

CALS 환경을 지원하는 데이터 모델링 방법론 개발

김중인*, 김철한**, 우훈식**, 임동순***

A Data Modeling Methodology for CALS Environment

Joong-In Kim, Cheol-Han Kim, Hoon-Shik Woo, Dong-Soon Yim

Abstract

For CALS environment, sharing information among business processes within the enterprise and between enterprises have been increased through common database. To support integrated data environment, data of the local legacy system must coincide with data of the global system to guarantee data integrity and transparency. This study is concerned with the implementation of this distributed database system under CALS environment. Based on existing IDEF methods, we developed a modified method which newly includes concepts of the enterprise structure, the integration of legacy databases, the distribution of data, and the distributed applications.

* 홍익대학교 경영정보학과

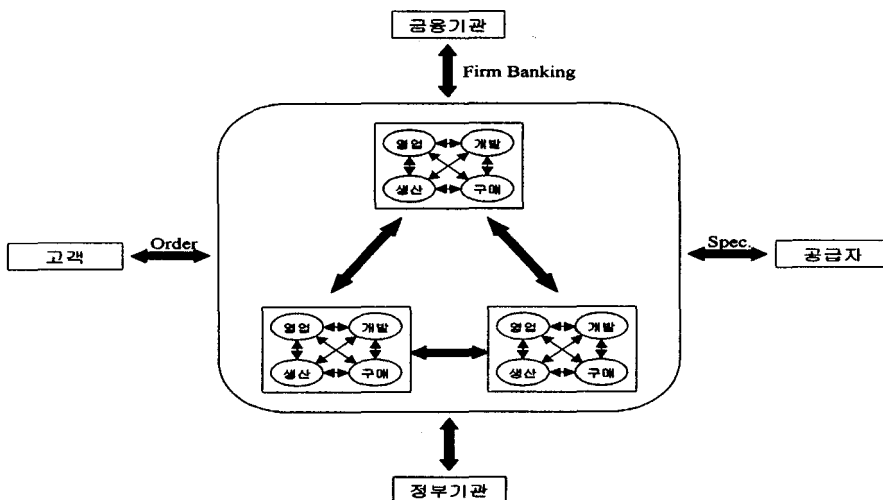
** 대전대학교 컴퓨터정보통신공학부

*** 한남대학교 산업공학과

1. 서론

기업 활동의 리드타임을 줄이기 위하여, 종래의 단일 기업 내에서 단일 기능에 의한 수행보다는 <그림 1>에 나타난 바와 같이 관련된 기업들간의 조직이나 기능이 함께 수행되는 추세에 있다. 즉, 기업들은 개발과 생산 부서의 통합(정보 흐름, 제어 흐름, 물류 흐름의 통합), 공급자와 생산자간의 통합, 설계와 생산의 통합, 그리고 기업 내 정보자원의 통합 등을 통한 시장에서의 대응시간 단축을 도모하고 있다. 개별기업은 하나의 독립된 조직에서 특정 제품에 관련된 기업들간의 관계에서 존재하는 하나의 부분으로 인식되어, 마치 하나의 가상기업처럼 활동하게 된다. 이러한 기업환경의 변화에 따라 조직 및 사업장이 지역적, 기능적으로 분산되고 이를 지원하는 정보시스템도 정보기술의 발달과 저렴한 하드웨어 가격으로 인하여 분산되면서, 분산 처리 및 분산

데이터베이스 개념이 도입되고 있다. 이러한 환경은 분산된 데이터를 통합된 하나의 시스템으로 처리하여 사용자에게, 사이트에 관계없이 권한을 가진 데이터에 접근할 수 있도록 하여야 하므로, 이를 지원하는 시스템의 구축 방법도 종래의 중앙집중 환경하의 개발 방법과는 달리해야 한다. 특히 <그림 1>과 같이 기업 내(Intra-Enterprise) 및 기업간(Inter-Enterprise)의 업무 프로세스의 연계와 정보의 교환/공유를 목적으로 하는 CALS 및 가상기업(Virtual Enterprise)환경의 이질 분산형 통합 데이터베이스(IDB: Integrated Database) 어플리케이션 시스템은 업무 기능과 데이터가 분산, 통합 또는 공유되는 클라이언트/서버 기반의 분산 어플리케이션과 분산 데이터베이스 시스템으로 구성된다. 이러한 분산 통합 시스템의 개발을 위하여 본 연구에서는 다음과 같은 특징을 갖는 방법론을 개발하였다:



<그림 1> 분산환경하의 기업활동

1.1 조직의 구조에 따른 분산시스템 개발방법론

기업의 실제 환경을 고려하지 않은 분산과 통합되지 않은 분산 어플리케이션 및 데이터 베이스는 막대한 예산의 낭비와 기술적인 어려움에 봉착되어 실패를 부르기 쉽다. 실제로 많은 조직이 중앙 집중적인 형태보다는 분산된 형태를 갖고 있기 때문에 조직 구조에 맞게 적용할 수 있는 분산 시스템의 개발과 통합된 시스템의 개발이 매우 중요하게 요구된다. 본 연구에서 제시된 방법론은 실제로 분산된 형태를 갖고 있는 조직 구조에 적용할 수 있는 분산 시스템의 개발 방법론이다.

1.2 기존 이질 분산형 데이터베이스들의 하향식 통합 방법론

어떤 조직에 이미 여러 개의 이질적인 데이터베이스 시스템(Legacy Heterogeneous Data-base Systems)들이 존재하고 있는 상황에서 기존의 이질 분산 통합 데이터베이스 개발 방법론들은 기존의 지역 데이터베이스를 기반으로 하여 상향식(Bottom-Up)으로 통합하도록 한다. 하지만 실제 환경에서는 이러한 통합작업은 역 공학(Reverse Engineering)적인 각 지역 스키마(Local Schema 또는 Local Data Model)의 분석, 이질적인 지역 스키마의 공통/표준 지역 스키마(Common/Canonical Schema)로의 변환(Schema Translation 또는 Mapping), 공유될 스키마(Export Schema)의 추출, 추출된 스키마를 통합(Schema Integration)하여 전역 스키마(Global Schema) 또는 연합 스키마(Federated Schema) 등에 소요되는 막대한 노력으로 인하여 실제적인 적용 면에서는 현실

성이 부족한 것으로 인식되고 있는 실정이다. 한편, 순 공학(Forward Engineering)적인 하향식(Top-Down) 개발 방법론은 상대적으로 수월하지만 동질의(Homogeneous) 분산 데이터베이스 시스템의 구축에만 적용할 수 있는 제한을 지니고 있다. 그러나, 본 연구의 개발 방법론은 기존에 구축된 대부분의 데이터베이스가 관계형이므로 관계형 스키마를 공통/표준 스키마로 하는 하향식 개발 방법론이면서도 이질적인 분산 데이터베이스를 통합할 수 있도록 하는 현실적인 방법론이다.

1.3 기업 내 및 기업 간의 업무 프로세스와 데이터의 분산 분석 및 설계 방법론

본 연구의 방법론은 분산 분석 단계에서 기능/프로세스/조직 연관 분석표, 데이터/조직 연관 분석표 및 기능/데이터 연관 분석표를 작성하여 분산 설계 단계에서 이용한다. 이러한 3가지 연관 분석표는 기존의 정보공학(IE: Information Engineering) 방법론이나 중앙/분산 데이터베이스 설계 기법들에서도 적용하고 있지만, CALS와 가상기업 환경과 같이 다수의 참여 기업간의 거래처리, 즉 기업간의 업무 프로세스(Inter-Organizational Business Process)와 기업간에 교환/공유되어야 할 데이터의 분산 분석 및 설계를 위한 연관 분석표는 아직 제시되지 않은 실정이다. 그러나, 본 방법론의 연관 분석표들은 CALS와 같은 분산 환경 하에서의 기업간 통합을 위한 업무 프로세스와 데이터의 분산 분석 및 설계를 지원한다.

1.4 분산 어플리케이션과 분산 데이터베이스를 위한 통합 개발 방법론

분산 어플리케이션들은 사용자가 요구하는 작업을 수행하기 위하여 필요한 데이터를 분산 데이터베이스로부터 전송 받아 트랜잭션을 수행하거나, 데이터를 생성하고 변경하여 데이터베이스에 저장한다. 따라서 분산 어플리케이션과 분산 데이터베이스 시스템은 밀접하게 연계되어 있으며, 이러한 두 시스템의 통합은 개발 과정에서 함께 고려되어야 한다. 본 연구의 방법론은 ERP(Enterprise Resource Planning)와 같은 전사 업무(Enterprise-Wide)를 대상으로 하는 기업 내 통합뿐만 아니라 CALS와 같이 기업간에도 통합된 분산 어플리케이션과 분산 데이터베이스의 개발을 위한 통합 개발 방법론이다.

2. 기존 연구

분산 데이터베이스 시스템을 구축하는 방법으로 두 가지 경우를 생각할 수 있다. 첫째로는 같은 형태의, 즉 동질의(Homogeneous) 지역 데이터베이스 시스템으로 분산 데이터베이스를 구축하는 방법으로, 이는 분산 데이터베이스 시스템을 사용하고자 하는 목적을 정한 후, 이 목적에 부합되는 데이터베이스 시스템만을 분산 데이터베이스의 구성요소(즉, 지역 데이터베이스 시스템)로 참여시키는 방법이다. 즉, 분산 데이터베이스 시스템에서 필요한 기능을 보유한 DBMS를 선정하고 선정된 DBMS를 일괄 구입하여 분산 데이터베이스 시스템을 처음부터 새롭게 구축하는 방법을 말한다. 이 형태의 분산 데이터베이스 시스템

을 동질형 분산 데이터베이스 시스템이라고 한다.

두 번째 방법으로는 서로 다른 형태의, 즉 이질의(Heterogeneous) 지역 데이터베이스 시스템을 통합하여 분산 데이터베이스 시스템을 구축하는 방법으로서, 이는 기존에 독자적으로 개발된 서로 다른 형태의 지역 데이터베이스 시스템을 통합하여 새로운 분산 데이터베이스 시스템으로 확장하는 방법이다. 이러한 형태의 분산 데이터베이스 시스템을 이질형 분산 데이터베이스 시스템이라고 한다.

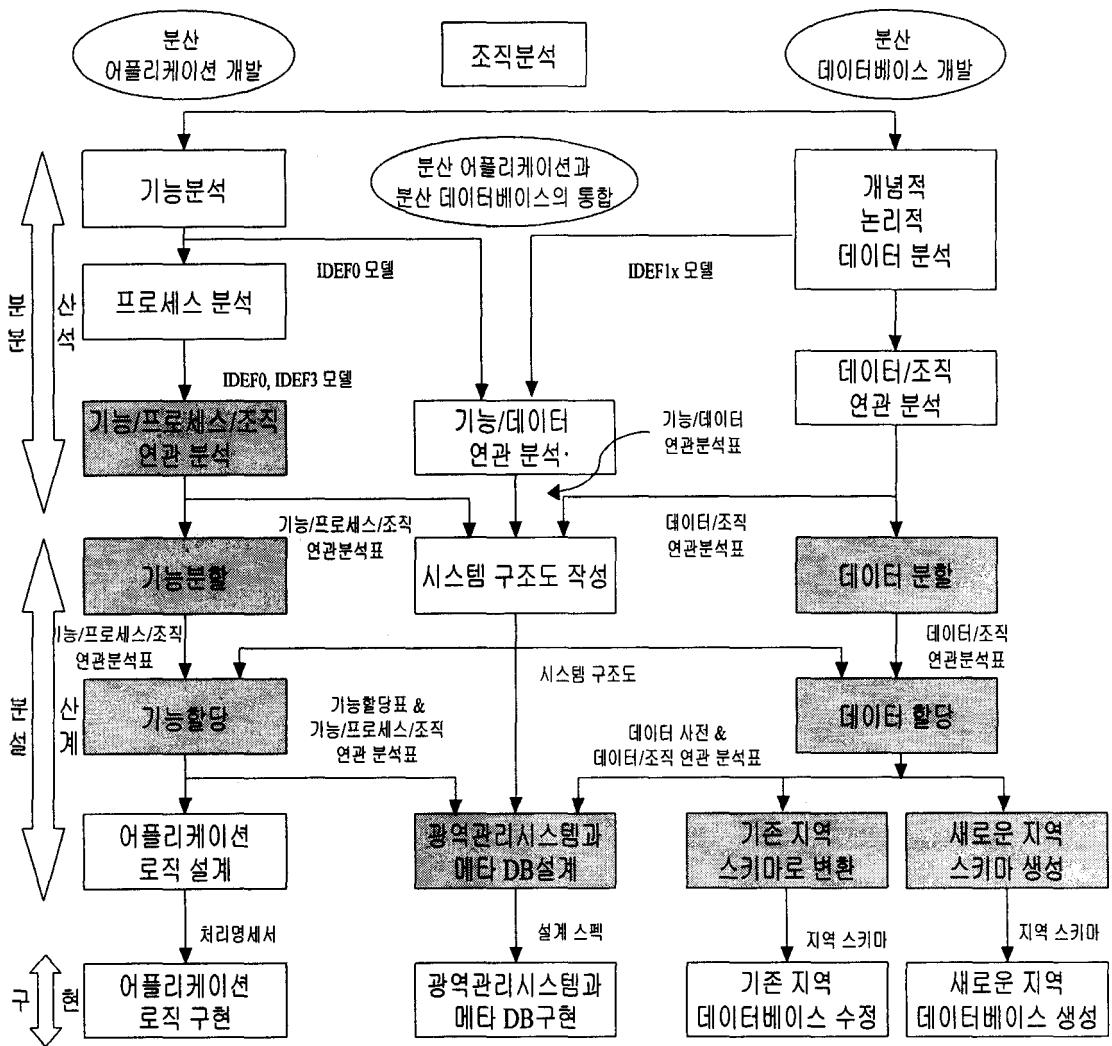
이질형 분산 데이터베이스 시스템 구축 방법은 지역 데이터베이스 시스템의 자치성(Autonomy)을 보장할 수 있으므로 CALS 및 가상기업 환경과 같이 새로운 지역 데이터베이스 시스템의 추가 및 제외가 용이한 시스템, 즉 확장성(Scalability)이 큰 분산 데이터베이스 시스템을 구축하는데 적합하다. 또한, 많은 기업 및 기관에서 분산 데이터베이스 시스템을 도입하기 전에 업무 특성에 맞는 데이터베이스 시스템을 독자적으로 구축하여 사용하고 있는 경우가 대부분이다. 따라서 각 기업은 이미 구축되어 있는 데이터베이스 시스템의 자치성을 완전하게 보장한 상태에서 분산 데이터베이스 시스템을 구축하려 한다.

본 연구에서 개발된 방법론의 대상인 이질형 분산 데이터베이스 시스템의 필요성에 대한 연구 방향에 대한 설명은 [Ceri and Pelagatti 1984], [Kim 1989], [Sheth and Larson 1990]과 [Sheth 1991]에 제시하였다. 이러한 참고문헌들의 내용을 종합 정리한 참고문헌으로 [Litwin et al. 1990]와 [Thomas et al. 1990]이 있다. 이질형 분산 데이터베이스 시스템의 형태를 분류하고 각 형태에 따른 특

성을 분류 소개한 내용이 참고문헌 [Bright et al. 1992], [Kim 1992]에 나와 있다. 이질형 분산 데이터베이스 시스템의 문제 중에서 데이터 모델 변환에 관한 내용은 [Papazoglou and Marinos 1990]에 나와 있다.

하향식 분산 데이터베이스의 설계는 전역

스키마(Global Schema)의 설계, 분할(Partition, Fragmentation) 방법의 설계, 할당(Allocation, Deployment)의 설계, 각 사이트에서 물리적 구조(Internal Schema) 설계의 4단계로 이루어진다. 전역 스키마와 물리적 구조는 중앙집중형 데이터베이스에서 개념 스키마(Conceptual



<그림 2> 방법론의 작업절차 및 입출력 자료

Schema)와 물리적 스키마의 설계와 같은 방법으로 설계된다. 그러므로 분산 데이터베이스의 고유한 문제는 분할 방법의 설계와 할당의 설계이다. 수평분할(Horizontal Partitioning)과 수직분할(Vertical Partioning)에 관한 참고문헌으로는 [Ceri et al. 1982], [Ceri et al. 1983a], [Ceri et al. 1983b], [Chang and Cheng 1980], [Navathe et al. 1982] 등이 있다.

3. 본론

이 장에서는 본 연구에서 개발된 개발 방법론에 관한 것으로, 본 연구의 방법론이 IDEF 방법론을 분산형 시스템 개발 방법론으로 확장한 것이므로, IDEF 모델들의 작성절차, 표현방법 및 메커니즘과 동일한 부분은 그대로 수용하므로 여기에서는 설명을 생략하고 그 중에서 변경, 확장된 부분이나 새로 추가된 사항들만을 설명하도록 한다.

3.1 방법론 Overview

방법론의 전체적인 작업 절차와 각 작업별 입출력 자료를 요약하면 <그림 2>와 같다. 조직분석 작업을 시작으로 하여 분산 어플리케이션 개발의 시작이 되는 기능 분석과 분산 데이터베이스 개발의 시작이 되는 데이터 분석은 동시에 수행될 수 있으며, 분산 어플리케이션과 분산 데이터베이스의 통합은 각각의 설계 작업 후에 이루어진다. <그림 2>에서 어둡게 칠해진 작업들은 그 작업의 내용 또는 절차가 기존 연구들과는 다르거나 그 작업이 본 연구의 방법론에서만 독창적으로 추가된 부분들을 나타내며, 나머지 부분들은 기존 연

구들에서 제시된 일반적인 작업과 절차들을 그대로 적용하고 있다. 각 단계별로 상세한 내용은 아래에 설명한다.

3.2 조직 분석: 조직단위와 사업장의 파악 및 연관 분석

분산 분석 단계는 기업의 업무행위(즉, 기능과 프로세스)가 발생하는 조직단위(Organization Unit), 예를 들어 전산실, 영업부, 구매부 등과 같은 사업장(Location), 즉 지역적 위치를 조직도로부터 파악하여 사업장별로 유형(예, 본사, 공장, 물류 창고, 영업센터, 해외지사 등)을 분류하는 일로부터 시작되는데, 조직단위 및 사업장 유형은 분산/집중 처리 분석의 기준이 된다. <표 1>은 사업장과 사업장 유형의 연관 분석표의 예를 보여주며, <표 2>는 조직단위와 사업장 유형의 연관 분석표의 예를 보여준다.

3.3 분산형 어플리케이션 개발

3.3.1 분산 분석 단계

1) 현행(As-Is) 및 개선(To-Be) 업무 기능 분석

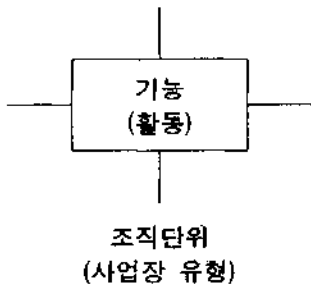
IDEFO 기능(활동) 모델을 작성하며, 이때 <그림 3>과 같이 각 기능을 수행하는 조직단위를 메커니즘(Mechanism)으로 모델링 하는데, 여러 사업장 유형별로 존재하는 조직단위에 대해서는 괄호를 사용하여 사업장 유형을 명시해 준다.

<표 1> 사업장과 사업장 유형의 연관 분석표

조직단위 \ 사업장 유형	본사	공장	영업소	해외지사
사장	○			
비서실	○			
관리본부	전산실	○	○	
	총무부	○	○	
	경리/회계부	○	○	○
	구매부	○		
영업본부	○		○	○
마케팅부	○			
기획실	○			
생산부		○		
생산관리부		○		
품질관리부		○		

<표 2> 조직단위와 사업장 유형의 연관 분석표

사업장 유형	사업장	주소
본사	서울 본사	서울시 송파구 방이동
공장	반월 공장 안성공장	경기도 안산시 반월공단 경기도 안성군 대덕면
영업소	경인영업소 중부영업소 영남영업소 호남영업소	
해외지사		



<그림 3> 조직단위를 포함한 IDEF0 모델의 기능

2) 현행 및 개선 업무 프로세스 분석

IDEF0 모델에서 파악된 업무 기능들의 순서는 IDEF3 프로세스 모델을 사용하여 작성하며, 이때 IDEF0 모델과의 일관성(Consistency) 유지에 의한 모델 통합(Model Integration)을 위하여 IDEF3의 UoB(Unit of Behavior)를 IDEF0의 기능으로 표현한다.

3) 기능, 프로세스, 조직단위 및 사업장 유형의 연관분석

앞에서 작성된 IDEF0 모델과 IDEF3 모델을 근거로 하여 기능, 프로세스, 조직단위 및 사업장 유형들 간의 연관 분석표를 작성하는데, 이때 이들을 <표 3>과 같이 하나의 분석표에 간략히 나타낼 수 있다면 더욱 효과적일 것이다. 이 연관 분석표에서는 IDEF0와 IDEF3 모델의 최하위 계층의 기능들에 대하여 하나의 조직 단위 내 및 기업 내에서만 수행되는 지역적인 기능들(Intra-Organizational Functions, Local Activities)과 여러 조직단위 간 및 기업간에 걸쳐서 수행되는 전역적인 기능들(Inter-Organizational Functions 또는 Global Activities)과 기능들의 수행순서를 순서-특성 형식의 기호를 사용하여 동시에 나타낸다. 즉, 표에서 각 기능들은 수행되는 순서대로 1, 2, 3 번 순으로 나타내며, 행(Row)으

로 동일한 번호가 여러 조직단위에 걸쳐서 나타나는 기능들은 다시 조직단위 내에서만 수행되는 지역적인 기능들(L로 표시됨)과 조직단위간 또는 기업간에 걸쳐서 전역적으로 수행되는 기능들(G로 표시됨)으로 분류된다. 한편, 동일한 번호가 열(Column)로 여러 기능들에 걸쳐서 나타나면 동시에 수행되는 기능들(Parallel/Concurrent Activities)을 나타낸다. 단, 각 기능들의 수행순서 중 동기성/비동기성(Synchronous/Asynchronous)과 AND/OR/XOR (Exclusive OR) 논리는 IDEF3 모델을 참조한다. 상위 기능들에 관해서는 최하위 기능들을 그룹화하여 보면 그 순서와 특성을 쉽게 유출할 수가 있다.

예를 들어, <표 3>에서 기능 A1n을 구매업무라고 가정하면 구매업무는 본사와 공장의 구매 부서에서 각각 수행하는 업무지만 각 사업장마다 독특한 특색이 있으므로 지역적인

<표 3> 기능/프로세스/조직 연관 분석표

기능 (UoB) \ 조직			기업 A		기업 B	
			조직단위 (사업장유형) A1	조직단위 (사업장유형) An	조직단위 (사업장유형) B1	조직단위 (사업장유형) Bm
최상위 기능 (UoB) A0	상위 기능 (UoB) A1	하위 기능 (UoB) A11	1-L			
		하위 기능 (UoB) A1n		2-L	2-L	2-L
	상위 기능 (UoB) An	하위 기능 (UoB) An1		3-G	3-G	
		하위 기능 (UoB) An2		3-L		

기능(L)에 해당된다. 또한, 기능 An1을 제품관리업무라고 가정하면, 제품관리업무는 본사, 영업소, 공장 및 해외지사 등에서 수행하는 업무이지만 그 내용은 전체적으로 동일하므로 전역적인 기능에 해당하여 향후 시스템 설계시에 전역적인 어플리케이션 서버를 통해 조직간 또는 기업간에 공통으로 수행할 수 있을 것이다. 즉, 이와 같은 형태의 분산분석은 시스템 설계 단계에서의 분산/집중 설계에 꼭 필요하게 된다.

한편, 분석 단계에서 파악된 업무 영역의 모든 분야에 대하여 CALS 시스템을 구축할 필요는 없을 것이다. 따라서, 표3의 기능, 프로세스, 조직단위 및 사업장 유형들 간의 연관 분석표를 근거로 정보시스템을 도입할 대상과 범위 및 우선 순위를 결정하고, 기능 및 프로세스에 대한 설계 작업으로 넘어간다.

3.3.2 분산 설계 단계

1) 기능 분할

여기서는 최하위 계층의 각 기능들을 다음 2가지 원칙 중의 하나를 이용하여 분할 (Partitioning, Clustering)한다:

원칙 1: 하나의 조직에만 관련된 기능들과 여러 조직간에 걸쳐서 수행되는, 즉 공유되는 기능들로 분할한다

원칙 2: 순서적으로 수행되는 기능들(Sequential Functions)과 동시에 수행되는 기능들로 분할한다

이러한 기능 분할 원칙의 결정은 바로 다음에 설명하는 기능의 할당(Allocation, Deployment) 작업과 매우 밀접하게 관련되어

있고, 분할과 할당 작업은 순차적인 작업이 아니라 함께 고려해야 하는, 즉 동시에 수행되는 작업이다..

2) 기능 할당

여기서는 분할된 최하위 계층의 기능들을 아래에 제시한 2가지 할당 원칙중에서 하나를 적용하여 시스템 구조도상의 각 사이트에 할당하고, 그 내용은 각각 <표 4>와 <표 5>와 같은 기능 할당 표에 정리한다:

<표 4> 기능 할당 표(기능 할당 원칙 1)

기능(서버 타입) 사이트	지역 기능 (Local Application Server)	전역 기능 (Global Application Server)
사이트1	A111 A114	A112 A122
사이트 N		

원칙 1: 하나의 조직에만 관련된 기능들은 각 지역 어플리케이션 서버(또는 각 지역 내의 클라이언트 PC)에 할당하고, 여러 조직간에 걸쳐서 수행되는, 즉, 공유되는 기능들은 전역 어플리케이션 서버에 할당한다.

원칙 2: 순서적으로 수행되는 기능들(Sequential Functions)은 전역 어플리케이션 서버에 할당하고, 동시에 수행되는 기능들

은 각 지역 어플리케이션 서버(또는 각 지역내의 클라이언트 PC)에 할당한다. 이렇게 할당을 하면 각 지역 어플리케이션 서버에서 동시에 수행될 기능들에 대한 병렬 프로세싱(Parallel Processing)이 가능하게 되어 시스템의 수행 성능이 향상된다.

3) 어플리케이션 로직 설계

분석 단계에서 파악된 기능들 중에서 최하위 계층의 기능(The Lowest Level Function, Leaf Function)들에 대하여 처리명세서(Process Spec., Mini-Spec.)를 작성한다. 처리명세서는 기존에 널리 알려져 있으므로 중복을 피하기 위하여 본문에서는 생략한다.

<표 5> 기능 할당 표(기능 할당 원칙 2)

기능(서버 타입) / 사이트	동시 수행 기능 (Local Application Server)	순차적 기능 (Global Application Server)
사이트1	A111 A114	A112 A122
사이트 N		

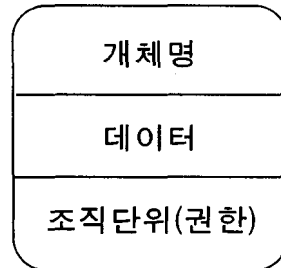
3.4 이질 분산형 통합 데이터베이스 개발

3.4.1 분산 분석 단계

1) 현행 및 개선 데이터 모델 분석

개념적 데이터 모델(Conceptual Data Model, Conceptual Schema)로서 IDEF1 정보모델을

작성하며, 관계형 데이터베이스의 구축을 목적으로 하여 개념적 모델을 더 상세화한 논리적 데이터 모델(Logical Data Model)로서 IDEF1x 데이터 모델을 작성한다. 즉, 대부분의 기존 데이터베이스가 관계형이므로 본 방법론에서는 관계형 모델을 공통/표준 스키마로 하였다. 이때 <그림 4>와 같이 각 데이터와 연관된 조직단위와 그 접근권한(Privilege 또는 Access Right)을 모델링 하는데, 여러 사업장 유형별로 존재하는 조직단위에 대해서는 괄호를 사용하여 사업장 유형을 명시해 준다. 조직단위의 역할은 각 개체에 대해 수행하는 연산(Operation)의 가장 기본단위인 CRUD(Create, Read, Update, Delete)를 척도로 하여 표현한다.



<그림 4> 조직단위와 접근권한을 포함한 IDEF1/1x 모델의 개체

2) 데이터와 조직단위 및 사업장 유형의 연관 분석

데이터와 조직단위/사업장 유형간의 연관 분석표를 <표 6>과 같이 CRUD 척도를 사용하여 작성한다. CRUD Matrix는 데이터가 어느 조직단위/사업장에서 어떻게 사용되는지를 알 수 있는 좋은 방법이다.

<표 6> 데이터와 조직단위/사업장 유형간의 연관분석표

조직 데이터	사업장 A			사업장 B		
	조직단원 A1	조직단원 An	조직단원 B1	조직단원 Bn
데이터 1	CRUD		RU			R
데이터 2	R		CRUD	RU		RU
⋮						
데이터 n	R		RU	CRUD		RU
⋮						
데이터 m	RU		CRUD			

3.4.2 분산 설계 단계

1) 데이터 분할

IDEF1x 모델의 개체, 즉 전역 릴레이션(Global Relation)들을 분할들로 나누는 작업에는 크게 수평분할(Horizontal Partitioning)과 수직분할(Vertical Partitioning)의 두 가지 방법이 있다. 본 방법론은 이미 기존에 잘 알려진 이 두 가지 분할 방법을 그대로 적용한다. 따라서 여기서는 중복을 피하기 위하여 이에 관한 설명을 본문에서는 생략한다.

2) 데이터 할당

여기서는 분할된 데이터를 아래에 제시한 3가지 할당 원칙중의 하나를 적용하여 시스템 구조도상의 각 사이트에 할당하고, 그 내용은 <표 7>과 같은 자료사전(메타 데이터)에 정리한다. 자료사전은 데이터의 저장위치뿐만 아니

라 데이터의 자료구조(Data Structure)를 정의하고 검색하기 위한 것으로서 논리적 데이터 모델인 IDEF1x를 기반으로 하여 작성한다.

원칙 1: 분할된 데이터들을 각각의 데이터가 생성(Create)되는 사이트에 중복 없이 할당한다. 즉, 데이터는 한 번만 생성해서 공유한다 또는 One Fact in One Place의 원칙에 따라 중복 없이 데이터를 생성하는 사이트에만 저장하고, 이를 공유한다.

원칙 2: 앞의 원칙을 적용하여 생성되는 사이트에만 중복없이 저장하고, 추가로 다른 사이트에서 검색(Read, Retrieval)만을 행하는 데이터는 검색하는 사이트에도 중복하여 저장한다.

원칙 3: 첫번째 원칙을 적용하여 생성되는 사

<표 7> 자료 사전(메타 데이터)

분할 (개체)	속성	Type	길이	형식	제한범위	유일성	Null허용	기본값	저장위치 (사이트)	설명
거래처	거래처코드 상호명	Int Char	4 20	9999 XXXX	0001-9999	Y N	N N	.	.	.
자재										

이트에만 중복 없이 저장하고, 다른 사이트로부터의 검색 요구가 갱신 요구(Update Request)보다 많은 데이터는 해당 사이트에도 중복하여 저장한다.

첫 번째 원칙은 같은 데이터에 대하여 여러 사본을 둬으로써 야기되는 비일관성(Inconsistency) 문제를 해결하는데 목적이 있다. 즉, 데이터를 중복 저장하게 되면 갱신 요구를 처리할 때마다 그 데이터의 모든 사본에 대해 동일한 갱신을 수행해야만 하므로 여러 사이트에 분산된 중복데이터의 일관성을 유지하기가 어렵게 되기 때문이다.

그러나 분산 데이터베이스에서는 데이터 중복성이 바람직한 특성으로 생각되는 이유들이 있다. 첫째로, 작업들을 처리하는데 있어서 작업이 제기된 사이트 내에서 처리될 수 있는 지역성(Locality)은 작업에 의해서 요구되는 사이트에다 필요한 데이터를 저장하면 높아지기 때문이다. 이를 위해서는 데이터의 중복 저장이 요구된다. 둘째로, 데이터가 중복되면 사이

트 하나가 고장이 나서 작동하지 못하더라도 다른 사이트에서 필요한 데이터가 중복 저장되어 있으므로 작업의 실행이 중단되는 일이 없게 된다. 즉, 시스템의 가용성(Availability)이 증가하게 된다. 그러나 첫번째 원칙에서는 이 경우 각 사이트마다 모든 관련 작업들을 수행할 수 없게 된다.

한편, 중복도를 감소시키자는 일반적인 원칙은 옳은 점이 있기 때문에 어느 정도의 중복을 허용하느냐 하는 것을 결정해야만 한다. 따라서 두번째 원칙과 세번째 원칙은 중복을 허용하여서 검색 작업이 데이터의 사본들 중의 아무 것이나 하나에 대해서 처리될 수 있도록 함으로써 지역성과 가용성을 증가시킬 수 있다. 세번째 원칙은 두번째 원칙보다 지역성과 가용성은 상대적으로 증가되지만 일관성 유지는 더 어렵다.

3) 기존 지역 스키마로의 변환 및 새로운 스키마 생성

활당된 데이터의 분할이 해당 지역 사이트의 데이터베이스에 이미 존재하지만 스키마가

다를 경우에는(즉, 관계형 스키마가 아닐 경우에는) 기존 스키마로 변환하여 준다. 또한, 할당된 데이터 분할 중에서 존재하지 않는 데이터가 있을 경우에는 기존 데이터베이스의 스키마와 데이터를 수정한다. 만일 지역 사이트에 데이터베이스 자체가 존재하지 않을 때는 할당된 데이터 분할들을 저장하는 새로운 데이터베이스를 생성한다.

3.5 분산 어플리케이션과 분산 데이터베이스의 통합 절차

3.5.1 분산 분석 단계

1) 기능/데이터 연관분석

분산 어플리케이션들은 사용자가 요구하는 작업을 수행하기 위하여 필요한 데이터를 분산 데이터베이스로부터 전송받아 트랜잭션을 수행하거나, 데이터를 생성하고 변경하여 데이터베이스에 저장한다. 따라서 요구되는 업무 기능에 필요한 데이터를 기능/데이터 연관분석

표 CRUD 척도를 이용하여 작성하여 파악한다(<표 8>).

3.5.2 분산 설계 단계

1) 통합 시스템 구조도(Architecture) 설계

시스템 구조는 기능/프로세스/조직 연관 분석표, 데이터/조직 연관 분석표, 기능/데이터 연관 분석표 등을 기반으로 하여 여러 가지 다양한 형태로 설계될 수 있다. 분산 어플리케이션과 분산 데이터베이스를 통합한 시스템의 일반적인 구조도의 한 예로서 <그림 5>의 구조도를 제시하였으며, 이 구조는 기업내의 전사적인 통합 시스템 구조로서 적합하다. <그림 5>에서 사용자의 질의를 수행하는 처리절차는 다음과 같다:

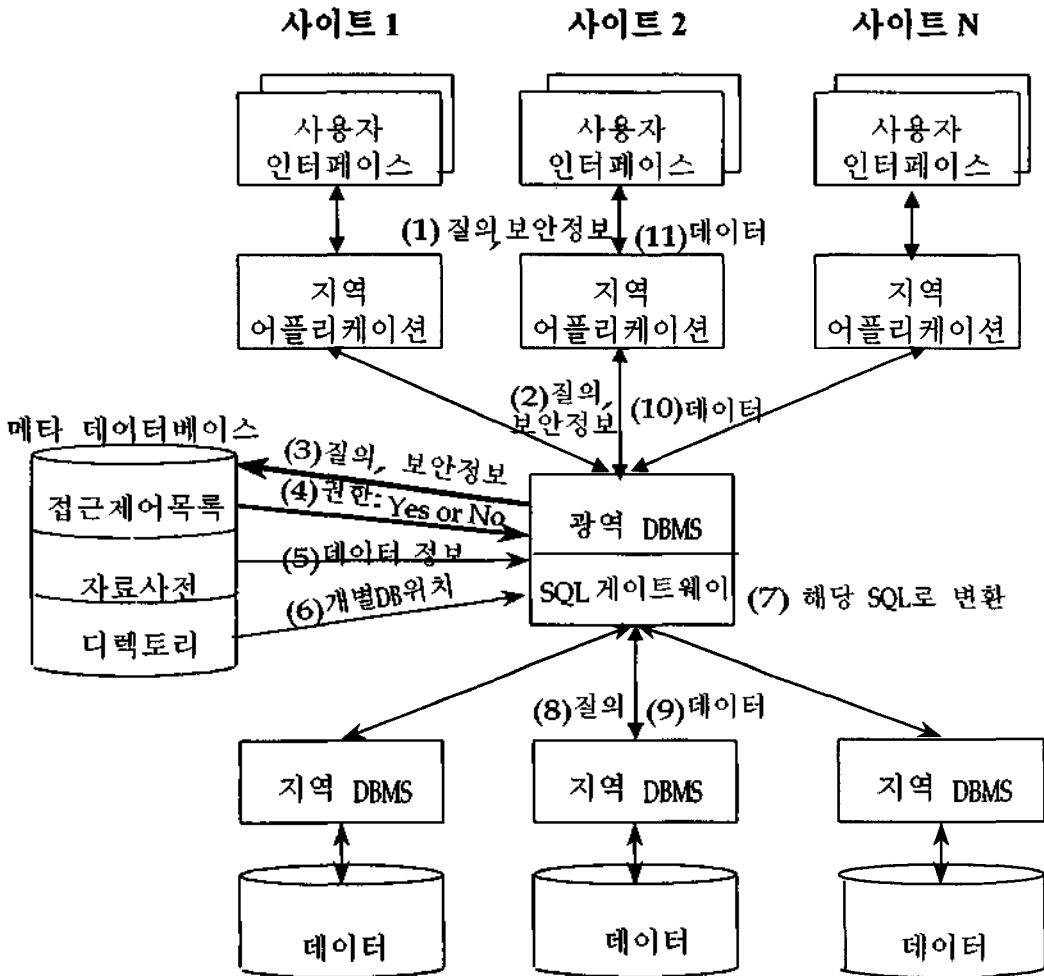
- ① 사용자 인터페이스는 사용자의 질의와 사용자의 보안정보(ACI: Access Control Information)를 지역 어플리케이션에 보낸다.

<표 8> 기능/데이터 연관 분석표

기능(UoB)		데이터	기업 A			기업 B		
			데이터	데이터	데이터	데이터
기능 A0	기능 A1	기능 A11	CD					RU
		기능 A1n	RU		CRUD			
	기능 An	기능 An1						
			R		R	CRUD		
		기능 Ann	R			R		CRUD

- ② 지역 어플리케이션은 요청받은 트랜잭션을 수행하는 과정에서 필요한 데이터를 데이터베이스로부터 전송받기 위하여 전역 데이터 관리 시스템(GDMS: Global Data Management System)의 전역 DBMS에 질의와 보안정보를 보낸다.
- ③ 전역 DBMS는 메타 데이터베이스(전역 데이터베이스)의 접근제어목록(ACL: Access Control List)에 질의와 보안정보를 보내어

- 사용자의 요구한 정보에 대한 접근권한 여부를 확인한다.
- ④ 접근권한 여부가 확인된다. 만일 접근권한이 있으면 다음 단계를 수행하고, 권한이 없으면 질의가 기각된다.
- ⑤ 접근권한이 있을 경우 자료사전(Data Dictionary)으로부터 데이터의 정보를 확인한다.
- ⑥ 또한, 전역 디렉토리(Global Directory)로부



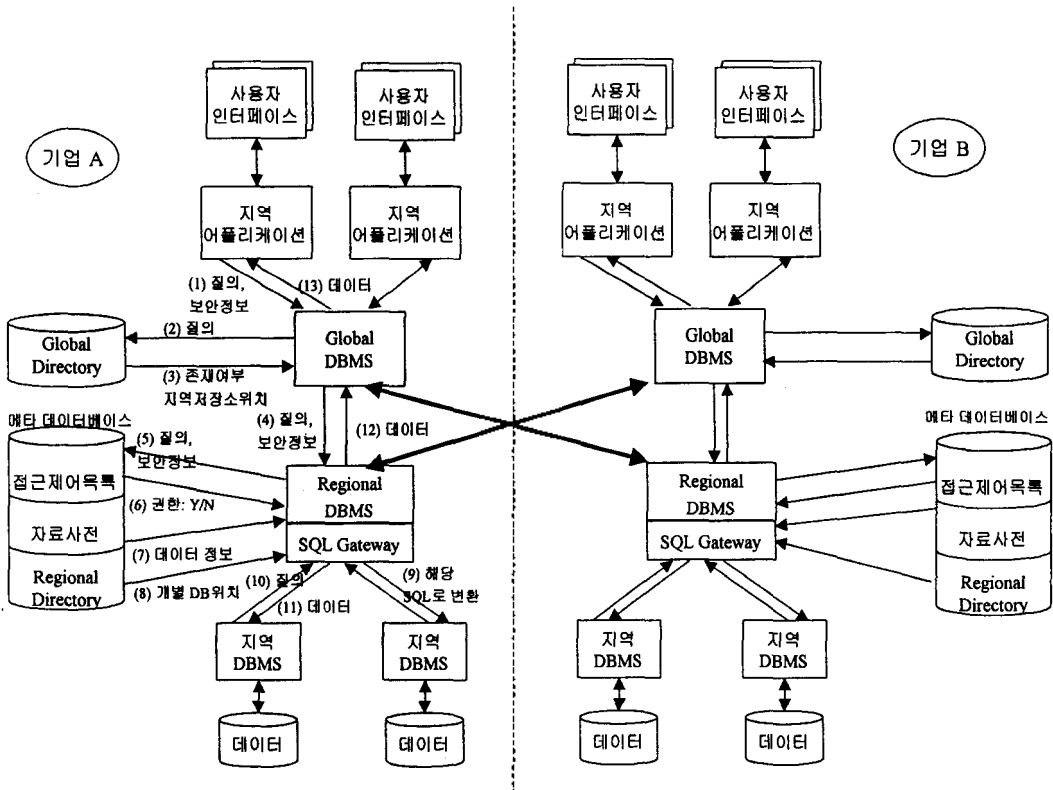
<그림 5> 분산 통합 시스템 구조도 및 질의 처리 절차(기업 내 통합)

터 해당 데이터를 저장하고 있는 지역 데이터베이스의 위치(사이트, 노드)를 확인한다.

- ⑦ 전역 DBMS와 해당 지역 DBMS의 SQL이 다를 경우에는, SQL 게이트웨이(Gateway)를 통하여 주어진 질의를 지역 DBMS의 SQL로 변환한다.
- ⑧ (해당 SQL로 변환된) 질의를 지역 DBMS에 전송한다.
- ⑨ 지역 DBMS가 데이터를 검색하여 전역 DBMS로 전송한다.
- ⑩ 전역 DBMS가 데이터를 지역 어플리케이션에 전송한다.
- ⑪ 지역 어플리케이션이 데이터를 가져오트

랜잭션을 수행하여 사용자 인터페이스에 결과를 보낸다.

분산 어플리케이션과 분산 데이터베이스를 통합한 시스템의 일반적인 구조도의 다른 예로서 CALS 및 가상기업 환경에서 기업내의 전사적인 통합 시스템뿐만 아니라 기업간의 시스템을 함께 통합한 시스템 구조와 질의 처리 절차는 <그림 6>과 같으며, 이러한 통합 데이터베이스(IDB) 구조는 미 국방부의 통합 데이터 환경(IDE: Integrated Data Environment) 개발 프로젝트에서 구축중인 IWSDB(Integrated Weapon System Data Base) 시스템 구조이다. <그림 5>와 <그림 6>의 차이점은 전역 데이터 관리 시스템을 Global DBMS(기업간)

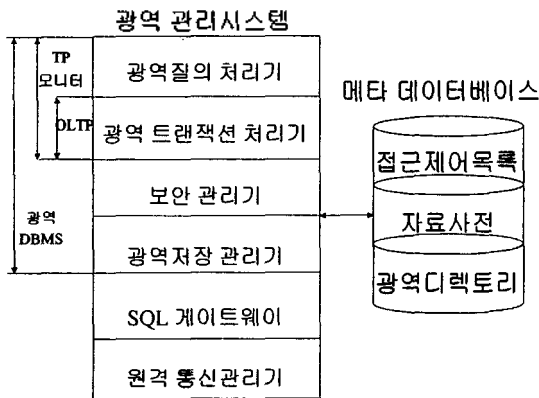


<그림 6> 분산 통합 시스템 구조도 및 질의 처리 절차(기업 내 및 기업간 통합)

와 Regional DMBS(기업내)로 분리하여 기업간에 정보를 공유할 수 있도록 기업간의 Global DBMS를 추가한 것이다.

2) 전역 데이터관리시스템 및 메타 데이터 베이스 설계

일반적으로 전역 데이터관리시스템(GDMS: Global Data Management System)과 전역 데이터베이스의 역할을 하는 메타 데이터베이스는 <그림 7>과 같이 구성된다(여기서는 크게 분류하였으며 구성요소들을 더 자세하게 분류할 수 있을 것이다).



<그림 7> 전역 데이터 관리 시스템과 메타 데이터베이스의 구성요소

전역 데이터 관리 시스템의 설계 및 구현은 업무 프로세스 및 데이터의 분석 및 설계와 무관하게, 요구되는 소프트웨어의 기능들을 설계하고 구현하게 되므로 본 방법론에서는 다루지 않는다. 반면에, 메타 데이터베이스는 업무 프로세스와 데이터의 분석 및 설계 작업과 관련이 있다. 즉, 접근권한 확인을 위한 접근 제어 목록은 데이터/조직 연관 분석표와 기

능/데이터 연관 분석표를 기반으로 하여 설계하고, 데이터의 저장 장소 확인을 위한 전역 데이터 디렉토리는 자료사전과 데이터/조직 연관 분석표를 기반으로 설계한다.

4. 결론

분산시스템의 성공은 데이터를 한번 생성하여 여러 번 사용할 수 있는 분산형 통합 데이터베이스 시스템의 구축이 관건이라 할 수 있다. 분산 시스템의 성공 전략은 해당 자료를 정보기술 표준에 입각하여 디지털 형태로 전환하는 것이고, 또한 이를 분산된 관련 조직, 즉 정부와 관련 기업들이 공유할 수 있도록 논리적으로 통합된 분산 통합 데이터베이스를 구축하여 제품의 전 수명주기에 활용하는 것이다. 따라서 본 연구에서 개발된 방법론은 분산형 통합 데이터베이스 어플리케이션 시스템의 구축, 개발에 활용될 수 있을 것이다.

본 연구의 방법론은 조직의 구조에 따른 분산 시스템 개발 방법론으로서 기존의 방법론과 차별화된 특징은 다음과 같이 요약될 수 있다:

- 1) 기존 이질분산형 데이터베이스들의 하향식 통합 방법론
- 2) 기업 내 및 기업간의 업무 프로세스와 데이터의 분산 분석 및 설계 방법론
- 3) 분산 어플리케이션과 분산 데이터베이스를 위한 통합 개발 방법론
- 4) 고유한 기능/프로세스/조직 연관 분석표, 기능 분할 방법, 기능 할당 방법, 데이터 할당 방법, 지역 스카마와의 통합 방법의 제시
- 5) 전역관리시스템과 메타(전역) 데이터베이스 설계와의 연계 방법 제시

참고문헌

- [김덕현 1996] 김덕현, 통합데이터 환경: 개념 및 구현기술, 한국CALIS/EC 학회지, 제1권, 제1호, pp. 69-91, 1996.
- [문송천, 김유성 1994] 문송천, 김유성, 의뢰자-제공자 데이터베이스, 집현전, 1994.
- [신선우 등 1996] 신선우, 손해용, 김철환, 국방 CALS를 위한 IWSDB 구축에 관한 연구, 한국 CALS/EC 학회지, 제1권, 제1호, pp. 93-115, 1996.
- [우훈식 등 1997] 우훈식, 장석찬, 백종명, 주경준, CALS 통합 데이터베이스 구현 방안에 관한 연구, 한국경영과학회/대한산업공학회 97 춘계공동학술대회, pp.641-644, 1997. 4.
- [Chang and Cheng 1980] Chang, S.K. and Cheng, W.H., A Methodology for Structured Database Decomposition, IEEE Transactions on Software Engineering, Vol.6, No.2, 1980.
- [Bright et al. 1992] Bright, M.W., Hurson, A.R. and Simin, H.P., "A Taxonomy and Current Issues in Multidatabase Systems," IEEE Computer, Vol.25, No.3, 1992.
- [Ceri et al. 1982] Ceri, S., Negri, M. and Pelagatti, G., "Horizontal Partitioning in Database Design," Proceedings of ACM SIGMOD Conference on Management of Data, 1982.
- [Ceri et al. 1983a] Ceri, S., Gottlob, G. and Pelagatti, G., "Joining Fragmented Relations in Distributed Databases," Technical Report DEPM-83-9, Dept. of Electronics, Politecnico, Milano, 1983.
- [Ceri et al. 1983b] Ceri, S., Navathe, S.N. and Wiederhold, G., "Distribution Design of Logical Database Schemas," IEEE Transactions on Software Engineering, Vol. 9, No. 4, 1983.
- [Ceri and Pelagatti 1984] Ceri, S. and Pelagatti, G., Distributed Databases: Principles and Systems, McGraw-Hill, 1984.
- [Doumeingts 1992] Doumeingts, G. and Chen, D., State-of-the-Art on Models, Architecture and Methods for CIM System Design, Human Aspects in Computer Integrated Manufacturing, Olling, G.J. and Kimura, F.(Eds.), Elsevier Science Publishers B.V. (North-Holland), pp.27-40, 1992.
- [Kim 1992] Kim, Y.S., Atomic Transaction Scheduling in Tightly Coupled Heterogeneous Technology, 1992.
- [Kim 1989] Kim, W., "Research Directions in Heterogeneous Databases," IEEE Office Knowledge Engineering, Vol. 3, No. 2, 1989.
- [Litwin et al. 1990] Litwin, W., Mark, L. and Roussopoulos, N., "Interoperability of Multiple Autonomous Databases," ACM Computing Surveys, Vol. 22, No. 3, 1990.
- [Monarchi 1992] Monarchi, D.E. and Puhr, G.I., A Research Typology for Object-Oriented Analysis and Design, Communications of the ACM, Vol.35, No.9, pp.35-47, 1992.

-
- [Navathe et al. 1982] Navathe, S.B., Ceri, S., Wiederhold, G. and Dou, J., "Vertical Partitioning for Physical Distribution and Design of Databases, Technical Report STAN-CS-82-957, Stanford University, 1982.
- [Papazoglou and Marinos 1990] Papazoglou, M.P. and Marinos, L. "An Object-Oriented Approach to Distributed Data Management," International Journal of Systems and Software, 1990.
- [Sheth 1991] Sheth, A.P., "Federated Database Systems for Managing Heterogeneous Distributed, and Autonomous Databases, Tutorial Notes of 17th International Conference on the Very Large Databases, 1991.
- [Sheth and Larson 1990] Sheth, A.P. and Larson, J.A., "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases," ACM Computing Surveys, Vol. 22, No. 3, 1990.
- [Thomas et al. 1990] Thomas, G., Thompson, G.R. et al., "Heterogeneous Distributed Database Systems for Production Use," ACM Computing Surveys, Vol. 22, No. 3, 1990.
- [Weston 1990] Weston, R.H. "Steps towards enterprise-wide integration: a definition of needs and first generation open solutions", International Journal of productionResearch, 1993, Vol31, No.9, pp. 2235-54

저자소개

김중인

한양대학교 산업공학과 학사 및 석사
Arizona State University 산업 및 경영시스템공학과 박사
한국전자통신연구원 선임연구원
현재 홍익대학교 경영정보학과 조교수
관심분야: 전자상거래, ERP, 시스템분석및설계

김철한

한양대학교 정밀기계공학과 학사 및 석사
포항공대 산업공학과 박사
현재 대전대학교 정보시스템공학과 조교수

우훈식

한양대학교 산업공학과 학사
Iowa State University 산업공학과 석사 및 박사
현재 대전대학교 정보시스템공학과 전임강사

임동순

한양대학교 산업공학과 학사
한국과학기술원 산업공학과 석사
Iowa State University 산업공학과 박사
현재 한남대학교 산업공학과 부교수