

선형 공정표를 생성하는 시스템 프로토타입 개발을 위한 기초 연구

A Preliminary Research for Developing System Prototype Generating Linear Schedule

류 한 국*

Ryu, Han-Guk*

Department of Architectural Engineering, Changwon National University, Changwon,
GyeongNam-Do, 641-773, Korea

Abstract

Linear scheduling method limits to present works of work breakdown structure as a form of lines and was often developed manually. In other words, linear schedule could not utilize activity, work breakdown structure, and etc. information of network schedule such as CPM(Critical Path Method) and has been used only for reporting or confirming construction master plan. Therefore, it is necessary to develop system which can automatically generating the linear schedule based on the network schedule having many accumulated and useful construction schedule information. Thus, this research has an effort to establish data process model, data flow diagram, and data model in order to make linear schedule. In addition, this research addresses the system solution structure, user interface class diagram and logic diagram, and data type schema. The results of this paper can be used as a preliminary research for developing linear schedule generating system prototype by utilizing the network schedule information.

Keywords : Linear Schedule, Data Process Model, Data Flow Diagram, Data Model, System Solution Structure, Class Diagram

1. 서 론

1.1 연구의 배경 및 목적

초고층 건축물과 같이 건설공사 중에는 공사 기간 내내 다수의 유사한 액티비티들이 연속해서 반복적으로 발생한다. 이러한 건설공사의 수행을 위해 시공자들은 연속적으로 발생하는 동일한 작업에 대해 고정적으로 자원을 할당하고 유지될 수 있도록 계획하고 관리한다. 이를 위해 액티비티들을 같은 공간에 겹치지 않고 동일한 작업조를 구성하여 작업을 수행한다.

이러한 건설공사의 작업들을 계획하는 것은 전통적인 네트워크 기법으로도 가능하지만, 선형 공정표를 사용하면 더 효과적으로 계획하고 관리할 수 있다. 선형 공정표를 사용하여 건설공사를 계

획하기 위해서는 건설 공사를 선형으로 표현하여야 한다.

선형 공정계획 모델은 반복적인 액티비티를 표현하기에 편리하고, 충돌이 발생할 수 있는 액티비티 식별에 도움이 된다. 바차트(막대그래프) 또는 네트워크 공정표는 액티비티들이 복잡하게 얽혀 공간-시간 관계를 쉽게 확인하기가 어렵다. CPM 공정표는 액티비티간의 관계가 복잡하게 연결되어있다. 즉, 액티비티들간에 SS관계(start-to-start), FF관계(finish-to-finish) 등의 릴레이션이 규정되어 있어 작업의 선후행 네트워크를 파악하는 것이 매우 복잡하다. 이러한 단점들 때문에 선형계획법이 사용되게 되었다. 그러나 기존의 선형 공정계획은 마스터 스케줄과 같이 작업분류체계의 대분류 레벨의 작업을 사선의 형태로 표현하는 데 국한하고 수작업으로 작성하였다. 즉, 네트워크 공정표의 정보를 활용하지 못하고 독립적으로 선형 공정표를 작성하여 보고용 또는 전체 공정계획 확인용으로 사용하고 있는 실정이다.

따라서 기존에 축적된 자료가 많은 네트워크 공정표를 기반으로 선형 공정표를 자동으로 생성할 수 있는 시스템 개발이 필요하다. 이에 본 연구는 선형 공정표 구현을 위한 데이터 프로세스 모델, 데이터 플로 다이어그램, 데이터모델을 구축하고 시스템 프로

Received : April 26, 2010

Revision received : May 31, 2010

Accepted : June 7, 2010

* Corresponding author: Ryu, Han-Guk

[Tel: 82-55-213-3804, E-mail: hgryu@changwon.ac.kr]

©2011 The Korea Institute of Building Construction, All rights reserved.

토 타입의 시스템 솔루션 구조, 클래스 다이어그램, 데이터 타입 스키마를 개발하고자 한다. 즉, 네트워크 공정표를 선형 공정표로 변환하기 위한 시스템 프로토타입 개발의 선행 연구를 수행하는 것이 본 연구의 목적이다.

1.2 연구의 범위 및 방법

본 연구의 절차와 방법은 Figure 1과 같다.

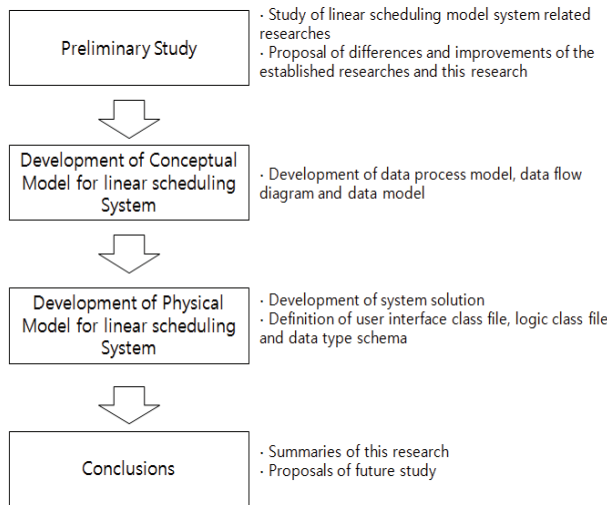


Figure 1. Research process and method

본 연구는 공정계획 시스템 개발과 관련된 연구를 고찰하여 공정계획 시스템 개발 연구의 현황을 파악하고 본 연구와의 차이점과 개선점을 제시한다.

네트워크 공정표를 선형 공정표로 변환하는 시스템 프로토타입 개발을 위한 선행 연구로써 데이터 프로세스 모델, 데이터 플로 다이어그램을 구축하고 선형 공정표 생성을 위한 데이터 모델링을 실시하여 개념적 모델을 제시한다.

개념 모델을 기반으로 하여 네트워크 공정표의 선형 공정표로의 변환을 위한 물리적 모델을 개발한다. 시스템 솔루션을 구축하고 사용자 인터페이스 클래스 파일, 로직 클래스 파일, 데이터 타입 스키마를 정의하여 궁극적으로는 네트워크 공정표 정보를 활용하여 선형 공정표로 변환하는 시스템 프로토타입 개발을 위한 이론 체계를 수립한다.

2. 예비적 고찰

2.1 공정계획 시스템과 관련된 기존 연구

건축공사의 공정계획 시스템과 관련된 기존 연구의 주요 내용을 정리하면 다음과 같다.

Moselhi and Nicholas[1]는 건설 계획과 스케줄링을 위한 통

합된 하이브리드 방식의 전문가 시스템을 제안하였다. 관계형 데이터베이스, 지식 베이스, 통제 기능, 전통적인 네트워크 소프트웨어, 포트란 언어로 작성된 인터페이스 프로그램을 통합하였다. 하이브리드 인공지능 환경을 적용하는 프로토타입 시스템은 최종 사용자 인터페이스를 통하여 추가되는 작업들내에서 작업 로직을 결정하도록 한다. 그러나 전문가시스템 서브모듈에 장착된 stand-alone 셋으로 구성된 다른 모듈은 이러한 영향을 받지 않은 작업 기간을 실제 기간으로 변경하고 서브모듈은 오버타임, 현장 협소, 작업자 재배치, 학습효과, 날씨 상태의 결과에 따른 생산성 레벨의 변화를 고려하였다. 사례 연구를 통하여 시스템의 기본적인 특징을 나타내고 건설관리의 교육과 훈련, 건설 클레임의 분석과 준비, 계약변경 관리 등의 다른 영역에도 적용할 수 있는 가능성을 제시하였다.

Shaked and Warszawski[2]는 모듈화된 다층 건축물 공사에서 진도 공정계획을 위한 전문가 시스템을 개발하였다. 효과적인 공정계획의 목적을 제안하고 건축물에 적용할 수 있는 몇 가지 규칙을 제안하였다. 개발된 시스템은 다양한 관리 목표 관점에서 실행 계획과 사용자 요구 데이터 측면에서 융통성을 추구하였다. 이는 RC구조의 공동주택 건물에 적용할 수 있으며 프로젝트 속성의 입력 프로세싱, 작업량 추정과 이를 수행하기 위한 액티비티 생성, 액티비티의 시간과 자원 할당, 공정계획의 결정을 지원하였다. 또한 개발 시스템은 규칙 기반, 데이터베이스 알고리즘, 상황 기반 지식, 사용자 화면을 제시하였다.

Mulholland and Christian[3]은 건설 프로젝트가 시간 제약에 따라 복잡하고 높은 불확실성의 환경에 노출되어 있음을 직시하고 건설 공정계획에서 이러한 불확실성을 고려하고 정량화하는 시스템적 방법을 제안하였다. 과거 프로젝트로부터 습득된 교훈을 활용하여 전형적인 입력정보와 기대 출력정보를 포함하는 공정 리스크 평가 과정을 HyperCard와 Excel을 이용하여 개발하였다.

Wang and Demsetz[4]는 공사단계의 작업기간은 계획과 다르게 발생할 수 있지만 공사계획단계의 CPM 공정표는 이러한 다양성을 반영하는 데 한계가 있음을 역설하였다. 이에 계획 대비 실적 결과를 통합하기 위하여 시뮬레이션과 규칙 기반의 공정계획 모델을 개발하였다.

El-Rayes[5]는 초고층 건축물, 다수의 주택건설, 고속도로, 파이프라인 설치공사, 다리, 터널, 철도, 비행장 활주로, 하수관거 배관 공사 등의 반복적인 건설 프로젝트를 스케줄링하기 위한 객체 지향 모델(LSCHEDULER)을 개발하였다. 개발된 객체 지향 모델의 분석, 설계, 개발 단계의 개관을 제공하고 이러한 단계는 반복적인 건설 프로젝트를 스케줄링하기 위한 모델을 제공하고 스케줄링 요구사항을 충족하도록 설계되었다. 모델은 반복 액티비티의 자원 계획, 반복 작업의 공정 최적화, 반복 및 비반복 공정계획 기법의 통합을 위한 절차를 제공하였다. 모델은 메뉴, 다

이알로그 박스, 윈도를 포함한 사용자 인터페이스를 윈도우용 애플리케이션으로 구현하였다.

2.2 기존 공정계획 기법 및 연구와 본 연구의 차이점과 개선점

기존 공정계획 기법과 기존 연구와의 차별성과 개선점을 고찰하기 위하여 우선 기존 공정계획 기법의 특징을 정리하면 다음과 같다.

바차트 기법은 판독이 용이하고 이해하기 쉬우며, 명확하고 간단하게 현재 공정을 표시하고, 계획과 진행의 비교가 용이하다는 장점을 가지는 반면에 액티비티간의 연계 및 공사공간 수요의 파악 등이 어렵고 선행공종에 대한 체크포인트 불확실하다는 단점이 있다.

네트워크 공정계획 기법은 액티비티 상호간의 연결관계가 분명하고 주공정선을 파악하기 쉬우며 컴퓨터 프로그램의 도움으로 최적화 공정이 가능하다는 장점이 있는 반면에 작성하는데 많은 경험과 지식이 필요하고 시각적인 이해도가 바차트에 비해 떨어지며 반복공정에 부적합하다는 단점이 있다.

LOB(Line of Balance) 공정계획 기법은 반복 작업이 많은 공사에서 공간적인 요소를 반영할 수 있는 장점이 있는 반면에 공사기간을 산정하는 체계적인 계산법이 개발되어 있지 않고 실제 공사 관리 측면에서 활용도가 떨어진다는 단점이 있다.

그러나 지금까지 건설공사의 공정계획은 건설공사의 기간에 대한 중요성과 일정분석 등의 장점 때문에 비네트워크 기법보다는 네트워크 기반의 공정계획 프로그램을 사용해왔다. 또한 2.1절에서 살펴본 바와 같이 공정계획과 관련된 다수의 연구들이 수행되고 시스템이 개발되었으나 본 연구와 같이 네트워크 공정표의 데이터를 이용하여 새로운 선형 공정표를 생성하는 연구는 다소 부족한 것으로 판단된다. 따라서 본 연구의 네트워크 공정표를 선형 공정표로 변환하는 방법론을 통하여 네트워크 공정표와 LOB 공정표의 장점을 확보할 수 있을 것으로 판단된다.

3. 네트워크 공정표의 선형 공정표로의 변환을 위한 개념 모델

3.1 데이터 프로세스 모델

선형 공정표 생성을 위한 업무 모델은 네트워크 공정표를 선형 공정표로 호환하기 위한 업무를 파악하여 체계적으로 모형화한 것으로 크게 기능 모델과 프로세스 모델로 구분할 수 있다.

기능 모델은 공정계획가가 자원을 활용하여 성과물을 공급하기 위해 해야 할 일을 체계적으로 모형화하는 것으로 업무영역, 기능, 프로세스 등으로 분류할 수 있다. 업무영역과 기능 부분은 공정계획 업무를 주요 영역별로 그룹화하고 추구하는 기능을 분류

한 것이며, 프로세스 모델은 공정계획 업무를 실행 가능한 단위로 분해하여 업무 기능을 논리적인 업무활동으로 세분화하여 하부 단위를 프로세스로 정의하고 이 프로세스를 체계적으로 모형화한 것이다. Figure 2는 네트워크 공정표를 선형 공정표로 변환하는 시스템 프로토타입 개발을 위한 프로세스 모델이다.

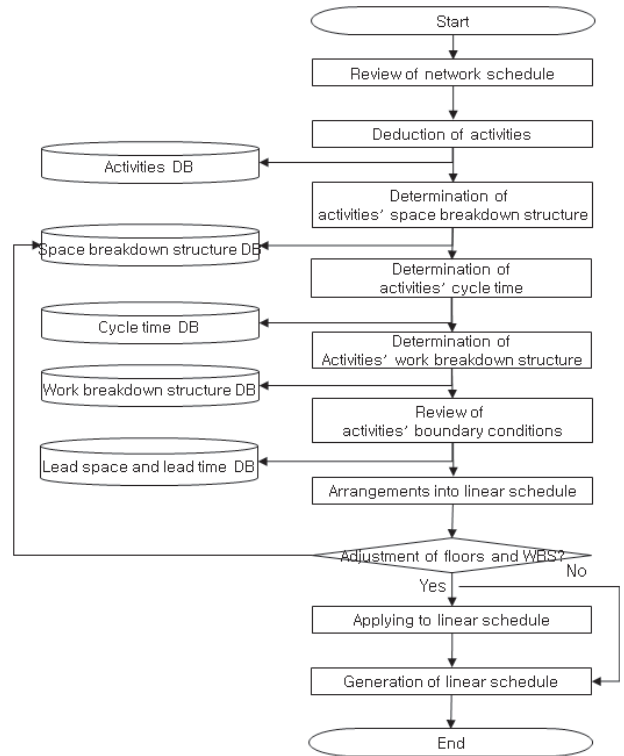


Figure 2. Process model for developing system prototype

공정계획가는 네트워크 공정표의 작성한 내용을 검토하고 액티비티를 도출하여 액티비티 데이터베이스에 저장하고 액티비티의 작업분류체계와 시간 정보 및 공간 정보를 도출하여 액티비티의 공간분류체계와 작업분류체계의 사이클 타임¹⁾을 결정하여 공간분류체계의 데이터베이스, 사이클 타임의 데이터베이스, 작업분류체계의 데이터베이스에 저장한다. 또한 액티비티간의 경계조건²⁾을 분석하여 리드스페이스와 리드타임을 설정하여 데이터베이스에 저장한다. 이를 바탕으로 선형 공정표로 액티비티를 정렬할 수 있는 기반이 마련되었으므로 이를 x축(시간축), y축(공간

- 1) 사이클 타임은 가장 작은 단위의 관리가 가능한 공간에 대한 작업 기간으로 정의한다. 선형 공정계획표상에서 사이클 타임은 한 개층 분량의 작업을 수행하는 데 필요한 작업 기간으로 정의할 수 있다.
- 2) 선형 공정표상에서 선·후행 작업간에 최소한의 리드스페이스와 리드타임을 만족시키는 조건을 경계조건으로 정의한다. 따라서 작업의 관계를 변경하거나 추가할 경우 경계조건을 만족하여야 한다.

축) 좌표로 구성된 2차원 평면에 선형으로 액티비티를 표현한다. 선형 공정표상에서 층수 및 작업분류체계의 조정이 있을 시에는 이를 반영하여 선형 공정표를 생성한다.

3.2 데이터 플로 다이어그램

3.1절에서 살펴본 데이터 프로세스 모델을 기반으로 하여 네트워크 공정표의 데이터를 선형 공정표상에서 적용하기 위한 데이터 플로 다이어그램은 Figure 3과 같다.

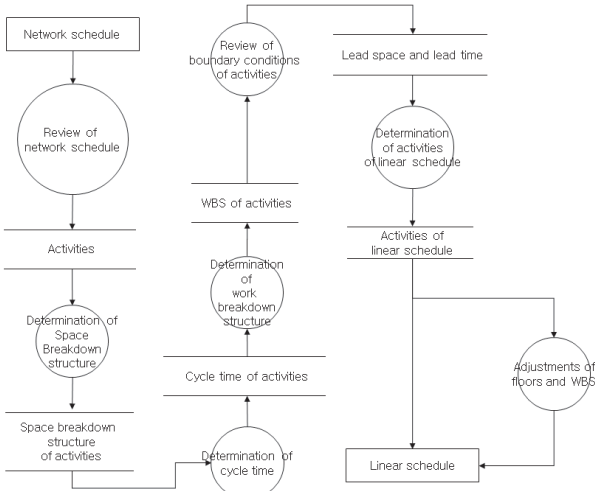


Figure 3. Data flow diagram for developing system prototype

데이터 플로 다이어그램은 데이터가 각 프로세스를 따라 흐르면서 변환되는 모습을 도식화한 것으로 정보시스템의 분석과 설계에 매우 유용하게 사용되며 시스템의 모형화 도구로서 가장 보편적으로 사용하고 있다. Figure 3의 데이터 플로 다이어그램은 공정계획가와 시스템 개발자 사이의 데이터의 의미와 흐름을 전달하기 위한 용도로 사용된다.

데이터 플로 다이어그램의 구성요소는 외부 개체, 데이터 플로, 데이터 저장, 업무 프로세스로 표현된다. 사각형(□)은 외부의 개체를 나타낸 것으로 프로세스 처리과정의 데이터 발생의 출처와 결과를 나타낸다. 데이터 흐름도상에서 프로세스의 상호 관련성을 표시한다. 화살표(→)는 데이터의 흐름을 나타내며 데이터 플로 다이어그램의 구성 요소들 간의 인터페이스를 나타낸다. 대부분의 데이터 흐름은 프로세스들 사이를 연결하지만, 데이터 저장소로부터 데이터 흐름을 나타내기도 한다.

상하선 박스(=)에 명기된 내용은 데이터 저장소로써 저장되어 있는 정보의 집합이다. 데이터 저장소는 파일 형태 등으로 단순한 데이터의 저장을 나타내는 것이지 데이터의 변동을 표시하지 않는다. 즉, 데이터가 도출되어 변화되지 않은 데이터 저장을 위한 행위를 의미한다.

원형(○) 내부에 명기된 내용은 프로세스로 데이터의 변경 및 조정을 위한 행위로 데이터를 원하는 데이터로 변환하여 출력시키기 위한 과정으로 원안의 내용은 프로세스가 수행하는 일 또는 프로세스를 수행하는 행위자를 기술한다. 프로세스는 자체적으로 데이터를 생성할 수 없고 항상 입력되는 데이터가 있어야 한다. 프로세스는 항상 새로운 가치를 추가해야 한다.

3.3 선형 공정표의 호환을 위한 데이터 모델

선형 공정표의 호환을 위한 데이터 모델이란 네트워크 공정표를 선형 공정표로 호환하기 위한 정보 및 자료를 데이터 관점에서 체계적으로 모형화 것이다. 정보 및 자료는 공정계획 시 필요로 하는 데이터들의 집합인 개체(Entity)³와 개체의 속성(Attribute)⁴으로 구성된다.

Figure 4와 같이 개체 간에 존재하는 상호간의 연관성과 대응 개체수(Cardinality)⁵를 고려하여 데이터모델링을 수행하였다.

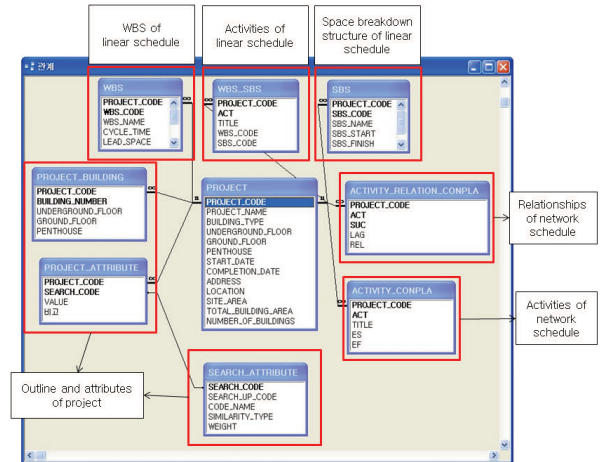


Figure 4. Data model for developing system prototype

데이터 모델은 프로젝트를 중심으로 스타스키마의 형태를 띄며 개체는 프로젝트 개체, 프로젝트 빌딩 개체, 프로젝트 속성 개체, 작업분류체계(WBS) 개체, 공간분류체계(SBS) 개체, 작업분류체계와 공간분류체계를 통합한 통합 개체, 액티비티 개체, 액티비티 릴레이션 개체, 액티비티 개체 등으로 구성된다.

프로젝트 개체는 프로젝트의 개요와 속성정보를 포함하며 프로젝트의 정보를 입력하거나 기존의 공정계획 정보를 이용하기 위

- 3) 개체(Entity)란 실제 현상에 존재하거나 또는 존재하는 것으로 가정하는(즉, 추상화를 통하여 식별할 수 있는) 것들(즉, 실제 현상을 구성하는 구성물들)을 지칭한다.
- 4) 속성(Attribute or property)이란 개체를 설명하는 특성, 즉 개체에 대하여 알고자 하는 정보들이다.
- 5) 대응개체수(Cardinality)는 관계를 구성하는 개체들이 상대방 개체에 몇 개씩 대응하는 가를 나타내는 개념이다.

하여 활용된다. 건축 공사의 작업 수행을 위해 네트워크 공정표의 작업분류체계를 공간분류체계를 통합하여 액티비티를 생성한다. 액티비티 개체는 액티비티의 작업 기간과 착수일, 완료일 속성을 갖고 액티비티 릴레이션 개체는 액티비티의 선후행 릴레이션 속성을 갖는다.

4. 네트워크 공정표의 선형 공정표로의 변환을 위한 물리 모델 개발

4.1 선형 공정표로의 변환을 위한 시스템 솔루션

네트워크 공정표를 선형 공정표로 변환하기 위한 시스템 프로토타입은 Microsoft .NET Framework 1.1, Microsoft Visual Studio 2003, Microsoft Visual C#, Microsoft Access Database를 기반으로 개발할 것이다.

선형 공정표의 호환을 위한 시스템솔루션 구조는 Figure 5와 같다.

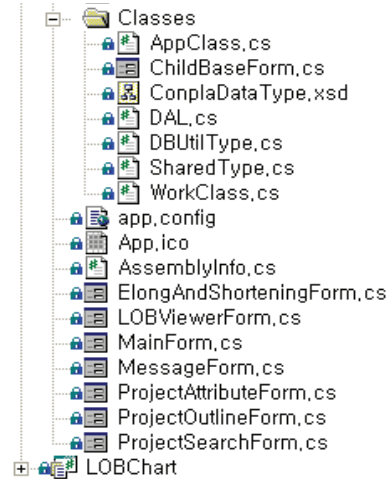


Figure 5. System prototype structure

선형 공정표의 생성을 위한 시스템 솔루션의 파일리스트는 사용자 인터페이스 클래스와 로직 클래스로 구성되며 4.2절에서 선형 공정표로 변환을 위한 사용자 인터페이스 클래스의 파일을 설명하며 4.3절에서 선형 공정표로 변환을 위한 로직 클래스의 파일을 설명한다.

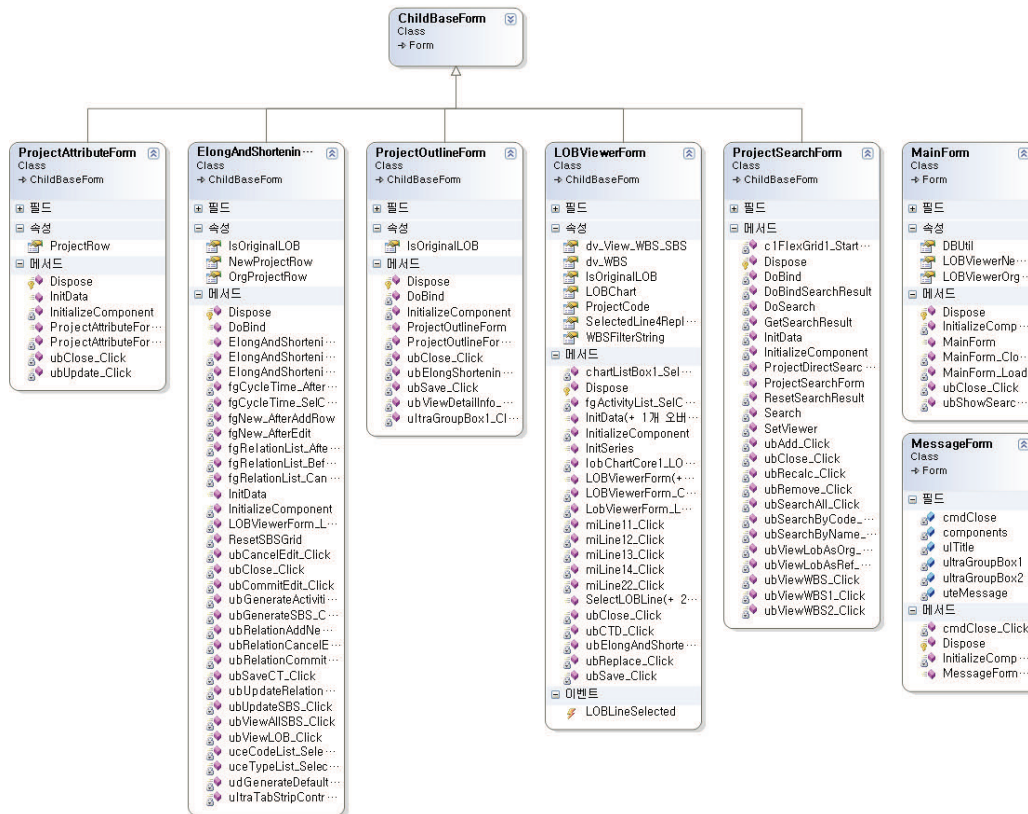


Figure 6. User interface class diagram

4.2 사용자 인터페이스 클래스의 파일

선형 공정표로 변환을 위한 사용자 인터페이스 클래스는 사용자의 입력과 출력을 관리하는 윈도우 폼 클래스이다. Figure 6은 선형 공정표로 호환하기 위한 시스템 프로토타입의 사용자 인터페이스 클래스 다이어그램이다. 주요한 사용자 인터페이스 클래스의 파일들을 설명하면 다음과 같다.

- ElongAndShorteningForm.cs는 건축물의 물리적인 층수 증가 또는 감소에 따른 선형 공정표상의 선형을 반영하기 위한 폼 클래스로서 구체적으로 사이클 타임의 정의, 공간 분류체계의 생성, 관계성 생성 작업을 수행한다.
- LOBViewerForm.cs는 네트워크 공정표를 선형 공정표의 선형을 나타내기 위한 LOB 차트를 보여주는 뷰어 폼 클래스이다. 각 공간분류체계의 선형 정보와 엑티비티의 공정정보를 나타낸다.
- MainForm.cs는 MDI(Multi Documents Interface)⁶⁾형태의 사용자 인터페이스에서 MDI 부모창의 기능을 하는 폼 클래스이다.
- MessageForm.cs는 기본적인 메시지 표시기능이 있는 폼 클래스이다.
- ProjectAttributeForm.cs는 건축공사 프로젝트의 여러 가지 속성을 정의할 수 있는 폼 클래스이다.
- ProjectOutlineForm.cs는 건축공사 프로젝트의 개요를 볼 수 있는 폼 클래스로서 새로운 프로젝트로 복사할 수 있으며 확장과 축소를 수행할 수 있다
- ProjectSearchForm.cs는 건축공사 프로젝트를 검색할 수 있는 폼 클래스로서 직접 검색과 유사성 검색을 할 수 있으며 검색된 프로젝트의 작업분류체계를 선택하여 선형 공정표인 LOB 라인을 볼 수 있다.

4.3 로직 클래스의 파일

네트워크 공정표를 선형 공정표로 변환하기 위한 시스템솔루션의 로직 클래스는 클래스 폴더 하위의 클래스들로 구성되어있으며, 직접적으로 사용자 인터페이스를 구성하지 않고 일반적인 업무 로직만을 수행한다. Figure 7은 선형 공정표로 호환하기 위한 시스템 프로토타입의 로직 클래스 다이어그램이다.

6) 초창기의 윈도우 애플리케이션은 간단한 문서나 그림 등을 제작하는 오피스 관련 애플리케이션이 대부분이었으나 최근에 들어서 인터넷의 보급과 .net의 개발로 인한 웹애플리케이션 등 다양한 온라인 중심의 애플리케이션이 개발되고 있다. 윈도우 애플리케이션의 성격에 따라 윈도우 구성 형태를 선택하는데, 윈도우는 구성상 단일 윈도우(SDI)와 다중 윈도우(MDI)로 나눌 수 있다. SDI는 오브젝트나 윈도우의 내용이 단순하거나 일대일 관계일 때, 오브젝트가 주요한 표현이나 한 개의 단위로 사용될 때 적합하다. 반면에 MDI는 일정한 오브젝트 셋을 관리하거나 주 윈도우에 종속되는 자식 윈도우들을 표현할 때 적합하다.

주요한 로직 클래스 다이어그램의 파일들을 설명하면 다음과 같다.

- AppClass.cs: 어플리케이션의 진입점을 포함한 어플리케이션 클래스이다.
- ChildBaseForm.cs: 어플리케이션의 모든 자식폼의 상위클래스로서 추가적인 방법과 멤버를 포함한다.
- ConplaDataType.xsd: 데이터를 표현하는 형식화된 데이터셋 클래스로서 주로 데이터의 입력과 출력 관련 정보전달 인자로 사용한다.
- DAL.cs: DAL(Data Access Layer) 클래스로서 데이터베이스를 대상으로 SQL문을 수행하고 데이터베이스의 연결 설정과 실질적인 연결을 관리한다.
- DBUtilType.cs: 데이터베이스의 테이블 단위 메서드들과 업무에 사용되는 코드를 생성하는 클래스이다.
- SharedType.cs: 공통적으로 쓰이는 타입이나 열거형 등을 정의해놓은 파일이다.
- WorkClass.cs: 업무로직을 담당하는 클래스이며 층수 증가 및 삭제, 작업분류체계의 삭제 또는 변경 등의 주요 작업을 수행한다.

4.4 선형 공정표로의 변환을 위한 데이터 타입 스키마

Table 1은 네트워크 공정표를 선형 공정표로 변환하기 위한 시스템 프로토타입의 데이터 타입 스키마이다.

Table 1. Data type of system prototype generating linear schedule

Entity	Attribute	Primary Key	Data type
PROJECT	PROJECT_CODE	○	string
	PROJECT_NAME	-	string
	BUILDING_TYPE	-	int
	UNDERGROUND_FLOOR	-	int
	GROUND_FLOOR	-	int
	PENTHOUSE	-	int
	START_DATE	-	datetime
	COMPLETION_DATE	-	datetime
	ADDRESS	-	string
	LOCATION	-	int
	SITE_AREA	-	double
PROJECT_B UILDING	TOTAL_BUILDING_AREA	-	double
	NUMBER_OF_BUILDINGS	-	int
	PROJECT_CODE	○	string
	BUILDING_NUMBER	○	string
	UNDERGROUND_FLOOR	-	int
PROJECT_A	GROUND_FLOOR	-	int
	PENTHOUSE	-	int
	PROJECT_CODE	○	string

Entity	Attribute	Primary Key	Data type
TTRIBUTE	SEARCH_CODE	○	string
	VALUE	-	string
SEARCH_LAT TRIBUTE	SEARCH_CODE	○	string
	SEARCH_UP_CODE	-	string
	CODE_NAME	-	string
	SIMILARITY_TYPE	-	string
	WEIGHT	-	double
SBS	PROJECT_CODE	○	string
	SBS_CODE	○	string
	SBS_NAME	-	string
	SBS_START	-	int
	SBS_FINISH	-	int
WBS	PROJECT_CODE	○	string
	WBS_CODE	○	string
	WBS_NAME	-	string
	CYCLE_TIME	-	double
	LEAD_SPACE	-	int
	LEAD_TIME	-	int
	CT_START	-	int
	CT_FINISH	-	int
CYCLE_TIME2	-	double	
ACTIVITY	PROJECT_CODE	○	string
	ACT	○	string
	TITLE	-	string
	ES	-	datetime
	EF	-	datetime
WBS_SBS	PROJECT_CODE	○	string
	ACT	○	string
	TITLE	-	string
	WBS_CODE	-	string
	SBS_CODE	-	string
ACTIVITY_R ELATION	PROJECT_CODE	○	string
	ACT	○	string
	SUC	○	string
	LAG	-	double
	REL	-	string
VIEW_PROJ ECT_ATTRIB UTE	PROJECT_CODE	-	string
	SEARCH_CODE	-	string
	VALUE	-	string
	CODE_NAME	-	string
	SIMILARITY_TYPE	-	int
	WEIGHT	-	double
	SEARCH_UP_CODE	-	string
	MAX_VALUE	-	string
	MIN_VALUE	-	string
	SEARCH_SCOPE	-	double
VIEW_SBS	PROJECT_CODE	○	string
	SBS_CODE	○	string
	SBS_NAME	-	string
	SBS_START	-	int
	SBS_FINISH	-	int
	AUTO_GENERATED	-	boolean
	TYPE_CODE	-	string
	SECTION_CODE	-	string
VIEW_SBS_ SECTION_LI ST	PROJECT_CODE	-	string
	SECTION_CODE	-	string
	MAX_SBS_START	-	int

Entity	Attribute	Primary Key	Data type
VIEW_WBS	MAX_SBS_FINISH	-	int
	COUNT	-	int
	PROJECT_CODE	○	string
	WBS_CODE	○	string
	WBS_NAME	-	string
	CYCLE_TIME	-	double
	LEAD_SPACE	-	int
	LEAD_TIME	-	int
	CT_START	-	int
	CT_FINISH	-	int
	CYCLE_TIME2	-	double
VIEW_ACTIV ITY	MIN_SBS	-	int
	MAX_SBS	-	int
	MIN_ES	-	datetime
	PROJECT_CODE	-	string
	ACT	-	string
	TITLE	-	string
	WBS_CODE	-	string
	SBS_CODE	-	string
	PROJECT_CODE	-	string
	ACT	-	string
	SUC	-	string
VIEW_ACTIV ITY_RELATI ON	LAG	-	double
	REL	-	string
	TITLE_ACT	-	string
	WBS_CODE_ACT	-	string
	WBS_CODE_ACT	-	string
	SBS_CODE_ACT	-	string
	SBS_START_ACT	-	int
	SBS_FINISH_ACT	-	int
	TITLE_SUC	-	string
	WBS_CODE_SUC	-	string
	WBS_NAME_SUC	-	string
VIEW_SEAR CH_ATTRIB UTE	SBS_CODE_SUC	-	string
	SBS_START_SUC	-	int
	SBS_FINISH_SUC	-	int
	TYPE_ACT	-	string
	TYPE_SUC	-	string
	SEARCH_CODE	-	string
	SEARCH_UP_CODE	-	string
	CODE_NAME	-	string
	DESC	-	string
	VALUE	-	string
	SIMILARITY_TYPE	-	int
CODE_MAS TER	WEIGHT	-	double
	CODE_CATEGORY	○	int
	CODE_ID	-	int
DESC	-	string	

네트워크 공정표를 선형 공정표로 변환하기 위한 시스템 프로토타입의 데이터 개체는 Figure 4와 같이 구성된 데이터 모델을 기반으로 구성된다. PROJECT 개체는 프로젝트 코드, 프로젝트 명, 건축물 종류, 지하층수, 지상층수, 펜타하우스 층, 착공일, 준공일, 위치, 연면적, 동수 등의 속성 정보를 갖는다.

PROJECT_BUILDING 개체는 프로젝트 코드, 빌딩 동, 지하층수, 지상층수, 지상층수, 펜타하우스 층의 속성정보를 갖는다. WBS 개체는 프로젝트 코드, 작업분류체계 코드, 작업분류체계 이름, 싸이클 타임, 리드스페이스, 리드타임, 싸이클 타임을 산정하기 위한 착수공간 완료공간 등의 속성정보를 갖는다. View_기타 개체들은 액티비터를 선형 공정표에 그래픽적으로 표현하기 위해 존재하는 개체이다.

Table 1의 개체의 속성 중에서 PROJECT, ACTIVITY, ACTIVITY_RELATION, PROJECT_BUILDING 등의 속성은 네트워크 공정표 데이터베이스에 있는 속성정보를 활용하고 네트워크 공정표 데이터베이스에 없는 SBS, WBS_SBS 등의 속성정보는 사용자가 프로젝트의 특성을 고려하여 작성하여 시스템을 적용해야 한다.

5. 결론 및 향후 연구방향

일련의 반복적인 액티비터들이 포함되는 건설공사로는 대규모 고속도로 건설공사, 고층건물, 제방, 방파제, 하수관거 공사 등이 있다. 기존 연구에서도 지적한 바와 같이 기존 CPM 기반의 공정계획은 이러한 유형의 건설공사를 효과적으로 계획하거나 관리하는 데 부족하다. 또한 공정표상의 프로젝트의 종료 시점에 초점을 맞춘 비네트워크 기법(Barchart, LOB, LS, LSM 등)을 적용한 선형 공정표 생성 시스템은 프로젝트 기간의 시각화에 초점을 맞추어 개발되었다. 즉, 네트워크 공정 데이터를 활용하여 새로운 선형 공정표를 생성하는 시스템은 거의 전무하다.

이에 본 연구는 네트워크 공정표를 선형 공정표로 변환하는 시스템 프로토타입을 개발하기 위한 선형 연구로서 네트워크 공정표의 데이터 정보를 분석하여 선형 공정표 구현을 위한 데이터 프로세스 모델을 구축하고 데이터 플로 다이어그램, 데이터 모델을 구축하였다.

구축한 데이터 모델을 기반으로 선형 공정표로 변환하기 위한 시스템 프로토타입의 시스템 솔루션과 사용자 인터페이스 클래스와 로직 클래스 다이어그램을 개발하였다.

네트워크 공정표상의 데이터 정보가 선형 공정표의 데이터 정보로 이용하거나 새로운 데이터 정보를 생성하여 네트워크 공정표의 데이터베이스 정보를 활용하는 데이터 타입 스키마를 제안하였다. 본 연구 결과를 기반으로 네트워크 공정표를 선형 공정표로 생성하는 시스템 프로토타입을 개발하여야 할 것이다.

요 약

기존의 선형 공정계획은 마스터 스케줄과 같이 작업분류체계의 대분류 레벨의 작업을 사선의 형태로 표현하는 데 국한하고 수작업으로 작성하였다. 즉, 네트워크 공정표의 정보를 활용하지 못하고 독립적으로 선형 공정표를 작성하여 보고용 또는 전체 공정계획 확인용으로 사용하고 있는 실정이다.

따라서 기존에 축적된 자료가 많은 네트워크 공정표를 기반으로 선형 공정표를 자동으로 생성할 수 있는 시스템 개발이 필요하다. 이에 본 연구는 선형 공정표 구현을 위한 데이터 프로세스 모델, 데이터 플로 다이어그램, 데이터모델을 구축하고 시스템 프로토타입의 시스템 솔루션 구조, 클래스 다이어그램, 데이터 타입 스키마를 개발하고자 한다. 즉, 네트워크 공정표를 선형 공정표로 변환하기 위한 시스템 프로토타입 개발의 선형 연구를 수행하는 것이 본 연구의 목적이다.

키워드 : 선형공정표, 데이터 프로세스모델, 데이터 플로 다이어그램, 데이터 모델, 시스템 솔루션 구조, 클래스 다이어그램

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2009-0072277).

References

1. Moselhi O, Nicholas MJ. Hybrid expert system for construction planning and scheduling, *Journal of Construction Engineering and Management* 1990;116(2):221-238.
2. Shaked O, Warszawski A. CONSCHEDE: Expert System for scheduling of modular construction projects, *Journal of Construction Engineering and Management* 1992;118(3):488-506.
3. Mulholland B, Christian J. Risk assessment in construction schedules, *Journal of Construction Engineering and Management* 1999;125(1):8-15.
4. Wang W, Demsetz LA. Model for evaluating networks under correlated uncertainty-NETCOR, *Journal of Construction Engineering and Management* 2000;126(6):458-466.
5. El-Rayes K. Object-oriented model for repetitive construction scheduling, *Journal of Construction Engineering and Management* 2001;127(3):199-205.