제18회 한국정보처리학회 추계학술발표대회 논문집 제9권 제2호 (2002.11)

e-비즈니스 에이전트지향 컴포넌트기반 개발 프로세스에 관한 연구

김성원*, 현창문**, 신호준***
*안양대학교 전자계산학과
**탐라대학교 컴퓨터공학과
***대구가톨릭대학교 컴퓨터공학과
e-mail:swkim@aycc.anyang.ac.kr, cmhyun@cheju.tamna.ac.kr, 98521002@amare.ac.kr

A Study on the e-Business Agent Oriented Component Based Development Process

Soung-Won Kim*, Chang-Moon Hyun**, Ho-Jun Shin***
*Dept. of Electronical & Electronic Engineering, Anyang University
**Dept. of Information & Communication, University of Tamna
***Dept. of Computer Engineering, Catholic University of Daegu

요 약

현재의 소프트웨어 개발 패러다임은 전체 시스템에서 독립적인 업무 또는 기능을 수행하는 단위이고 유지보수에 필요한 교체 가능한 모듈로 재사용성과 대치성을 제공하는 컴포넌트기반 개발(CBD:Component Based Development)이 연구되고 있다. 기업이나 조직은 웹을 통한 비즈니스 기능의 확대에 따른 효과적이고 빠른 개발과 적용을 요구함에 따라, 에이전트를 통한 유연한 서비스를 제공하고자 한다.

본 논문에서는 비즈니스 영역에 대한 에이전트의 효과적인 개발과 적용을 위해 컴포넌트, e-비즈니스 에이전트, 자원, 관리 관점을 고려하여, e-비즈니스 에이전트지향 컴포넌트기반 개발 프로세스(ebA-CBD:e-business Agent oriented Component Based Development)를 제안한다. 특히, e-비즈니스 에이전트의 식별과 명세를 위해 ebA-CBD 매트릭스, ebA Architecture Model과 ebA-Spec.을 제시하며, 사례를 통해 프로세스의 이해성 및 사용성을 제공하고자 한다.

1. 서론

현재 IT 분야로 일축되고 있는 비즈니스 영역에서 최우선 과제로 생각하고 있는 것은 비즈니스 개념을 적절히 표현하는 빠른 개발을 요구하고 있으며, 이러한 e-비즈니스 영역에 에이전트의 활용은 간과 할 수 없는 부분이다. 따라서, 빠른 개발시간과 변경에 대한 높은 대체성과 유연성을 가진 컴포넌트 기술을 활용한 에이전트 서비스 제공은 중요하다.

본 논문에서는 e-비즈니스 에이전트, 컴포넌트, 자원, 관리의 4가지 관점에서 각각의 특징을 정의하고 이에 대한 컴포넌트기반 개발 방법을 기반으로 ebA-CBD 프로세스를 제안한다. e-비즈니스 에이전트를 ebA(e-business Agent)로 명명하고, 제안된 프로세스에서 에이전트 관점을 중심으로 에이전트의 요구사항 분석을 통해 필요한 에이전트 식별과 에이전트간의 관계를 정의한다. 이를 위해 ebA-CBD 매트릭스, ebA Architecture Model 모델을 제시하며, 각각의 식별된 e-비즈니스 에이전트는 ebA-Spec.에 따라 정의한다. 또한, 사용자별 목록 서비스에 대한 사례를 통해 e-비즈니스 에이전트에 명세과정을 제시한다.

2. 관련 연구

2.1 소프트웨어 에이전트

소프트웨어 에이전트는 좀더 구체적이고 특정적인 에이전트 종류로써, 환경과 상호 작용할 수 있는 자동화된 소프트웨어 엔티티로 정의되며, 소프트웨어를 사용하여 구현되는 에이전트다. 이는 자율성을 가지고 있고, 다른 엔티티에 반응할 수 있다. 이러한 엔티티는 사람과 기계 및 다양한 환경과 플랫폼에서 다른 소프트웨어 에이전트를 포함한다[1].

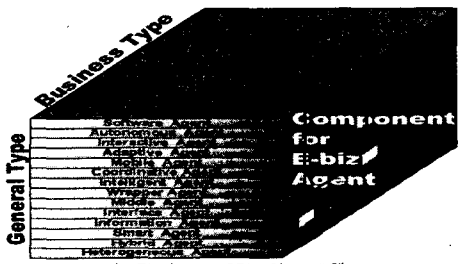
기본적으로 소프트웨어 에이전트는 소프트웨어에 대한 설계 패턴이다. 툴, 언어와 환경은 특별히 에이전트 기반 패턴을 지원하기 위해 개발될 수 있다. 그러나, 에이전트 설계 패턴은 객체지향 툴, 언어와 환경 혹은 다른 툴, 언어와 환경을 이용하여 또한 구현될 수 있다. 이는 자율적이고, 상호작용하며, 적응할 수 있는 소프트웨어 엔티티를 지원할 능력이 있다. 에이전트 기반 툴은 설계 패턴이 소프트웨어에서 상속받았기 때문에 명시적으로 프로그램을 작성하는 것보다 더 바람직하다. 다시 말하면, 객체 기술은 가능한 에이전트 기반 기술을 사용할 수 있지만, 에이전트가 요구하는 자율성, 상호 작용성, 적응성은 현재 객체기술로 지원되지 않는다. 이러한 특징은 객체지향 접근에 부가될 수 있으며, 에이전트를 위한 설계 패턴과 에이전트 기반 소프트웨어는 전반적이고 직접적으로 지원되지 않는다.

2.2 ebA-CBD 참조 모델

컴포넌트 참조 아키텍처의 구성을 위해서 식별된 일반 에이전트의 타입과 e-비즈니스 기능 속성으로 에이전트를 분류를 한다. 이를 기반으로 e-비즈니스 에이전트를 위한 컴포넌트와 개발 프로세스의 메타 아키텍처로써 ebA-CBD 참조 아키텍처는 다음(그림 1)과 같다. 참조 아키텍처는 14개의 일반적인 에이전트 타입과 11의 세부적인 비즈니스 에이전트 타입으로 분리된 측면으로 이루어져 있으며, 도메인 지향적인 컴포넌트 아키텍처이다[2].

두 가지 영역은 상호 독립적이며, 상호 참조의 성질을 가지고 있으며, 각각의 영역에 따라 일반적인 에이전트 타입은 수평적인 특징을 가지면, 특정 e-비즈니스에 관련된 에이전트 타입은 수평적인 특징을 가진다. 또한, 일반 에이전트 타입은 에이전트 플랫폼, 에이전트 프레임워크, 애플리케이션으로 대응된다. 에이전트 기반의 시스템이나 애플리케이션을 비즈니스 영역에 적용하거나, e-비즈니스 영역에서 에이전트 기술을 도입할 경우에 참조 아키텍처를 통해 필요한 컴포넌트를 조립하여 구성 가능하다.

개발된 컴포넌트는 이 참조 아키텍처에 따라 분류되며, 일반적인 에이전트의 타입과 비즈니스 속성에 따라서 하나의 영역에 배치된다. 또한, 에이전트 타입을 지향하는 시스템이나 비즈니스 영역에 필요한 에이전트를 응용하고자 할 경우, 이와 연관된 컴포넌트를 비즈니스 영역에서 식별하여 조립함으로 시스템 구성이 가능하다.



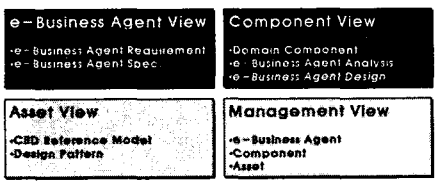
(그림 1) ebA-CBD 참조 모델

3. ebA-CBD 프로세스

3.1 프로세스의 4 View

개발 프로세스를 정의하기 이전에 e-비즈니스 도메인을 대상으로 목표한 시스템이나 소프트웨어를 개발하기 위해서는 4가지 관점에서 개발 프로세스를 고려할 수 있다. 컴포넌트 기반으로 e-비즈니스 에이전트를 구성하는 것은 에이전트의 크기에 따라서 하나의 컴포넌트나 그 이상의 컴포넌트로 구성될 수도 있지만, 컴포넌트로 구성하기 어려운 부분도 존재한다. 본 논문에서는 e-비즈니스 에이전트의 기능성에 초점을 두고 이전 연구에서[2] 제시된 CBD 참조 모델을 기반으로 다음(그림 2)과 같이 4가지 관점에서 프로세스를 고려한다.

e-business agent 관점은 개발하고자하는 대상이 되는 부분으로써, 일반적인 컴포넌트 개발 단계에서는 도메인 영역을 포함하여, 도메인 분석 단계에 포함하거나, 컴포넌트 분석, 설계 단계에 포함하지만, 에이전트 영역 분류를 기반으로 작성된 CBD 참조 모델에 따라서 에이전트를 식별하기 위해서 e-비즈니스 에이전트의 요구사항과 명세에 대한 부분의 고려가 필요하다.



(그림 2) 프로세스의 4 View

또한, Component 관점은 에이전트 구성이 가능한 부분이 컴포넌트로 존재하지 않을 경우 새로운 컴포넌트의 개발에 초점을 두는 부분이다. 따라서, e-비즈니스 도메인에 특화된 컴포넌트 분석, 설계 과정이 새롭게 요구된다.

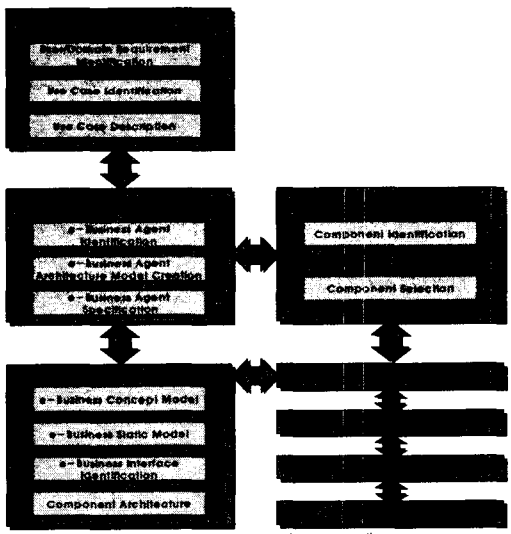
Asset 측면은 CBD 참조 모델을 구성한 일반적인 에이전트와 비즈니스 측면의 에이전트를 고려하여, 요구사항, 분석, 설계 단계에서의 명세 및 모델 작성에 도움을 주고, 에이전트 분류를 통한 컴포넌트 식별을 가능하게 한다. 또한, 각각의 영역별 에이전트 구성을 위한 설계 패턴을 제공함으로써, 유용성을 제공하는 측면이다.

마지막으로, Management 측면은 앞의 세 가지 측면에 대한 기술적인 관리, 조직적인 관리 작업을 들 수 있다. 기술적인 관리는 에이전트 관점, 컴포넌트 관점, 자산 관점의 행위를 감독하고, 개발되는 부산물과 요구되는 행위로 작성되는 목표 소프트웨어에 대한 보증을 한다. 조직 관리는 본 논문에서 고려하지 않는 부분이며, 인증된 부산물, 자산 등의 충분한 자원에 대한 테스팅과 기술적인 행위간의 조화를 위해 인적, 물적 자원을 결정한다.

3.2 ebA-CBD 프로세스 정의

프로세스에서 고려해야할 네 가지 관점을 기반으로 작성된 e-비즈니스 에이전트 도메인을 위한 컴포넌트기반 개발 프로세스는 다음(그림 3)과 같다. 사용자와 도메인 요구사항을 기반으로 e-비즈니스 에이전트 요구사항을 식별하여, 에이전트 명세를 개발하며, 이를 통해 컴포넌트를 작성하거나, CBD 참조 모델을 기반으로 컴포넌트를 선택하여 조립함으로써 목표로 하는 시스템이나 소프트웨어를 구축한다.

본 논문에서는 e-비즈니스 에이전트 요구사항 분석과 에이전트 관점의 프로세스 단계인 e-비즈니스 에이전트 스펙 개발 단계에 초점을 두고 기술한다.



(그림 3) ebA-CBD 전체 프로세스

3.2.1 ebA 요구사항 식별

e-비즈니스 도메인과 사용자, 시스템, 기반환경 등을 고려하여 문제 영역을 분석하고 요구사항 시나리오를 작성함으로써, 개발 목표를 명확히 정의한다. 사용자의 관점에서 시스템이 어떻게 동작하는지 결정하여 전체 시스템 예측 활동은 스토리보딩 기법을 사용하여 요구사항 시나리오를 작성한다. 또한, 작성된 요구사항 시나리오를 기반으로 사용사례를 식별하고, UML(Unified Modeling Language)의 Use case Diagram을 작성한다. 이는 목표 시스템의 영역을 정의하고 전체 업무 영역 중에 어느 업무 기능을 책임질지 명확히 하는 것이며, 행위자와 기능간

의 관계 및 부가적인 설명이 된다. 요구사항 식별단계에서는 후보가 될 수 있는 에이전트에 대한 이해와 에이전트를 기반으로 구성되는 기능적인 요소나 시스템 전체의 구성 파악이 주목적이다.

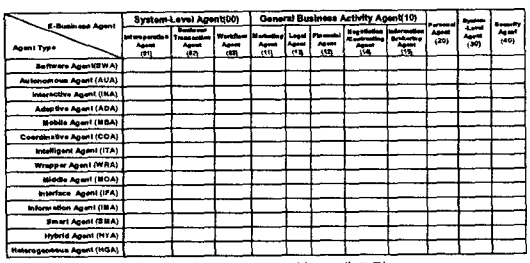
3.2.2 ebA 명세 개발

식별된 요구사항을 기반으로 개발될 에이전트에 대한 식별을 통해 에이전트 아키텍처 모델과 각각의 에이전트에 대한 명세 작성을 주요 단계로 구성된다. 이 단계를 통해 얻은 부산물은 구성될 컴포넌트의 식별 및 새로운 컴포넌트의 개발에 주요한 자원이 된다.

(1) ebA 식별

요구사항을 통해 얻어진 에이전트의 기본적인 요구사항을 통해 e-비즈니스 에이전트의 CBD 참조 모델에 어느 영역에 위치하는지를 식별하고 식별된 내용을 다음(그림 4)의 참조 모델 매트릭스를 이용하여 체크리스트를 작성한다. 또한, 작성된 매트릭스에 해당하는 에이전트의 이름을 에이전트의 사용 사례 기술서를 참조하여 명명하며, 이는 에이전트 명세 아키텍처 모델과 스펙 생성에서 목표 에이전트로 정의한다.

참조 모델 매트릭스에서 일반적인 에이전트 타입은 이름기반 식별자와 e-비즈니스 에이전트 타입은 수치기반 식별자를 정의하며, 조합되는 부분에 여러개의 에이전트나 컴포넌트가 식별될 수 있으므로 순차코드를 고려한다. 두 코드의 조합은 요구되는 에이전트의 식별코드로 사용된다. 예를 들어, 식별된 일반적인 에이전트 타입이 "Software Agent"이고 e-비즈니스 에이전트 타입이 "Marketing Agent"이고, 이 영역에서 식별된 첫 번째 에이전트일 경우에 식별코드는 "SWA-11-01"이 된다. 식별된 영역에 대한 분석 패턴, 설계 패턴이 존재한다면, 분석, 설계 과정에 적용 가능하다.



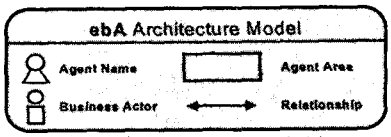
(그림 4) ebA-CBD 참조 매트릭스

(2) ebA Architecture Model 생성

식별된 e-비즈니스에이전트는 각각의 기능에 따라 에이전트 타입에 대응되며, 구별될 수 있는 특징을 가지게 된다. 동일한 에이전트 타입을 이라도 목적에 따라 다른 에이전트와 협동이나 협상 등의 기능을 수행하게된다. 이러한 다른 에이전트와의 관계 및 상호 작용하는 동적인 성질을 명세하기위한 부분이 에이전트 아키텍처 모델을 생성하는 것이다. 본 논문에서는 비즈니스 사용자와의 관계 및 식별된 에이전트를 에이전트 영역에 정의하고 각 에이전트간의 관계를 다음(그림 5)과 같이 ebA Architecture Model로 제시한다.

(3) ebA-Spec. 작성

작성된 ebA Architecture Model을 기반으로 각각의 에이전트 별로 에이전트의 특징, 공유되는 자원, 접근에 대한 정보, 모델 정보 등을 명세한다. 이는 에이전트의 기능적, 비기능적인 특징을 표현하는 것으로써, 기능적인 명세는 UML의 class diagram, sequence diagram을 이용하여 각 에이전트마다의 속성과 행위를 기본적으로 정의한다. 또한, 정의된 에이전트간의 비동기적으로 전송되는 메시지와 이벤트에 대한 부분을 표현하며, 다음 <표 1>은 ebA-Spec.을 명세항목과 설명으로 나타낸 것이다.



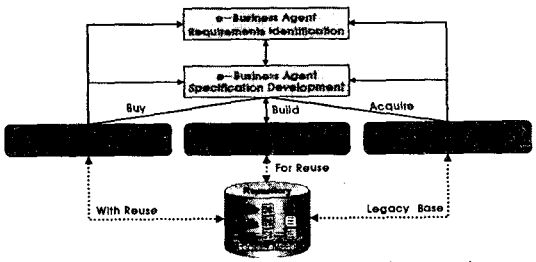
(그림 5) ebA Architecture Model

<표 1> ebA-Spec. 목록

Item	Description
Agent Name	식별된 ebA 이름
e-business Type	ebA-CBD 참조모델에서 e-비즈니스 타입
General Agent type	ebA-CBD 참조모델에서 e-비즈니스 타입
Identification Code	ebA-CBD 참조 매트릭스에서 식별 코드
Access Information	에이전트가 접근, 저장, 갱신할 정보
File Information	에이전트가 동작중에 생성하는 파일정보
Related Agent	협상이나 협동하거나 메시지를 주고받을 에이전트
Class Diagram	에이전트의 속성 및 행위 표현
Sequence Diagram	에이전트간의 행위적인 순서 표현

3.3 프로세스간의 관계 및 재사용성

e-비즈니스 에이전트에 대한 요구사항을 식별 및 명세를 한 다음(그림 6)과 같이 프로세스는 컴포넌트 활용, 컴포넌트 개발 및 기존 에이전트 자산의 획득으로 세분화된 단계로 나눌 수 있다. 이는 재사용 측면이나 기술적인 측면에서 프로세스간의 다른 관점을 가지며, 공유되는 리파지토리에서 재사용 대상이 다르다.



(그림 6) ebA-CBD 프로세스간의 관계 및 재사용성

컴포넌트 활용 프로세스 측면은 요구명세와 컴포넌트 아키텍처 모델에서 요구되는 컴포넌트를 상용 컴포넌트(COTS: Commercial Of The Shelf) 사이트에서 일반적으로 구매하게 된다. 조직의 컴포넌트 개발 성숙도가 높을 경우 in-house 형태로 구축된 컴포넌트 리파지토리에서 쉽게 획득할 수 있다. 이는 기존 컴포넌트로 목표 시스템을 구축하는 형태의 컴포넌트 재사용이다(with reuse). 컴포넌트 개발 프로세스는 e-비즈니스 에이전트 명세를 통해 식별된 영역에 적절한 컴포넌트가 없을 경우 요구사항과 명세를 기반으로 컴포넌트 개발 단계를 거친다(for reuse). 재사용과 목표 시스템을 구성하기 위한 적절한 컴포넌트를 생성하기 위한 프로세스가 수반되며, 리파지토리에서 기존의 분석, 설계된 내용이 있을 경우 참조가 가능하다. 마지막으로, 에이전트 자원의 획득은 요구사항 분석과 명세 개발 단계에서 도출된 내용을 기반으로 기존의 에이전트 시스템에서 특정 기능이나, 모델, 코드 정보 등의 에이전트 자산을 통해서 시스템을 구축하거나 새로운 컴포넌트의 생성을 하는 경우이다(legacy base).

4. 사례 연구

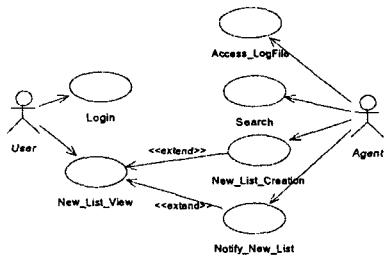
4.1 사용자 목록 서비스의 ebA 요구사항 식별

ebA-CBD에서 제시된 첫 번째 단계로써 다음(그림 7)은 상용 컴포넌트 리파지토리에서 새롭게 추가하고자 하는 시스템 기능 및 요구사항을 스토리보딩 기법에 따라

표현한 것이다. 도메인 요구사항 명세서를 바탕으로 일반 사용자와 에이전트, 두 개의 액터가 존재하며, 각각의 액터에 의해 접근되는 사용사례는 다음(그림 8)과 같이 도식화하였다.

> 웹상에서 구축된 상용 컴포넌트 리파지토리에서는 비정기적으로 다양한 사용자가 검색을 통해 요구에 맞는 컴포넌트 식별 작업을 한다. 적절한 컴포넌트가 있을 경우 이를 선택하고 구매를 한다. 이러한 시스템에서는 사용자 로긴을 통해 정보에 접근가능하고 운영 조직에 따라 분류기준을 정의하고 이를 기반으로 컴포넌트 카테고리별 관리한다.
>
> 새로운 시스템에서는 사용자가 로긴했을 경우에 상용 컴포넌트 리파지토리에 새롭게 등록된 컴포넌트 목록을 자동적으로 제시하는 서비스를 하고자 한다. 이는 사용자가 최후 로그아웃된 시점에서 다음 로그인하는 시점사이에 갱신된 컴포넌트 목록을 보여주는 것으로, 각각의 사용자를 대상으로 이러한 서비스를 에이전트를 통해서 개발 및 제공 하고자 한다. 또한, 이 시스템을 '사용자 목록 서비스'라고 지칭한다.

(그림 7) 도메인 요구사항 명세서

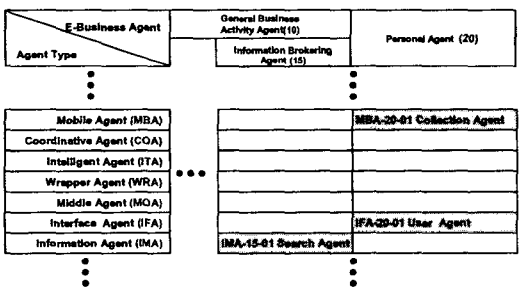


(그림 8) 사용자 목록 서비스의 Use Case Diagram

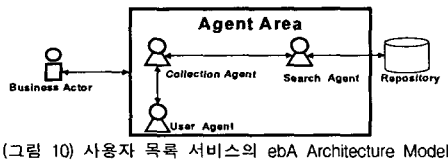
4.2 사용자 목록 서비스의 ebA 명세 개발

요구사항 식별 단계에서 사용자 로그파일에 대한 접근과 검색, 새로운 컴포넌트 리스트의 생성, 새로운 컴포넌트 리스트를 사용자에게 알려 주는 서비스가 에이전트로 대응되는 부분으로 식별되었다. 이를 기반으로 에이전트 이름을 명명하고 식별코드를 부여한 것을 다음(그림 9)과 같이 ebA-CBD 참조 매트릭스로 표현하였다. 여기서, 리파지토리에서 새롭게 등록된 컴포넌트의 검색 위한 Search Agent, 주기적으로 검색된 목록과 정보의 갱신을 위한 Collection Agent, 사용자의 로그파일 관리와 알림 서비스를 위한 User Agent가 식별되었다.

비즈니스 액터와 3개의 에이전트간의 관계 및 구성을 식별하기 위한 간략한 Architecture model은 다음(그림 10)과 같으며, 다음 <표 2>는 본 논문에서 제안된 ebA-Spec.을 기반으로, User Agent에 대한 ebA-Spec 항목을 요약하였다.

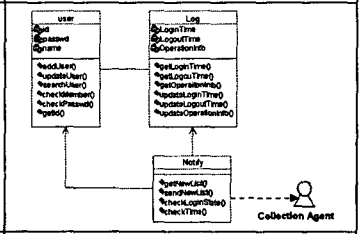
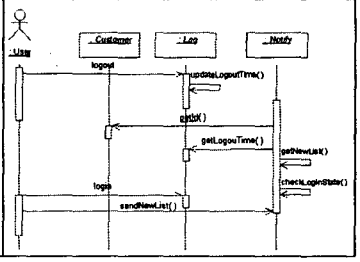


(그림 9) 사용자 목록 서비스의 ebA-CBD 매트릭스



(그림 10) 사용자 목록 서비스의 ebA Architecture Model

<표 2> 사용자 목록 서비스 ebA-Spec.

Item	Description
Agent Name	User Agent
e-business Type	Personal Agent
General Agent type	Interface Agent
Identification Code	*IFA-20-01*
Access Information	User Id, User Login Time, Logout Time
File Information	User Log File
Related Agent	Collection Agent
Class Diagram	
Sequence Diagram	

5. 결론 및 향후연구

비즈니스 영역에 적용가능한 에이전트를 컴포넌트 기반으로 개발하기 위해 본 논문에서는 e-비즈니스 에이전트 개발을 위해 4가지 관점에서 각각의 특징을 정의하고 이에 대한 컴포넌트기반 개발 방법을 적용하여 ebA-CBD 프로세스를 제안하였다. 전체 프로세스 중에서 에이전트 관점을 중심으로 에이전트의 식별과 에이전트간의 관계를 정의하기 위해 ebA-CBD 매트릭스, ebA Architecture Model 모델을 제시했다. 또한, 전체 에이전트에 대한 명세는 ebA-Spec.으로 정의함으로써, e-비즈니스 에이전트의 정보 제공을 체계적이고 직관적인 인식이 가능하도록 모델정보를 포함하여 정의하였다. 체계적인 프로세스 도입과 e-비즈니스 에이전트의 ebA-CBD 참조 모델을 기반함으로써, 분석된 에이전트 영역의 컴포넌트 식별을 쉽게 함으로써 개발에 효율성을 제공한다. 또한, 스펙을 제공함으로써 적용가능한 컴포넌트 선택의 가이드라인이 되고, 컴포넌트 생성을 위한 기반 모델로써 재사용 된다. 향후 연구로써는 ebA-Spec.을 기반으로 컴포넌트의 개발과 조립에 관한 상세한 방법 정의가 요구되며, 구현 사례를 통해 기존 방법과의 비교, 검증이 필요하다.

참고문헌

[1] OMG Agent Platform Special Interest Group, "Agent Technology Green Paper", http://www.objs.com/agent/, 2000.

[2] Ho-Jun Shin, Haeng-Kon Kim, "CBD Reference Architecture through E-Business Agent Classification," Proceedings 2nd International Conference on computer and Information Science, 2002.

[3] Martin L. Griss, "Agent-Mediated E-Commerce Agents, Components, Services, Workflow, UML, Java, XML and Games...", Proceedings of the Technology of Object-Oriented Languages and System, Keynote Presentation, 2000.

[4]Hyacinth S. Nwana, "Software Agents:An Overview", Knowledge Engineering Review, VOl. 11, No 3, pp. 1-40, 1996.

[5] 김행곤외, "컴포넌트 저장소 형상관리 시스템에 관한 연구", ETRI 최종연구보고서, 2000.