

# CORBA 기반의 이종분산환경용 제품구조 및 구성관리 시스템 개발

김선호<sup>1</sup> · 권용성<sup>2</sup> · 주경준<sup>3</sup> · 정석찬<sup>4</sup>

<sup>1</sup>명지대학교 산업공학과 / <sup>2</sup>PTC Korea / <sup>3</sup>ETRI 컴퓨터소프트웨어기술연구소 / <sup>4</sup>동의대학교 경영정보학과

## Development of the CORBA-based Product Structure and Configuration Management System for Heterogeneous Distributed Environments

Sun-Ho Kim<sup>1</sup> · Yong-Sung Kwon<sup>2</sup> · Kyung-Jun Ju<sup>3</sup> · Seok-Chan Jeong<sup>4</sup>

In this research we have developed a prototype which manages a product structure and configuration of a product under distributed heterogeneous computing environments. First, the data-schema for product structure and configuration which can represent options and effectivity is proposed. Second, a technique to use JAVA and CORBA is suggested for the exchange of product structure among multiple servers in CITIS environments. Functions of the prototype related to option-based product structure are described to help readers' understanding.

### 1. 서론

설계단계에서 설계자가 CAD를 이용하여 제품을 설계하면 다음 단계로서 제품구조(product structure) 및 제품구성(configuration) 정보를 생성하게 된다. 제품구조는 부품간의 상하 조립관계를 나타내는 것으로서 상위부품, 하위부품, 소요수량 등의 정보를 가지고 있다. 제품구성은 한 제품의 구성 또는 조립사양을 나타내는 것으로서 선택사양, 대체품 등의 정보를 포함하고 있다. 제품구조에는 parts list 구조와 BOM (Bill of Materials) 구조가 많이 이용되고 있으나 (ISO 10303-44, 1994) parts list 구조가 중간조립품의 재고를 파악하기 어려운 단점이 있어 점차 BOM으로 대체되고 있다.

제품구조 및 구성정보를 현장에서 제대로 활용하기 위해서는 일반적인 BOM 구조뿐만 아니라 보다 복잡한 형태의 BOM 구조를 효율적으로 관리할 수 있어야 한다. 즉, 업체에서 관리하는 모델이 다양해지고 선택사양이 많아지면 관리해야 할 제품의 구성 종류는 기하급수적으로 많아지며 그와 관련하여 BOM 정보 역시 제품의 구성별로 모두 관리를 해야 하므로 중복되는 데이터가 많아진다. 또한, 한 부품이나 중간조립품에 대한 설계가 변경되면 이와 관련된 모든 BOM 정보를 찾아 전부 수정하여야 하는 불편함이 따르게 된다. 따라서 이러한 문

제점을 해결하기 위하여 구성을 모두 고려하여 하나의 구조로 표현할 수 있는 BOM이 필요하다.

제품정보는 한 업체에서 모두 관리하는 경우도 있지만 여러 업체에 분산시켜 관리하는 경우도 있다. 또한 분산되어 있는 정보를 각각 관리하는 업체들간의 환경, 즉 OS나 정보가 저장된 DB 구조가 서로 다를 경우도 쉽게 찾아볼 수 있다. 예를 들어 CITIS (Contractor Integrated Technical Information Service) 환경에서는 여러 업체간에 제품 정보를 공유하고 통합화된 환경을 구성하게 된다 (MIL-STD-974, 1993; 정석찬 외 3인, 1997). 따라서 CITIS 환경하에서는 제품정보 및 제품구조정보, 그리고 관련된 도면, 문서정보 등을 효율적으로 관리하는 기능이 필요하다. 이를 위해서는 이종분산환경에서 데이터를 교환할 수 있도록 시스템이 구축되어야 한다. 최근에 인터넷이 많이 활용되고 있으며, 플랫폼에 관계없이 web상에서 활용될 수 있도록 JAVA 언어가 많이 활용되고 있다. 이종분산환경에서 데이터를 교환하기 위해 객체지향 개념의 CORBA, DCOM 등의 미들웨어가 이용되고 있다 (OMG, 1998).

본 연구에서는 CITIS 환경에서 활용되는 PDM 시스템의 일부인 제품구조와 구성관리 시스템을 개발하였다. 이를 위하여 우선, 제품구성을 모두 고려하여 하나의 구조로 표현할 수 있는 데이터 구조를 설계하였다. 특히 제품구성 관리에서 주요 관리대상인 옵션(option)과 유효성(effectivity)을 효율적으로

관리할 수 있는 방안을 제시하였다. 또한 분산환경하에서 제품구조 정보의 교환방법과 이를 위한 JAVA와 CORBA의 활용 기술을 제시하였다. 이 개념들을 근거로 하여 제품구조와 구성 관리를 위한 프로토타입을 개발하였다.

## 2. 제품구조와 구성 관리

### 2.1 제품구조와 구성 관리의 개념

제품구조는 부품간의 상하 조립관계를 나타내는 것으로서 상위부품, 하위부품, 소요수량 등의 정보를 가지고 있다. 제품 구성은 한 제품의 구성 또는 조립사양을 나타내는 것으로서 옵션(option)과 같은 변형품(variant) 정보, 대체품 및 유효성(effectivity) 정보 등을 포함하고 있다(김선호, 권용성, 정석찬, 조장혁, 1998; 김선호, 1999). 제품구조는 데이터의 구조 관점에서 볼 때, 일반적으로 BOM 구조, parts list 구조로 구분된다(ISO 10303-44, 1994). BOM 구조는 최종제품, 중간조립품, 부품, 원자재의 조립관계를 표현하고 있으며 상하관계, 소요수량, 조립공정순서 등의 정보를 가지고 있다. parts list 구조는 최종제품, 중간조립품, 부품간의 조립관계를 표현하고 있다. 한 상위부품의 바로 아래에 동일한 하위부품이 여러 개 있을 경우, BOM 구조는 하나의 하위부품명과 그 하위부품의 소요수량정보를 가지게 된다. 그러나 parts list 구조는 하위부품 각각의 정보를 가지게 되며 소요수량이 1이 된다. 또한, parts list 구조는 조립관계를 나타내므로 바로 위의 상위부품과의 관계뿐만 아니라 더 위의 상위부품과의 관계 정보도 가지게 된다. 일반적으로 parts list 구조는 부품간의 조립관계를 표현하므로 CAD 3D 모델러에서 많이 이용되고 있으며 BOM 구조는 부품의 소요수량 파악과 재고관리에 적합하여 PDM, ERP 시스템에서 많이 활용되고 있다.

지금까지 제품구조 및 제품구성을 고려하는 BOM에 대한 연구가 많이 진행되어 왔다. 고석완(1998), Trappey, et al.(1995), Chung, et al.(1994)은 BOM 정보와 제품정보를 동적으로 표현하기 위하여 Relational DB 대신에 객체지향 개념을 이용하여 BOM 시스템을 설계하였다. 그러나 이러한 연구에서는 제품구조에 대한 내용을 중심으로 설계하였으며 제품구성의 개념은 별로 고려되지 않고 있다.

제품구성의 기본인 변형품을 표현하는 방법으로 지금까지 modular BOM, family BOM, matrix BOM, add/delete BOM 등이 소개되어 있다(Tersine, 1994; 김선호, 문희석, 1997; 이한표, 이춘열, 이국철, 1995). Modular BOM은 제품구조 상위의 주요부품들을 공통부품 그룹과 변형품 그룹으로 구성하고 상위레벨에서 변형품 조합의 BOM을 구성하는 방식이다. Family BOM은 modular BOM과 유사하나 최상위부품을 유사한 그룹으로 분류하는 것이 다르다. 이 방식들에서는 제품구조 상위레벨의 주요 부품들의 소요량만 파악할 수 있고 하위레벨의 단위부품

까지 소요량을 파악하기 위해서는 별도의 그룹별 BOM 구조를 다시 만들어야 하는 불편함이 있다. matrix BOM은 부품의 수량을 제품별로 테이블의 형태로 나열하고 변형품도 명시하는 방식이나 제품의 구조가 표현되지 않아 중간조립품의 정확한 소요량을 파악하기 어려운 단점이 있다. add/delete BOM은 변형품이 생길 때마다 부품을 해당부품을 교체, 추가, 삭제하는 방식이다. 그러나 부품을 추가, 삭제하면서 제품구조를 변형해야 하는 번거로움이 있다.

최근에는 변형품을 표현하는 방법으로서, 변형품 그룹에 generic code number를 주고, 변형품을 구분하는 색깔이나 기능 등의 특성에는 parameter 값을 주는 generic BOM이 제시되었다(Erens, 1996; Hegge, Wortmann, 1991). 여기서는 parameter가 상하 부품간에 상속되도록 정의하여 상위에서 변형품을 정의하면 하위부품에서 그 변형품이 선택되도록 구성하였다. McKay et al.(1996)은, 제조의 관점(assembly view)에서 개별적인 제품들로 표현된 기존의 데이터베이스 모델과, 영업의 관점(sales view)에서 변형품이 많은 product family를 나타내는 모델을 데이터의 중복 없이 통합하여 표현할 수 있는 framework-based product data modeling 기법을 제시하였다.

여기서는 이러한 변형품을 표현하는 방법으로서 피쳐(feature)와 옵션을 고려하여 제품구조 및 구성을 표현하는 옵션 BOM을 제시하였다. 여기서는 피쳐는 옵션부품들을 대표하는 이름으로서 실제부품이 아닌 가상부품으로 처리된다. 그래서 사용자는 피쳐별로 옵션부품을 선택하여 제품구성을 하도록 되어 있다.

### 2.2 피쳐와 옵션을 고려한 옵션 BOM 구조

제품의 일반적인 BOM 구조는 <그림 1>에서와 같이 부품의 상하 관계와 하위부품의 수량으로 표현된다. 제품들 중에서 판매를 목적으로 제조되는 최상위부품을 일반적으로 모델이라고 부르며 하나의 모델에는 사용자의 요구에 따라 변형품이 존재하게 된다. 즉, 선택할 수 있는 부품 사양이 존재하여 한 모델 내에서 선택적으로 제품을 구성할 수 있게 된다. 이때 이러한 부품을 옵션이라고 한다. 이러한 옵션이 많아질 경우 가능한 제품구성의 수가 옵션 수에 비례해서 증가하게 된다. 예를 들어, 자동차의 한 모델에, 차체 색깔 10종류, 엔진배기량 3종류, 트랜스미션 4종류, 브레이크 2종류의 옵션이 있을 때, 240종류의 제품구성이 존재하게 된다. 이러한 변형품을 고려

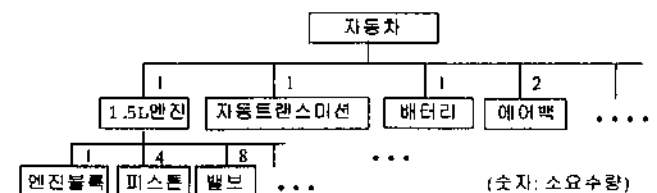


그림 1. 일반적인 BOM의 구조.

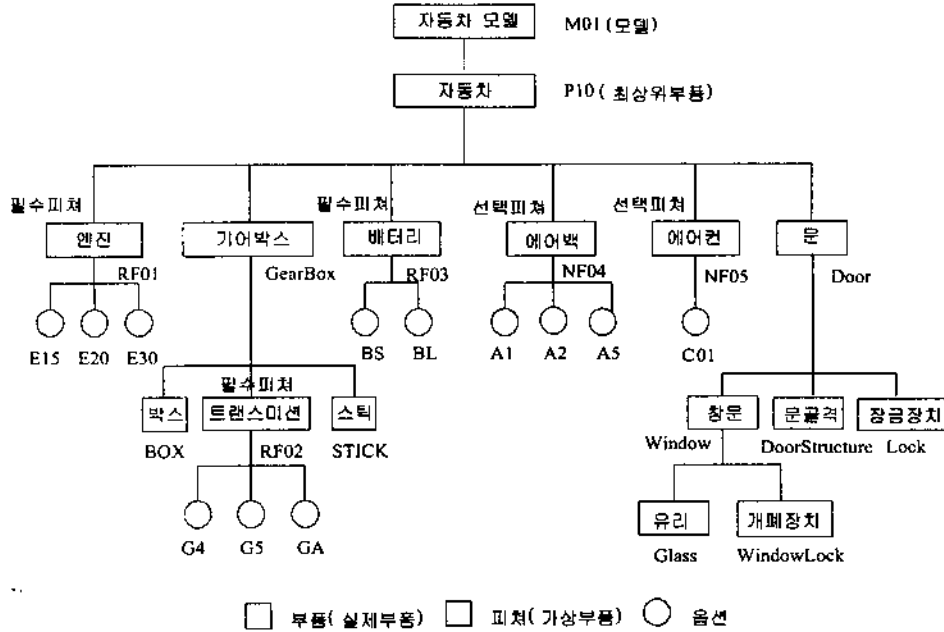


그림 2. 피쳐와 옵션을 고려한 옵션 BOM 구조.

하지 못하는 일반적인 BOM 구조로 저장할 경우 각 제품구성 별로 BOM 데이터를 저장 관리해야 한다. 이때, 중복되는 데이터의 양이 매우 많아지고, 한 부품을 수정할 경우 각 제품구성 별로 그 부품을 찾아 수정해야 하는 불편함이 따르게 된다.

이러한 문제점을 해결하기 위하여 여기서는 여러 가지의 옵션을 하나의 모델로 표현할 수 있는 옵션 BOM을 제시한다. 옵션 BOM에 대한 예제가 <그림 2> (편의상 수량은 생략함)에 나타나 있다. 이 예에서는 최상위제품인 자동차는 엔진, 기어박스, 배터리 등으로 구성된다. 여기서 여러 개의 옵션들을 대표하는 품목명을 나타내기 위하여 피쳐(feature)라는 가상부품이 도입되었다. 엔진, 트랜스미션, 배터리, 에어백, 에어컨은 가상부품에 해당하는 피쳐들이다. 이 피쳐는 그 아래 실제부품인 옵션들로 구성된다. 여기서 엔진 피쳐는 1.5(E15), 2.0(E20), 3.0리터(E30)의 옵션으로, 트랜스미션 피쳐는 4단(G4), 5단(G5), 자동(GA) 옵션으로 구성된다. 피쳐는 다시 필수피쳐와 선택피쳐로 구분된다. 필수피쳐는 하나의 옵션을 의무적으로 택해야 하는 경우이고, 선택피쳐는 옵션을 택하지 않아도 관계없는 경우이다. 이 예에서 엔진, 트랜스미션, 배터리는 옵션을 반드시 하나 택해야 하는 필수피쳐이며 에어백, 에어컨은 택하지 않아도 관계없는 선택피쳐이다.

2.3 옵션 BOM의 데이터 모델

이러한 옵션 개념을 나타내는 E-R 다이어그램이 <그림 3>에 요약되어 있다. 하나의 모델은 여러 개의 피쳐를 가지게 되므로 이것을 표현하기 위하여 모델-피쳐 엔티티가 존재한다. 모델별로 피쳐의 구성이 달라질 수 있으므로 여기서는 모델번호와 피쳐부품번호가 키(key)가 된다. 그리고 이 엔티티에는 각

피쳐가 필수인지 선택인지를 판단하는 피쳐종류 속성이 포함된다. 또한 각 피쳐는 여러 가지 옵션들을 가지고 있으므로 이것을 표현하는 피쳐-옵션 엔티티가 존재한다. 여기서 피쳐-옵션 엔티티의 키는 모델번호, 피쳐부품번호, 옵션부품번호가 되며 모델번호, 피쳐부품번호가 모델-피쳐 엔티티로 연결시켜주는 외부 키(foreign key) 역할을 하게 된다. 그리고 BOM 구조를 나타내기 위하여 BOM 정보 엔티티가 모부품번호, 자부품번호, 수량 등의 속성을 가지게 된다. 피쳐가 고려된 데이터 구조하에서 일반적인 방식으로 BOM을 전개할 경우 가상부품인 피쳐가 포함되어 전개되는 문제점이 있다. 이 문제를 해결하기 위하여 우선, 피쳐를 실제부품과 구별하는 속성을 부품정보 엔티티에 포함하여야 한다. 여기서는 이 속성을 부품종류라고 칭하였으며 최상위부품, 중간조립품, 부품, 원자재, 부자재, 가상부품 등으로 구분된다. 그래서, 부품의 종류가 가상부품이면 피쳐로 간주하고 BOM 전개시 제외할 수 있도록 한다. 여기서 부품종류에 옵션 부품을 별도로 정의하지 않는 이유는 한 모델에서 피쳐 내의 옵션이었던 부품이 다른 모델에서는 일반부품으로 사용될 수도 있기 때문이다.

BOM 전개시 피쳐를 제외하기 위해서는, BOM 정보 엔티티의 모든 자부품번호마다 피쳐인지를 확인하기 위하여 부품정보 엔티티를 탐색하여야 한다. 이것은 외부키를 이용하여 SQL 문장으로 부품정보 엔티티를 다시 검색하게 되므로 전개시간이 길어지는 단점이 있다. 그래서 전개 시간을 줄이기 위하여, 다른 엔티티를 검색할 필요가 없도록 BOM 정보 엔티티에, 모부품과 자부품간에 관계를 나타내는 연결관계 속성을 추가하였다. 이러한 연결관계 속성은 BOM 등록시에 부품정보 엔티티의 부품종류를 참조하여 자동 생성된다. 연결관계 속성은 상하관계로서 부품과 부품, 부품과 피쳐, 피쳐와 옵션 등의 관

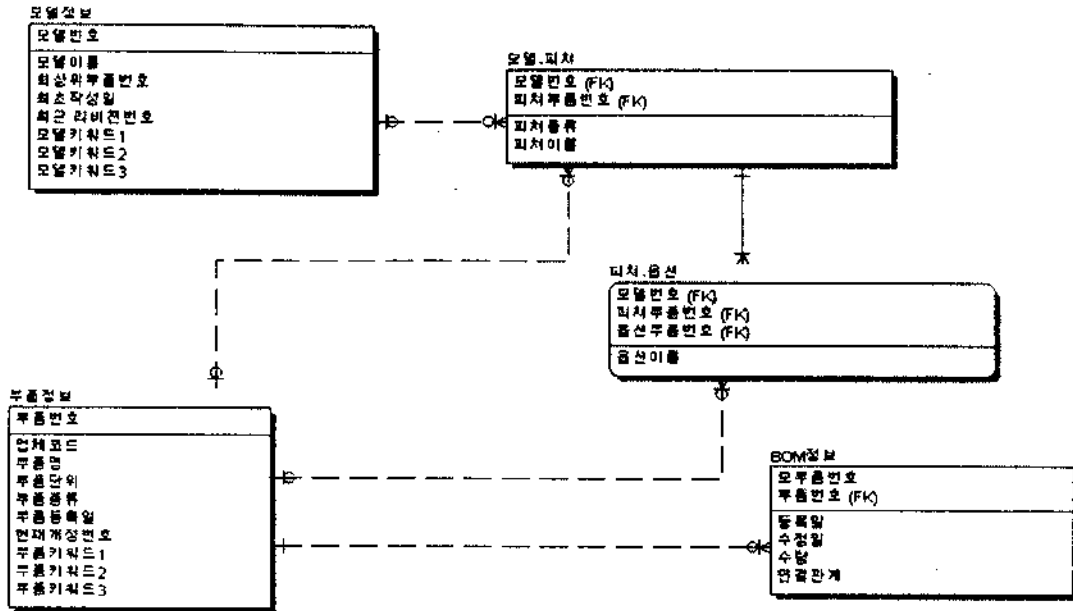


그림 3. 옵션관리를 위한 ER Diagram.

계를 나타내어 BOM 정보 엔티티의 자부품마다 피쳐(가상부품)가 포함되어 있는지를 즉시 알 수 있다. 이 속성은 다음과 같은 값을 갖게 된다.

McT 모부품이 모델이고 자부품이 최상위부품인 경우. 자부품이 최상위부품인 경우  
 모부품은 실제로 존재하지 않으며 모델과 최상위부품은 동일한 부품을 나타낸다.

PrP 모부품과 자부품이 모두 부품인 경우.

PtF 모부품이 부품이고 자부품이 피쳐인 경우.

FrO 모부품이 피쳐이고 자부품이 옵션인 경우.

위의 연결관계속성 값에서 가운데의 "t"를 중심으로 좌측이 모부품, 우측이 자부품을 나타내며, M, T, P, F, O는 각각 모델, 최상위부품, 부품, 피쳐, 옵션을 나타낸다.

### 2.4 유효성 관리

유효성이란 제품을 구성하는 요소들의 변경된 사항에 대해서 실제로 시스템에 변경사항이 적용되어 정보의 효력을 발생시키는 것을 말한다. 유효성 관리를 통해 제품구조 내에 포함되어 있는 부품의 일시적 혹은 영구적인 변경에 대한 정보를 관리할 수 있다. 유효성을 적용하는 방법은 크게 3가지로 구분할 수 있다.

- 가) 제품이 생산되는 동안의 날짜에 의한 방법
- 나) 생산되는 제품의 일련 번호에 의한 방법
- 다) 생산되는 제품에 대한 lot numbers에 의한 방법

본 연구에서는 날짜에 의해 적용하는 방법만을 고려하였다. 날짜에 의한 방법은 정확한 제품구성을 유지할 뿐만 아니라 재고량을 정확히 예측하는 데 이용된다. 만일 이러한 날짜가 관리되지 않을 경우 시스템은 다른 부품이 계속 사용되는 것

표 1. 유효성 정보를 관리하기 위한 테이블 구조

Eff_ Number	모델 번호	소스 부품 번호	대상 부품 번호	부품 사용 시작일	부품 사용 중지일	Effectivity 종류	수량	플래그
1	M01	P03	P31	98.3.1	98.3.15	Substitute	100	F
2	M01	P31	P32	98.3.16	98.3.31	Substitute		P

으로 가정하기 때문에 부품의 현재 재고량이 틀려지게 된다. 날짜를 고려한 유효성의 테이블 구조는 <표 1>과 같다.

위의 <표 1>에서와 같이 저장된 정보는 M01이라는 모델 내의 부품 P03을 부품 P31로 교체하여 98년 3월 15일까지 임시로 사용하겠다는 내용이다. 이것은 유효성 관리를 위한 프로그램이 가동될 때 유효성 정보를 관리하는 테이블에서 부품사용시작일, 부품사용중지일과 현재 날짜를 서로 비교하여 일치하는 날짜가 발견되면 BOM 정보 테이블에서 해당되는 부품을 교체하게 된다. 플래그는 유효성 정보의 이력을 관리하기 위한 것으로 P(Planning), P(Processing), F(Finished)로 구분된다. 날짜에 대한 유효성을 BOM 정보 테이블에 적용시킬 때 최하위부품과 중간조립품에 따라 적용 방법이 약간 다르다. 최하위부품은 그 부품에 대해 유효성만 적용하면 되고, 중간조립품은 그 부품을 포함하여 하위부품들까지 유효성을 적용시켜야 한다.

## 3. 이종분산환경에서의 제품구조 및 구성관리

### 3.1 CITIS 환경에서의 제품구조 정보교환

CITIS(Contractor Integrated Technical Information Service)는 조달자가 시스템 및 제품에 관련된 데이터를 필요로 할 때 주계약

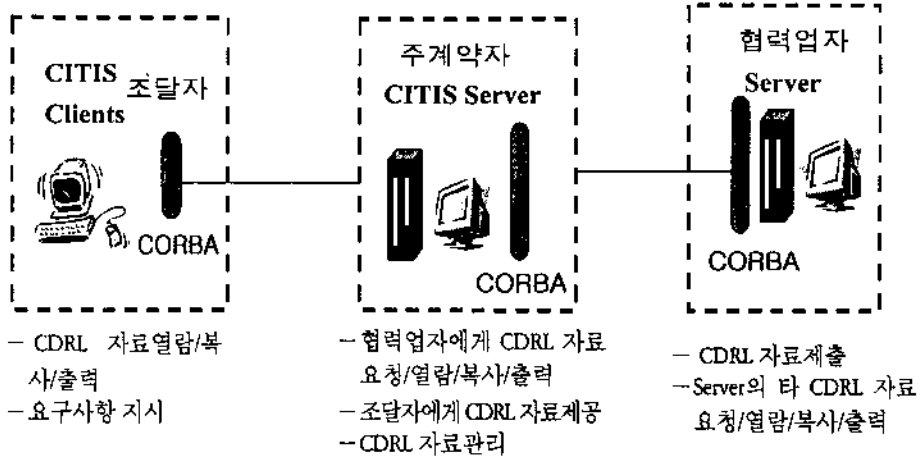


그림 4. CITIS 환경에서의 정보교환.

자가 분산되어 있는 협력업체의 각종 데이터베이스로부터 정보를 취합하여 제공하는 서비스이다(MIL-STD-974, 1993; 정석찬 외 3인, 1997). 이것을 위해서 조달자와 주계약자, 주계약자와 협력업자 사이에서 필요한 정보와 기능은 다음과 같다(<그림 4> 참조).

1) 조달자는 계약자 요구사항 (Contractor Data Requests List, CDRL)에 따라 필요한 제품구성 자료를 주계약자로부터 열람/복사/출력하고, 요구사항을 지시한다.

2) 주계약자는 협력업자에게 필요한 형상품의 성능(performance) 속성을 규정하고, 납품하는 제품/부품에 대한 필요한 자료를 요청 및 출력, 복사 및 열람할 수 있게 된다. 또한 계약에 따라 필요한 정보를 조달자 및 협력업자에게 제공하게 된다.

3) 협력업자는 주계약자가 요청한 제품/부품의 성능사항 및 자료를 제공한다. 또한, 주계약자에게 설계변경 요청을 하게 된다.

이러한 환경에서 제품구조와 구성은 정보를 취합하는 데 중요한 역할을 한다. 일반적으로 협력업자는 제품이나 부품을 설계 및 제조하고, 주계약자는 부품을 납품받아 조립하거나 완제품을 만들어 납품하게 된다. 그러므로 고객인 조달자는 완제품에 대한 전체적인 제품구조정보뿐만 아니라, 그와 관련된 부품이나 문서정보 등을 관리하게 된다. 이러한 정보를 보여주기 위하여 주계약자는 협력업자가 보관중인 부품의 제품구조 데이터를 그대로 가져와 최종제품의 제품구조를 구성할 수 있다(Tersine, 1994). 그리고 협력업체에서 관리하고 있는 한 부품의 도면이나, 문서정보 등이 변경되면 정부나 주계약자가 보관중인 최종제품의 도면, 문서정보 등도 변경되어야 한다. 또한, 그 반대의 경우도 마찬가지이다.

예를 들면, <그림 5>에서 보는 바와 같이 조달자가 제품 A의 제품구조를 보고자 했을 때 조달자 클라이언트는 주계약자 CITIS 서버로 접근(access)하게 된다. 그러나 제품 A는 부품 B와 부품 C로 조립되는데 이 부품 B, C는 협력업체 1과 협력업체 2로부터 공급된다. 그러나 주계약자 CITIS 서버에는 외부로부터 공급되는 부품의 데이터를 모두 갖고 있지 않으므로 협력업체 1과 협력업체 2의 서버로 접근한다. 각 협력업체 서버에

서 부품 B, C의 제품구조정보를 가져오면 주계약자 서버나 조달자 클라이언트에서 이것을 다시 제품 A에 조립하여 완성된 제품구조를 만든다. 결국, 조달자는 이 완성된 제품구조정보를 볼 수 있게 된다(Yarabe, et al., 1997).

위와 같이 주계약자 서버는 협력업체에서 만들어서 가져온 완제품에 대한 부품번호만을 가지고 있고 그 부품이 가지고 있는 제품구조정보는 주계약자 서버에서는 관리를 하지 않고 있다. 따라서 협력업체에서 생산된 부품에 대한 제품구조정보를 보기 위해서는 협력업체에 필요한 정보를 요구하고 협력업체는 요구된 정보에 대하여 정보를 조합하여 주계약자 서버로 정보를 넘기게 된다. 이렇게 외부업체에서 작성된 부품에 대한 제품구조정보를 보기 위해서는 다음과 같은 절차에 따라서 제품구조정보를 얻게 된다(<그림 6> 참조).

- ① 현재 보려고 하는 부품을 부품마스터에서 찾고 이 부품이 어느 협력업체에서 관리되고 있는지를 알아낸다.
  - ② 협력업체에 대한 CAGE(Commercial and Government Entities) 코드를 찾아낸 후 이 정보를 가지고 이 업체의 서버명을 찾아낸다.
  - ③ 협력업체의 서버명을 통하여 CORBA에 부품에 대한 제품구조정보를 호출하게 된다.
  - ④ CORBA는 이렇게 호출된 정보를 서버명과 함수를 통하여 부품이 있는 서버에 접속을 하게 된다.
  - ⑤ 주계약자 서버에서 받아온 부품번호를 가지고 협력업체 서버의 DB에서 그 부품번호를 찾는다.
  - ⑥ 부품을 찾은 후 그 부품이 가지고 있는 제품구조정보를 찾아낸다.
  - ⑦ 찾아낸 제품구조정보를 CORBA에 넘겨준다.
  - ⑧ CORBA는 이렇게 받은 제품구조정보를 다시 주계약자 서버에 돌려주게 되고 받은 제품구조정보를 가지고 주계약자 서버에서는 부품의 제품구조정보를 구성하게 된다.
- 현재 <그림 5>에서는 부품의 제품구조정보만 취합되는 것을 보여주고 있지만 실제로는 이것뿐만 아니라 도면, 문서 등의 정보가 같이 취합되어야 한다. 이러한 기능을 위해서는 문

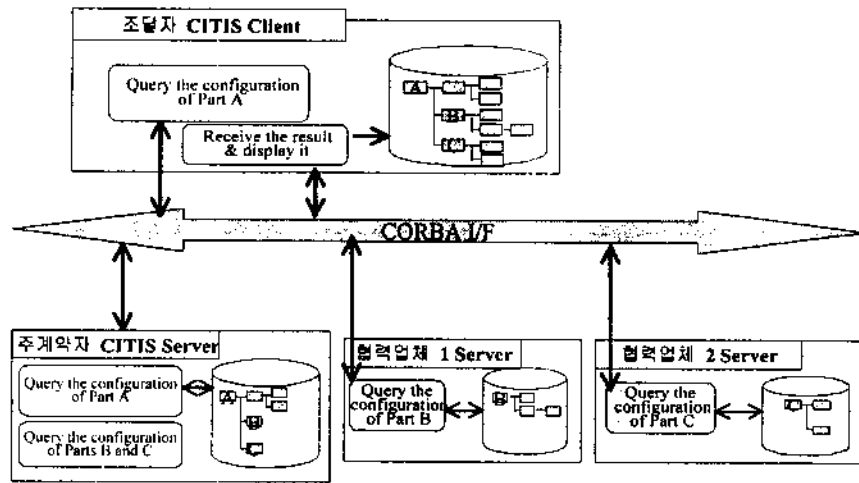


그림 5. CORBA를 이용한 제품구조 조합기능.

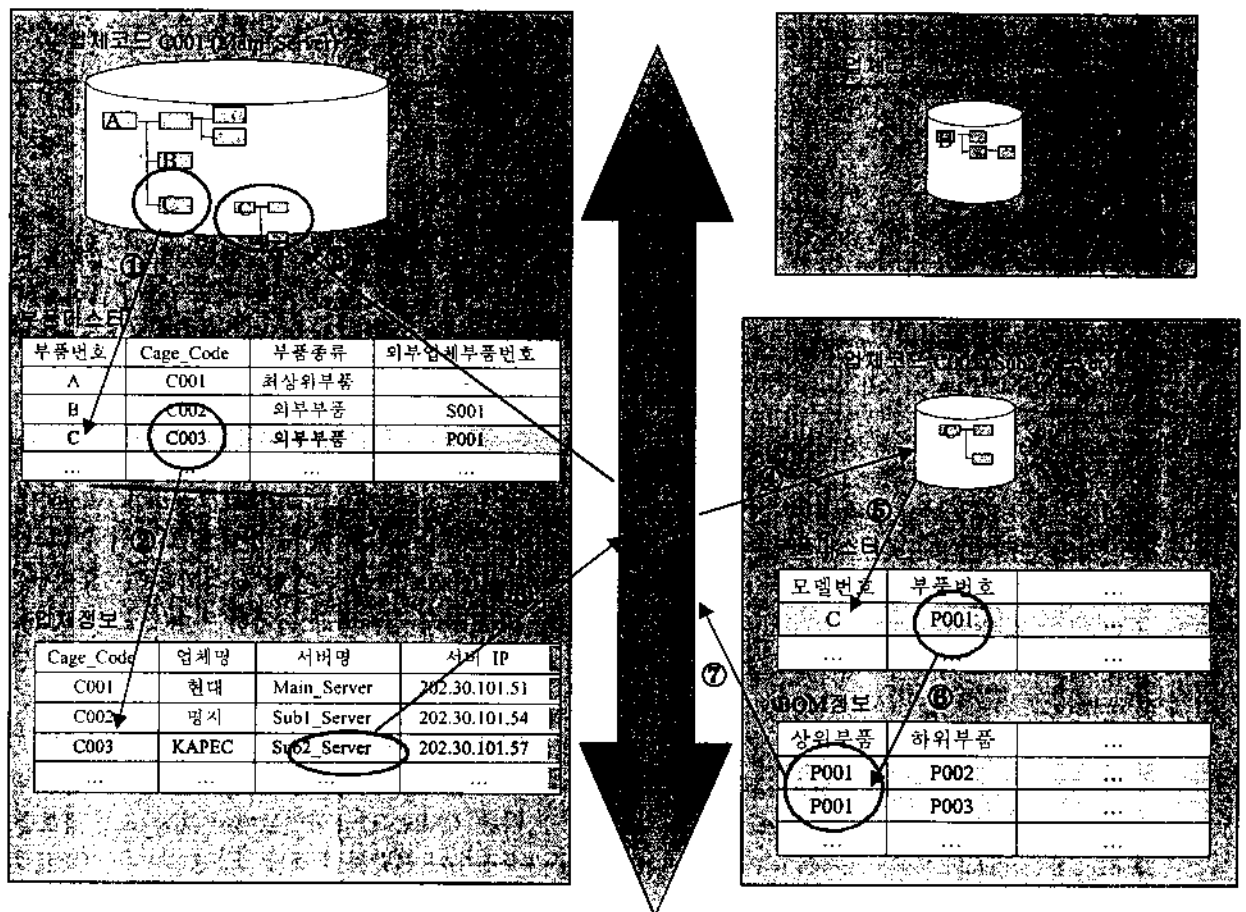


그림 6. 제품구조정보의 조합절차.

서관리, 워크플로우 관리기능과 통합되어 운용되어야 한다.

3.2 JAVA 프로그램과 CORBA 활용

이중분산환경에서 제품구조 데이터를 공유 및 교환하기 위하여 OMG에서 제정한 CORBA 표준을 이용하였다. 먼저 시스템 설계 단계에서 클라이언트와 서버 상호간에 연동이

필요한 객체들간의 인터페이스를 IDL (Interface Definition Language)이라는 언어로 정의를 하게 된다(OMG, 1998; Orfali, Harkey, 1997). <표 2>는 현재 개발된 제품구조 및 구성관리 시스템에서 IDL로 정의된 CORBA interface들이다. 이중에서 BOM 데이터를 교환하는 PS\_BOM\_Info의 IDL 사례를 소개하면 다음과 같다.

표 2. 제품구조 및 구성관리 시스템을 위한 CORBA Interface 정의

Interface 명	Interface 설명
PS_Part	객체간에 주계약자 서버 또는 이와 관련된 타 서버들의 DB 내에 존재하는 부품정보, 부품리비전 정보를 서로 공유하여 사용하기 위한 Interface이다. 이 Interface 내에는 부품에 대한 상세를 등록, 수정, 삭제, 조회 등을 할 수 있는 Operation들이 정의되어 있다.
PS_Model	객체간에 주계약자 서버 또는 이와 관련된 타 서버들의 DB 내에 존재하는 모델정보, 모델리비전 정보를 서로 공유하여 사용하기 위한 Interface이다. 이 Interface 내에는 모델에 대한 상세정보를 검색하거나 수정, 삭제, 등록 등을 할 수 있는 Operation들이 정의되어 있다.
PS_BOM_Info	객체간에 주계약자 서버 또는 이와 관련된 타 서버들의 DB 내에 존재하는 BOM를 서로 교환하거나 BOM 정보를 전개하기 위한 Interface이다. 이 Interface 내에는 BOM 정보를 구성, 전개, 수정, 조합 등의 기능을 가진 Operation들이 정의되어 있다.
PS_Relate_Doc_Info	객체간에 주계약자 서버 또는 이와 관련된 타 서버들의 DB 내에 존재하는 부품정보와 그와 관련된 문서정보간의 Relation 관계를 관리하기 Interface이다. 이 Interface 내에는 부품정보에 문서를 등록시키거나, 수정 또는 삭제시키는 Operation들이 정의되어 있다.
PS_Contract_Info	주계약자 서버가 관리하는 타 업체들의 상세한 업체정보를 클라이언트와 함께 공유하기 위한 Interface이다. 이 Interface 내에는 업체들에 대한 상세한 정보를 검색할 수 있는 Operation들이 정의되어 있다.

```

interface PS_BOM_Info {
struct BomInfoStruct {
    short level;
    string parent_part_number;
    string part_number;
    string original_date;
    string update_date;
    short quantity;
    string relation_type;
};
typedef sequence <BomInfoStruct> BomInfo;
boolean getPomInfo (in string Parent_Part_No, in string
    Part_No, out BomInfo bom);
boolean getMBomInfo (in string Model_Code, out BomInfo
    bom);
boolean updateBomInfo (in string P_Part_No, in string
    Part_No, in BomInfo m_struct);
boolean insertBomInfo (in BomInfo m_struct);
boolean deleteBomInfo (in string P_Part_No, in string
    Part_No);
boolean displayBomInfo (in string Model_Code, out BomInfo
    bom);
boolean getqp_bom_part (in string Part_Type, out PartMaster
    p_struct);
boolean getq_model_code (out ModellInfo m_struct);
....
};

```

<그림 7>은 JAVA와 CORBA를 이용해 객체간에 서로 BOM

정보를 주고받을 수 있도록 프로그램을 만드는 과정을 표현한 그림이다. 그림 상단에서와 같이 서버쪽의 프로그램과 클라이언트쪽 프로그램을 JAVA 언어를 이용하여 작성하고, IDL로 이 둘간의 인터페이스를 정의한다. 정의한 IDL을 JAVA 언어로 바꾸기 위해 pre-compile하게 되면 JAVA 파일들이 생기게 되며, 이 파일들을 서버와 클라이언트의 소스파일들과 함께 compile 하면 클라이언트와 서버에 JAVA 클래스 파일들이 생성된다.

PS\_BOM\_Info의 IDL 사례에서, 클라이언트에서 생성된 객체 A에는 서버로 부터 Model\_Code를 매개변수로 하여 BOM 정보를 요청하는 메소드(Method)인 getMBOMInfo()를 호출하고 서버로부터 받아온 BOM 정보를 화면에 출력시키는 메소드인 displayBOMInfo() 등이 있다. 객체 B에는 요청받은 BOM 정보를 DB로부터 검색하고 접근하는 메소드인 getMBOMInfo()가 존재하여 클라이언트쪽으로 요구하는 정보를 전달해 준다. 객체 A가 ORB에 BOM 정보를 제공받을 수 있도록 요청을 하면, ORB는 클라이언트의 호출을 받아 클라이언트가 원하는 서비스를 제공해 줄 수 있는 객체 B가 존재하는 서버를 찾는다. 그 다음, ORB는 클라이언트에서 보낸 요청 정보를 해당 구현 객체 B에게 전달하여 해당 객체로 하여금 DB로부터 BOM 정보를 찾도록 객체 B를 Invoke시킨다. 객체 B는 요청 받은 BOM 정보를 다시 ORB를 통해서 객체 A로 보내준다.

#### 4. CORBA 기반의 제품구조 및 구성관리 시스템 개발

##### 4.1 시스템 개발 환경

본 연구에서는 조달자와 주계약자 사이의 관계를 CITIS의 기본단위로 보고 프로토타입을 개발하였다. 비록, JAVA가 속

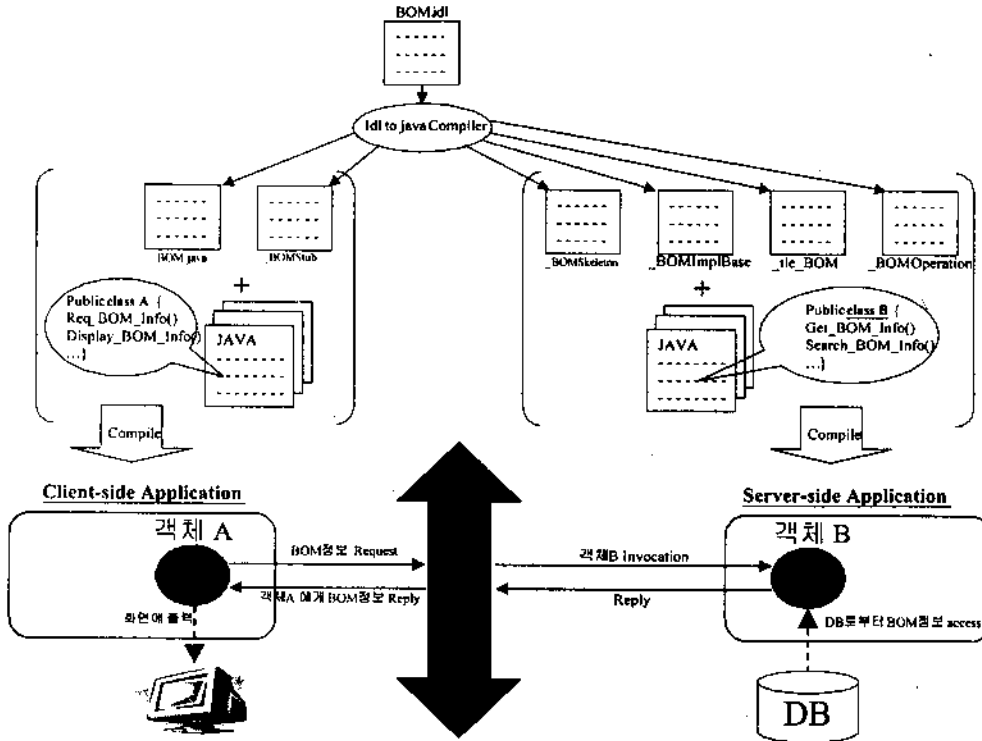


그림 7. JAVA와 CORBA를 이용한 객체간 연동.

도는 느리지만 미래에 web용으로 쉽게 변환시킬 수 있도록 조달자, 주계약자, 협력업자에게서 사용되는 모든 프로그램을 JAVA application으로 구현하기로 하였다. 또한 이종분산환경 하에서 정보를 공유하기 위하여 CORBA가 사용되었다. <그림 8>, <그림 9>는 CITIS의 클라이언트와 서버의 시스템 구조를 나타낸 그림이다. 이 그림 중에서 DM은 문서관리 모듈을, WM은 워크플로우 모듈을 나타내고 있으며 PS/CM이 제품구조 및 구성관리 모듈을 나타낸다.

<표 3>은 제품구조 및 구성관리 모듈의 클라이언트와 서버의 프로토타입을 개발하는 데 사용된 환경을 보여주고 있다.

표 3. 제품구조 및 구성관리 시스템의 시스템 개발환경

	Client	Server
Hardware	pentium급 이상의 PC	pentium급 이상의 PC
Software	Client Application : Jbuilder C/S Version. 도면 Viewer : 성우 View. 문서Viewer : MS-Word, 아래아 한글	Server Application : JDK1.1.4 CORBA : OrbixWeb 3.0
OS	Windows 95	Windows NT4.0
DBMS	없음	Oracle7.3.3 workgroup Server

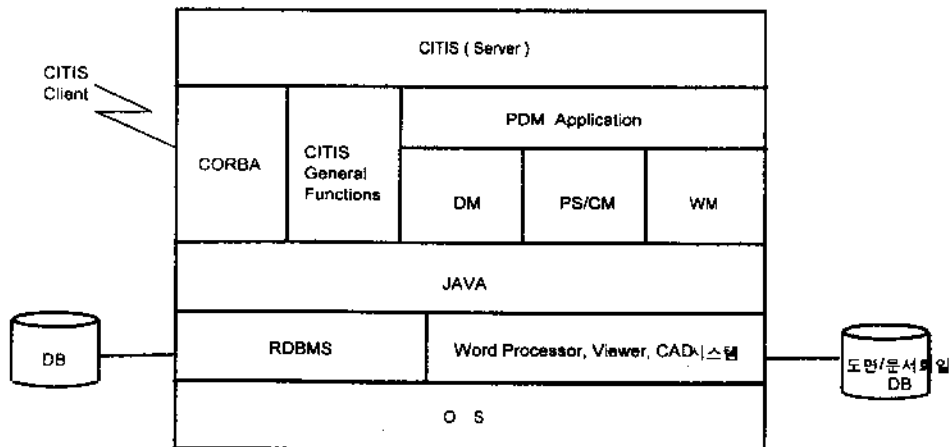


그림 8. 서버의 소프트웨어 구조.



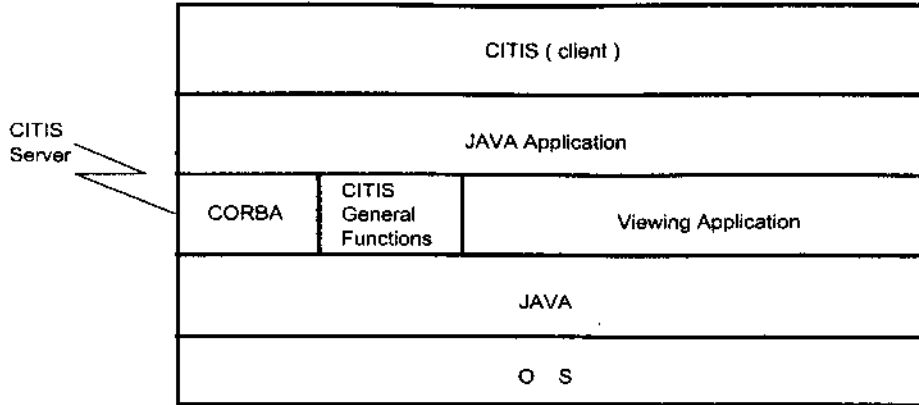


그림 9. 클라이언트의 소프트웨어 구조.

4.2 시스템 기능

현재 개발된 제품구조 및 구성관리 시스템의 ER 다이어그램은 <그림 10>과 같다. 여기서, “BOM 정보”는 BOM 구성정보를 저장하고 있는 엔티티이다. “BOM 정보”에서 ‘모부품번호’는 상위부품을, ‘부품번호’는 하위부품을 나타내며 이 두 부품을 이용하여 제품구조를 표현한다. 이 엔티티에서 ‘연결관계’는 앞에서 설명하였듯이 연결관계 속성을 나타내며 MrT, PtP, PtF, FtO의 값을 이용하여 부품 사이의 관계를 표시한다. “모델-피쳐”는 각 모델별 피쳐리스트를 저장하고 있는 엔티티이다. 이

엔티티는 “BOM 정보”에 피쳐가 등록될 때 생성되며 ‘피쳐종류’는 피쳐가 선택피쳐인지, 필수피쳐인지를 구분하는 데 사용된다. “피쳐-옵션”은 피쳐별 옵션리스트를 저장하고 있는 엔티티이다. “BOM 정보”에 옵션부품이 등록될 때 생성되며 “BOM 정보”의 ‘모부품번호’가 ‘피쳐부품번호’로, ‘부품번호’가 ‘옵션부품번호’로 매칭된다. “부품정보”는 부품들의 마스터정보를 보관한다. 이 엔티티는 “BOM 정보”와 일대다 관계를 유지하며 BOM을 구성하는 부품들의 기초정보를 제공한다. “부품 Revision 정보”는 부품의 Revision된 항목들만을 따로 모아서 관리함으로써 데이터의 중복을 피하고 관리의 효율성을 높

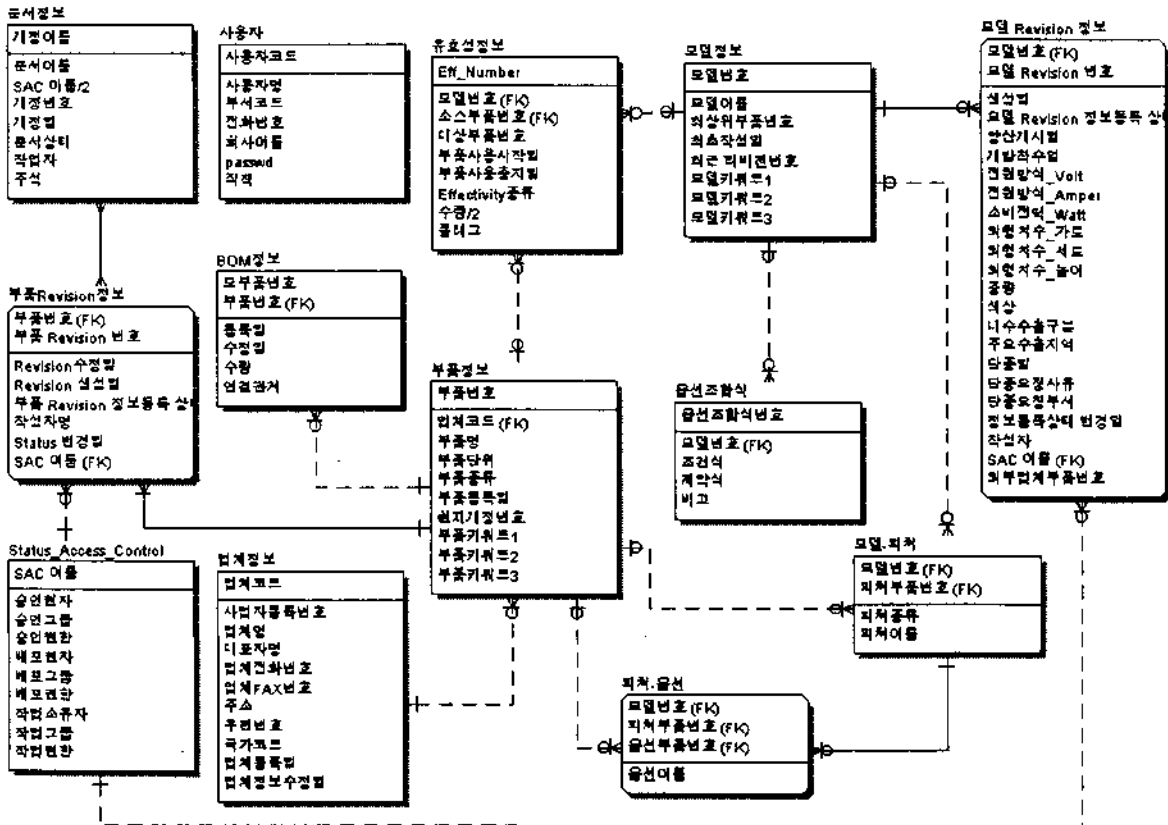


그림 10. PS/CM 시스템을 위한 ER\_Diagram.

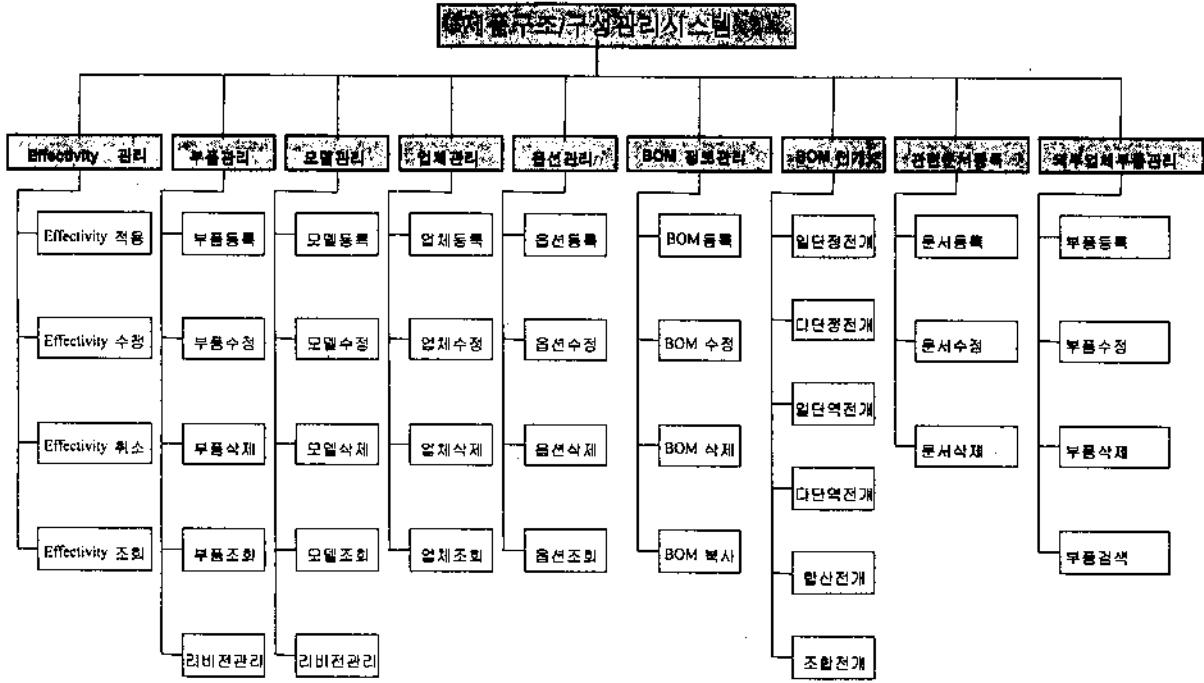



그림 11. 제품구조 및 구성관리 시스템의 기능구조도.

일 수 있도록 하였다. “유효성 정보”는 유효성에 대한 정보를 관리하는 엔티티이다. 이 엔티티의 속성 중 ‘플래그’는 현재 유효성의 상태를 계획, 진행, 종료의 값으로 표시하였으며 적용된 이후의 이력까지 관리할 수 있도록 하였다. 이 그림에서 본 개발 내용에 설명되지 않은 엔티티들이 포함되어 있는데, 이것은 참고적으로 관련성이 있다는 것을 보여주기 위함이다. 특히, “문서정보” 엔티티는 문서정보관리 모듈과 연결시키기 위하여 설계되어 있다.

그리고 시스템의 기능은 <그림 11>과 같이 부품관리, 모델 관리, 업체관리, 옵션관리, 유효성 관리, BOM 정보관리, BOM 전개, 관련문서등록, 외부업체 부품관리 모듈로 구성되어 있다. 여기서 각 모듈의 기능은 생략하며 BOM 전개 기능을 좀 더 자세하게 소개한다.

BOM을 전개하는 기능으로는 일단 정전개, 다단 정전개, 일단 역전개, 다단 역전개, 합산 전개가 있다. 이 전개 기능을 수행하기 위해서는 일반적인 BOM 전개방식과는 다르게 옵션을 먼저 선택하여 제품구성을 하여야 한다. 또한, 이 시스템에는 특별히 조합전개가 추가되어 있다. 이 기능은 분산된 협력업체의 서버에 존재하는 BOM 정보를 검색하고 이것을 조합하여 조달자에게 보여주는 기능이다. 협력업체의 서버로부터 가져오는 부품은  모양의 아이콘이 옆에 표시되어 구별할 수 있다. 외부업체 부품은 BOM 등록시에 외부업체 부품으로 등록하게 되어 있다. 외부업체 부품은 부품종류에서 외부부품을 따로 선택하여 전개할 수도 있으며 모델 안에 포함시켜 전개할 수도 있다. 그리고, 옵션을 선택하지 않고 모든 피쳐와 옵션을 포함한 전체의 제품구조를 보여주는 전개기능도 가지고 있다. <그림 12>는 BOM 전개를 위하여 옵션을 선택하는 화면

을 보여주고 있다. 이 예에서는 다단 정전개를 위하여 피쳐인 변속기를 선택하고 이변속기에 속하는 옵션 중에 자동기어를 선택하고 있다. 선택한 옵션들은 우측 상단에 나타나며 전개 버튼을 누르면 다단 정전개가 이루어진다.

## 5. 결 론

여기서 개발된 내용은 두 가지로 요약될 수 있다. 우선, 하나의 데이터 구조하에 모든 옵션을 표현할 수 있는 BOM 구조를 제시하였다. 그리고 CITIS 환경에서 제품구성 관리의 운용 개념을 제시하였으며 프로토타입을 개발하였다. 여기서 제시한 옵션 BOM 구조는 변형품을 표현하기 위한 하나의 방법으로서, 기존의 표현 방법과 다른 점은 가상부품인 피쳐를 이용하였다는 것이다. 이 옵션 BOM 구조는 기존의 모듈러(modular) BOM과 유사한 구조를 가지고 있어 전개 속도는 모듈러 BOM과 유사할 것으로 예상된다. 이 시스템은 설계분야에서 단독으로도 운용될 수도 있으나 좀 더 효율성을 높이기 위해서는 도면 및 문서관리 시스템과 함께 운용되어야 한다. 현재 이 시스템은 CITIS의 한 모듈로서 설계 및 개발되었으며 도면 및 문서관리 모듈과 연결될 수 있도록 관련 문서 정보를 입력하는 엔티티가 설계되어 있다.

추후 연구과제로서, 옵션이나 부품 선택시 다른 구성 부품에 대해 제한을 주는 사양(option constraints)이 존재하므로 이것을 표현할 수 있는 기능이 필요하다. 또한, CAD의 3D 모델러로부터 생성되는 STEP 파일에서 조립 데이터를 읽어 제품구조 및 구성관리 모듈의 BOM 데이터로 변환하는 기능도 부가할

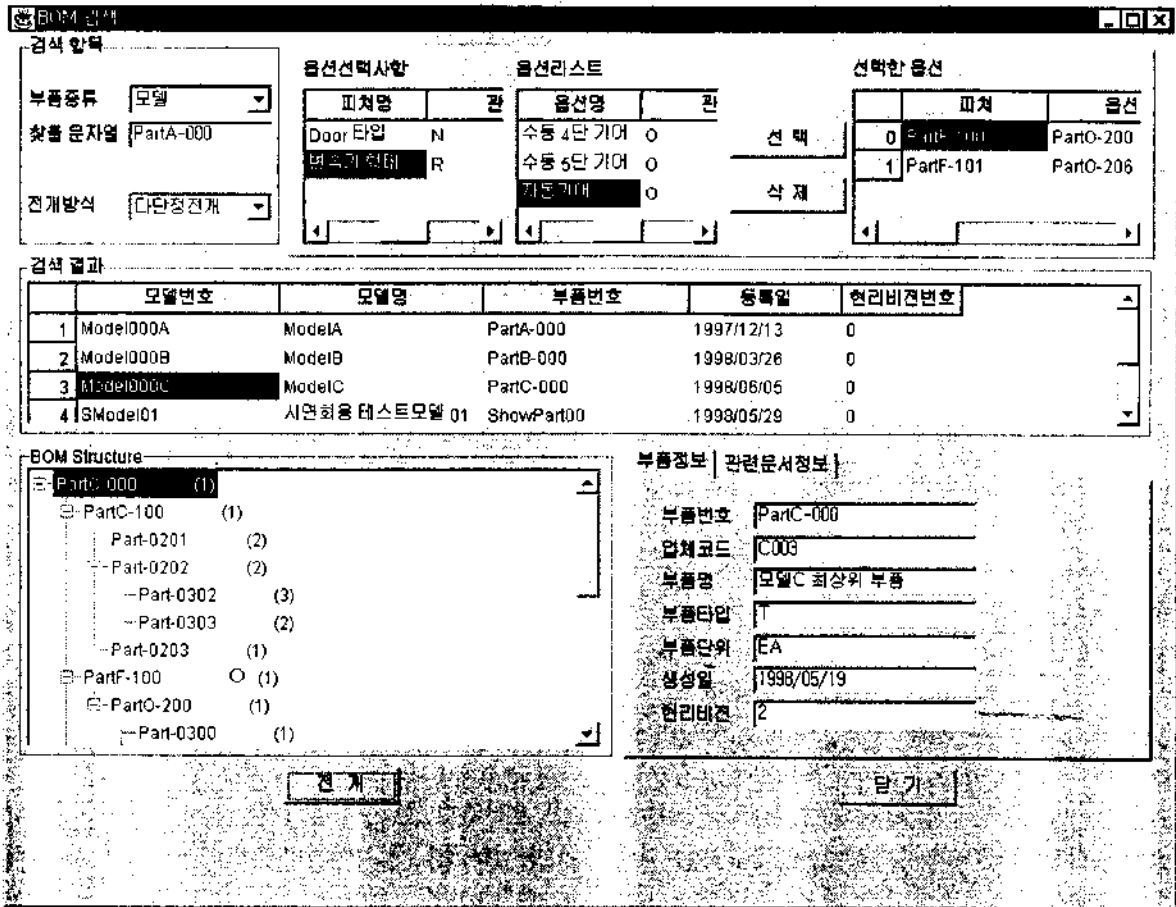


그림 12. BOM 전개기능 화면.

계획이다. 이것은 설계자가 조립 데이터를 생성한 후 BOM 데이터를 다시 생성하는 이중의 작업을 줄이기 위함이다. 또한, 현재 개발된 프로토타입을 web상에서 활용할 수 있도록 applet으로 변환하는 작업도 추진할 계획이다.

참고문헌

고석완, 김선호, 정석찬 (1998. 9), Option을 고려한 객체지향형 Product Structure 시스템 설계, *대한산업공학회지*, 24(3), 457-473.  
 김선호 (1999. 5), 제품구성 관리기 개발, *ETRI 위탁연구보고서*.  
 김선호, 권용성, 정석찬, 조장혁 (1998), 이종분산환경하에서의 Configuration Management 시스템 설계, *대한산업공학회/경영과학회'98 춘계학술대회 논문집*.  
 김선호, 문희석 (1997), Group Technology를 이용한 설계정보관리시스템 개발. *한국정밀공학회지*, 14(1), 58-68.  
 이한표, 이춘열, 이국철 (1995), Family BOM 데이터베이스 구조에 대한 대안: 목적별 BOM 연결 구조의 간접표현 방안, *대한산업공학회'95 추계학술대회 논문집*, 195-202.  
 정석찬, 우훈식, 백종명, 주경준 (1997), CITIS(Contractor Integrated Technical Information Services) 구현에 관한 고찰, *한국경영과학회/대한산업공학회'97 춘계공동학술대회*.  
 Chung Y., and Fischer G. W. (1994), A Conceptual Structure and Issue for an

Object-Oriented Bill of Materials(BOM) Data Model, *Computers and Industrial Engineering: An International Journal*, 26(2), 321-339.  
 Erens, F. J. (1996), The Synthesis of Variety Developing Product Families, Ph.D thesis, *Eindhoven University of Technology*.  
 Hegge, H. M. H., Wortmann, J. C. (1991), Generic bill-of-material: a new product model, *International Journal of Production Economics*, 23, 117-128.  
 ISO 10303-44 (1994), Industrial Automation Systems and Integration - Product Data Representation and Exchange, Part 44: Integrated Generic Resources: Product Structure Configuration.  
 McKay, A., Erens, F., Bloor, M.S. (1996), Relating Product Definition and Product Variety, *Research in Engineering Design*, 63-80.  
 MIL-STD-974 (1993), Contractor Integrated Technical Information Service, DoD.  
 OMG (1998), PDM Enabler : Joint Proposal to the OMG in Response to OMG Manufacturing Domain Task Force RFP1.  
 Orfali, R., Harkey, D. (1997), Client/Server Programming with JAVA and CORBA, Wiley.  
 Tersine, R. J. (1994), *Principles of Inventory and Materials Management*, 4th Edition, PTR Prentice Hall.  
 Trappey, A. J. C., Peng, T. K. and Lin, H. D. (1995), An Object-Oriented Bill-Of-Materials System For Dynamic Product Management, *the 3rd International Conference on Computer Integrated Manufacturing*, 1, 629-636.  
 Yatabe, S., Tanaka, Y., Hasegawa, K. (1997), An approach to Engineering and Manufacturing Data Sharing among heterogeneous PDM systems, *Proceedings of 21 Century Commerce & C.I.T.S Expo 1997*, Orlando, Florida, Oct.

**김 선 호**

서울대학교 산업공학과 학사

Pennsylvania State University 산업공학과 석사

Pennsylvania State University 산업공학과 박사

현재: 명지대학교 산업공학과 교수

관심분야: PDM, workflow management, B2B EC

**권 용 성**

명지대학교 산업공학과 학사

명지대학교 산업공학과 석사

현재: PTC Korea, Global Service 부원

관심분야: PDM, 객체지향응용

**주 경 준**

고려대학교 산업공학과 학사

고려대학교 산업공학과 석사

삼성전자 생산관리부장

현재: ETRI 컴퓨터소프트웨어기술연구소 전

자상거래연구부 팀장

관심분야: 전자상거래, SCM

**정 석 찬**

부산대 기계설계학과 학사

일본 오사카 부립대학 경영공학과 석사

일본 오사카 부립대학 경영공학과 박사

ETRI 컴퓨터소프트웨어기술연구소 전자상거

래연구부 선임연구원

현재: 동의대학교 경영정보학과 조교수

관심분야: 전자상거래, SCM