

3계층 표현 방식 아키텍처에서 UML 기반 컴포넌트를 이용한 시스템 모델링 기법

임성빈⁰ 송치양 문창주 백두권

고려대학교 컴퓨터학과 소프트웨어 시스템 연구실
E-mail : bluesol@swwsys2.korea.ac.kr

The System Modeling Method with a UML-based Component in the architecture using Triple Layer Expression Formula

Sung Bin Yim⁰ Chee Yang Song Chang Joo Moon Doo Kwon Baik
Software System Lab., Dept. of Computer Science & Engineering, Korea Univ.

요 약

최근 들어 소프트웨어의 점점 복잡화, 대형화가 되고 있는 추세에 의해서 컴포넌트의 중요성이 부각되면서, 컴포넌트를 이용한 시스템의 모델링(Modeling) 기법들이 등장하고 있다. 그리고, 이런 모델링 기법들을 적용한 응용 분야가 점차 확대되어 가고 있는 가운데, 효과적으로 시스템 모델링을 할 수 있도록 하고자 기존의 복잡한 다 계층의 모델링 표현 방식의 구조를 탈피한 3계층 표현 방식의 아키텍처를 가진 모델링 기법을 제안한다. 그리고, 하나의 작은 예제를 통해서, 실제 적용되는 과정을 보면서, 기존의 방법과의 차이점에 대해서 관찰 및 분석, 그리고, 추후 적용의 기대효과에 대해서 알아본다.

1. 서 론

현재, 컴포넌트라는 시스템을 구성하면서, 물리적이고, 독립적이면서, 대체가 가능한 부품이라는 의미의 작은 단위의 소프트웨어 기술이 등장하고 있다. 그리고, 이 컴포넌트라는 기술을 적용하기 위해서 실행모듈, 라이브러리 파일, 도큐먼트, API(Application Programming Interface) 등을 처리 할 수 있는 모델링의 기술이 필요로 하게 되었다. 따라서, 이런 많은 요소들을 포함하고, 충족하기 위해서 복잡한 구조의 모델링 기법을 필요로 한다. 현재, 컴포넌트를 이용한 시스템을 모델링하기 위해서는 UI layer, Application Layer, Business Layer, Data Layer 등의 여러 세부영역으로 나누어진 여러 개의 층(layer)을 가지고 시스템의 분석 및 설계를 필요로 한다. 그리고, 이렇게 많은 계층 구조를 사용하다 보니 상호 계층간의 관계나 계층을 분리하는데, 명확한 구분이나 관계에서 어려움을 가지게 된다. 따라서, 본 논문에서 이런 어려움을 탈피하고 단순한 모델링 환경을 만들고자 컴포넌트를 이용한 모델링을 위한 3계층 표현 방식의 아키텍처를 제안한다.

2. 관련연구

(1) CBSD Process

CBSD(Component Based Software Development) Process는 현재 소프트웨어 공학 및 소프트웨어 산업분야에서 재사용(Reuse)의 개념을 담고 있는 컴포넌트라는 소프트웨어 개발 기술에 대해서 중요성이 점점 높아만 가고 있는 가운데, 이 컴포넌트 기술을 이용하여 효율적이고, 실용적으로 소프트웨어를 개발 할 수 있도록 컴포넌트 기반의 소프트웨어 개발 방법을 말한다. 즉, 이런 CBSD Process는 다양각색의 요구사항과 개발하는데 어려움과 복잡성, 그리고, 문제점들을 쉽게

해결 할 수 있는 표준화된 하나의 절차들로 이루어진다. 이런 환경에 대한 표준 개발 절차로서 CBSD Process는 현재, 요구사항에 맞게 개발하기 위해서 컴포넌트 개념과 장점을 충분히 반영 할 수 있는 실제 개발절차들에 대한 구체화된 사항들을 제시하고 있고, 이를 반영하고 있는 여러 개발 방법들이 존재하고 있다.

(2) UML

UML은 Booch, Rumbaugh, Jacobson에 의해서 개발되어진 객체지향 분석 및 설계 방법으로 객체지향 개발을 위한 통일된 모델링 언어(Unified Modeling Language)이다. 이 UML은 소프트웨어를 시각화 및 명세화 하여 생성하고 문서화하기 위한 표준화된 언어이며 구현과 상관없이 소프트웨어의 전 개발 과정에 걸쳐 프로세스에서 사용되어지는 방법이다. 특히, 클래스나 아키텍처, 객체에 대해서 관계나 상호작용, 내부행동 등을 여러 종류의 다이어그램을 통해서 표준화된 그래픽한 표기법을 사용하여 나타낸다.

3. 3계층 표현 방식의 아키텍처

3.1 개념

3계층 표현 방식(Triple Layer Expression Formula)의 아키텍처는 3개의 층으로 구성된 표현 형태를 가진 아키텍처를 의미하는 것으로, 사용자와 연관이 되는 User Related Layer와 주요기능을 담고 있는 Primary Business Layer, 그리고, 추가적인 기능이나, 종속적인 기능을 가진 Secondary Business Layer로 구성되어져 있다. 자세한 세부사항은 다음과 같다.

(1) User Related Layer (사용자 관련 층)

실제 시스템이 외부 환경과 접촉하는 External Interface

관련 부분으로 일반적으로 사용자나 운영자가 다루는 UI(User Interface)나, 윈도우(window), 폼(Form), 메뉴(Menu), 버튼(Button) 등의 그래픽에 관련된 컴포넌트를 담고 있는 층이다.

(2) Primary Business Layer (주된 층)

시스템이 제공하고자하는 주요 서비스나, 기능에 대한 내용을 담고 있는 층으로 실제 시스템에서 가장 중요한 역할들만을 모아 놓은 층이라고 생각 할 수 있다. 즉, User Related Layer를 통해서 들어온 데이터나 정보를 가지고, 실제 요구사항에 맞게 핵심기능(Core Business)이나 중앙 제어 기능을 가진 컴포넌트를 담고 있는 기능 중심의 층이다.

(3) Secondary Business Layer (종속적인 층)

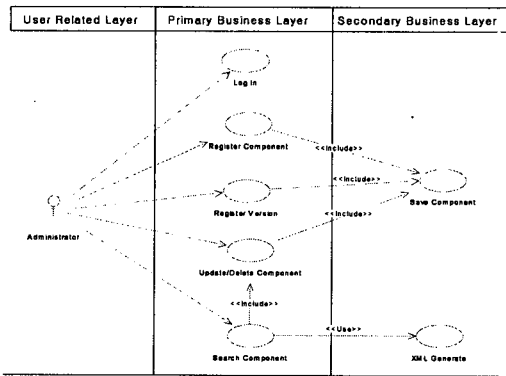
이 층은 위의 Primary Business Layer에서 작성한 시스템의 주된 기능에 대해서 데이터베이스에 저장을 해주는 기능과 같은 보조기능을 수행하는 컴포넌트나, 시스템의 외부적으로 보이지 않고 내부적으로 특정한 알고리즘이나 기능을 가지고 있으면서, 시스템 내부 인터페이스를 통해서 Primary Business Layer에 종속된 컴포넌트를 위한 층이다.

3.2 작은 적용 예제

위에서 살펴보았던 3계층 표현 방식을 이용해서, 하나의 작은 적용 예제로서 작은 시스템을 선택하여, UML 기반의 환경에서 시스템을 분석 및 설계하는 과정을 알아보자. 참고적으로, 본 논문에서는 UML에 기반한 CBSD Process of the Work Flow Model[1]을 이용하여 컴포넌트의 정보를 등록하는 작은 컴포넌트 정보 등록기를 예제로 선택하였다.

(1) 3계층 표현방식을 이용한 Use Case Diagram

본 논문에서 적용하고 있는 예제는 단순히 컴포넌트 정보를 등록 및 조회, 검색하는 기능을 가지고 있는 시스템이다.

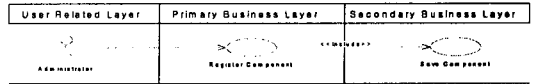


<그림1> 컴포넌트 정보 등록기의 Use Case 다이어그램

위의 <그림1>에서 구축하고자하는 시스템에 대해서 요구사항을 분석하여 3계층 표현 방식의 형태로 사용자 부분, 시스템의 주된 기능 부분, 종속적인 부분으로 나누어서 Use Case와 액터를 잡고 전개도를 그렸다.

(2) 기능별로 나눈 부분 Use Case 다이어그램

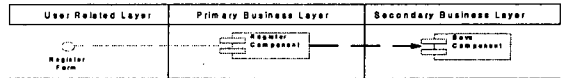
본 예제를 가지고 좀더 자세하게 적용예제를 접근하기 위해서, 특정 기능 예제로서 위의 <그림1>에서의 컴포넌트 정보 등록에 관한 부분만을 다음 <그림2>와 같이 추출하였다.



<그림2> 정보 등록 부분 Use Case 다이어그램 추출된 부분 Use Case Diagram을 가지고 일반적인 UML의 프로세스처럼 각각의 Use Case에 대한 시나리오를 작성한다.

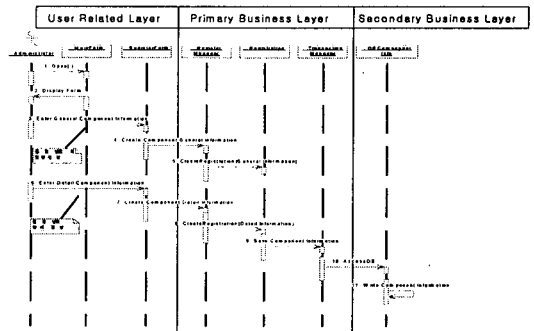
(3) Component Diagram

작성된 시나리오를 중심으로 다음 <그림3>과 같이 필요한 컴포넌트를 추출하고 3계층 표현방식으로 구분하였다.

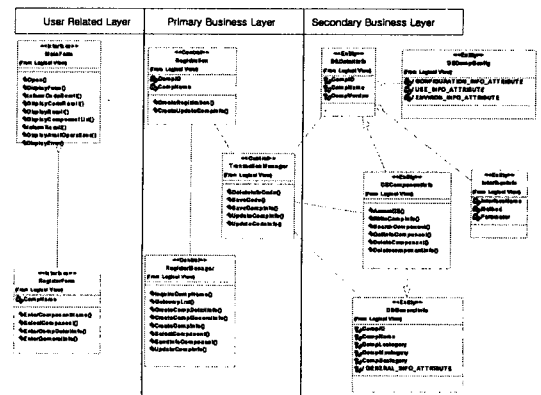


<그림3> 컴포넌트 다이어그램

이때, 컴포넌트의 의미가 일반 패키지의 의미처럼 보일 수 있으나, 복잡한 시나리오나 포괄적인 의미를 가지고 있는 Use Case라면, 컴포넌트를 더 세분화 해야한다. 또는, 위의 Use Case 단계에서 점점 반복의 프로세스에 의해서 좀더 세분화하고 명백한 Use Case 다이어그램을 작성해야 한다.



<그림4> 컴포넌트의 동적 설계



<그림 5> 컴포넌트의 정적 설계

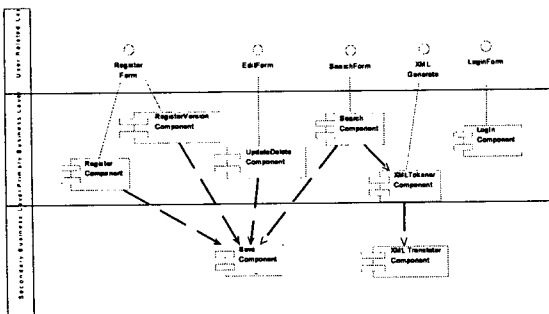
(4) 동적 정적 컴포넌트 설계

위의 <그림3>에서 3계층 표현방식에 의해서, 기능별로 한

선상에 위치한 컴포넌트를 가지고, 동적인 설계로서 Use Case의 시나리오를 바탕으로 순차 다이어그램(Sequence Diagram)을 작성하고, 작성된 순차 다이어그램에서 사용되어졌던 객체나, 클래스를 가지고, 클래스 다이어그램을 작성한다.

(5) Integrated Component Diagram

컴포넌트의 정적, 동적 설계까지의 단계를 마친 후에 모든 Use Case 다이어그램으로부터 나누어진 각각의 기능별 컴포넌트들을 가지고 모아서, 3계층 표현방식의 구조를 이용해서 통합된 컴포넌트 다이어그램을 작성한다. <그림1>의 Use Case 다이어그램의 내용을 가지고 3계층 표현방식에 실제 적용되어진 통합 컴포넌트 다이어그램이 <그림6>과 같다.



< 그림 6 > 3계층 표현방식의 통합 컴포넌트 다이어그램

(6) Component Specification

위의 <그림6>의 통합된 컴포넌트 다이어그램을 통해서, 컴포넌트들 간의 관계를 알 수가 있었다. 그러나, 통합된 컴포넌트 다이어그램은 설계하고자 하는 전반적인 시스템의 구조도로 자세한 사항을 표시할 수가 없다. 따라서, 컴포넌트의 내부 상태와 다른 컴포넌트의 관계를 좀더 세부적으로 명시하고자, 본 논문에서 제시하는 순수 3계층 표현방식과는 좀 다르지만, 각각의 컴포넌트에 대해서 역시 3계층으로 구성된 다음과 같은 형식의 컴포넌트 명세 방법을 제시한다.

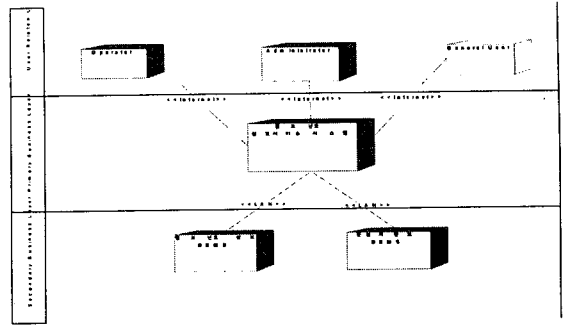
On Entry :
On Execution :
On Exit :

<그림7> 3계층 구조의 컴포넌트 명세

- On Entrance : 컴포넌트에 입력에 관련된 부분으로 사전조건(Pre-Condition)이나, 어느 컴포넌트가 입력과 관계가 되어 있는지에 관한 사항을 명시해 준다.
- On Execution : 데이터나 정보를 외부로부터 받은 후에 컴포넌트의 내부 동작이나 실행이 되는 내부 알고리즘, 그리고, 그에 따른 상태 등을 명시해준다.
- On Exit : 컴포넌트를 거쳐 나가는 데이터에 대한 사후 조건(Post-Condition)이나, 컴포넌트 수행 후에 연이어서 받아줄 컴포넌트에 관한 정보를 명시해준다.

(7) Deployment Diagram

마지막 단계로서 만들고자하는 시스템의 환경에 관련된 배치상태를 나타내주는 방식으로 역시 3계층 표현방식을 이용해서 <그림8>과 같이 나타내었다.



<그림8> 3계층 표현방식의 배치 다이어그램

3.3 3계층 표현방식의 적용 기대 효과

지금까지 컴포넌트의 정보 등록을 하는 시스템의 분석 및 설계된 내용을 가지고, 3계층 표현방식의 아키텍처를 이용한 간단한 작은 모델링 적용 예제를 보았다. 이 적용 예제를 보면서, 무엇보다도, 3계층 표현방식을 모델링에 이용한다면, 먼저, 매우 간단해지고, 이해하기가 쉬워진다는 것을 알 수가 있다. 요구사항이나 기능을 3개의 표현방식 계층에 각각 기능 및 형식에 맞게 넣어주어 쉽게 CBSD Process를 통해서 분석 및 설계를 할 수가 있었다. 그리고, 본 논문의 컴포넌트 기반의 소프트웨어 개발(CBSD)에서는 객체보다는 큰 단위의 컴포넌트를 사용한 관계로 오히려 깔끔하게 시스템을 설계하고 컴포넌트를 찾아 배치하였으며, 여러 세부영역으로 나누어진 여러 개의 층을 가지고 시스템을 모델링 하는 기법보다는 다른 층과의 연계성을 쉽게 할 수 있고, 명확한 구분을 여러 계층에서 사용되었던 모델링 계층 방식보다는 3계층 표현방식이 훨씬 분석 및 설계의 모델링 하는 부분에서 효율적이라는 것을 발견 할 수 있었다.

4. 결 론

본 논문에서는 효율적으로 컴포넌트 기반의 시스템 모델링을 하는 방법의 하나로서 3계층 표현방식의 구조를 제안하였다. 그리고, 하나의 작은 예제를 통해서, 실제 적용되는 과정을 보면서, 기존의 방법과의 차이점에 대해서 관찰 및 분석, 그리고, 추후 적용의 기대효과에 대한 고찰을 하였다. 본 논문에서 제시하는 3계층 표현방식의 구조가 모든 시스템의 모델링 방식에 적용하지는 못하였다. 따라서, 현재 존재하는 CBSD Process의 방식에 대해서 적용 및 분석을 할 수 있도록 좀더 연구하고, 이런 장점들만을 모아놓은 좀 더 좋은 세부절차의 CBSD Process를 개발하는 것이 추후 주요 연구과제로 제안된다.

5. 참고문헌

- [1] 임성빈, 송치양, 문창주, 백두권, "UML에 기반한 CBSD Process의 Work Flow Model", KIPS2000(춘계), 한국정보처리학회, CD발간, 2000.
- [2] Desmond F. D'souza and Alan C. Wills, "Objects, Components, and Frameworks with UML", Addison-Wesley, 1998.
- [3] Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language UserGuide", Addison-Wesley, 1999