전사적 자원 관리 시스템을 위한 기업 리파지토리 구축

이희석* 이 제* 이충석* 조창래* 손주찬** 백종명**

Implementing an Enterprise Repository for Enterprise Resource Planning System

Heeseok Lee* Jay Lee* Choongseok Lee* Changrae Cho* Joochan Sohn** Jongmyung Baik**

Abstract

Currently, many companies are trying to be more competitive by implementing the most appropriate software package, called ERP (Enterprise Resource Planning). ERP can support the company's business process. A repository, which can store and retrieve enterprise-wide information, is a key component of such software package. This paper implements an enterprise repository for the ERP. A well-known repository standard, IRDS (Information Resource Dictionary System), is employed for the repository architecture. A meta schema that can help develop database, business applications, models, and workflow is proposed. To illustrate the practical usefulness, a real-life ERP prototype is built within the framework of the proposed repository.

I. 서 론

급변하는 새로운 경영환경은 현재 가지고 있는 정보기술의 변화와 경영혁신이 끊임없이 요구된다. 그러나, 이러한 혁신은 전체적인 계획에 따라 이루어져야 한다. 즉, 필요 시 마다 개별적으로 각기 구축되어진 정보시스템은 이질적인 모델로 인하여 생산성 향상과 정보 재사용성에 제약을 받게 된다. 이러한 문제점에 전략적으로 대처하기 위해서는 변화를 통합적으로 관리할 수 있는 기법이 필요하다. 지금까지 이질적인 시스템 환경에 저장되어 있는 다양한 기업 내 정보 자산을 통합적으로 관리하고 재사용하기 위하여 CASE (Computer Aided Software Engineering) 사용 시스템 개발, 시스템의 배치 (Configuration) 와 버전관리 (Version Control), 통합모델관리를 사용하였다. 이러한 기술들은 메타 정보를 활용하여 이질적이거나 분산된 자원을 통합적으로 관리하려는 것이다.

이러한 메타 정보에 관한 기존 연구는 대부분 단일한 CASE 환경 또는 데이터베이스의 기술 정보 (Descriptive Information) 만을 포함하는 협의의 메타 데이터에 관한 것이다. 최근에 이러한 한계를 극복하기 위해 중요한 기술로 대두되고 있는 것이 전사적 자원 관리 시스템 (ERP, Enterprise Resource Planning)이다. ERP 는 기업의 생산, 자재, 영업, 인사, 회계 등의 업무를 통합 관리해주는 대형 경영관리용 패키지 소프트웨어이다. ERP 의 수요자는 공급자가 제공하는 패키지 가운데 자신의 기업환경에 적합한 모듈만 부분적으로 선택하여 시스템을 유연하게 구축할 수 있으며, 또한 새로운 시스템으로 확장할 수 있다. ERP 의 장점인 이러한 시스템 통합성과 유연성으로, 기업은 정보시스템을 자체 개발해야 하는 부담을 줄일 수 있어 ERP 를 사내 정보 인프라로 구축, 활용할 수 있다[16].

이러한 ERP 를 위해 필요한 것은 지식 자산관리의 프레임워크인 리파지토리 (Repository)의 구축이다. 리파지토리는 기업의 경영전략에서 구현까지의 모든 정보를 통합적으로 관리하여 유연성, 포괄성, 개방성을 제공하여 특정 개발 환경에 독립적으로 존재할 수 있는 메타 정보 관리 기법[12][14]이다. 즉, 이러한 리파지토리를 ERP 와 같은 패키지방식의 개발도구에 활용함으로써, 전사적 관점에서 기업의 자원과 지식의 효과적 통합 관리가 가능하다. 실제로 SAP (Systems, Applications and Products in Data Processing)과 같은 선진 ERP 업체의 경우에도 ERP 시스템에서 리파지토리의 비중을 중요시 하고 있다. 또한, 비즈니스 개념을 객체지향적으로 모듈화 한 비즈니스 객체가 리파지토리에 저장되는 방식 [17]을 통해 특정 업무 영역의 표준지식을 표현하고 있다.

본 연구의 목적은 ERP를 위한 통합 지식 저장소인 리파지토리를 구축하는 것이다. 이를 위해 리파지토리 구조화 방법론이 요구되는데, 아직까지 이에 대한 표준안이 미비하며, 다만 IRDS (Information Resource Dictionary Systems) [2]와 같은 개념적인 표준 프레임워크 및 표준 기술명세서 정도만이 있을 뿐이다. 본 연구에서는 (i) ERP 환경에 맞게 실세계의 요구사항을 충실히 반영함과 동시에, (ii) 재사용성에 용이한 계층화 개념을 가진 다차원 모델의 장점을 살린 IRDS가 제시하고 있는 4계층 (Level) 프레임워크를 적용하는 방식을 통하여 ERP용 리파지토리를 설계하고자 한다.

II. IRDS 기반 리파지토리

리파지토리에 관한 정의는 [표 1]과 같이 J. Martin, T. Moriarty, R. Mcgaughey & M. Gibson [12], 그리고 A. Tannenbaum [20]에 의하여 제시되고 있으며, 본 연구에서는 리파지토리를 정보 공유의 틀로써 모델과 구현도구에 독립적으로 활용되어질 수 있는 조직화된 자료 구조 통합체로 보았다. 즉, 기업에 의해서 사용되거나 생성된 통합된 정보 데이터베이스의 틀로 본 것이다.

[표 1] 리파지토리 정의

정의자	정 의
J.Martin (1989)	정보 공학 요소들 간 다양한 정보와 정보들간의 연결을 포함
T. Moriarty (1993)	각 공급업체로부터 제공되는 특수한 유틸리티가 아니라, 기업의 전략적 관리를 위한 시스템
R. Mcgaughey & M. Gibson (1993)	정보저장을 위한 빈 그릇으로 비유할 수 있으며, 만일 기업모형의 컴포넌트 들이 채워진다면 정보백과사전 이라고 할 수 있음
A.Tannenbaum (1994)	리파지토리 이외에 존재하는 이질적인 모델타입에 연계되어질 수 있는 하나의 통합된 저장 공간
본 연구(1998)	정보 자원의 공유를 위해 조직화 되어진 저장 구조 통합체로 활용되어질 목적성을 갖고 있으며, 모델과 구현도구에 독립적으로 활용될 수 있음

리파지토리 활용의 장점은 (i) 특정 개발 환경에 독립적으로 적용가능하며, (ii) 이질적인 개발 도구들을 상호 유기적으로 연결하여 통합적인 개발 접근이 가능하며, (iii) 장기간의 사용 시 기업 정보의 축적이 가능해진다는 것이다. 궁극적으로 리파지토리는 조직이 가지고 있는 모든 지식자산을 통합적으로 저장 관리 할 수 있는 기반 기술로 활용될 수 있다.

통합을 목적으로 한 메타 정보 개념에 관한 연구는 [표 2]와 같다. 기존의 원시적인 형태의 데이터베이스 자료구조 및 레코드, 필드 정보 등 단순한 데이터 관리용 메타 정보인 자료사전 (Data Dictionary) [11]과, 이를 포괄하여 단일한 CASE 도구상에서 계획에서 분석, 설계, 그

* 본 연구는 한국과학기술원 ETRI 수탁연구과제 (98-17-P00540)로 수행되었습니다.
* 한국과학기술원 기업정보시스템연구실 서울 동대문구 청량리동 207-43 (02-958-3615, hlee@msd.kaist.ac.kr)
** 한국전자통신 연구원 대전 유성구 어은동 1번지

리고 구현에 이르기까지의 모든 정보를 통합 관리하는 메타 정보 체계인 백과사전 (Encyclopedia) [15]으로 발전하였다. 그러나, 기업의 전산환경이 이질적으로 발전하기 쉬우므로, 단일한 CASE 환경에서의 메타 정보 체계는 그 한계가 있다. 이에 자료구조를 통합적으로 구조화하며, 전사적인 관점에서 기업 전 조직의 경영전략에서 구현까지 모든 정보를 통합적으로 관리하기 위해 특정 개발 환경에 독립적으로 존재할 수 있는 메타 정보 관리 기법으로써 리파지토리 개념 [12][14] 이 필요하다. 이 개념은 (i) 리파지토리를 관리하고 사용자가 리파지토리에 접근할 수 있는 기능을 제공하는 리파지토리 시스템과, (ii) 기업의 지식 자산을 저장하는 리파지토리로 구분할 수 있다. 협의의 리파지토리는 기업 내 존재하는 다양한 지식 자산을 통합 모형화 하여 관리하기 위한 일종의 메타 데이터베이스 (Meta-database)로써 데이터베이스 및 어플리케이션, 다양한 모델링, 그리고 각종 기술정보 등의 각 객체와 그들 간의 관계를 통합적으로 저장할 수 있게 한다.

[표 2] 메타 데이터 관리 방법의 전이

방법 / 관점	자료사전 (Dictionary)	백과사전 (Encyclopedia)	리파지토리 (Repository)
자료범위	물리적 데이터베이스의 메타데이터	시스템 분석 설계의 결과	전사적 정보
구 조	비범용성	특정 비즈니스 영역에 제한적 사용	범용 구조 지향
공급업자와의 관계	종속적	종속적	독립적
표준화 작업	DBMS 업체별	CASE 업체별	IRDS / CDIF / PCTE / ATIS등 다양한 표준기관
모델지원	특정 모델만 지원	특정모델을 지원하나 모델 호환 일부 가능	다양한 모델 지원

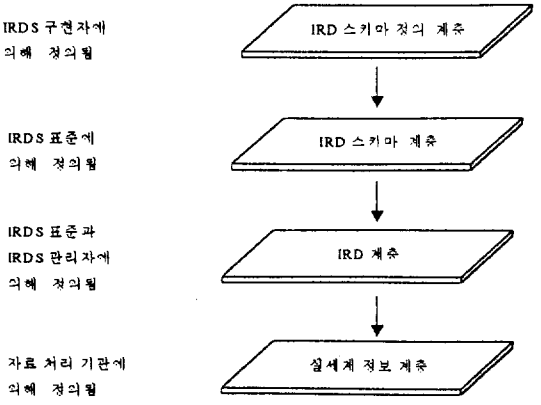
ERP로 대표되는 전사적 자원 관리시스템을 기업에 적용하기 위해, 본 논문은 두 가지의 고려 점을 염두에 두었다. 첫째, 리파지토리는 자료 구조의 틀을 제공하기 위한 표준화가 중요하므로 IRDS 표준에 의거하여 모델정보와의 인터페이스를 포함하는 ERP용 리파지토리를 설계한다. IRDS 표준안은 계층의 개념을 통해서 이러한 인터페이스 문제를 해결하고 있다. 본 연구에서도 IRDS의 계층 개념에 입각한 표준안을 사용, 분석에서 구현까지의 일관성 및 각 모델의 독립성을 보장할 것이다. 둘째, 이러한 IRDS 아키텍쳐하에서 ERP 구현이라는 특성에 맞게 리파지토리의 메타데이터 스키마를 설계하고 이를 구현한다. 현존하는 리파지토리 표준안은 다양한 구현 도구로부터의 이질적인 정보 통합이 주요한 관건이다. 이들 주요 표준안을 비교 요약하면 [표 3]과 같다. IRDS 4계층 구조는 [그림 2]와 같은 다차원 모델구조로서, 개체-관계모델을 기반으로 한다. 한 계층의 정보는 다음 하위계층에 저장된 정보를 설명하고 제어한다. IRDS 표준안은 계층 개념을 통해서 2계층을 하나의 쌍으로 보아 상위층은 정의부 (Definition), 하위층은 인스턴스로 이루어진다. 즉, 다음 하위 계층의 정보에 관한 객체/개체의 타입을 그 상위 계층에서 정의하게 되고, 하위 단계는 상위단계에서 정의된 객체/개체 타입에 관한 실제정보 또는 그 값을 갖게 되는 것이다. IRDS 표준안은 다차원 모델하에서 계층간의 일관성을 유지함으로써, 계층간의 인터페이스 문제를 해결하고 분석에서 구현까지의 일관성과 각 모델의 독립성을 보장하고자 한다.

[표 3] 리파지토리 표준안 비교

표준안 / 특성	IRDS	CDIF	PCTE	ATIS
표준화 기관	ANSI, ISO	EIA, ANSI	ECMA	DEC, AT
주요사항 및 특성	공유적 리파지토리 자료구조	CASE 도구 메타 데이타와 CASE 도구간의 데이터 전송	-객체 관리시스템 -본질적으로 소프트웨어 공학 환경의 기저를 위해 연구 시작 -기술지향적	-객체지향 패러다임 기반 표준안
구 조	4계층 구조 1) IRD 스키마 정의 계층 2) IRD 스키마 계층 3) IRD 계층 4) 실세계정보 계층	4계층 구조 -메타메타 모델 계층 -메타모델 계층 -모델 계층 -데이터계층	-객체타입 -객체 (연결, 속성, 관계)	-객체타입 -객체
모 델 통 합	개념적 가이드라인 제시	지원하지 않음	지원하지 않음	객체지향 관점에서만 지원
미비점	개념적 가이드라인 만을 제시	계층 간 정보공유와 상호참조 불가능, PCTE와 연계 않됨	미세한 데이타 공유에 부적합	단계별 상호참조 불가능

*EIA: Electronic Industries Association
*ECMA: European Common Tool Environment
*DEC: Digital Equipment Corporation
*AT: Atherton Technologies
*CRUD: Create, Read, Update, Delete

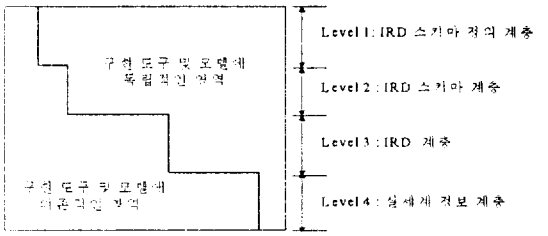
본 논문에서 집중 분석되어질 내용은 위의 그림 중 IRDS의 핵심인 IRD 스키마 계층과 IRD 계층 (IRD 계층쌍 2-3계층)이다. 따라서 이 부분에 대한 집중적인 연구와 설계가 필요하다. 이에 앞서, IRDS 표준안의 본질적인 프레임워크에 관한 고찰이 필요하다. ATIS는 IRDS에 객체 지향적 관점으로 흡수되어지는 과정이 있었는데[20], 이는 IRDS가 표준안으로서 통합적 체계를 견지하기 위한 방향성을 시사한다. 이를 위한 이상적인 IRDS 4계층 구조는 각 계층별로 모델 및 구현 도구에 독립적인 것이다. 현실적인 IRDS 구조 사상과 대비하면, 각 계층마다 구현도구 및 모델에 의 존적인 영역이 없고 모두 독립적인 영역만으로 사상 될 수 있다. IRDS가 제시하는 본연의 구조적 한계로 인해 현실적으로는 하위계층으로 갈수록 모델과 구현 도구에 의존적일 수 밖에 없다 [그림 3]. 특히 4계층이 의존성이 큰 것은 현실적인 정보 처리 단계의 내용인 인스턴스를 담고 있는 것에 기인한다.



[그림 2] IRDS 4계층 기본 모형

반면, 4계층의 상위 계층인 3계층은 4계층의 인스턴스에 관한 정의를 하고, 이를 제어하므로 4계층보다 추상적

어며 독립적인 영역이 커지게 된다. 이러한 2계층 쌍구조가 상위 계층에도 연쇄적으로 적용되므로, [그림 3]에서와 같이 전체 구조 사상체계는 상위층으로 갈수록 모델 및 구현도구에 독립적인 영역이 커져 가는 구조로 볼 수 있게 된다.

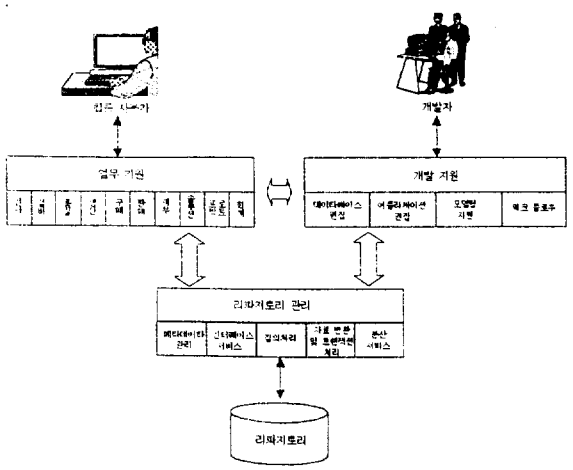


[그림 3] 현실적인 IRDS 구조 사상

III. ERP 리파지토리 구축

ERP 시스템은 [그림 4]와 같이 어플리케이션 편집, 개발 지원, 리파지토리 관리, 리파지토리의 4부분으로 구성되며, 세부기능은 [표 5]와 같다. 업무 지원 부분은 인사, 설비, 품질, 생산, 구매, 판매, 재무, 회계, 산업 솔루션 관리, 프로젝트 관리 등의 기업의 업무 영역에 관련한 부분이며, 리파지토리 관리는 각 프로그램 모듈의 타입 정보, 모듈과 데이터베이스 간의 관계, 모듈 및 데이터베이스의 버전 관리, 시스템 배치 관리, 보안 및 권한 등, 시스템의 관리 측면에서 필요한 정보를 관리한다.

개발 지원 부분은 크게 어플리케이션 편집, 데이터베이스 편집, 모델링 지원; 워크플로우의 4가지 기능을 제공한다. 어플리케이션 편집 분야는 각 업무를 대중소로 분류한 후, 응용프로그램들을 해당되는 각 분류체계에 할당하는 방법을 사용한다. 데이터베이스 편집 분야의 경우, 이질적인 데이터베이스 관리 시스템을 사용하는 분산환경에도 적용 가능할 수 있는 리파지토리가 설계가 요구된다. 이를 위해서 Microsoft SQL Server 와 Oracle 의 메타데이터 스키마를 분석하여 공통적으로 제공하는 기능과, 데이터베이스 관리 시스템이 데이터를 관리하는 방법들을 조사하여 포괄적으로 적용될 수 있게 한다. 모델링 지원분야의 경우, 리파지토리를 사용하는 경우, 개체 관계도를 사용하든, 객체 지향모형을 사용하든 모델간의 교환이 가능해야 한다. 워크플로우 분야는 조직의 정보와 밀접한 관계를 갖고 있다. 조직과 관련하여 사용자, 부서, 업무, 권한 등에 대한 내용을 관리하는 것이므로 데이터베이스 분야나 어플리케이션 분야의 보안 문제와의 관계가 필수적이다.



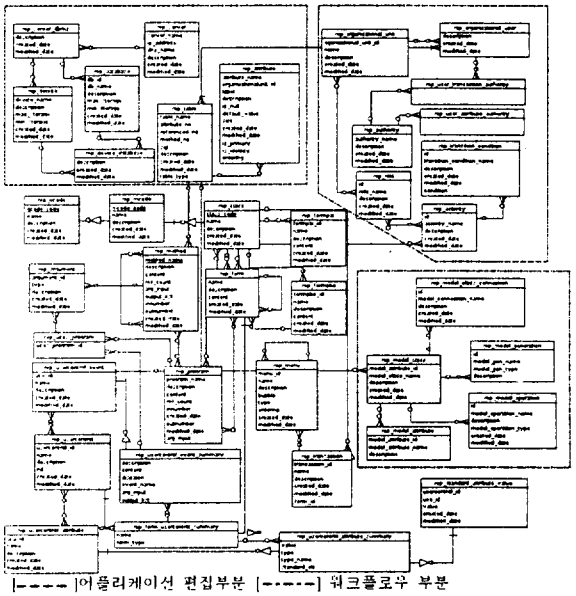
[그림 4] ERP 시스템 아키텍쳐

[표 5] ERP 시스템 세부기능

서브시스템	세부기능	설명
업무지원	인사관리	인사계획, 조직 개발, 인사정보관리, 근태관리, 시간관리,급여관리, 출장여비 관리 등의 종합적인 인사관리
	설비관리	보전일정관리, 예방보전, 보전 능력 계획, 보전정보관리, 서비스
	품질관리	생산된 상품의 불량 원인 관리 및 사후 간접 서비스 관리
	생산관리	생산 계획 수립, 작업지시 및 작업실적 등록, 생산 모델의 정의, 재고관리, 품질관리, 발주관리
	구매관리	구매요구관리, 견적관리, 거래계약관리, 청구서조회, 창고관리
	판매관리	거래 조회, 견적, 수주, 출하, 청구 등의 판매/유통 관리, 판매분석
	재무관리	간접비관리,제품원가계산 및 관리에 대한 다양한 원가관리, 업적관리, 수익성분석
	회계관리	일반회계, 외상매출관리, 외상매입관리, 어음관리, 예산관리
	프로젝트관리	프로젝트 일정관리, 프로젝트 정보관리
	산업솔루션관리	석유, 은행, 병원, 보험 등 업종 특유의 요구사항을 충족시키기 위한 기능만을 별도로 모듈화하여 관리
개발지원	테이터베이스편집	이질적 분산 환경하에서의 적용, 해당 데이터 타입 정보관리, 해당 테이블과 관련한 속성 등의 정보 관리 기능 지원
	어플리케이션편집	업무 분류, 사용자 관점에서의 화면 관리, 실행을 위한 트랜잭션 처리, 데이터베이스와의 연동되는 프로그램 구성 지원
	모델링지원	정보시스템 개발 시 모델의 생명주기 관리에 대한 정보, 모델 변화 관리 정보, 배치관리, 모델간 정보 교환 기능 지원
	워크플로우	비즈니스 프로세스 자동화, 조직의 사용자, 부서, 권한, 보안 등에 관한 내용 관리를 위한 지원
리파지-토리관리	메타데이터관리	전사적 관점에서의 데이터 요소들간의 표준화, 무결성 유지, 자료값의 범위 제한, 자료 유효성 명시 등의 리파지토리 관리
	인터페이스서비스	IRD 스키마 간의 인터페이스 및 사용자 인터페이스 처리상태 보고, 자료 입출력 결과 표현, 에러 메시지출력, 도움기능출력에 필요한 리파지토리 관리 기능
	질의처리	질의/답변 처리를 위한 리파지토리 관리
	자료변환 및 트랜잭션 처리	IRD간 적절한 데이터 처리를 위한 자료변환과 트랜잭션 확약을 위한 리파지토리 관리
	분산서비스	이질적인 데이터베이스 하에서의 정보 교환을 위한 리파지토리 관리
라파지-토리	자료구조 통합체	IRDS 표준안에 근거 기업의 정보 및 자원을 전사적 관리를 위한 구조적 자료 통합체

당해 년도에는 가장 많이 활용될 수 있는 구현 부분의 리파지토리를 먼저 설계했다. 설계 안에 포함된 내용은 어플리케이션 및 사용자 인터페이스, 데이터베이스 등이다. 이러한 구현 관련 요구사항을 개발 중인 ERP 시스템의 현실에 맞게, 수정 보완하여 구현용 리파지토리 메타 데이터 스키마를 설계했다 당해 년도 연구 결과 산출된 ERP용 리파지토리 시스템의 IRDS 제 2 계층은 60여 개이다. 연구 결과 제 3 계층의 인스턴스 정보는 500여 개로 산출되었다. 이에 해당하는 객체/개체는 사원, 업무 등이 있다.

IRDS 표준안에 따르는 리파지토리 스키마에서 실세계 정보를 관리하기 위한 계층이 바로 메타 데이터 계층이며 IRDS 프레임워크로 보면 제 3 계층이 되는 것이다. 이것의 추상화된 계층이며 관리 계층은 제 2 계층이다. 제 2 계층을 관리하는 모델정보가 메타 모델 정의 계층이며 IRDS 프레임워크에서 보면 제 1 계층이 되는 것이다. 이 제 2 계층 리파지토리 스키마는 ANSI IRDS에서 주로 사용이 되는 Entity Relationship Model (ERM) 을 사용하여 작성하였다[21]. 이와 같은 내용을 담은 개발 지원 분야의 IRD 계층에 대한 전체 리파지토리 스키마 중 일부 리파지토리 스키마를 Power-Designer의 CDM (Conceptual Data Model)을 이용하여 [그림 5]와 같이 표현하였다.



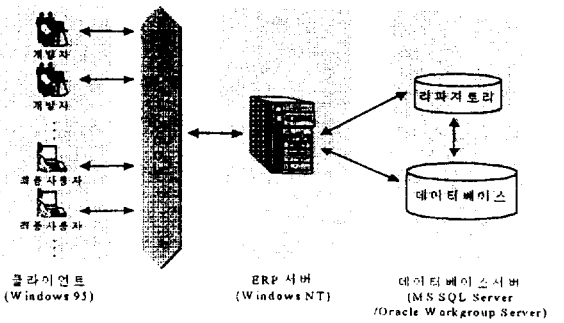
[그림 5] 리파지토리 제 2 계층 스키마

[그림 5]에서는 본 논문의 편의상 전체적인 스키마[18] 중 일부만을 도시하고 있으며, 전체 객체/개체는 500 여 개이다. SAP의 전체 스키마와 비교시 그 차이는 본 연구 관점에서는 SAP R/3는 다국화를 지원하는 모듈의 구성으로 스키마가 부분적으로 중복되었을 것으로 본다. 이에 비해 본 연구 논문은 업무 범위에 있어 중소 기업의 리파지토리 표준 정보 시스템 구축을 지향했음을 밝히는 바이다. 이는 SAP R/3와 같은 시스템을 도입시 부딪히는 고비용의 부담을 고려해 볼 때, 중소 기업형에 효율적으로 그 비용의 감소와 함께 각 기업에 맞는 적절한 모듈을 선택하여 시스템을 단기간에 구축할 수 있으며, 이를 통해 중소 기업이 자체 정보시스템을 개발하는 부담을 줄여 정보를 관리할 수 있는 효과성을 중시한 것이다. 또한 향후 본 논문에서 구현한 ERP용 리파지토리 정보 시스템의 연구 내용을 확대 응용하여 지식관리의 프레임워크인 지식 리파지토리 시스템으로 적용되어질 수 있다고 본다. 앞에서 작성된 리파지토리 스키마를 활용하여 리파지토리 기반 ERP 시스템을 구축하였다. ERP 시스템 구축을 위하여 운영시스템으로 클라이언트의 Windows 95와 서버의 Windows NT를 이용하였으며, 클라이언트와 서버의 분산 미들웨어로 양자가 기본적으로 지원하는 DCOM 을 활용하였다. 구현시스템의 구성도는 [그림 6]과 같으며, 각 시스템 구성요소별 이용제품 목록은 [표 6]과 같다.

클라이언트의 기능이 수행되기 위해서는 ERP 서버와 DCOM 을 이용한 통신이 요구되며, 그 과정은 다음과 같다. 클라이언트의 사용자는 어플리케이션 편집기, 데이터베이스 편집기, 워크플로우 편집기 등을 선택하여 작업을 수행하게 되면, 각 편집기는 DCOM 을 통하여 ERP 서버에 서비스를 요청하게 되고, ERP 서버는 데이터베이스 서버에 있는 리파지토리에 미들웨어인 Transaction Server 를 통하여 접근하여 필요한 메타 정보를 참조하여 요구처리를 수행하게 된다.

IV. 결 론

리파지토리를 이용한 시스템 개발의 장점은 특정 개발 환경과 개발 도구에 독립적이며 유연하게 대처하여 통합적인 개발 접근이 가능하다는 점이다. 본 논문은 ERP 로 대표되는 전사적 자원 관리를 한국 중소기업에 적용하기 위해, IRDS 표준을 따라 지식 자산 관리 프레임워크인 리파지토리의 강점을 활용하여 리파지토리를 설계하였다. 결과적으로, ERP 에서 리파지토리는 기업의 전략적인 차원에서 전사적 자원 계획을 통해 기업의 정보, 자원을 표준적인 체계 속에서 저장하게 될 자료 저장 구조 통합체로써 그 가치가 있다. 이 리파지토리 설계로 기대되는 효과는 기업 정보 및 지식의 구조화된 틀로서 기업 자산의 인프라를 제공할 수 있다는 것이다. 또한, 본 연구의 성과를 지식 관리의 측면에 적용시켜 기업에 필요한 지식관리 프레임워크를 보이고 이를 통한 지식 리파지토리를 구현하는 연구로 확장 시킬 수 있다.



[그림 6] ERP 구현 시스템 구성도

[표 6] 시스템 구성요소별 이용 제품 목록

아키텍처 구성요소	기 능	구성소프트웨어	비 고
클라이언트	개발지원	어플리케이션 편집기 워크플로우 편집기 데이베이스 편집기	C++ 사용
	업무지원	비즈니스 어플리케이션	VB 스크립트 사용
미들웨어	분산 컴포넌트 지원	DCOM	—
	트랜잭션 관리	MS Transaction Server 2.0	—
ERP 서버	리파지토리 관리 어플리케이션 서버	ERP 서버 모듈	C++ 사용
데이터베이스 서버	리파지토리 데이터베이스	Oracle Workgroup Server 7.3 MS Sql Server 6.5	—

향후 연구되어져야 할 과제들 중 중요한 것은 워크플로우 분야와 모델 분야에 관한 세부적인 연구이다. 워크플로우 분야는 비즈니스 환경변화에 조직 내 업무를 신속하게 처리하기 위한 비즈니스 프로세스 자동화를 위해 필요한 것이다. 이것은 워크플로우 시스템이 지향하는 바이며, 주요 연구 방향은 (i) 주어진 프로세스에서 조직도 등에 명시된 각 사용자의 역할, (ii) 비즈니스 규칙에 근거한 비즈니스 절차의 경로와 이의 제어, (iii) 비정형적 워크플로우 관리 등이다. 모델분야에 대한 연구는 분석, 설계 등 정보시스템 개발 시 모델의 생명주기 관리에 대한 정보를 리파지토리에 저장하고 이를 ERP 리파지토리 시스템에서 관리하기 위해 필요하다. IRDS 에서, 이 중 2 계층인 IRD 스키마 계층의 모델분야에 대한 리파지토리 스키마는 지원분야의 데이터베이스나 어플리케이션과 기본적인 연관관계를 위해서 스키마를 설계했으며, 본 모델은 많은 양의 추가적인 확장이 요구된다.

또한 IRDS의 IRD 정의 스키마 계층과 IRD 스키마 계층, IRD 계층과 연계한 모델 전체적인 관점에서의 연구가 필요하다. 또한 구현되어야 하는 분석 설계용 리파지토리는 개략적인 버전으로 테스트를 거쳐야 할 것이다.

주) 참고 문헌 및 Full Paper는 저자에게 요청시 송부함.