

객체지향형 설계정보관리시스템 모델링

오태훈*, 김선호**

The Object-Oriented Modeling for Drawing Information Management Systems

Tae-Hoon Oh, Sunn-Ho Kim

Abstract

In recent years, the manufacturing industry begins to demand the PDM which integarates design and manufacturing information concurrently to reduce the time to market of new products. However, current commercialized PDM tools have drawbacks in representing by relational data modeling a variety of data types such as CAD files, images, audio, long text, etc. For this reason, object-oriented approaches are suggested as an alternative. In this paper, using one of object-oriented approaches, Rumbaugh's OMT(Object-Modeling Technique), we propose a data model of drawing information management systems essential for PDM. The scope of the model is limited to the information management for drawings, parts, and product structures. Static, dynamic and functional models are suggested in detail. In the models, the drawing status is classified into on-progress, approved, disposed, and released. Also, parts are classified into products, assemblies, and components.

†) 본 연구는 한국학술진흥재단 연구지원(01 E-0717)으로 수행되었음.

* 명지대학교 산업공학과

** 명지대학교 산업공학과 교수

1. 서론

기업의 목표변화가 70년대의 생산비 감축에서 80년대 생산품질 향상으로 변화해 왔으며, 90년대에는 설계에서 출하까지 소요시간, 즉 제품개발시간(Time-to-Market)의 감축이 기업의 최대 핵심 이슈로 등장했다. 기업이 경쟁시장에서 경쟁력을 유지하기 위해서는 보다 값싸며 좋은 품질을 갖는 제품을 보다 빠르게 경쟁시장에 공급해야한다[D. A. Taylor,1995]. 이를 위해서는 최초의 제품 개념설계에서부터 생산 및 출하 단계까지의 소요시간을 전체적으로 단축해야 한다. 이에 따라 동시공학(Concurrent Engineering)이라는 엔지니어링 작업을 순차적이 아닌 병렬적으로 처리한다는 관리기법이 도입되기 시작하였다. 동시공학을 실현하기 위해서는 우선적으로 설계정보와 제조 관련정보가 결합된 종합적인 PDM(Product Data Management)이 필요하다[김선호 등, 1996; 이충화 등, 1996; 최열현, 1995; CIMdata,1995; HP, 1993].

PDM은 조직내 관련 구성원 모두에게 필요한 모든 데이터를 적절한 시점에 원하는 곳으로 알맞는 형식으로 제공하는 것으로, 제품 개발에 필요한 모든 정보인 이미지 정보, 도면, 문서, 제품구성, 엔지니어링 정보, 업무프로세스, 프로젝트 관리에 관한 정보 등을 개념설계부터 제품 전체의 라이프사이클에 걸쳐 정보를 공유하여 관리하는 것이다. PDM에서는 CAD/CAM 정보, 이미지, 오디오, 긴 문장 등 다양한 멀티미

디어 정보가 복합적으로 처리되어야 한다.

그러나, 기존의 관계형 데이터베이스 형태는 정수나 문자, 문자열 등과 같은 기본적인 데이터형만을 지원하여 프로그램에서 볼 수 있는 일반적인 형태인 이미지, 음성, 긴 문장과 같은 구조가 일정치 않고, 많은 양의 데이터의 저장과 검색을 지원하지 못하고 있다. 또한, 복잡한 구조를 가지는 구조를 표현하려면 함수적 종속성(functional dependency)을 가지는 열(row)들을 모두 테이블(Table)로 나누어 저장하므로, 이를 검색할 경우 수많은 결합(join)을 통해 튜플(tuple)들을 검색해야 하는 어려움이 있다. 설계환경과 같은 장기간 동일한 일을 하는 경우에 데이터베이스에서 장기간 트랜잭션을 지원해야만 하는데 관계형 데이터베이스의 트랜잭션 모델은 장기간 뿐만아니라 객체와 스키마의 버전과 변화 통보(change notification)와 같은 기능을 현세계에 나타내고 관리하는 일을 지원하지 못한다[최영근 등,1995; W. Kim, 1990].

이에 반해서 객체지향 데이터베이스는 설계시의 장기간 트랜잭션의 처리, 시간 개념, 설계변경에 따른 버전 갱신, 변화통보의 개념, 멀티미디어와 같은 비구조적인 객체 처리, 일반화(generation) 및 집산화(aggregation)같은 의미개념의 표현, 복합 객체(composite object) 계층 구조를 가진 엔지니어링 데이터의 처리가 가능하다[D. A. Taylor,1990;W. Kim,1990]. 결국 PDM에서 정보를 원활히 처리하기 위해서는 객체지향적 데이터베이스와 객체지향적 방법

론에 입각한 데이터 모델링이 요구된다.

본 연구에서는 PDM에서 무엇보다도 중요한 설계정보관리 영역을 객체지향적 패러다임을 이용하여 설계정보관리 요소들을 '객체'의 관점에서 분석하고 이를 바탕으로 설계정보관리시스템을 설계하기 위한 분석모델링을 하였다. 객체지향 모델링방법론으로는 Rumbaugh의 OMT(Object Modeling Technique) 방법을 이용하였다.

2. 관련된 기존연구

제품개발시간을 줄이기 위해 제품 설계에서 생산에 이르기까지 발생하는 복잡하고 다양한 데이터들을 효율적으로 통합 관리하고자 하는 통합정보시스템에 대한 연구들이 계속적으로 이루어지고 있다[삼성HP,1993; 이충화,1996; 김철호 등,1994].

CAD 도면과 같은 벡터 그래픽 정보, 여러종류의 화상정보, 수식 및 테이블 등의 도구, 그리고 동화상과 음성정보 등 한 문서 내에서 여러 가지 형태의 자료를 공유하여 통합적인 관리를 하고자 하는 기술정보관리시스템(Technical Information Management System)에 관한 연구가 있었으며[신영식 등,1995], 도면과 관련된 기술정보를 효율적으로 관리하고자 하는 도면정보관리시스템에 대한 개발 연구가 있었다[김선호 등, 1994]. 그러나, 이들 시스템들은 기존 관계형 데이터베이스가 갖고 있는 몇가지 문제점들을 가지고 있으면서 다양한 형태의 자료들을 공유하며 관리하고 있다. 관계형데

이터베이스가 갖는 문제점을 극복하기 위해 새로운 개념인 객체지향데이터베이스가 나타나게 되었고, 생산정보시스템을 객체지향접근방법을 이용해 모델링하는 연구가 제시되었다[김광수 등,1992]. 또한, 제조시스템의 운영에 있어서 기본정보로 널리 사용되어지고 있는 BOM을 Chung와 Fisher가 객체지향 데이터베이스를 사용하는 기본개념으로 제안하였다[Y. Chung et al.,1991].

최근에는 설계와 제조분야에서 개별적인 자동화 관련 시스템간의 데이터 교환을 위해 국제표준기구(ISO)에서 제정한 표준 제품 데이터 모델(STEP : Standard for the Exchange of Product model data)을 객체지향 데이터베이스로 구현하기 위한 연구가 진행되고 있다[유상봉 등,1995].

3. 객체지향 방법론

3.1 개요

객체지향 방법론은 객체지향 패러다임(paradigm)을 기반으로 하고 있으며, 여러 가지 개발 방법들이 개발되어 있다. 대부분의 객체지향 방법은 소프트웨어를 객체 및 그 구조를 정의하는 정적관계(Static Relation) 부분과 동적 행위(Dynamic Behavior)부분으로 파악한다.

정적관계는 객체 및 클래스들 사이에 관계가 문제 영역이 존재하는 시간 동안 시간의 흐름에 관계없이 항상 고유하게 유지되는 것이다. 객체 및 클래스들 사이의

정적관계들은 연관관계(association), 상속관계(inheritance), 집단관계(aggregation) 등이 이에 속한다. 연관관계는 가장 일반적인 개념으로써 객체 및 클래스 사이의 의미적인 관련성(relationship)을 말하며, 계승관계나 집합관계는 연관관계의 특별한 경우라고 할 수 있다. 연관관계는 객체들 사이의 의미적 관련성을 표현하는데, 그 관계가 양방향으로 존재한다. 즉, 연관화는 어느 시점에서 보느냐에 따라 상대적인 관계를 나타낼 수 있다. 계승관계는 하나 이상의 클래스들로부터 속성이나 행위를 공유하는 관계를 의미하며, 상위 클래스의 속성과 행위와는 별개로 자신 나름대로의 속성과 행위를 가질 수 있다. 집합관계는 여러 클래스들을 모아서 새롭게 클래스를 생성하는 것이다. 이때 내부에 포함된 기존의 클래스는 독립된 클래스로 존재하지 않고 자기를 포함하는 클래스의 일부분이 된다.

동적행위는 정적 관계의 상대적인 개념으로 시간의 흐름에 따라 변하는 관계를 말한다. 모든 객체는 서로 다른 객체와 일정 시간동안 상태 변화를 일으키게 하는 메시지(message)를 주고받으면서 자신의 상태(state)가 변하거나 다른 객체의 상태를 변화시키게 된다. 이러한 동적변화를 위해 객체들은 항상 외부 또는 자기 자신에게서 들어오는 메시지를 기다리게 되고, 메시지가 들어오면 그에 적절한 행위를 일으키게 된다. 이 행위는 다른 객체에게 메시지를 보내는 행위를 포함하고 있을 수 있다. 다시 여기에 보내진 메시지에 따라 객체들이 새로운 행위를 하게된다. 이렇게 동

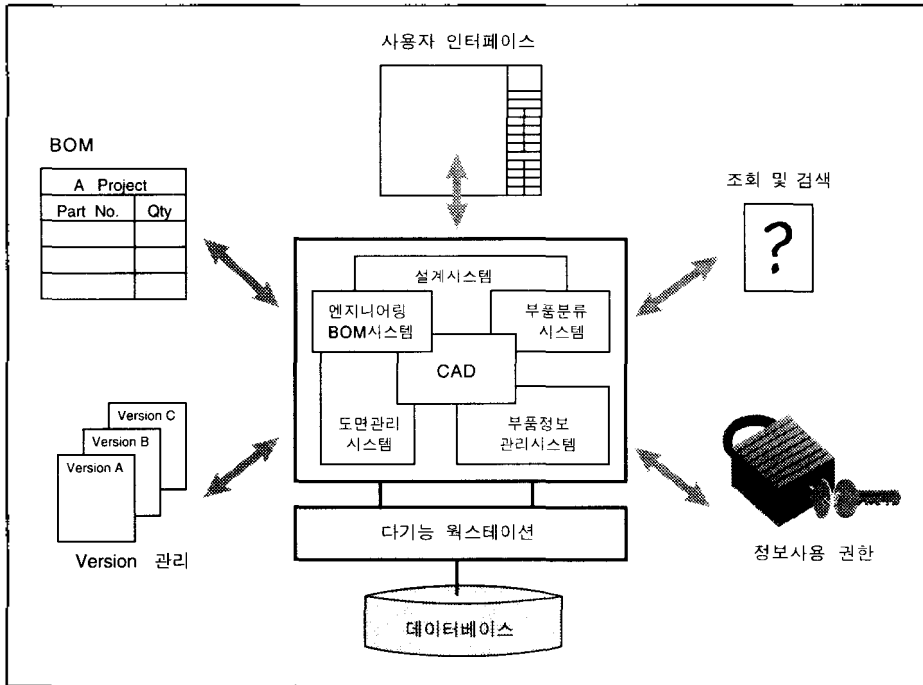
적 행위는 시간이 지남에 흐르면서 연속적으로 이루어지는데, 이러한 연속적인 흐름이 모여 어떤 하나의 작업을 구성한다[최성운 등, 1996; 최영근 등, 1995; D. A. Taylor,1990; J. Rumbaugh et al., 1991].

3.2 사용된 모델링 기법

Rumbaugh의 OMT(Object Modeling Technique)는 그래픽적 표기법을 이용하여 객체를 모델링하는 방법으로 분석(Analysis), 시스템 설계(System Design), 객체 설계(Object Design), 구현(Implementation)의 4 단계로 나누어지며, 전 과정에 추상화(abstraction), 캡슐화(encapsulation), 모듈화(modulation), 계층화(hierarchy) 등의 일관된 객체 지향 개념을 적용한 것이다.

OMT 방법은 분석과 설계모델을 설정하는데 있어 3가지 모델, 즉 객체와 클래스를 중심으로 이들의 관계성(relationship)을 분석하는 객체 모델링(object modeling), 객체의 행위를 중심으로 사건과 반응에 대한 상태의 변화를 분석하는 동적 모델링(dynamic modeling), 데이터의 변환 과정에 따른 처리 과정과 데이터의 흐름을 나타내는 기능 모델링(functional modeling) 등을 적용한다[J. Rumbaugh et al.,1991].

OMT는 사용하기 쉬우며, 문제들을 이해하기 쉽고, 사용자와의 대화를 쉽게 할 수 있다. 또한 기업 업무를 모델링하기 쉽고, 데이터베이스 설계에 유용하게 사용될 수 있는 장점을 가지고 있다[최영근 등, 1995].



<그림 1> 설계정보시스템 기능적 측면[삼성HP, 1993]

여기서는 설계정보에 관련된 객체와 클래스를 추출하여 그들간의 관계를 연관화(association), 집단화(aggregation), 일반화(generalization) 관계를 규명하여 클래스의 속성과 연산을 함께 표현하여 정적구조인 객체모델링을 생성하였으며, 도면과 관련된 상태 즉, 진행상태, 승인상태, 폐기상태와 관련된 사건들을 상태다이아그램(State Diagram)을 이용하여 표현함으로써 동적모델링을 생성시켰다. 또한, 프로세스(process), 데이터흐름(data flow), 제어 흐름(control flow), 데이터 저장소(data dictionary), 행위자들로 구성된 기능 모델링을 생성하기 위해서 부품정보와 도면정

보관리에 관련된 다수의 프로세스들간의 데이터 흐름을 분석하여 데이터 흐름 다이어그램(DFD: Data Flow Diagram)을 이용하여 표현하였다.

4. 객체지향 설계정보관리시스템 모델링

4.1. 설계정보관리시스템

설계정보관리시스템은 도면 및 BOM(Bill of Material) 등의 기술관련 자료를 저장 및 관리하며, 데이터간의 연관성을 부여하여 부품과 제품에 관련된 정보를

<표 1> 도면정보관리 모듈에서 식별된 객체들

객체명	객체설명
도면정보	도면관리시스템에서 관리하는 상위 개념의 객체로, 검토이력, 수정이력, 승인이력, version, 폐기이력, 출도이력 등의 정보를 소유하고 있으며, 도면번호, 도면명, 설계자, 도면상태, 설계착수일, 설계완료일, 도면크기, CAD파일명 등의 속성정보를 갖는다. 또한 도면과 함께 수정, 삭제, 조회, 출력되며, 부품정보와 BOM에서 도면정보가 참조된다.
검도이력	도면에 대한 검토요청시 작성되는 문서로, 도면번호, 도면명, 검토자, 검토신청일, 검토내용 등의 속성을 갖는다. 도면수정시 참조되어지며, 검토확인된 도면에 대한 승인요청을 한다.
수정이력	가상탐인 검토자들이 작성한 도면 검토내역에 따라 도면수정시 작성되는 문서로, 도면번호, 도면명, 수정자, 수정사유, 수정내용, 수정번호 등을 속성으로 갖는다. 수정된 도면에 대한 검토요청을 한다.
승인이력	가상탐인 승인자들이 도면승인 요청시 작성하는 문서로, 도면번호, 도면명, 승인자, 승인일 등의 속성을 갖는다. 승인시 도면에 대한 version이 발생하며, 배포 및 출도가 발생된다.
Version	승인된 도면에 대한 변경요청으로, 새로 작성된 수정도면을 승인할 때 작성되는 문서이다. 속성으로 version번호, 버전갱신일, 변경사유, 변경내용 등을 갖는다.
폐기이력	도면의 폐기 승인이 발생할 때 작성되는 문서로, 도면번호, 도면명, 폐기자, 폐기일, 폐기사유 등의 속성을 갖는다. 신규도면 작성시 조회가 가능하며, 복사요청시 도면복사가 가능하다.
출도이력	도면의 출력 요청시 작성되는 문서로, 도면번호, 도면명, 출도자, 출도일, 출도매수, 출도용도, 배포처 등의 속성을 갖으며, 도면출력 요청시 도면을 출력한다.
사용자	관리자, 설계자, 검토자, 승인자, 승인자로 역할이 주어지며, 사용권한에 따라 가능한 업무를 수행하는 사원이다. 사원번호, 소속부서코드, 사용지명, 비밀번호, E mail 주소 등의 속성을 가지고 있다.
사용권한	관리자는 사용자들을 생성/조회/삭제와 그 외 객체에 대한 모든 행위를 할 수 있는 권한을 가지고 있다. 설계자는 도면에 대한 생성/읽기/쓰기/복사 등의 권한을 갖으며 도면작성과 수정시 이력 이력을 작성할 수 있다. 검토자는 도면에 대한 읽기만 가능하며 검토시 검토이력을 작성한다. 승인자는 도면에 대한 읽기만 가능하며 승인요청시 승인이력을 작성한다. 권한여부를 체크할 수 있도록 사용자 생성 및 삭제 여부, 도면에 대한 생성, 삭제, 읽기, 쓰기, 복사 여부 등의 속성을 갖는다.

서비스하는 시스템이다. 이것은 도면정보관리시스템, 부품정보관리시스템, 모델정보관리시스템, 엔지니어링 BOM관리시스템, 군분류시스템 등으로 구성된 통합시스템으로 볼 수 있다. 그 기능과 연관관계를 살펴보면 <그림1>과 같이 요약될 수 있다.

모델정보관리시스템은 제품번호, 제품명, 규격, 소비자 가격, 제품의 외형사진 등의 일반적인 제품정보와 제품의 조립, 분해

방법을 알 수 있는 총 조립도, 제품의 구성품목 및 소요수량을 알 수 있는 부품리스트를 관리한다.

부품정보관리시스템은 부품코드번호, 부품명, 공급처, 납기일, 단가 등 부품과 관련되어 설계자에게 필요한 정보를 관리하며, 현 부품이 그려져 있는 도면, 부품의 기술 사양을 담고 있는 문서와의 관계를 관리한다. 또한, BOM시스템과 연관되어

<표 2> 부품정보관리모듈에서 식별된 객체들

객체명	객체 설명
부품정보	부품정보관리시스템에서 관리하는 상위 개념의 객체로, 부품이력을 소유하고 있으며, 부품번호, 부품명, 도면번호, 적용모델명, 부품규격, GT코드정보, 삭제정보, 공급업체정보, 대치품정보, 부품등록일 등의 속성을 갖는다. 또한 부품생성품, 조립품, 부분품들로 구성된다. 또한, 부품관련 정보를 생성, 수정, 조회, 삭제, 출력할 수 있다.
GT-CODE	부품이력을 구성하는 요소로 군분류코드 정보를 관리하며, 각 군분류코드 및 코드설명 등의 속성을 가진다. 또, 군분류코드를 생성, 수정, 삭제, 조회 등의 기능을 수행한다.
공급업체	부품이력을 구성하는 요소로 부품의 공급업체정보를 관리하며, 공급업체코드, 공급업체명, 업체부품코드, 업체품명, 등록일 등의 속성을 가진다. 또한 생성, 수정, 삭제, 조회 기능 갖는다.
대치품	부품이력을 구성하는 요소로 부품의 대치정보를 관리하며, 대치품코드, 대치품명, 삭제사유, 등록일 등의 속성을 갖으며, 대치된 날짜와 대치된 생산번호에 따라서 부품을 대치하는 기능을 갖는다.
담당사	부품을 담당하는 사원으로서, 사원번호, 성명, 소속부서코드, 비밀번호, 권한 등의 속성을 가지고 있으며, 부품생성시 군분류코드를 이용하여 유사부품을 조회할 수 있다.
생산품	업체가 납품하는 부품으로, 부분품과 조립품 개개가 생산품이 될 수 있다. 이는 부품의 한 종류로서, 부품번호, 생산품코드, 생산품명 등의 속성을 갖는다. 또한, 모델정보와 BOM정보에 참조되어진다.
조립품	단품인 부분품들을 조립하거나 조립품들이 조립품을 말하며, 이는 부품의 한 종류로서 부품번호, 부품명, 하부조립품 및 부분품정보 등의 속성을 가지며, 모델정보와 BOM정보에 참조되어진다.
부분품	단품을 말하며, 부품의 한 종류로서 원재료등도 포함되며, 부품번호, 부품명 등의 속성을 갖으며, 모델정보와 BOM정보에 참조되어진다.

<표 3> BOM정보관리모듈에서 식별된 객체들

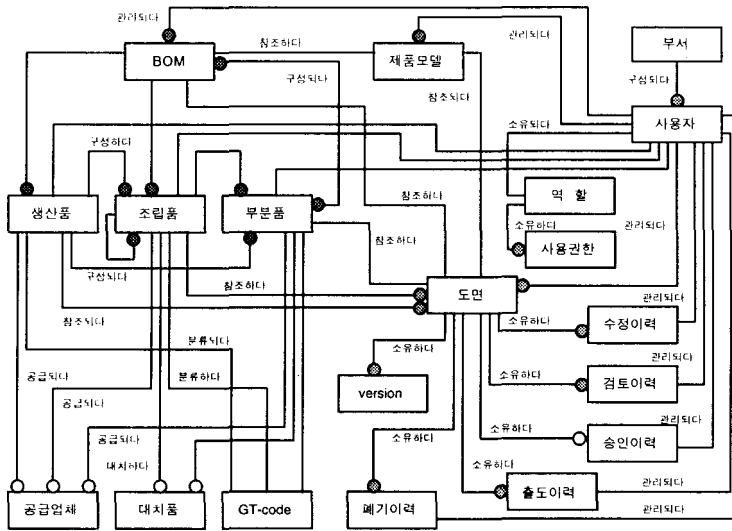
객체명	객체 설명
BOM	부품구성을 관리하기 위한 객체로서, 모부품, 자부품, 부품의 위치, 수량 등의 속성으로 구성되어 있으며, 이는 다시 BOM 클래스를 객체로서 갖는다. BOM 정전개 및 역전개 기능 수행을 할 수 있다.

BOM정보를 관리하고, 군분류시스템과 연관이 되어 군분류코드정보를 관리한다.

도면정보관리시스템은 도면과 관련된 정보를 관리하는 시스템으로, 도면번호, 도면명, 부품혹은 조립품의 이름, 작성자, 검토자 작성일자 등의 정보를 관리하며, 도면의 생성, 수정, 삭제, 조회 등의 기능을 수행한다. 또한, 도면정보시스템의 중요한 기능으로 버전관리가 있으며, 일단 검토를 거쳐 출도되면, 이 도면은 작성자라 할지라도

수정할 수 없는 상태가 된다. 도면의 수정을 위해서는 같은 도면에 수정번호를 추가하여 버전이력을 기록하여야만 개정도를 작성할 수 있게 관리한다.

엔지니어링 BOM관리시스템은 부품의 속성과 모품번호, 자품번호, 소요수량 등을 입력하는 엔지니어링 BOM 입력기능과 부품의 소요수량을 정전개 및 역전개 방법으로 탐색하여 부품구상 목록을 보여주는 BOM 처리 기능로 구성되어 있다.



<그림 2> 클래스 연관성 다이어그램

군분류코드관리시스템은 부품의 군분류코드를 작성하거나 군분류코드 수정, 조회, 삭제 등의 기능을 수행한다. [삼성HP, 1993; 김선호 등,1996]

4.2 객체 모델링

객체 모델링은 실 세계 분해 영역을 표현하는 객체 및 클래스의 식별과 이들의 관계를 연관화, 집산화, 일반화 관계를 중심으로 규정한다. 여기서 클래스의 성질을 나타내는 속성과 연산을 함께 표현함으로써 시스템의 정적인 구조를 표현한다.

4.2.1 모델 구축을 위한 객체도출

설계정보관리시스템에는 크게 3가지 모듈, 즉 도면 객체와 관련된 도면정보관리 모듈, 부품 객체와 관련된 부품정보관리 모

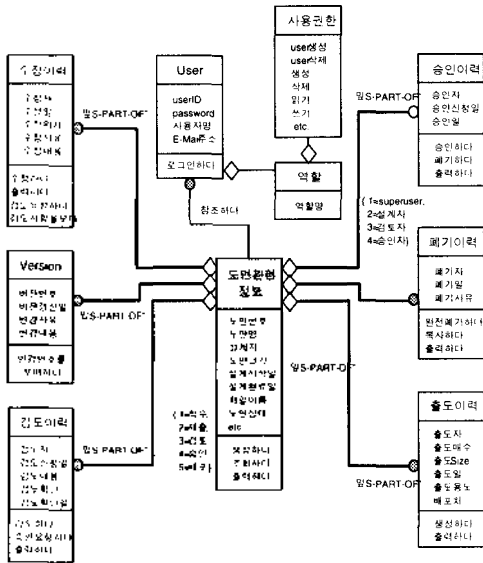
듈 및 BOM 모듈로 나눌 수 있다: 객체 모델링 기법을 적용하기 위해서, 위에서 설명된 설계정보관리시스템에 대한 클래스 객체를 식별한 것이 <표1-3>과 같다.

4.2.2 연관성 정의

식별된 클래스들의 연관성을 식별하기 위해서는 각 클래스간 즉, 이미 정의된 객체 클래스를 중심으로 객체, 클래스의 상호 연결 관계, 다중성을 정의해야한다. 이는 클래스간에 전달되는 메시지를 표현함으로써 시스템의 흐름과 클래스간의 연관 관계를 파악할 수 있기 때문이다. 설계정보관리시스템에 관련된 클래스들간의 연계성을 다이어그램을 나타낸 것이 <그림2>와 같다.

집산화(aggregation)는 서로 관련이 있는 여러개의 객체(components)를 묶어

한 개의 상위 객체를 표현하는 것으로, "a-part-of"의 관계성이 있다. 다른 객체로



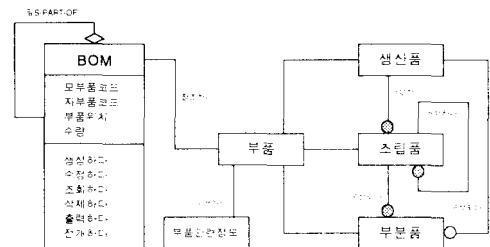
<그림 3> 도면관련정보클래스의 집산화

구성되는 객체를 복합 객체(composite object)라고 하며, 집산화는 복합 객체의 종속 성분을 모델링하기 위해 사용되어지며, 복합/성분(assembly/component) 클래스 관계를 통해 복합 속성 계층(composite attribute hierarchy)을 형성한다[. Rumbaugh et al.,1991].

객체의 속성이 임의의 클래스를 그 영역으로 할 수 있는 복합객체(composite object)에 대한 표현으로 도면관련정보클래스를 예로들면, 기존의 관계형 데이터베이스에서의 모든 정보는 각 정보별로 여러 개의 테이블을 만들고 도면번호를 Primary key로 하여 다른 테이블과 연결시켰다. 그러나 객체지향형 도면정보관리모듈에서는

수퍼클래스인 도면관련정보에서 수정이력, 검토이력, 승인이력, 폐기이력, 출도이력, 폐기이력, 버전이력 등 일련의 작업흐름을 기치면서 작성되는 이력사항들을 한 도면에 종속시켜 관리하게 된다. 즉, <그림3>과 같이 도면관련정보클래스는 수정이력클래스, 검토이력클래스, 승인이력클래스, 폐기이력클래스, 출도이력클래스, 버전이력클래스 등과 Composite Relationship을 가지게 된다.

BOM 객체들은 <그림4>에서 보듯바와같이 전형적으로 원재료의 집합이나 시브조립품 객체들로서 구성되어질 수 있다. BOM 객체는 3가지 객체 즉, 부분품(component), 조립품(assembly), 생산품(product)를 갖는다. 부분품 객체들은 원재료 및 단품으로써 설계되어진 객체들이며, 조립품 객체들은 부분품들 또는 어떤 서브조립품들로 구성된다. 또한 생산품(product) 객체들은 인체가 납품하는 부품들로, 부분품과 조립품 개개가 생산품이 될 수 있으며, 조립품 및 부분품 객체들로 구성된다.



<그림 4> BOM 클래스의 집산화

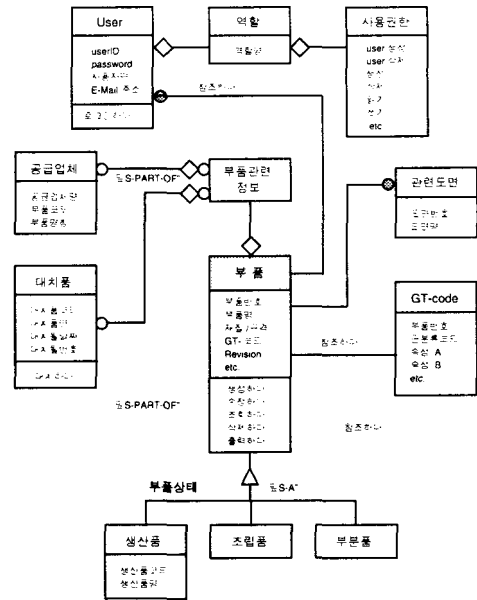
4.2.3 속성식별과 상속성 정의

클래스 관계성 다이어그램에서 작성된 객체모델을 바탕으로 객체의 연산과 인스턴스 변수 등 객체 특성을 기술하여 클래스에 관련된 연산 및 클래스간의 상호연관 관계를 표현할 수 있다. 이렇게 클래스에 연산과 객체의 특성을 함께 표현하면 데이터 모델과 프로그래밍 언어의 구현을 쉽게 처리할 수 있게 한다.

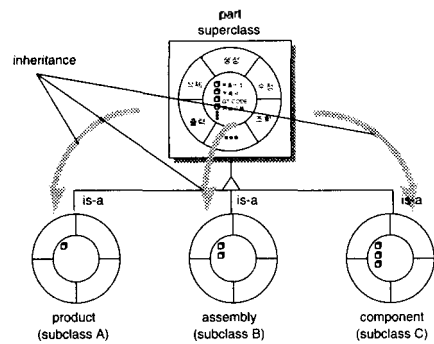
<그림5>에서 보는바와 같이 생산품클래스, 조립품클래스, 부품클래스는 각각 속성들로 부품번호, 부품명, 재질, 규격, GT-코드 등을 공통적으로 가지며, 생성하다·수정하다·조회하다·삭제하다·출력하다 등의 함수들 또한 공통적으로 가지고 있다. 이렇게 공통된 속성과 행위들을 가지고 있는 클래스들을 슈퍼클래스인 부품클래스와 같이 추상클래스를 생성하여 상속성 개념인 일반화(generation)관계를 적용할 수 있다. 예를들면, <그림6>에서와 같이 부품은 슈퍼클래스가 되며, 생산품, 조립품, 부품품은 서브클래스가 되어 부품에 있는 속성 및 행위들이 서브클래스들로 상속된다.

또한, <그림5>에는 부품과 관련된 정보들인 공급업체클래스, 대치품클래스, 관련도면클래스들을 부품관련클래스로 클래스들을 종속시킴으로써, 클래스 집단화를 형성한 것을 보여주고 있다. 부품클래스는 클래스 속성으로 부품관련클래스를 가지며 집합객체를 구성한다.

<그림7>은 설계정보관리시스템에 있는 부품정보, 도면정보, BOM정보에 대한 이들 클래스의 속성과 클래스 상속성을

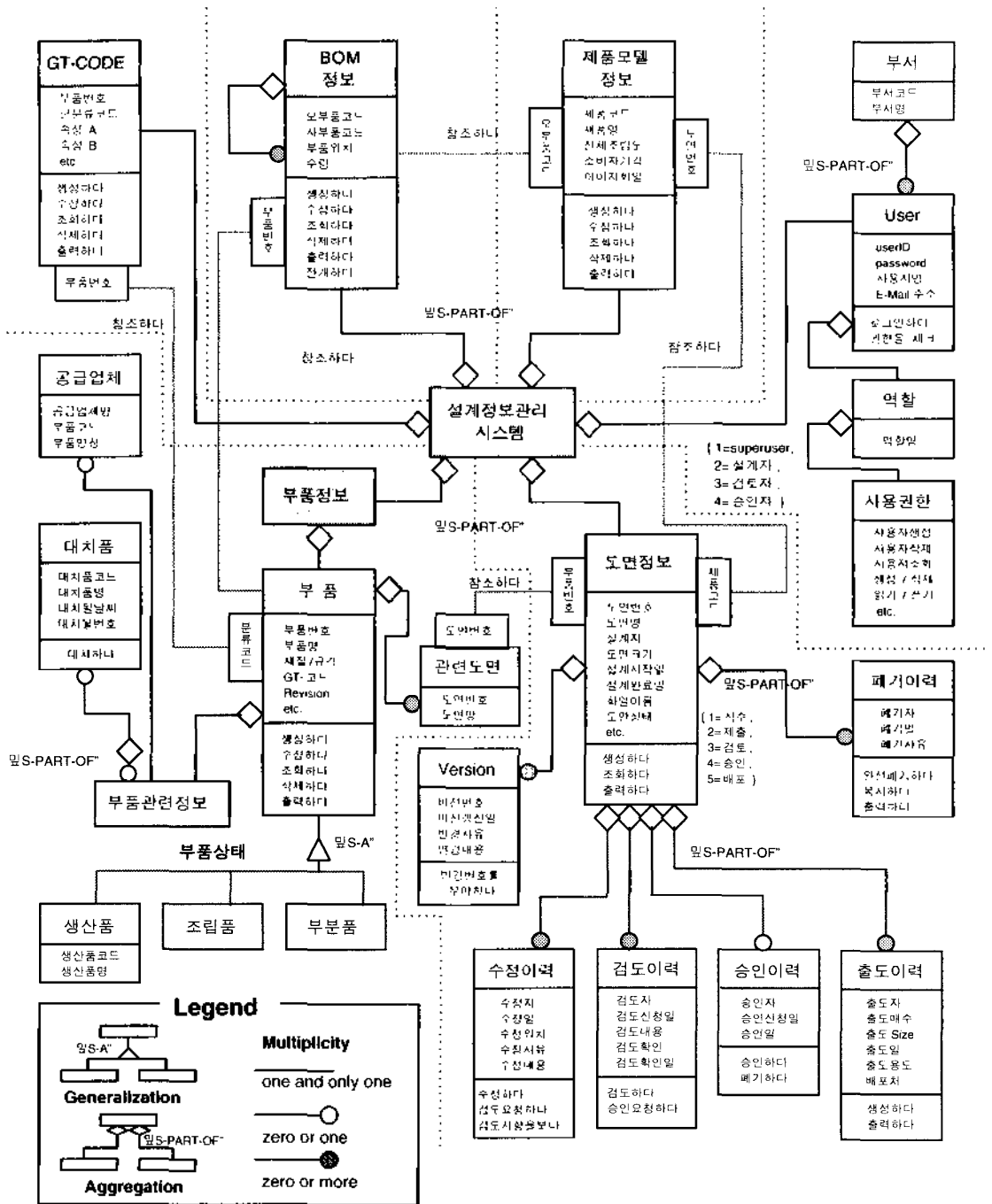


<그림 5> 부품관련클래스의 집단화 및 부품클래스의 일반화



<그림 6> 부품클래스의 상속성

하나로 표현한 객체 모델 다이어그램을 나타낸 것이다.



<그림 7> 설계정보관리시스템의 객체 다이어그램(Object diagram)

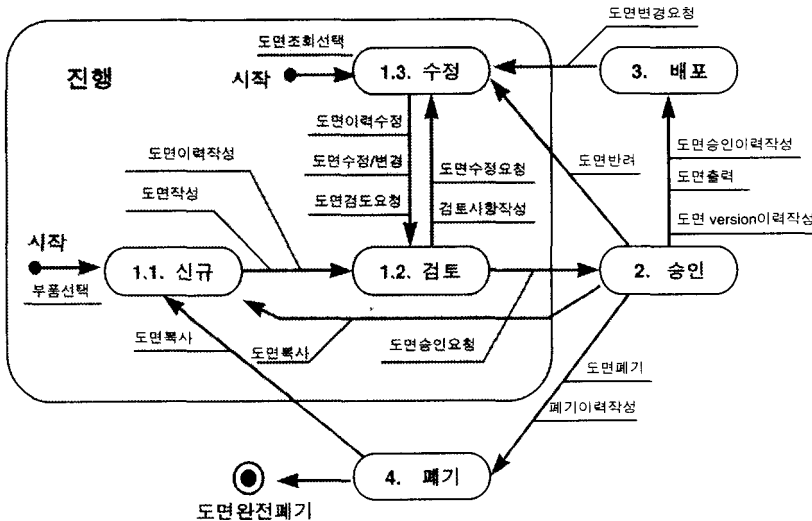
4.3 동적 모델링

동적 모델링은 객체 상태가 변환되어 가는 과정을 중요시하며, 시스템에서 병행 (concurrent)으로 활동하는 객체들이 수행하는 순서, 제어의 흐름, 그리고 상호작용하는 관계를 다루는 모델이다. 즉, 객체의 행위를 중심으로 객체가 변화하는 상태를 정의하고, 시간이나 사건의 발생에 따라 어떻게 상태 변화의 흐름이 이어지는가에 대한 객체의 생명주기(object-life-cycle)를 분석한다.

설계정보시스템에서 도면을 예로 들면, 도면은 진행상태, 승인상태, 폐기상태, 배포 상태 등 4종류의 상태로 구분할 수 있다. 진행상태는 다시 신규상태, 수정상태, 검토 상태 등을 서브상태를 갖고 있는 수퍼상태

를 구성한다. 작성자가 신규도면을 선택하면, 도면에 대한 이력을 작성해야 한다. 기존에 유사한 도면이 승인도면이나 폐기도면으로 존재한다면 도면을 복사하여 도면을 작성한다. 유사도면이 존재하지 않으면 작업자는 도면을 새로 작성한다. 또한, 검토된 도면을 수정하기 위해서 도면을 조회한 후, 도면수정을 하며, 도면수정이력을 작성한다. 도면작성 및 수정을 다 마치면 설계자는 가상팀(virtual team)으로 구성된 검토자들에게 검토요청을 한다. 이때까지 도면은 “진행”상태를 유지한다. 검토자들은 검토요청을 통해 도면을 검토하며, 검토사항을 작성한다. 이러한 검토사항을 토대로 수정자는 다시 도면을 수정한다.

수정이 다 되면, 다시 검토를 거치고 검토자들은 승인요청을 승인자에게 한다.



<그림 8> 도면에 대한 상태 다이어그램(State diagram)

승인요청된 도면은 “진행” 상태에서 “승인” 상태로 변한다. 승인자는 검토자들의 검토 요청에 의해 도면에 대한 승인 작업을 한다. 승인자가 검토사항을 검토한 후, 도면에 대한 승인을 거부하고 다시 도면을 수정하게 할 수 있으며, 이때 도면의 상태는 “승인”에서 다시 “진행” 상태로 변화된다.

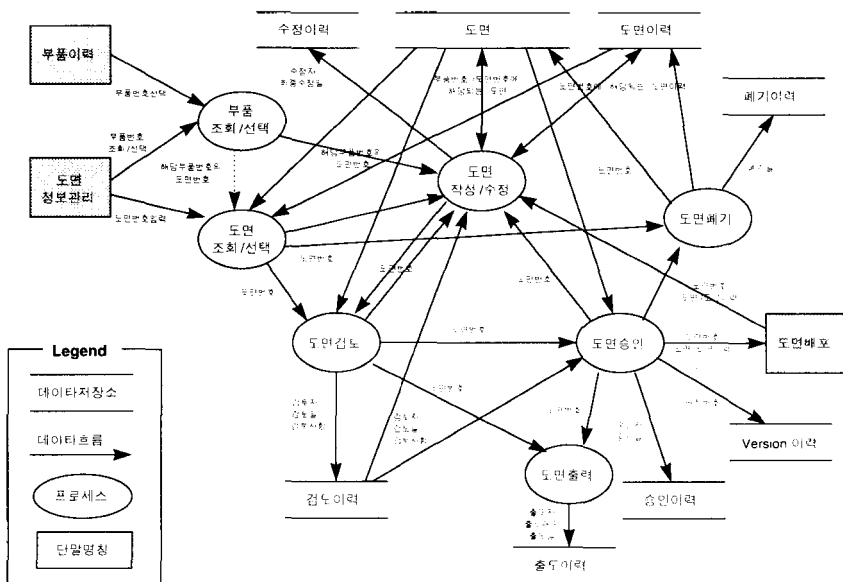
승인자가 도면에 대해 승인을 하게 되면, 도면은 “승인” 상태에서 “배포” 상태로 되며, 배포 대기를 하게 된다. 배포 대기 에 있는 도면은 일정 기간이 지난 후 배포날짜에 맞춰 관련부서로 배포를 하게 된다. 배포된 도면에 대해 다시 관련부서의 수정요청시에는 “배포” 상태에서 “진행” 상태로 변하며 도면을 수정하게 된다. 이 수정요청된 도면에 대해 수정작업이 완료되어 승인이 이루어지면 도면은 버전

(version)이 증가하게 된다.

또한, 승인자가 도면에 대한 폐기 승인을 하게되면 도면은 “폐기” 상태로 변한다. 이때 도면은 완전히 폐기되는 것이 아니라 폐기라는 상태로 일정기간 보관이 된다. “폐기” 상태에 있는 도면을 완전폐기시키면, 폐기상태로 데이터베이스에 존재하던 도면의 이력과 도면이 완전히 삭제된다. <그림8>은 이와같이 도면에 대한 상태변화에 대한 번이 과정을 보여준다.

4.4 기능 모델링

이 기능 모델링 단계에서는 값과 값 또는, 함수에 의한 값의 기능적 종속성을 데이터 흐름 다이어그램(DFD : Data Flow Diagram) 프로세스의 순서나 객체의 구조



<그림 9> 도면정보모듈에 있어서 데이터 흐름도(Data flow diagram)

에 부관하게 한 시스템 내의 여러값들이 어떻게 처리되어 지는가를 정형적으로 나타낸 것으로 DFD를 이용하여 데이터 흐름을 중심으로 처리하는 과정을 표현한다. <그림9>는 도면정보모듈에 대한 DFD를 나타낸다.

5. 결론 및 향후 발전방향

본 연구에서 분석모델링된 설계정보시스템은 관계형 데이터베이스상에서 구현된 시스템을 객체지향 패러다임을 이용하여 '객체' 관점에서 분석하였으며, 시스템에 관련된 객체와 클래스를 추출하여 클래스들 사이의 관계를 연관화(association), 집산화(aggregation), 일반화(generation) 관계로 정의하여 클래스의 속성과 연산을 함께 표현한 정적인 객체모델링을 생성하였다. 또한 객체의 각 상태와 그와 관련된 사건들을 상태다이아그램을 이용하여 동적모델링을 하였으며, 부품정보관리모듈과 도면정보관리 모듈등 설계정보관리시스템 각 모듈에 관계된 다수의 프로세스들간의 데이터 흐름을 분석하여 데이터 흐름 다이어그램을 이용하여 기능 모델링을 하였다.

객체지향으로 모델링된 시스템의 특징으로 첫째, 객체의 속성과 이에 대한 연산이 하나로 캡슐화시켜 변경으로 인한 영향을 지역화 할 수 있고, 한 객체가 다른 객체의 종속성을 최소화 할 수 있다는 점, 둘째, 공동된 속성들을 명확히 표현하여 상속을 시킴으로써 재사용성을 높일 수 있다

는 점이다.

그러나, 제시된 객체지향 설계정보시스템은 작업의 흐름(workflow)을 고려하지 않고 설계하였기 때문에 동시공학적인 변을 모델링하지 못했으며, BOM에서 구성관리(Configuration Management) 또한 고려하지 않았다. 앞으로 PDM시스템의 구현을 위하여 이러한 점들을 반영하는 연구를 계속할 계획이다.

참고문헌

- [김광수 등, 1992] 김광수, 김철환, “객체지향적 접근방법을 이용한 생산정보시스템 모델링방법”, 대한산업공학회 '92 추계학술대회발표회 논문집, pp192-201, 1992.
- [김선호 등, 1994] 김선호, 윤희철, “Technical Document Management System을 위한 도면정보 관리시스템 개발”, IE Interfaces 산업공학, Vol.7, No.3, pp213-225, 1994.
- [김선호 등, 1996] 김선호, 신용하, 문희석, “동시공학적인 도면정보관리시스템 개발”, IE Interfaces 산업공학, Vol.9, No.1, pp41-52, 1996.
- [김철호 등, 1994] 김철호, 임종근, 정운영, 김우승, “선계 DATA의 통합을 위한 EDB시스템의 구조에 관한 연구”, IE Interfaces 산업공학, Vol.7, No.3, pp137-145, 1994.
- [삼성HP, 1993] 삼성HP, CIM 실천전략(I)(II), (주)컴퓨터리엔지니어링, 1993.
- [신영길 등, 1995] 신영길 외3명, “기술정보관리를 위한 통합솔루션의 효율적 구조”, IE Interfaces 산업공학, Vol.8, No.3, pp197-202, 1995.
- [유상봉 등, 1995] 유상봉, 서효원, 고영욱, “STEP을 이용한 생산 시스템의 제품 데이터 교환”, IE Interfaces 산업공학, Vol.8, No.3, pp75-95, 1995.
- [이충화, 1996] 이충화, “CALs/PDM을 이용한 연구개발 통합시스템 구축 사례”, IE 매거진, Vol.3, No.1, pp58-62, 1996.
- [이충화 등, 1996] 이충화 외 7명, “제품개발 프로세스 개선을 위한 동시공학시스템 K-CE 개발”, IE Interfaces 산업공학, Vol.8, No.3, pp25-37, 1996.
- [최성운, 1996] 최성운, 노홍석, 계원성, “객체지향 소프트웨어 개발 방법론 동향”, 정보과학회지, Vol.14, No.10, pp4-11, 1996.
- [최열현, 1995] 최열현, “CE 실현을 위한 PDM 시스템”, 컴퓨터월드 12월호, pp162-167, 1995.
- [최영근 등, 1995] 최영근, 허계범, 객체지향 소프트웨어 공학, 한국실리콘, 1995.
- [CIMdata, 1994] CIMdata, Prodcut Data Management : A Technology Guide, 1994.

- [D. A. Taylor, 1990] David A. Taylor, Object-Oriented Technology : A Manager's Guide, Addison-Wesley Publishing Company, 1990.
- [D. A. Taylor, 1995] David A. Taylor, Business Engineering with Object Technology, John Wiley & Sons, Inc., 1995.
- [J. Rumbaugh et al., 1991] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorensen, Object-Oriented Modeling and Design, Prentice-Hall International, Inc., 1991.
- [HP, 1993] HP, Product Data Management : Understanding the Fundamental Technology and Business Concepts, Hewlett-Packard Company, 1993.
- [Y. Chung et al., 1994] Y. Chung, G. W. Fischer, "A Concepture and Issues for an Object-Oriented Bill of Materials(BOM) Data Model", Computers & Industrial Engineering, Vol.26, No.2, pp321-339. 1994.
- [W. Kim, 1990] Won Kim, Introduction to Object-Oriented Databases, MIT Press, 1990.

저자소개

오태훈

명지대학교 산업공학과 대학원생
관심분야 : 객체지향형 PDM 설계 및 개발

김선호

서울대학교 산업공학과 학사
Pennsylvania State University 산업공학 석·박사
국방과학연구소, 한국기계연구원 근무
현재 명지대학교 산업공학과 부교수
관심분야 : PDM 개발, 객체지향형 모델링, CAT, tolerance optimization