아키텍쳐 스타일 기반의 컴포넌트 조립 및 지원도구의 개발

이승연 1) 이지원 권오연 신규상
한국전산통신연구원, 컴퓨터공학 연구팀
(coral, jihyun, ookwon, gsshin@etri.re.kr)

Component Assembly Based on an Architecture Style and Development of a CASE Tool

Seung-Yun Lee 2) Ji-Hyun Lee Oh-Cheon Kwon Gyu-Sang Shin
Component Engineering Research Team, ETRI

요 약
다양한 응용 소프트웨어를 개발함에 있어 컴포넌트 기반의 개발기술이 퍼고게 발전하고 있는 가운데, 실제로 개발된 컴포넌트들을 재사용하고, 이들을 조립하여 지달한 시스템을 개발하기라는 노력이 주목받고 있다. 컴포넌트의 개발과 컴포넌트 조립을 통한 응용 소프트웨어 시스템 개발을 위해서는 이들이 서로 정확하게 결합하여 작동할 수 있는 아키텍처 기반의 조립환경이 지원되어야 한다. 본 논문은 Enterprise JavaBeans (EJB)의 조립을 지원하는 아키텍처 기술론 및 이의 지원도구에 대하여 기술한다. 아키텍처 스타일의 하나인 Chiron-2(C2)를 기반으로 아키텍처를 기술하고, 이를 다이어그램 형식과 텍스트 형식으로 표현할 수 있는 아키텍처 모델링의 구조 및 이의 프로토타입에 대해 설명한다.

1. 서론

본 논문은 컴포넌트의 개발과 컴포넌트 조립에 의한 응용 소프트웨어의 개발을 효과적으로 지원하기 위하여 EJB 기반의 컴포넌트를 개발하고, 이를의 CASE 도구의 프로토타입을 기술한다. EJB의 조립을 지원하는 아키텍처 기술론인 Chiron-2(C2)에 기반하여 정의하고 이를 다이어그램 형식과 텍스트 형식으로 기술하고 구현, 탐색 및 형태의 모델링을 지원하는 아키텍처 모델러의 프로토타입에 대해 설명한다.

2. 관련연구

2.1 컴포넌트 조립
컴포넌트 조립 (Component Assembly)은 컴포넌트 기반 소프트웨어 개발 (Component-Based Software Development: CBSD)의 구현적 목표로서, 소프트웨어를 구성하는 단위 컴포넌트들은 조립을 통해 다른 컴포넌트들과 상호 작용한다. CBSD에 있어 컴포넌트 조립은 같은 모듈을 기반으로 개발된 단위 컴포넌트간이나 이중 모듈 컴포넌트간의 아키텍처선 레벨 조립과, 이중 서비스로 전개되어 있는 컴포넌트간의 시스템 레벨 조립, 그리고 조립된 컴포넌트간의 상위레벨에서의 조립 또한 포함된다[3]. 이러한 조립을 지원하기 위해서 컴포넌트는 인터페이스 (Interface)를 기반으로 결합된다. 인터페이스는 정보 은닉 (Information Hiding) 원리에 따라 컴포넌트의 자료를 독립적으로 사용할 수 있도록 지원한다. 그러나, EJB나 COM+ 같은 컴포넌트 모델에서의 인터페이스는 컴포넌트의 모듈 기반에 대한 표현을 제공하지만, 컴포넌트의 비기능적인 속성들의 환상화는 제공하지 못한다. 또한, JavaBeans나 WaterBeans에서 표현되는 컴포넌트의 응용 환경이 제공되지 않고, 'Plug-and-Play' 방식의 손쉽은 조립을 할 수 있는 기술이 제공하지 않아야 한다. 이를 구축하기 위하여 아키텍처 레벨에서의 지원화는 주상위와 인터페이스를 제공해야 하며, 컴포넌트간의 메소드 호출방식

370
2.2 아키텍처 스타일(Architectural Style)

아키텍처 스타일은 자주 사용되는 관용적인 소프트웨어 시스템의 구조 패턴을 의미하여 컴퓨터과 커뮤니들 모델링한다. 아키텍처 스타일은 구조적인 구성 패턴을 공유하는 시스템들의 집합을 정의하며, 아키텍처의 설계에 사용되는 컴퓨터 태입, 커넥트 태일, 그리고 이들의 상호 연결 패턴에 대한 규칙들과 재료 조건들을 정의한다. 

架[4]. 아키텍처 스타일의 예로는 작업이나 데이터의 흐름을 나타내는 pipelines, pipe-and-filter, 컴퓨터의 상호작용을 나타내는데 온라인 event 방식이고, 여러 동신 프로세스, 데이터 저장 및 역할을 수행을 분할할 수 있도록 지원하는 blackboard 방식 등이 있으며, 현재 이러한 아키텍처 스타일 기반의 다양한 아키텍처 기술연구가 개발되고 있다 [5].

아키텍처 스타일은 EJB를 아키텍처 스타일의 경우 미국의 UCI 대학에 연구 개발중인 스타일로 메시지 기반의 계층 아키텍처를 나타낸다. 모든 컴퓨터와 거더는 top과 bottom을 가지고 있어 EJB 컴퓨터들은 사이의 통신은 커넥터를 통해 이루어지며 이메일에는 top, bottom 포트를 통해 일정한 규칙을 따라 상호작용할 수 있다. C2 컴퓨터의 구조는 EJB 컴퓨터들을 독립적으로 만들 수 있는 wrapper를 생성하여 컴퓨터들간의 조립을 지원하는데 용용할 수 있다.

3. 컴퓨터 조립을 지원하는 아키텍처 기술연구 및 지원 환경

본 장은 조립을 돕는 EJB 컴퓨터간의 상호작용을 유연하게 지원하는 모델과 이를 기술하는 언어, 이외 설계 및 구현을 지원하는 도구의 환경에 대해 기술한다.

3.1 C2 Style을 기반으로 하는 아키텍처 기술연구

EJB 컴퓨터의 조립을 지원하는 아키텍처는 C2가 기반의 스타일을 따르며, 각 컴퓨터마다 입력체 메시지와 관련된 wrapper를 생성하여 컴퓨터간의 예외적 요인과 함께 효율적이게 하겠다고 할 수 있다. 본 연구에서 컴퓨터의 상호작용을 개발하고 정의하는 아키텍처 기술연구는 각 컴퓨터의 독립적인 기술뿐만 아니라, 컴퓨터들과 거더들간의 조립 아키텍처도 가할 수 있다. 즉, 각 컴퓨터마다 독립적으로 작동하는 메소드 및 행위(behavior)가 별개하게 기술될 수 있으며, 세부적으로 기술된 컴퓨터들은 아키텍처 레벨에서 각 컴퓨터들의 상호작용을 추상적으로 표현할 수 있도록 지원한다. 또한, 정의하는 아키텍처 기술는 컴퓨터와 커넥터의 태일링이나 가능하며 기술하는 아키텍처의 타당성 결론은 보장한다. [그림 1]은 EJB 컴퓨터의 조립을 지원하기 위해 정의한 컴퓨터 및 아키텍처의 Semantic 모델이며 이는 아키텍처 모델을 기술하는 ADN(Architecture Description Notation)

과 컴퓨터 모델을 기술하는 IDN(Interface Definition Notation) 정의어로 나눌 수 있다.

그림 1 EJB 컴퓨터의 조립을 지원하는 아키텍처의 Semantic 모델

3.2 컴퓨터 조립 아키텍처 설계을 위한 지원환경

3.1에서 정의한 아키텍처 기술연구를 기반으로 컴퓨터를 조립할 수 있도록 지원하는 CASE 도구의 요구사항은 다음과 같다.

1) 편집 기능: C2 스타일을 기반으로 컴퓨터 조립을 지원하는 아키텍처는 다이어그램과 테스트를 이용하여 기술할 수 있다. 다이어그램 편집기에서 EJB 컴퓨터들을 조립이 높이 연결한 후, 특정 편집기를 사용하여 이들간의 조립 명세를 작성하고, 조립 아키텍처를 생성하거나, 테스트 기반의 편집기에서 EJB 스타일을 따르는 아키텍처 기술연구를 이용하여 조립 아키텍처를 기술할 수 있다. 동일한 어플리케이션 구축을 위해 작성된 컴퓨터 조립 아키텍처의 다이어그램 기술내용과 테스트가 컴퓨터를 기술내용은 상호 일치해야 한다.

2) 모델 검사 기능: 다이어그램이나 테스트로 작성된 아키텍처 모델은 구문검사와 의미검사를 통해 문법 및 의미검사를 수행할 수 있다. 구문 검사는 아키텍처 모델이 문법에 맞게 작성되었는지에 끌어들이며 의미검사는 타입검사와 형체(topology) 검사는 포함된다. 타입 검사를 아키텍처 모델의 메시지의 타입규칙에 맞게 작성되었는지를 끌어들이며 형체 검사와 아키텍처의 스타일을 규정하는 규칙에 따라 아키텍처 구성요소가 연결되고 구조화되어있는지를 따져다.

3) 아키텍처 모델의 컴퓨터 모델의 변환 기능: 아키텍처를 구성하고 있는, 새로운 컴퓨터의 필요할 경우에는, 이를 생성할 수 있도록 아키텍처 모델을 컴퓨터 모델로 변환할 기능이 필요하다. C2 기반의 아키텍처 분석 기법과 UML
기반의 컴포넌트 설계 및 구현 방법을 접목하여 앱

액터 모델의 모든 앱액터 구성요소가 컴포넌트 모델의 EJB 컴포넌트로 변환될 수 있다.

위의 요구사항을 만족하기 위해 포함되어야 하는 앱액터 기반 조립단위는 기존 구성은 크게 컨텍스, 모델링,기간, 컴포넌트 모델 변경기로 나눌 수 있다. 변경기들은 앱액터

데이터베이스 관련 테이블과 ADL 관련의 두 종류가 있으며 두 변경기들의 동기화를 맞추기 위한 작업이 필요하다. 모델링기는 변경기에 작성된 앱액터의 구조 및 타입간

사를 담당하며 컴포넌트 모델 변환기는 앱액터 모델의 UML 기반의 컴포넌트 모델로 변환할 때 이용된다.

4. EJB 컴포넌트 조립 지원 앱액터 모델의 구현

3장에서는 EJB 컴포넌트의 조립을 지원하기 위하여 C2 앱액터 모델을 응용하여 앱액터 기술연어를 정의하였고 이를 지원하는 도구의 기능을 살펴보았다. 이를

바탕으로 본 장에서는 실제 EJB 컴포넌트들의 조립을 지원하는 앱액터 모델의 구현 및 이론의 프로토타입 구현에 대해 기술한다.

[그림 2]는 앱액터 모델의 전체 구성도 및 서브 모듈간의 관계를 나타낸 것이다.

그림 2 앱액터 모델의 전체 구성도 및 서브 모듈간의 관계도

각 변경기는 M-V-C의 구조로 따르며 이들간의 모델링의

변환 및 동기화 지원에 의해 가능하다. EJB 컴포넌트의

조립을 지원하기 위하여 생성 푸블리셔로부터 기존의 EJB Jar

파일을 놓거나 새로운 앱액터 요소들을 푸블리시로부터 생성할 수 있으며 컴포넌트 조립을 위해 생성결합기

와 생성 EJB 생성기의 기능을 이용하여, 3장에서 정의한

IDN를 작성하여 컴포넌트 명세를 기록하고 이로부터 생성

되던 IDN을 이용하여 전체 앱액터를 구성하는 EJB 조

립 컴포넌트로 생성한다. 작성된 컴포넌트 명세 및 앱액터

명세는 모델 검사기를 통한 검사할 수 있으며, 조립을 위해

하여 새로운 모델을 배내는 컴포넌트는 UML 변경기의 기능

을 이용하여 UML 기반의 컴포넌트 모델로 변환함 수 있다.

[그림 3]은 EJB 컴포넌트 조립을 지원하는 앱액터 모델의 수행 화면이다.

그림 3 앱액터 모델러

5. 결론

본 논문은 EJB 기반의 컴포넌트 개발과 조립에 의한

동형 소프트웨어 개발의 효과적 지원하기 위해 앱

액터기술 기반으로 컴포넌트들이 서로 정렬하게 결합하여 작동할 수 있도록 지원하는 컴포넌트 조립기

장을 제안하고 이를 구성한 앱액터 모델의 프로토타

입을 기술하였다. 향후 본 연구에서 제안한 앱액

터 기술 연어의 확장이 필요하며 같은 모델의 컴포넌

트 조립뿐만 아니라 이중 모듈간의 조립 및 이중 서

비에 전개되어 있는 서비터래트의 조립 또한 지원될 수

있는 도구의 확장이 필요하다.

6. 참고문헌


Component Builder’s Guide, Pittsburgh, Pa.: Software

Engineering Institute, Carnegie Mellon University,

1999.


N. Taylor, A Language and Environment for

Architecture-Based Software Development, In the

Proceedings of the 21st International Conference on

Software Engineering (ICSCE 21), Los Angeles, CA.


John R., Robert S. and Kurt W., Technical Concepts of

Component-Based Software Engineering, Pittsburgh,

Pa.: Software Engineering Institute, Carnegie Mellon

University, 2000.


Proceedings of PrecOZ, 94 Conference. PrecOZ, Addison


[5] Mary Shaw, David Garlan, "Software Architecture -

Perspectives on and emerging discipline", Prentice-

Hall, 1995.