

EJB 컴포넌트의 모델링 및 생성 지원 도구의 설계

김민정⁰ 조진희 이우진 신규상
한국전자통신연구원 소프트웨어공학연구부 컴포넌트공학팀
(minjikim,chojh,woojin,gsshin@etri.re.kr)

Design of a supporting Tool for modeling and developing EJB Components

Min-Jung Kim⁰ Jin-Hee Cho Woo-Jin Lee Gyu-Sang Shin
Component Engineering Team, Software Engineering Department, ETRI

요 약

컴포넌트 소프트웨어 기술은 재사용성, 적시성, 유지 보수성 등이 업체의 경쟁력으로 대두되고 있는 정보 기술 업계에서 점차 각광을 받고 있다. 현재 다양한 컴포넌트 생성 지원 도구들이 제공되고 있지만 컴포넌트의 식별, 모델링, 코드 생성, 배치 등 전 개발 단계를 밀접히 연계하여 지원하는 도구가 드물다. 이 논문에서는 컴포넌트 생성에 연관된 모든 과정을 지원하는 컴포넌트 모델링 및 생성 지원도구의 설계를 제안한다. 컴포넌트 모델링은 영역 특성을 고려한 시스템 분할 관점에서 컴포넌트 플랫폼 아키텍처에 관계없이 논리적(logical) 컴포넌트의 식별 및 모델링을 지원한다. 코드 생성 부분은 플랫폼 아키텍처에 의존적이어야 하므로 이 논문에서는 우선적으로 EJB 컴포넌트의 코드 생성을 지원한다.

1. 서론

소프트웨어 산업이 급속하게 발전해감에 따라 정보 기술 업체간 경쟁이 더욱 심화되어 소프트웨어 재사용성, 적시성(time to market), 유지 보수성 등이 업체의 생명력으로 대두되면서 컴포넌트 소프트웨어 기술이 점차 각광을 받고 있다. 컴포넌트 소프트웨어 기술은 컴포넌트 단위로 독립적인 개발이 가능하므로 병행적/단계적 소프트웨어 개발을 통해 개발 기간을 단축시킬 수 있다. 그리고 컴포넌트의 수행은 SUN의 Enterprise JavaBeans(EJB) [1], OMG의 CORBA Component Model(CCM) [2], 마이크로소프트의 DCOM [3] 등의 컴포넌트 플랫폼 아키텍처상에서 이루어지므로 아키텍처가 제공하는 트랜잭션, 보안성, 지속성 등의 다양한 서비스를 이용할 수 있으며 또한 컴포넌트 단위로 수행되므로 컴포넌트의 대체 및 유지보수가 수월한 장점이 있다.

대표적인 컴포넌트 개발 지원 도구로는 Computer Associates사의 Cool:Joe, Compuware사의 Uniface, Select사의 SCF(Software Component Factory), TogetherSoft사의 Together 등이 있다. 이러한 도구들은 대부분 특정 컴포넌트 플랫폼 아키텍처만을 지원하고 있으며 몇몇 도구는 컴포넌트 모델링 및 코드 생성 부분이 밀접하게 연계되어 있지 않는 경우도 있다.

이 논문에서는 컴포넌트의 식별, 컴포넌트 모델링, 코드 생성, 배치 등 전체 과정을 지원하는 컴포넌트 생성 지원 도구를 설계한다. 그리고 컴포넌트 모델링에서는 영역 특성과 요구사항만을 고려하여 시스템을 분할하는 논리적인(logical) 컴포넌트 모델링과 특정 컴포넌트 모델에 맞춘 내부 컴포넌트 클래스 모델링으로 구분하여 컴포넌트의 이식성을 고려하고 있다. 이 논문에서는 EJB 컴포넌트 모델을 우선적으로 고려하여 논리적 컴포넌트를 EJB 컴포넌트로 생성, 배치하는 과정을 지원한다. EJB 컴포넌트의 배치(deployment)는 Java™ 2 Platform, Enterprise Edition (J2EE) [4]의 표준에 맞게 컴포넌트를 묶고 배치(deploy)하는 과정을 지원한다.

이 논문의 구성은 다음과 같다. 우선 제 2 장에서는 EJB 아키텍처에 대해 간략히 살펴본다. 제 3 장에서는 도구의 전체적인 기능 흐름을 활동도로 설명하고 제 4 장에서는 컴포넌트 모델링 및 생성 지원 도구의 각 구성 요소에 대해 간략히 설명한다. 마지막으로 제 5 장에서는 결론 및 향후연구방향을 제시한다.

2. Enterprise JavaBean (EJB) 아키텍처

Enterprise JavaBean (EJB) 아키텍처는 선(Sun)사에서 제안한 컴포넌트 기반 분산 비즈니스 어플리케이션의 개발 및 배치(deployment)를 위한 컴포넌트 아

키택처이다. 그림 1은 EJB 아키텍처의 일부 구성요소를 보여준다. EJB 컴포넌트는 홈/원격(Home/Remote) 인터페이스를 가지는 빈(bean)으로 이루어진다. 개발된 빈은 EJB 컨테이너에 배치(deploy)되어 동작한다. EJB 컨테이너는 트랜잭션(transaction), 명명(naming), 보안(security), 지속성(persistence) 서비스를 제공하는 EJB 서버상에 놓여 있다. EJB 서버에서 제공하는 다양한 기반 서비스들로 인해 빈 개발자는 빈이 제공하는 기능 위주로 개발하면 된다. EJB 서버가 제공하는 서비스의 선택은 배치자(deployer)가 빈을 EJB 컨테이너에 배치할 때 DD(Deployment Descriptor)에 기술하게 하여 빈의 이식성을 향상시킨다.

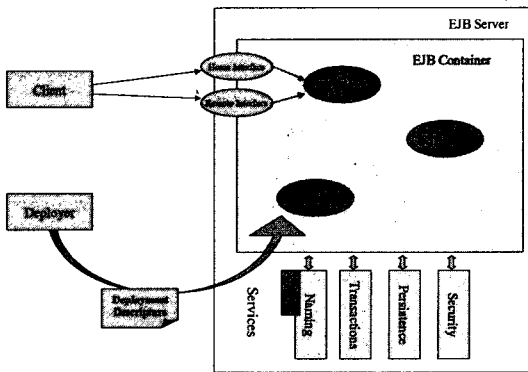


그림 1. EJB 아키텍처의 구성요소

외부의 클라이언트가 빈에 이용하기 위해서는 JNDI 명명 서비스를 이용하여 특정 빈의 홈 인터페이스에 접근하여 빈을 생성한 후 원격 인터페이스를 통해 빈이 제공하는 기능을 이용한다. EJB 빈의 유형에는 클라이언트와의 대화를 나타내는 세션(Session) 빈과 데이터베이스 데이터의 객체 뷰(view)를 제공하는 엔티티(Entity) 빈이 있다.

3. 컴포넌트 모델링 및 생성 프로세스

이 논문에서 제시하는 컴포넌트 모델링 및 생성 프로세스는 크게 논리적 컴포넌트의 식별 및 컴포넌트 인터페이스 명세 부분과 EJB 컴포넌트 모델링 및 생성 부분으로 나뉘어 볼 수 있다. 그림 2는 영역 모델 생성, 컴포넌트 식별 및 정의, 컴포넌트 구현 및 배치 과정을 지원하는 프로세스를 보여준다.

FOCUS 개발 방법론 [5]에서 제안된 공통성 및 가변성 분석 방법을 통해 얻어진 어플리케이션 도메인의 영역 정보를 바탕으로 use case 모델과 객체 모델을 생성한다. 객체 모델에 기술된 객체들의 연관 관계 정보와 use case에 나타난 객체들의 기능적 연관성 및 사용 횟수 등을 고려하여 논리적 컴포넌트 식별이 이루어진다.

식별된 논리적 컴포넌트들에 대해서 컴포넌트간의 의존성과 각 컴포넌트의 인터페이스를 명확히 정의하기 위해 컴포넌트 다이어그램과 순차도(sequence

diagram)를 작성하고 최종적으로 컴포넌트 인터페이스를 정의한다.

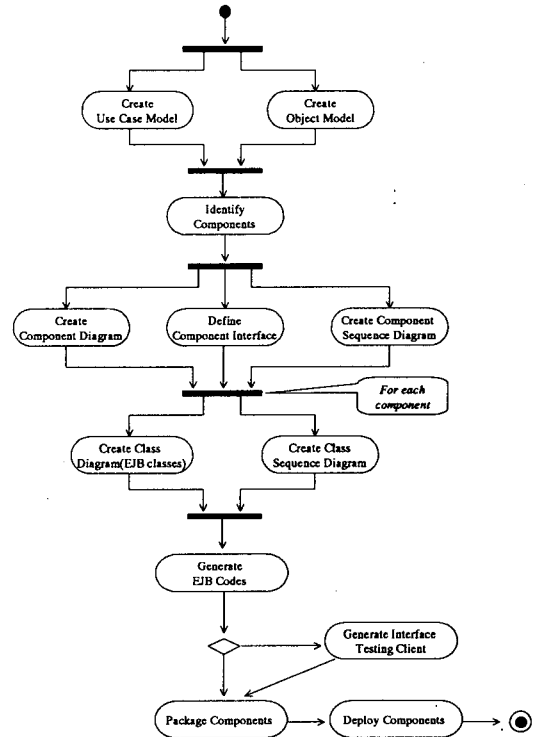


그림 2. EJB 컴포넌트 모델링 및 생성 도구의 활동도

컴포넌트 인터페이스가 정의되면 각각의 컴포넌트 별로 독립적으로 내부 설계 및 구현이 가능하다. 컴포넌트 내부설계 및 구현은 컴포넌트 플랫폼 아키텍처의 특성을 고려하여야 하므로 EJB, DCOM, CCM에 따라 조금씩 차이가 있다. 이 논문에서는 EJB 컴포넌트 설계 및 구현에 대해서만 언급한다. EJB 컴포넌트 내부 설계시에는 영역 분석에서 얻어진 객체들에 대해 EJB 특성에 맞게 클래스 다이어그램을 세분화하고 EJB의 홈/원격 인터페이스를 빈 클래스에 추가한다. 이때 세분화된 클래스의 인터페이스를 명확히 정의하기 위해 클래스간의 순차도를 작성하기도 한다.

EJB 컴포넌트의 내부 설계를 바탕으로 인터페이스 구현 코드의 자동 생성 및 클래스의 골격 코드 생성이 이루어지며 이들을 기반으로 하여 사용자가 비즈니스 로직을 직접 구현한다. 그리고 컴포넌트의 인터페이스의 기능 테스트를 원할 경우는 테스트 클라이언트 생성 과정을 거친다. EJB 컴포넌트를 어플리케이션 서버에 배치(deploy)하기 위해서는 먼저 DD 파일을 생성하고 Ejb-jar 파일로 컴포넌트를 묶어야 한다.

4. 컴포넌트 모델링 및 생성 지원 도구의 설계

컴포넌트 모델링 및 생성 지원 도구의 구성요소는

그림 3 에 나타나 있다. 각 구성요소에 대한 간략한 설명은 다음과 같다.

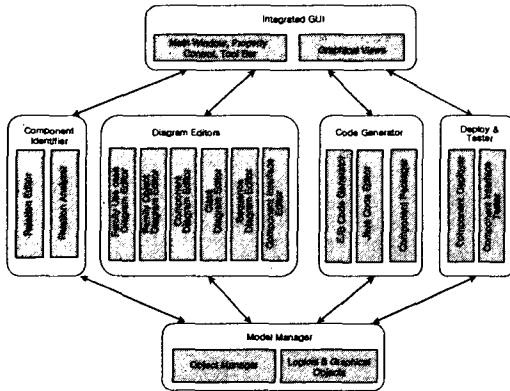


그림 3. EJB 컴포넌트 모델링 및 생성 도구 구성도

4.1 다이어그램 편집기(Diagram Editors)

다이어그램 편집기는 크게 영역 분석을 위한 모델러와 컴포넌트 설계를 위한 모델러로 구성된다. 영역 분석을 지원하기 위해 패밀리 유스케이스 다이어그램 편집기 및 패밀리 객체 다이어그램 편집기를 제공하며, 논리적 컴포넌트 모델링을 위한 컴포넌트 다이어그램 편집기, 인터페이스를 식별하기 위해 사용되는 순차도 작성기, 컴포넌트 내부 설계를 위한 클래스 다이어그램 편집기, 인터페이스 명세를 위한 컴포넌트 인터페이스 편집기로 구성된다.

4.2 컴포넌트 식별기(Component Identifier)

컴포넌트의 식별은 영역 객체 다이어그램의 객체간의 구조적 관계 정보와 use case 에 나타난 기능적 정보를 바탕으로 객체 의존 그래프로 생성한 후, 응집도 등을 고려한 결합 알고리즘을 수행하여 후보 논리적 컴포넌트들을 생성한다.

4.3 코드 생성기(Code Generator)

코드 생성기는 컴포넌트 내부의 EJB 클래스들의 설계 정보를 바탕으로 EJB 컴포넌트의 홈/원격 인터페이스와 빈 클래스와 도움 클래스들의 골격 코드를 생성한다. 생성된 골격 코드를 바탕으로 프로그래머는 Java 편집기를 이용하여 비즈니스 로직을 구현한다. 그리고 특정 EJB 서버에 독립적인 컴포넌트의 배치 정보를 지정하기 위해 워저드 방식의 설정기를 지원하여 손쉽게 DD 파일을 생성할 수 있다.

4.4 배치기 및 시험기(Deployer & Tester)

배치기는 특정 EJB 서버에 맞게 DD 파일을 수정할 수 있게 하고 실제 컴포넌트를 서버에 배치하는 과정을 지원한다. 그리고 테스터는 컴포넌트의 인터페이스를 웹 환경에서 테스트할 수 있는 클라이언트를 자동으로 생성해 준다.

4.5 모델 관리자(Model Manager)

모델 관리기는 영역 및 컴포넌트 모델러의 편집

대상이 되는 각 다이어그램의 논리적인 정보 뿐만 아니라 화면상에 다이어그램을 표현하기 위한 정보들의 관리기능을 제공한다. 특히, 단계별 산출물을 유스케이스 모델, 객체 모델, 컴포넌트 모델, 구현 모델, 배치 모델로 나누어 관리하게 되며, 각 편집기들은 객체 관리기를 통해 다이어그램 구성요소 객체에 접근하게 된다.

4.6 통합(integrated) GUI

현재 개발중인 통합 GUI 는 모델링 도구, EJB 코드 생성기, 배치 도구, 테스트 도구들이 통합된 형태로 각 단계별 산출물을 트리 형식으로 관리하게 된다. 그림 4 는 본 도구의 상위 단계 GUI 로서 왼쪽 창은 모델 정보 관리기를 나타내고 오른쪽 창은 다이어그램 편집 창을 나타낸다.

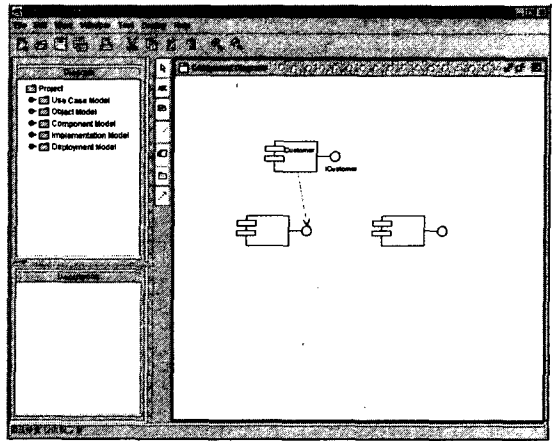


그림 4. 컴포넌트 모델링 및 생성 지원도구의 통합 GUI

5. 결론 및 향후연구방향

컴포넌트 모델링 및 생성 지원 도구는 컴포넌트의 식별, 모델링, 코드 생성, 배치 등이 밀접히 연관된 개발 전과정을 지원하고 있어 효율적인 컴포넌트 생성이 가능하다. 현재 설계를 바탕으로 Java 로 각 세부 기능들을 개발하고 있으며 이들을 취합하여 전체 지원 도구의 프로토타입을 생성할 계획이다.

참고문헌

- [1] Sun, *Enterprise JavaBeans Specification, Version 1.1*, Sun Microsystems Inc, 1999
- [2] OMG, *CCM Revised Submission*, OMG TC Document orbos/99-07-01, 1999.
- [3] Microsoft, *DCOM Technical Overview*, URL: http://msdn.microsoft.com/library/backgrnd/html/msdn_dcomtec.htm, 1996.
- [4] Sun, *Designing Enterprise Applications with the Java™ 2 Platform, Enterprise Edition, Version 1.0*, Mar. 2000.
- [5] 김수동, *컴포넌트 생성 및 추출을 위한 개발 방법론 II*, 한국전자통신연구원 위탁과제 중간보고서, 2000.