세, 이승연, 박수용 (jtkim, mjkim, tjpark, sylee)@selah.sogang.ac.kr,
syipark@ccs.sogang.ac.kr 서강대학교 소프트웨어 공학 연구실

Agent-oriented Software Modeling Method and Tool

Jintae Kim, Minjeong Kim, Injae Park, Seungyouon Lee and Sooyong Park, Software Engineering
Lab. Sogang Univ.

요약

에이전트에 대한 연구는 최근에 들어와 그 응용분야가 점차 확대되어 가고 있으며, 에이전트를 근간으로 하는 소프트웨어도 점점 복잡화, 대형화되고 있는 추세이다. 이에 따라 에이전트를 근간으로 하는 소프트웨어 개발 방식도 좀더 체계적이며 공학적인 접근이 요구되고 있다. 본 논문에서는 에이전트 기반의 소프트웨어를 효율적으로 모델링하기 위하여 에이전트 추출, Intra agent modeling과 Inter agent modeling 기법과 방법론을 지원하는 Agent-Oriented Modeling Tool에 대해 소개한다. 에이전트 추출은 실세계에 존재하는 객체들로부터 에이전트를 추출하는 방법을 사용했고, Intra agent modeling에서는 에이전트의 내부를 Goal, Belief, Plan, Capability로 나누어 각각의 모델링 방법을, Inter agent modeling에서는 다중 에이전트 시스템에서 에이전트의 Mobile과 에이전트간의 메시지 교환을 중심으로 한 모델링 방법을 제안한다.

1. 서론

현재 소프트웨어의 모습은 점차 복잡하고 대형화되어 가고 있으며, 이를 지원하기 위한 소프트웨어 환경 역시 이질적이고 분산된 실시간 환경을 지원하도록 요구되고 있다. 이렇게 시스템이 대형화되고 복잡해짐에 따라, 미래의 소프트웨어 환경에 적합한 하나의 해결책으로 부각되고 있는 것이 에이전트 지향의 소프트웨어이다.

이러한 에이전트 지향의 소프트웨어들이 개발됨에 따라 소프트웨어 공학적인 측면에서 에이전트 지향의 소프트웨어 개발을 위한 분석적인 접근 방법이 점점 중요해지고 있으며, 체계적인 개발 방법론에 대한 연구가 절실히 필요하게 되었다.

본 연구는, 실세계는 에이전트와 객체가 공존하며, 에이전트는 Active Object[1] 또는 분산 객체[2]의 모습이 라는 개념 하에, UML(Unified Modeling Language)[3,4]을 이용한 문제 영역 분석 과정에서 얻어진 객체들에 앞으로 기술될 기준들을 적용시켜 에이전트화 시킬 객체들과 새로 생성해야 할 에이전트들을 결정한다. 이렇

게 생성된 에이전트 중심의 시스템을 구체적으로 모델링하기 위해, 본 논문에서는 크게 두 부분으로 나누어 모델링을 하였다. 즉 에이전트가 가지는 속성들과 행동들을 표현하기 위한 *Intra agent modeling* 방법과, 여러 에이전트들 간의 메시지의 교환 및 이동성을 나타내기 위한 *Inter agent modeling*을 제안한다. 또한 간단한 전자상거래를 문제영역으로 잡아 모델링 방법을 적용시켜 보았다

적용 시나리오

구매자는 인터넷을 통하여 컴퓨터 부품을 구입하려고 한다. 구매자는 상품 구매 Agent에게 구입하고자 하는 컴퓨터 부품명과 사양을 입력한다. 상품 구매 Agent는 분산되어 있는 서버들을 이동하면서 각 서버의 DB에 저장되어 있는 상품들의 정보를 수집하고, 그 정보들을 비교하여 구입하려는 상품 중에서 최적의 가격을 가진 서버의 이름과 상품명, 상품 가격을 구매자에게 제시한다. 구매자는 Agent가 제시한 정보를 보고 그 상품을 구입할 지의 여부를 결정한다. 구매자가 구입을 결정하면 구매 Agent는 상품 구매에 대한 order를 보낸다.

[표 1] 전자상거래 적용 시나리오

2. 에이전트 지향의 모델링 방법

2.1 UML을 이용한 문제 영역 분석

본 논문에서는 문제 영역의 분석을 위해 기존의 검증된

방법인 UML[3,4]을 이용하여 문제 영역을 분석한다. 이를 통하여 시스템의 정적인 면과 동적인 면을 분석할 수 있으며, 에이전트화 할 수 있는 객체 및 새로 생성할 에이전트를 결정할 수 있도록 객체 지향의 분석 방법을 진행한다. 여기에 사용되는 각 다이어그램들은 UseCase diagram, Sequence diagram, Class diagram, Activity Diagram이다.

2.2 에이전트 추출

UML에 의해 분석한 시스템의 동적인 면과 정적인 면을 고려하여, 클래스 다이어그램에 다음의 기준을 적용하여 객체를 중에 에이전트로 만들어야 하는 것들을 선택한다. 각 기준들은 앞에서 말한 에이전트의 기본 성질과도 밀접한 연관성을 갖는다.

- | |
|--|
| <ol style="list-style-type: none"> 1. 내부 지식을 가져야 하며 그것이 지속적인 update가 필요한가? 2. 능동적으로 결정을 내리야 하는가? 3. 지능이 필요한가? 4. 외부 객체와 상호작용이 필요한가? 5. Cooperation이 필요한가? 6. Heterogeneous, 분산환경에서 작동해야 하는가? |
|--|

[표 2] Agent Selection Rule

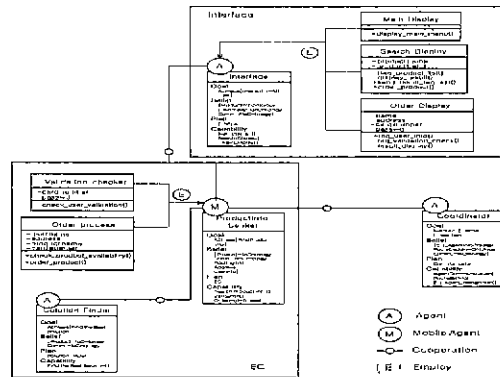
이러한 기준에 의해 추출된 에이전트는 Mobility가 필요한지의 여부에 따라 Mobile Agent 와 Agent로 구분한다.

2.2.1 Agent-Class Diagram

본 논문에서 제안한 에이전트-클래스 다이어그램은 실제 세계에 객체와 에이전트가 공존한다는 사실에 근거하여, 시스템을 구성하는 에이전트들과 클래스들의 협력 및 그들의 관계를 나타낸다. 지금까지 그러한 UML 기반의 Diagram들에 Agent Selection Rule을 적용하여 에이전트화 될 수 있는 부분을 추출하도록 한다. 6가지의 추출 기준을 적용하여 하나 이상의 클래스가 에이전트에 도달할 때 이를 그룹화 하여 에이전트로 나타낸다. Class Diagram과 Activity Diagram에서 3개의 에이전트(Interface Agent, EC Agent, Solution Finder Agent)를 도출해냈으며 하나의 에이전트(Coordinator Agent)를 지금까지 제시한 Diagram들과 시나리오에 적용하여 보다 원활한 작업의 수행을 위하여 생성하였다.

각각의 에이전트에 대해 [표 2]에서 제시된 선택기준을 적용시켜 Agent와 class가 공존하는 Agent-Class Diagram[그림 1]을 만든다.

Agent-Class Diagram



[그림 1] EC에서의 Agent-Class Diagram

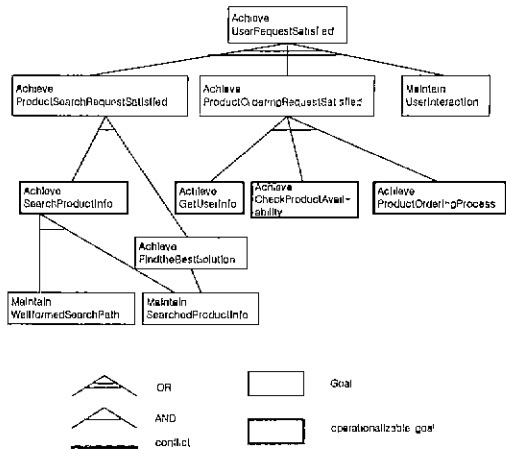
3. Internal Modeling

Agent의 내부대한 모델링 방법으로 BDI모델을 기초로 제시한다.

3.1 Goal Modeling

Agent가 궁극적으로 하고자 하는 일을 보여 준다.

Goal-HierarchyDiagram



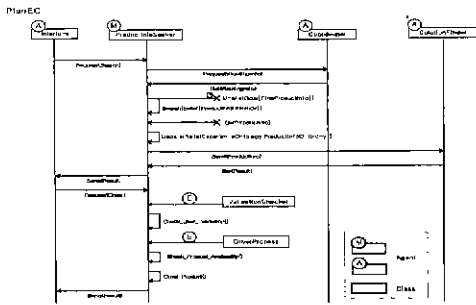
[그림 2] EC에서의 Goal-Hierarchy

3.2 Belief Modeling

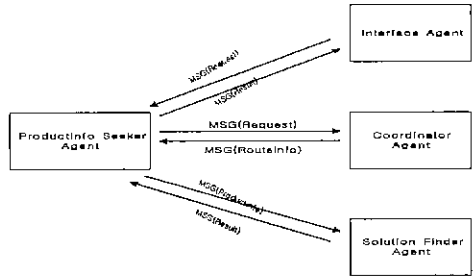
에이전트의 속성 중 Belief는 주어진 작업을 수행하는데 필요한 모든 지식 정보를 말한다. 이 Belief Modeling은 ontology를 사용하여 구현한다.

3.3 Plan Modeling

Plan은 에이전트가 Goal을 달성하기 위해 취하는 행동이다.



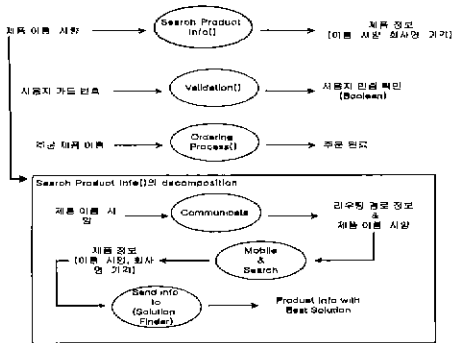
[그림 3] EC에서의 ProductInfoSeeker Agent에 대한 Plan Sequence Diagram



[그림 6] EC에서의 Agent Communication

3.4 Capability Modeling

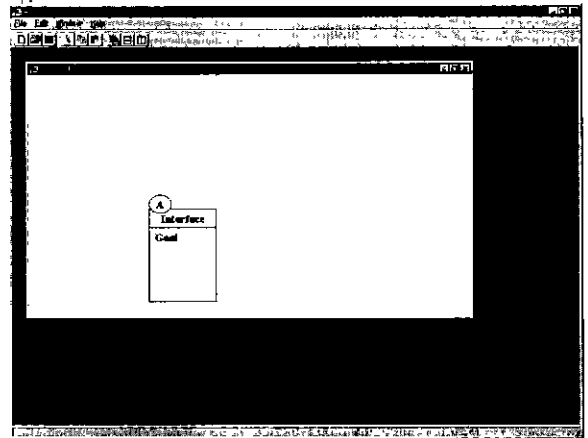
Capability는 에이전트가 수행할 수 있는 기능을 나타낸다.



[그림 4] EC에서 ProductInfo Seeker Agent의 Capability

5. 분석 도구

현재 서강대학교에서 위에서 제시한 모델링 방법을 지원하기 위한 Tool을 개발 중이며 구현언어는 델파이, 환경은 windows 98환경이다. 다음은 개발중인 tool의 일부분이다.



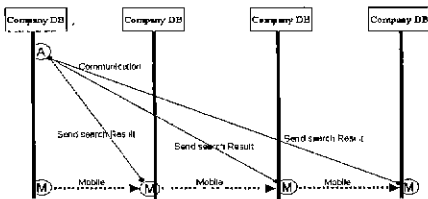
[그림 7] 개발중인 Tool의 모습

4. External Modeling

Agent와 외부환경과의 관계를 모델링한다.

4.1 Agent Mobile Modeling

Agent가 mobile을 나타낼 때 어떤 요소들을 표현해야 하는지 보여 주고 있다.



[그림 5] EC에서의 Agent Mobility

6. 결론

제한한 모델링 방법은 agent 소프트웨어 개발에 있어 분석할 수 있는 체계적인 방법론이라는 점에서 의미가 크며, 개발중인 지원 tool로 실제 방법론을 적용시킬 때 더욱 안정적인 모습을 갖추게 되리라 본다.

7. 참고문헌

- [1] N R Jennings, K P. Sycara, and M Woodriddle A Roadmap of Agent Research and Development In Journal of Autonomous Agents and Multi-Agent Systems 1(1), pages 7-36 July 1998
- [2] M.Schroeder, Are Distributed objects agents?, *International Bi-Conference Workshop on AGENT-ORIENTED INFORMATION SYSTEMS (AOIS'99)*, 1999
- [3] Paul Harmon and Mark Watson, *Understanding UML : The Developers Guide*, Morgan Kaufmann Publishers, 1998
- [4] Bran Selic, and Jim Rumbaugh, *Using UML for Modeling Complex Real-Time Systems*, 1998.