

새로운 부하관리시스템 개발을 위한 UML 적용 연구

이찬주* 김진호 박종배
 *건국대학교전기공학과 기초전력공학공동연구소 건국대학교전기공학과

An Application of Unified Modeling Language to Develop the New Load Management System

Chan-Joo Lee* Jin-ho Kim Jong-Bae Park
 Konkuk University EESRI Konkuk University
 jubily@kkuoc.konkuk.ac.kr gmji@plaza1.snu.ac.kr jbaepark@konkuk.ac.kr

Abstract - This paper presents a methodology for the application of Unified Modeling Language (UML) to develop the new load management system. Development of new load management system in competitive electricity market is very complex since it requires too much data of power system. For the efficient and flexible design to develop of new load management system, a UML approach in applied which is composed of a class diagram, package diagram using Rational Rose Unified Process.

2. 본 론

1. 서 론

지난 수 세기동안, 많은 진보된 종류의 소프트웨어 엔지니어링 기법들이 전력계통 공학분야에 적용되어왔다. 과거 전력계통 분야에 지배적으로 적용되어왔던 구조적 프로그래밍(Structured Programming) 기법은 데이터 처리의 효율성을 증진시켰으나 소프트웨어 개발의 대형화와 복잡화에 따른 유지보수의 어려움과 설계측면에서 많은 문제점들이 발생되었다. 더욱이 개인용 PC의 보급과 함께 IT 기술의 진보로 인해, 전력계통 역시 설계·모의·해석을 위한 GUI(Graphical User Interface) 기반의 응용소프트웨어 개발[1,2]이 대두됨에 따라서 기존의 프로그래밍 기법은 그 한계성이 부각되었다. 새로운 패러다임의 객체지향 프로그래밍(Object-Oriented Programming) 기법은 대규모 소프트웨어 시스템에서 발생할 수 있는 많은 문제들을 효율적으로 처리할 수 있을 뿐 만 아니라 설계된 컴포넌트의 재사용이 가능한 장점이 있기 때문에 전력계통 공학분야의 소프트웨어 개발에 적용한 많은 연구가 보고되고 있다[3-6]. 그러나 소프트웨어의 복잡화와 대형화에 따른 유지보수의 어려움과 사용자의 요구사항에 대한 능동적인 처리가 요구됨에 따라 소프트웨어 개발에 있어서 효율적인 설계의 중요성이 강조되었다. UML(Unified Modeling Language)은 1990년 후반에 Booch, Jacobson, 그리고 Rumbaugh를 중심으로 개발되었으며, 소프트웨어 중심의 산출물을 가시화하고, 명세화하며, 구축하고 그리고 문서화하는데 사용되는 비주요 모델링 언어이다[7]. UML은 많은 서로 다른 형태의 복잡한 시스템을 정의하는데 사용되며, 요구사항 분석에서부터 완성된 시스템의 테스트까지 전반적인 시스템의 개발에 적용된다. 또한 점진적이고 반복적인 개발 프로세스를 통해서 설계시 발생할 수 있는 위험요소를 최소화하고 시스템 설계를 최적화할 수 있는 장점을 지닌다.

본 논문에서 우리는 경쟁적 전력시장에서 전력가격 결정에 관련하여 중요 요소인 부하패턴을 모니터링하고, 사용자의 에너지 소비패턴을 관측할 수 있는 윈도우 기반의 부하관리시스템(Load Management System)의 개발을 위해 UML을 적용하여 설계하였고, 전력계통 소프트웨어 설계에 잠재적인 가능성을 제시하였다. 그리고 Rational Rose 2000을 사용하여 모델링하였고, 운영체제에 독립적이고 객체지향의 대표적 언어인 Java를 사용하여 구현하였다.

2.1 새로운 부하관리시스템 구성

경쟁적 전력시장에서의 전력가격은 매 시간마다 시장에서 결정되기 때문에 전력시장 참여자 및 에너지 소비자들은 가격에 대한 가변성과 불확실성에 대해 노출되어 있다. 따라서 부하를 소유한 사업자에 대해 불확실한 전력가격에 대해 능동적으로 대처할 수 있는 효율적인 부하관리 메커니즘의 중요성이 높아지고 있다. 경쟁적 전력시장에서의 새로운 윈도우 기반의 부하관리 시스템은 에너지 사용 패턴의 시각화를 통하여 계통 상태를 감시하거나 또는 특정 전기사업자의 관점에서 전체 전력계통을 보다 효과적이고 직관적으로 평가하여 에너지 사용자의 비용을 감소하는데 그 목적을 두고 있다. 본 논문에서 우리가 설계한 새로운 부하관리시스템의 개념은 다음 그림1과 같다.

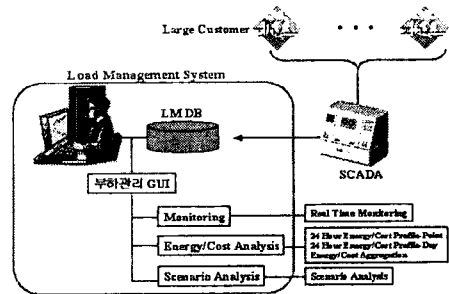


그림 1 새로운 부하관리시스템 개념도
 Fig. 1 New Load Management System Architecture

새로운 부하관리시스템은 위의 그림에서처럼 크게 2가지 기능으로 이루어져 있다. 우선, 대규모 수용가의 미터기로부터 측정된 SCADA의 데이터를 일정한 시간 간격을 두고 필요한 데이터를 호출·저장하는 부하관리시스템의 데이터베이스(Load Management DataBase, LMDB)와 저장된 데이터를 바탕으로 사용자 자신의 부하를 효율적으로 관리하도록 도와주는 GUI(Graphical User Interface)로 이루어진 부하관리시스템이다. 이러한 부하관리시스템을 통해서 사용자는 계통 상태를 모니터링(Monitoring) 할 수 있고 또한 이를 바탕으로 자신의 이용가능부하와 전력요금을 분석(Analysis) 할 수 있다. 그리고 부하관리시스템의 시나리오 기능을 통해서 소유하고 있는 부하에 대한 최적 에너지 사용계획 및 소비전력을 수립할 수 있다. 본 논문에서 설계한 부하관리시스템의 시나리오는 기존의 부하관리 프로그램에서 사용되고 있는 부하이전, 최대부하삭감, 자율절전요금제, 직접부하제어, 자가발전, 하계휴가보수기간조정요금제와 경쟁적 전력시장을 고려하여 공급자비교요금제, 실시간요

금액, 그리고 직거래요금제를 추가하였다. 또한 위의 그룹을 통해서 새로운 부하관리 시스템의 개발범위를 설정하고 사용자에 대한 요구사항을 분석하였다. 그리고 분석된 결과물에 대해서는 UML의 쓰임새도(Use-Case Diagram)를 통해서 시스템에 필요한 객체(Object)들을 추출하였고, 객체와 객체간의 상호관계를 순차도(Sequence Diagram)와 교류도(Interaction Diagram)를 활용하여 사용자의 동적인 관점에서 설계하였다. 본 논문에서는 개념적인 분석단계에 대한 설계를 생략하고 실제 부하관리시스템 개발에 적용될 수 있는 설계단계의 클래스도(Class Diagram)와 패키지도(Package Diagram)에 중점을 두었다.

2.2 Package Diagram

UML의 새로운 부하관리 시스템에 대한 사용자의 요구사항 분석과 개발 구성요소의 결정을 한 후, 논리적인 관점으로 시스템에서 연관성 있는 클래스간의 집합을 만들어 패키지도를 설계하였다. 클래스들을 그룹화 함으로써 모델을 보다 높은 차원에서 바라볼 수 있고, 각 패키지에 무엇이 포함되었나를 조사함에 따라 모델을 더 깊게 파악할 수 있으며, 분석단계에서 설정한 설계범위를 구체화할 수 있다. 본 논문에서 설계한 패키지도는 아래 그림2와 같다.

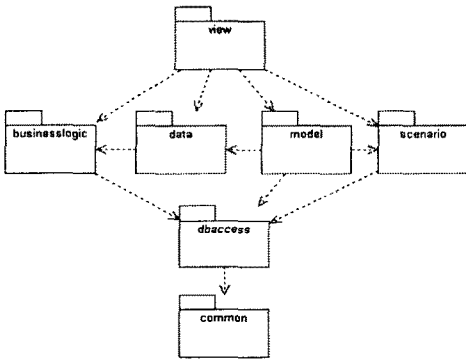


그림 2 부하관리시스템 패키지도
Fig. 2 Package Diagram for Load Management System

패키지의 구조는 객체지향 방법론의 MVC(Model-View-Controller) 구조론을 바탕으로 설계하였으며, 각각의 패키지에 대한 설명은 다음과 같다.

- **Common** : 부하관리시스템의 데이터베이스 가운데 공통적으로 사용되는 클래스의 집합이다. 이 패키지를 통해서 데이터베이스의 환경이 설정되고 접근경로가 결정된다.
- **DBAccess** : SQL(Standard Query Language)을 사용하여 데이터베이스에 접근, 기록된 데이터를 Read, Write, Delete, Insert, Update 하는 역할을 한다.
- **BusinessLogic** : GUI에서 사용자에게 정보를 제공하기 위해서 필요한 데이터를 잠시 보관하는 일종의 버퍼 역할을 하는 클래스의 집합이다. 이와 같은 패키지가 필요한 이유는 대규모 전력계통 데이터의 가중치를 덜어주어 효율적인 데이터 처리를 하기 위함이다.
- **Data** : GUI의 모니터링과 분석 기능을 수행하는데 필요한 데이터를 생성하고 각각의 수행결과를 보여주는 클래스의 집합이다.
- **Model** : GUI에서 생성된 테이블 모델을 위한 클래스의 묶음이다.

- **Scenario** : 시나리오 관련된 뷰에 해당하는 패키지이다. 즉, 시나리오에 대한 사용자 정의 화면에 해당하는 클래스의 집합이다.
- **View** : 가장 높은 계층에 해당하는 패키지로서 모든 화면 구성에 필요한 클래스의 집합이다.

Common 패키지는 가장 낮은 층(Low-Level Layer)에 위치하여 DBAccess패키지와 의존관계(Dependency)를 갖는다. 이것은 영속성을 갖는 계층데이터를 포함한 Common 패키지가 어떤 외부 작용에 의해서 변할 경우, 이 패키지에 의존적인 데이터 베이스를 형성하는 DBAccess 패키지가 변화하는 것을 의미한다. 그리고 부하관리 시스템의 사용자인터페이스에 해당하는 View 패키지는 BusinessLogic, Data, Model, Scenario 패키지에 의존적이므로 가장 높은 층(High-Level Layer)에 설계하였다. View, Model, Scenario 패키지는 사용자에게 제공하는 부하관리시스템의 사용자 인터페이스에 해당하는 것이므로, 설계자 혹은 개발자에 따라 상이하게 나타낼 수 있다. 본 논문에서는 설계한 부하관리시스템 패키지도 가운데 사용자에게 독립적이며 개발과정에 있어서 가장 중요시되는 데이터베이스 설계에 대해서 클래스도를 통해서 구체화하였다.

2.3 Class Diagram

시스템에 존재하는 객체의 타입과 타입간의 정적인 관계를 기술하는 클래스도는 분석단계에서 추출된 객체와 그들간의 관계를 형상화시킴으로써, 실제 개발할 부하관리시스템의 코드에 직접적인 영향을 미친다. 클래스도를 통해서 객체간에 주고받는 메시지의 구조를 이해할 수 있을 뿐만 아니라, 클래스의 행위(Behavior)와 속성(Attribute)을 알 수 있다. 그림3은 부하관리시스템의 데이터베이스 환경을 설정하는 Common 패키지의 클래스도이다.

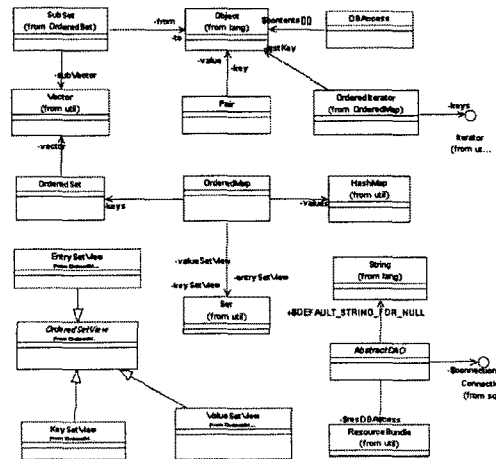


그림 3 Common 패키지의 클래스도
Fig. 3 Class Diagram of Common Package

Common 패키지의 클래스들은 부하관리시스템의 데이터베이스에서 사용되는 일반적인 클래스들로 구성되어 있다. 추상적인 AbstractDAO 클래스는 모든 영속적 관련 클래스들과 SQL의 Connection클래스를 통해서 연관관계를 갖는다. 데이터베이스의 접속은 OrderedIterator 클래스가 key 인터페이스를 통해서만 접근 가능하다. 그리고 KeySetView, ValueSetView, EntrySetView 클래스는 OrderedSetView 클래스로부터 상속을 받는다. 이와같이 설계한 부하관리시스템 Common

