

컴포넌트 기반 시스템 개발을 위한 방법론에 관한 연구

이지현* 김진삼 박창순

한국전자통신연구원 임베디드 S/W 기술센터 S/W 공학연구팀

e-mail : {jihyun, kjs, cpark}@etri.re.kr

Methodology for Component-Based System Development

Jihyun Lee* Jin-Sam Kim Chang-Soon Park

S/W Engineering Research Team, Embedded S/W Technology Center, ETRI

요 약

컴포넌트 기반 시스템의 개발 생산성과 유지 보수성을 향상시키기 위해 기존에 생성된 소프트웨어 컴포넌트를 재사용할 수 있는 컴포넌트 기반 시스템 개발 방법론을 제시하고 종합적이고 체계적인 형태로 재사용 프로세스를 구성하고 각 단계별로 필요한 작업과 핵심 산출물을 살펴보도록 한다.

1. 서론

비즈니스 시스템의 규모가 대형화되는 환경에서 엔터프라이즈 급의 시스템을 구축하기 위한 종합적이고 체계화된 방법이 요구되었다. 소프트웨어 컴포넌트의 등장을 통해 80 년대에 확산되던 정보공학 방법론과 객체지향 방법론 외에 컴포넌트 기반 소프트웨어를 생산하기 위한 컴포넌트 기반 소프트웨어 개발 방법론 [1]이 점차 부각되어 왔다. 현재 쓰이고 있는 대표적인 컴포넌트 기반 소프트웨어 방법론에는 Rational 사의 RUP(Rational Unified Process) [2], Computer Associates 사의 CBD96 [3], Compuware 사의 UNIFACE [4], PrinceTonSoftech 사의 Select Perspective [5] 등이 있다. 이들 방법론들은 UML(Unified Modeling Language)에 기반하여 모델링하고 있다는 공통점이 있지만 많은 기업들이 컴포넌트 기반 방법론을 도입하려 할 때 방법론마다 적용 분야가 정해져 있거나 개발하고자 하는 시스템의 크기가 달라 회사의 업무나 기반 환경에 직접 적용하기가 어려웠다. 따라서, 기존 시스템에 대한 확장성을 높이고 체계적으로 시스템 구축 방법을 제공할 필요가 있다. 본 논문에서는 컴포넌트 기반 시스템을 개발하기 위해 필요한 작업을 크게 컴포넌트

생성과 컴포넌트 조립이라는 두 가지 프로세스로 나누고 각 프로세스에서 수행되어야 하는 하위 업무를 갖춘 방법론을 제시한다. 더불어 각 단계별 산출물을 간략하게 제시하여 기업의 요구 사항이 다양화 되더라도 기존 결과물에 기반하여 확장성 있게 진행해 나가고 방법론의 적용을 통해 보다 쉽게 프로젝트를 진행해 나갈 수 있도록 단계별 수행 산출물을 제시하고자 한다.

본 논문은 제 2 장에서 컴포넌트 기반 개발 방법론을 구성하는 요소들에 대해 분류하고, 제 3 장에서 컴포넌트 개발 시스템 개발을 위한 두 가지 프로세스들을 기술한다. 4 장에서는 개발 프로세스를 구성하는 개발 단계와 각 단계별 산출물을 제시함으로써 컴포넌트 기반 시스템을 구현할 때 필요한 작업과 시스템 구성을 위한 재사용 가능한 결과물을 기술한다. 마지막으로 5 장에서 결론을 맺는다.

2. 컴포넌트 기반 개발 방법론의 구성 요소

일반적으로 컴포넌트 기반 시스템을 구축하기 위해 여러 단계와 절차가 필요하다.

본 논문에서 컴포넌트 기반 시스템을 개발하기 위

한 방법론을 프로젝트, 프로세스, 단계, 활동, 작업, 절차로 구분한다. 이들 요소들은 그림 1 에서와 같이 보이고, 각 구성 요소에서 수행해야 하는 기능을 바로 아래에 다음과 같이 기술한다.

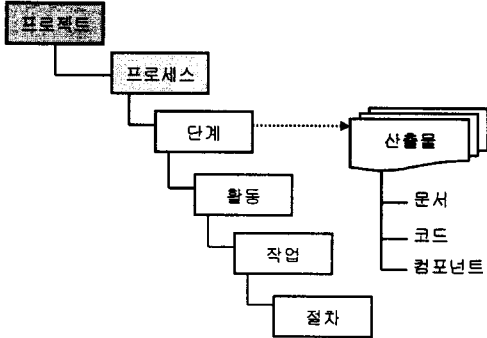


그림 1. 컴포넌트 기반 개발 방법론의 구성 요소

- **프로젝트**
분석된 요구 사항이 반영된 구축하고자 하는 시스템을 의미한다.
- **프로세스**
프로젝트를 수행하기 위한 병렬적이고 선택적으로 적용 가능한 수행 업무이다.
- **단계**
프로젝트를 수행할 때 의사 결정의 기점이 되는 단위이다.
- **활동**
단계를 구성하는 단위로 논리적으로 관련된 업무들의 집합이다.
- **작업**
개발자가 실제로 수행해야 하는 최소 업무 단위로 실제 입력물과 출력물이 정의된다. 작업의 결과는 단계별 산출물을 구성한다.
- **절차**
작업을 수행하기 위한 단위 순서를 의미한다.

산출물은 각 작업을 수행한 결과로 생성되는 문서, 구현 코드, 컴포넌트와 같은 결과물로 컴포넌트 기반 시스템 개발 방법에서 향후 다른 프로젝트를 수행할 때 재사용할 수 있게 저장되고 관리한다.

3. 컴포넌트 시스템 개발 프로세스

컴포넌트 기반 시스템 개발 방법론에서는 컴포넌트를 시스템의 구성 단위로 사용하여 시스템의 생산성과 유지 보수성을 높이고자 한다.

본 논문에서 제시하는 컴포넌트 기반 시스템 개발 방법은 프로젝트의 수행 과정을 크게 컴포넌트 생성 프로세스와 컴포넌트 조립 프로세스로 나눈다.

컴포넌트 생성 프로세스는 재사용 컴포넌트로부터 시스템의 기능 구현에 적합한 컴포넌트를 찾고, 찾을 수 없는 경우에 신규 컴포넌트를 생성하는데 필요한

작업들을 수행토록 한다.

컴포넌트 조립 프로세스는 기존의 컴포넌트들을 이용하여 기능과 규모 면에서 확장된 컴포넌트 기반 시스템을 구현하거나 단위 컴포넌트들을 통합하여 조립 컴포넌트를 보다 규모 있게 생성하기 위한 종합적인 업무들을 제공한다.

컴포넌트 기반 시스템을 구성하기 위해 두 가지 프로세스에서 단계적으로 수행되는 업무들은 표시하면 다음 표 1 과 같다.

표 1. 컴포넌트 기반 시스템 개발을 위한 각 프로세스별 구성 업무 (참조: O 는 수행을 의미, X 는 수행하지 않음을 의미)

단계적 업무	생성 프로세스	조립 프로세스
요구 분석	O	O
아키텍처 정의	X	O
아키텍처 참조	O	O
단위 컴포넌트 설계	O	X
단위 컴포넌트 구현	O	X
컴포넌트 개조	X	O
컴포넌트 전개	O	O
컴포넌트 시험	O	O
합성 컴포넌트 구현	X	O
컴포넌트 관리	O	O

컴포넌트 생성 단계는 신규 컴포넌트가 다른 컴포넌트와 연결되기 위해 필요한 위치와 행위 정보를 얻기 위해 아키텍처를 참조하는 업무 이외에 요구 분석, 단위 컴포넌트 설계/구현, 컴포넌트 전개/시험/관리의 추가적인 업무로 구성된다. 일반적으로 각 업무들은 순차적으로 이루어지고, 경우에 따라 선택적으로 선별된 업무들이 병렬적으로 수행됨으로써 새로운 컴포넌트를 생성하도록 한다.

컴포넌트 조립 단계는 기존에 이미 생성되어 있는 컴포넌트들을 재사용하여 사용자의 요구 사항에 맞는 시스템으로 조립하는 것을 목표로 한다. 조립 프로세스는 조립 시스템에 대한 요구 분석, 아키텍처 정의, 개조를 통한 재사용 컴포넌트의 테일러링, 컴포넌트 전개/시험/조립/관리와 같은 일련의 업무들로 구성된다.

각 프로세스를 구성하는 '단계적 업무'는 방법론의 구성 요소인 '단계'로 매핑 가능하다. 단계 별 작업에 관한 자세한 설명은 다음 장에 기술한다.

4. 컴포넌트 시스템 개발 단계와 산출물

본 논문은 컴포넌트 기반 시스템을 개발하기 위해 필요한 단계를 시스템 개발 관점에 따라 요구 분석 단계, 아키텍처 정의 단계, 단위 컴포넌트 설계 단계, 단위 컴포넌트 준비 단계, 컴포넌트 전개 단계, 컴포넌트 시험 단계, 컴포넌트 조립 단계, 컴포넌트 공급 단계의 총 8 개의 단계로 나눈다.

4.1. 요구 분석 단계

요구 분석 단계는 비즈니스 시스템을 사용하는 사용자가 필요로 하는 요구 사항을 이해하고 충분히 표현할 수 있도록 업무 파악에 필요한 자료를 수집하고 정리하는 업무를 포함한다.

요구 분석 산출물은 OMG 에서 공표한 모델링 표준 언어인 UML 을 사용하여 시스템 개발에 관련된 설계자, 개발자, 관리자들이 시스템 요구 사항에 대한 이해를 같이 하도록 한다. 요구 분석 산출물로는 유스케이스 다이어그램과 유스케이스 별 시나리오를 구성하는 유스케이스 모델 기술서를 생성하도록 한다.

4.2. 아키텍처 정의 단계

아키텍처 정의 단계는 사용자의 요구 사항에 기반해 시스템 아키텍처와 소프트웨어 아키텍처를 설계하고 설계된 아키텍처에서 필요한 재사용 가능한 컴포넌트들을 식별하는 업무를 포함한다. 특히, 아키텍처 정의 단계에서는 컴포넌트 기반의 시스템 개발하는데 필요한 논리적/물리적 구조를 명시하고 컴포넌트 뿐만 아니라 아키텍처 또한 재사용될 수 있도록 관리한다.

아키텍처 정의 단계에서는 운영 시스템과 데이터베이스 시스템 등이 포함된 시스템 아키텍처, 컴포넌트들간의 상호 연동에 필요한 소프트웨어 계층, 컴포넌트 시스템의 구성을 표현한 소프트웨어 아키텍처, 물리적인 패키징 구조를 정의한 정의서가 생성되도록 한다. 그리고, 시스템을 구성하는 객체들이 객체 모델 기술서에 포함되도록 산출물을 작성한다.

4.3. 컴포넌트 설계 단계

컴포넌트 설계 단계는 '아키텍처 정의 단계'에서 식별된 컴포넌트들 간의 상호 관계를 정의하고 정제함으로써 신규 컴포넌트의 구현 단위를 세분화한다.

신규 컴포넌트의 기능에 따라 내부 구조와 인터페이스를 구현 수준까지 설계하고 설계 기술 문서에 명시한다. 재사용 가능한 컴포넌트 중에 '요구 분석 단계'에서 수집된 요구 사항을 반영하기 위해 변경되어야 하는 컴포넌트를 4.4 절에서 설명될 '단위 컴포넌트 준비 단계'에서 개조하도록 지원한다. 개조 요구 사항은 설계 기술 문서에 추가적으로 정의하도록 한다.

본 단계의 산출물에는 컴포넌트의 기능과 인터페이스를 명시한 컴포넌트 명세서와 특정 플랫폼에서 수행하도록 선별된 트랜잭션 및 보안에 관한 정의서가 포함되도록 한다. 그리고, 컴포넌트를 배치할 때 필요한 관련 컴포넌트 식별명, 트랜잭션 처리를 위한 설정값 등을 포함하는 전개 기술서와 구현된 컴포넌트를 시험하기 위한 시험 설계서를 함께 포함하도록 한다.

4.4. 단위 컴포넌트 준비 단계

단위 컴포넌트 준비 단계는 '컴포넌트 설계 단계'에서 정의된 명세대로 컴포넌트를 구현하는 업무를 포함한다. 구현된 컴포넌트는 어플리케이션 서버에 배치하기 위한 '컴포넌트 전개 단계'를 거치고, 개조 요구 사항을 갖는 재사용 컴포넌트는 본 단계에서 컴포넌트

트 내부 구현을 변경할 수 있도록 지원한다.

단위 컴포넌트 준비 단계에서는 '컴포넌트 설계 단계'에서 작성된 컴포넌트 명세서, 전개 기술서, 시험 설계서의 내용을 단위 컴포넌트의 구현과 개조에 따라 수정 보완한다.

4.5. 컴포넌트 전개 단계

컴포넌트 전개 단계는 구현이 완료된 코드를 전개 기술서에 기반하여 어플리케이션 서버에 배치할 수 있도록 정해진 특정 플랫폼에 맞게 컴포넌트로 패키징하고 어플리케이션 서버에 배치하여 컴포넌트로 동작하도록 준비하는 업무를 포함한다. 부가적으로 기술할 사항이 생기면 '컴포넌트 설계 단계'에서 마련된 전개 기술서를 참조하여 추가 수정 사항은 보완한다.

4.6. 컴포넌트 시험 단계

컴포넌트 시험 단계는 컴포넌트의 기능을 검증하기 위해 클라이언트 프로그램을 생성하고 컴포넌트에 대한 구현을 검증하는 업무를 포함한다.

컴포넌트의 동작 중에 예외 현상이 발생하거나 코드를 추가적으로 수정할 상황이 발생했을 때 생성된 컴포넌트가 컴포넌트 명세서와 시험 설계 문서에 정의된 대로 생성되었는지를 검증하도록 한다.

컴포넌트 시험은 구현 코드의 변경 범위를 정하고 변경된 내용을 코드에 반영함으로써 이루어지고, 수정된 변경 사항들은 설계 기술 문서에 기록하여 구현과 설계 내용이 일치하도록 한다.

요구 사항에 맞게 새롭게 구현된 신규 컴포넌트나 개조 요구 사항에 맞춰 개조 변경된 컴포넌트들은 설계 기술 문서와 함께 재사용을 위해 컴포넌트 저장소에 저장되도록 한다.

4.7. 컴포넌트 조립 단계

컴포넌트 조립 단계는 아키텍처 정의의 정보와 준비된 단위 컴포넌트들에 기반하여 컴포넌트들을 합성된 컴포넌트로 조립하는데 필요한 코드를 구현하는 업무를 포함한다. 조립된 컴포넌트는 내부적으로 상호 연관 관계가 존재하고 외부로는 독립적인 한 개의 컴포넌트로 구성된다.

조립 컴포넌트는 '컴포넌트 전개 단계'를 통해 어플리케이션 서버에 전개되고 '컴포넌트 시험 단계'를 거쳐 통합 시험된다. 검증된 조립 컴포넌트는 단위 컴포넌트와 마찬가지로 컴포넌트 명세서나 관련 아키텍처 정의서와 같은 컴포넌트 설계 기술 문서와 함께 컴포넌트 저장소에 저장된다.

4.8. 컴포넌트 관리 단계

컴포넌트 관리 단계는 '컴포넌트 구현 단계'와 '컴포넌트 조립 단계'에서 생성된 컴포넌트들을 컴포넌트 명세에 관한 컴포넌트 명세서, 아키텍처 정의서와 같은 설계 기술 문서와 함께 저장하고 관리하는 업무를 포함한다. 따라서, 유사한 컴포넌트와 관련 있는 확장된 요구 사항이 있는 경우 유사 도메인에 관한 컴포넌트 기반 시스템을 구축하기 위해 재사용될 수

있다.

재사용 가능하게 컴포넌트들을 체계적으로 관리하기 위해 컴포넌트 저장 체계 및 컴포넌트 관리를 산출물로 생성하도록 한다.

5. 결론 및 향후 연구

본 논문에서는 컴포넌트 기반 시스템을 개발하는데 필요한 작업을 컴포넌트 생성과 컴포넌트 조립의 두 개의 프로세스로 구분하여 각 프로세스를 구성하는 단계들과 단계 별로 산출되어야 하는 결과물에 대해 제시하였다.

컴포넌트 기반 시스템을 구축하기 위한 개발 방법론은 개발자의 관점에 따라 순차적이거나 병렬적이며 선택 가능한 작업들로 구성될 수 있어야 한다. 그리고, 규모가 큰 방법론에서 흔히 나타나는 “개발자들이 실제로 방법론을 이용하기 어렵다”는 문제점을 해결할 수 있어야 한다.

본 논문에서 제시하고 구분한 생성 프로세스와 조립 프로세스, 그리고 각 프로세스 별 단계적 업무는 시스템을 개발하는데 필요한 설계 및 구현 단계들이 확장성 있고 점진적으로 수행될 수 있도록 지원한다.

그리고, 각 단계 별 산출물을 명명하여 프로젝트 진행 및 관리에 필요한 산출물을 제시한다.

본 연구는 프로젝트의 규모에 따라 프로젝트 준비, 수행, 종료에 필요한 작업과 단계 별 지침을 다듬는 작업 중에 있으며 사례 적용을 통해 개선시키는 연구가 진행 중에 있다.

참고문헌

- [1] Katharine Whitehead, “Component-Based Development: Principles and Planning for Business Systems”, Addison-Wesley, 2002.
- [2] “Rational Unified Process”, Rational Software Corporation, <http://www.rational.com/products/rup/index.jsp>.
- [3] “CBD96”, Computer Associates, <http://www.sterling.com/dbdedge1/dbd96.htm>
- [4] “UNIFACE Development Methodology”, <http://www.compuware.com>
- [5] “Select Perspective”, White Paper, Princeton Sofitech, 2000.