

# ERP와 워크플로우 시스템 : SAFE

백종명\*

박성진\*\*

## ◆ 목 차 ◆

1. 서론
2. ERP 시스템
3. 워크플로우 시스템

4. ERP와 WfMS의 비교
5. 시스템 예 : SAFE
6. 결론

## 1. 서론

오늘날 기업들은 치열한 경쟁속에서 세계화, 글로벌화(globalization)의 도전에 직면하고 있다. 경쟁적인 세계 시장에서 기업들은 보다 유연하고 신뢰할 수 있는 구조를 가진 더 좋은 비즈니스 솔루션을 찾고자 한다. 기업내 인프라와 조직의 대부분은 기업의 성장과 생존에 중요한 비즈니스 프로세스에 직,간접적으로 영향을 미치는 정보시스템에 의해 지원된다. 비즈니스 프로세스의 효율적 관리를 통해 비용 절감, 생산성 향상 그리고, 고객 서비스 개선 등의 효과를 가져올 수 있다.

세계 시장은 끊임없는 비즈니스 프로세스의 개선을 피할 수 없도록 하고 있다. 비즈니스 프로세스가 기업의 성공을 위해 필수적이기 때문에 프로세스 관리와 개선에 초점을 맞춘 정보 기술은 기업의 비전을 충족하고 경쟁력을 향상시키는 훌륭한 대안이 되어 왔다. 지난 20여 년 동안 비즈니스 프로세스를 개선하기 위한 두개의 명확한 솔루션이 제시되었는데 하나는 ERP(Enterprise Resource Planning) 시스템이며 또다른 하나는 바로 워크플로우 시스템(WfMS; Workflow Management System)이다.

WfMS와 ERP 시스템은 모두 비즈니스 프로세스

를 관리하며 기업 내에서 상호간에 밀접한 연관성을 갖는다. 이 두 시스템을 통한 프로세스 정의와 관리 는 부서 단위의 데이터와 애플리케이션에 관한 통합을 가능하게 한다.

WfMS와 ERP 시스템이 모두 비즈니스 프로세스에 관련되지만 접근 방식에는 분명한 차이가 있다. 먼저, WfMS는 특정 비즈니스 프로세스 구조를 적용하도록 커스터마이징하는 워크플로우 모델 그림 1을 제공한다.

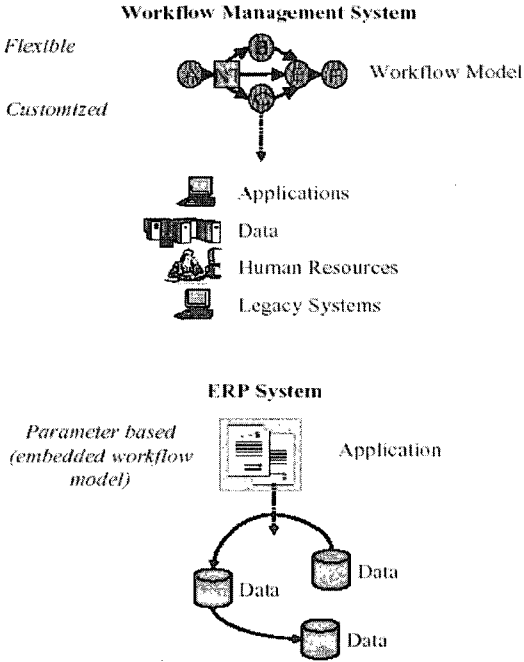
워크플로우는 기업 내에 이미 존재하는 비즈니스 프로세스에 기반하여 설계된다. 일단 모델 설계 과정이 완료되면 주어진 워크플로우안의 실제 단계들을 수행하도록 인스턴스들이 생성된다. 실행 과정동안 워크플로우 인스턴스들은 기존 시스템과 데이터베이스, 애플리케이션에 접근가능하고 사용자들과 상호작용할 수 있다.

워크플로우 시스템은 흐름 독립적(flow-independent)이다. 태스크 사이의 제어 흐름과 데이터 흐름은 워크플로우 설계 단계에서 그래픽결하게 표현된다. 이것은 애플리케이션이 하부의 워크플로우 시스템에 대해 독립적이어도록 한다. 워크플로우 애플리케이션은 워크플로우 시스템이 어떻게 태스크 실행을 스케줄링하고, 실행시에 데이터 흐름이 어떻게 되는지를 기술하는 독립적인 명세의 집합을 포함한다.

한편, ERP 시스템은 조립식 애플리케이션의 개념으로 구현된다. 그림 1 벤더들이 특정 업종을 위한

\* (주)유비넷 대표이사

\*\* 한신대학교 정보시스템공학과



(그림 1) WfMS와 ERP 시스템

애플리케이션을 개발하고 기업들은 필요에 의해 애플리케이션을 구매하며 이 안에 워크플로우 모델이 포함되어 있다. 애플리케이션들은 비즈니스 프로세스를 구성하는데 유연성을 허용하기 위해 매개변수화 되어 있다. 각 애플리케이션은 데이터와 정보 관점에서 기업 프로세스 제어에 필요한 로직을 포함한다. 기업 환경에 보다 적합하게 하기 위해서 애플리케이션은 종종 수천 개의 매개변수를 설정하기도 한다.

그림 1은 WfMS와 ERP 시스템간의 핵심적인 차이점을 보여준다. WfMS는 애플리케이션을 개발하기 위해 시스템이 직접 구현해야하는 작업들의 워크플로우/프로세스에 대한 정의와 모델을 갖는다. 시스템은 애플리케이션에 의해 접근되는 매개변수들을 설정함에 의해 작업을 구현한다. ERP가 갖는 매개변수가 많으면 많을수록 워크플로우 프로세스를 다양한 형태로 구성할 수 있는 유연성이 커진다. ERP 안의 워크플로우 모델은 애플리케이션이나 매개변수 테이블 안에 포함되기 때문에 밖으로 드러나지 않는다. 물론, WfMS의 경우, 개발자들은 WfMS 개발 플랫폼상의 도구들을 통해 워크플로우 모델을 직접 개발하기 때

문에 더욱 많은 유연성을 갖는다. ERP 시스템은 기업 전체에 걸쳐 동종의 데이터 인프라를 제공하는 데이터 중심(data-centric)이다.

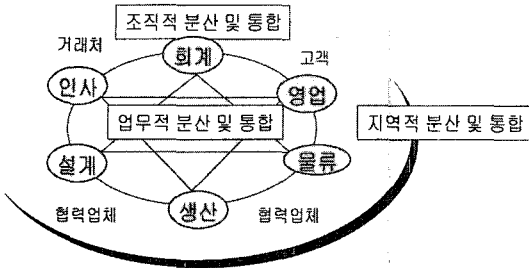
이 둘의 차이를 프로그래밍 언어와 비교해 볼 경우, WfMS를 가지고 작업하는 것은 개발자가 직접 비즈니스 워크플로우 모델의 그래픽 표현을 가지고 작업할 수 있는 비절차적 고수준 언어를 사용해서 프로그래밍하는 것과 유사하다. 그때, WfMS 개발 플랫폼은 하나의 워크플로우 프로세스를 실행하기 위하여 필요한 애플리케이션 컴포넌트, 데이터베이스, 연결부 등을 생성한다. ERP 시스템으로 작업하는 것은 애플리케이션과 데이터를 직접 다루는 절차적 3세대 언어를 가지고 작업하는 것과 같다.

WfMS 혹은 ERP 시스템의 적절한 선택은 기업에게 있어 매우 중요하다. 이 논문에서는 WfMS와 ERP 시스템의 특성과 두 시스템간의 유사점과 차이점을 보이며 그 사례로 SAFE 패키지 시스템을 소개한다. 두 시스템에 대한 비교는 이들 시스템의 구입과 통합에 관련해서 보다 정확한 결정을 내려야하는 시스템 구현자, 관리자, 기업들에게 도움이 될 수 있다.

## 2. ERP 시스템

컴퓨터와 정보기술의 발전은 기업 환경을 급속하게 변화시키고 있다. 끊임없이 변화하는 경영환경 속에서 기업이 경쟁력을 유지하기 위해서는 비즈니스 환경을 사전에 예측하고 이에 신속하게 대응할 수 있는 능력을 요구한다. 이를 위해 기업은 기존 단위 업무 및 부서별로 적용하던 정보시스템 구축 차원을 벗어나 모든 비즈니스 업무 영역을 통합적으로 지원할 수 있는 전사적 정보시스템인 ERP(Enterprise Resource Planning)를 구축하고 있다.

ERP 시스템은 그림 2에서 보는 바와 같이 기업의 생산, 자재, 영업, 인사, 회계 등의 업무를 통합, 최적으로 관리하는 경영관리용 패키지 소프트웨어로서 그 핵심은 경영 정보의 통합과 정보 흐름의 자동화이다. ERP 구축을 통해 기업은 치열한 글로벌 경쟁체제 하에서 정보 우위를 통한 경쟁력을 향상시킬 수 있으며 급변하는 새로운 경영환경에도 유연하고 신속하게 대처할 수 있다.



(그림 2) ERP 시스템 개요

표 1은 ERP 시스템이 갖는 다양한 기능들에 대한 구체적 특성들을 보여준다.

(표 1) ERP 시스템의 기능적 특성

기능적 특성	내용
업무 통합	- 기업 내의 생산, 물류, 재무, 회계, 영업, 구매 및 재고 등 기간업무 프로세스들을 통합적으로 연계 관리하도록 구성
표준 비즈니스 프로세스 모델	- BEST Practice에 의해 제공된 비즈니스 프로세스 모델을 제공 - BEST Practice는 세계에서 우수 기업이 채용하고 있는 프로세스 중에서 가장 효율적이고 체계화된 글로벌화된 비즈니스 프로세스
통합 데이터베이스 구축	- 데이터의 생성이 한번 발생하면, 같은 데이터를 다른 기능에서 발생시키지 않고 사용하는 것으로 데이터의 무결성을 보장
다양한 산업 및 비즈니스 형태 지원	- ERP는 다양한 산업형태를 지원하여야 하므로, 산업의 형태에 따라서 주문생산, 계획 생산, 반복 생산, 연속 생산 등 다양한 생산형태를 지원 - ERP를 도입하고자 하는 기업은 이러한 생산 형태 중에서 자사의 환경에 맞는 생산형태를 선택

한편, 클라이언트/서버, 분산객체 컴퓨팅, 인터넷/웹 등 계속되는 소프트웨어 신기술 등장은 ERP 소프트웨어에 대해서도 지속적으로 새로운 컴퓨팅 모델을 요구하고 있다. 그러나, 제한된 비용, 개발 인력과 기술력으로는 효율적인 기업 내 ERP 구축 작업도 쉽지 않은 상황이며 지속적인 ERP 소프트웨어의 보완과 확장을 통한 최신의 컴퓨팅 환경을 수용하는 것은 현실적으로 많은 어려움이 존재한다. 최근 적극 추진되

는 아웃소싱(outsourcing) 방안 또한 다양한 업종과 패턴을 갖는 기업 요구사항에 대한 맞춤형대의 구축 방법으로는 지속적인 유지보수 및 확장 등에 많은 어려움이 있다. 또, 고도화되고 복잡해져만 가는 소프트웨어 기술 환경은 서비스 공급자나 개발자들에게도 사용자의 요구 시기에 맞춰 최신의 애플리케이션을 개발하여야하는 어려움을 주고 있다.

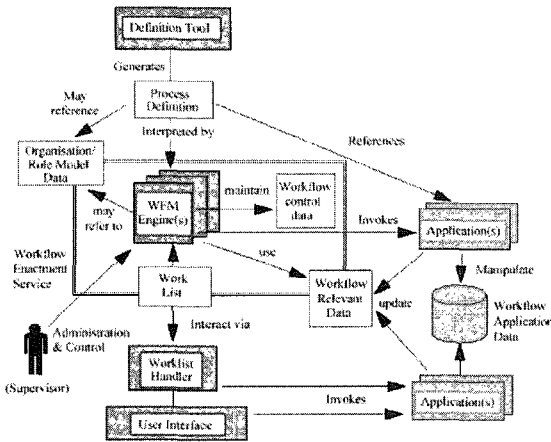
이에 대한 최선의 방안이 바로 ERP 패키지 시스템의 활용이다. ERP 패키지를 사용함으로써 기업은 ERP 초기 구축시간과 비용을 줄일 수 있을 뿐만 아니라 소수의 개발 인력만으로 지속적인 유지관리 및 확장이 가능하다는 이점이 있다. 즉, 사용자는 ERP 시스템 패키지 중에서 자신의 기업 환경에 적합한 모듈만 부분적으로 선택하여 시스템을 유연하게 구축하거나 기존 시스템을 확장할 수 있다. 또한, 최신의 소프트웨어 기술을 반영한 패키지 시스템을 지속적으로 공급받음으로써 최신 컴퓨팅 환경에 기반한 정보시스템을 사용할 수 있다. 이러한 시스템 확장성, 통합성 및 유연성 등의 ERP 패키지 시스템 이점으로 인해 기업은 정보시스템을 자체 개발해야 하는 부담을 줄일 수 있으며 손쉽게 ERP를 사내 정보 인프라로 구축, 활용할 수 있다.

### 3. 워크플로우 시스템

기업 내에는 여러 가지의 비즈니스 프로세스가 존재하고 이러한 업무 프로세스들은 특정한 작업 흐름을 갖는 단위 활동들로 구성된다. 작업 흐름 즉, 워크플로우(workflow)는 컴퓨터에 의해 자동으로 실행 및 관리되는 비즈니스 프로세스를 말하며, 워크플로우 관리 시스템은 기업의 워크플로우를 정의하고 그에 따른 비즈니스 프로세스를 생성 및 실행, 관리해주는 소프트웨어 시스템이다[1].

WfMS는 업무 흐름의 자동화, 정보 및 문서 전달의 전자화 그리고 일관적 데이터 접근 및 제어를 통해 비즈니스 프로세스의 개선, 통제, 관리, 공동 작업을 지원하는 소프트웨어로서 비즈니스 프로세스의 개선 측면에서 BPR의 한 가지 방법으로 생각되기도 한다.

그림 3은 일반적인 워크플로우 시스템에 대한 구



(그림 3) WfMC 워크플로우 시스템 구성도

성도를 보여준다.

WfMS는 비즈니스 프로세스의 자동화를 의미하며 이때 문서, 정보, 업무 등이 특정 사용자에서 다른 사용자에게로 사전에 정의된 규칙에 의하여 전달된다. 이러한 WfMS의 적용 대상은 기업 전체이므로 기업 내의 모든 업무와 자원을 프로세스 관점에서 통합하여 관리하게 된다. WfMS는 기업의 업무 흐름을 자동으로 처리함으로써 이에 따른 비용과 시간을 절약할 수 있는 솔루션으로 외국에서는 제조, 금융, 의료 등의 다양한 분야에서 폭넓게 도입되어 이에 관한 연구 및 개발이 활발하게 이루어지고 있다.[2-4]

워크플로우 시스템의 구조는 크게 비즈니스 프로세스를 모델링하는 모듈과 모델링된 비즈니스 프로세스를 알맞은 자료구조로 변환하여 저장하는 모듈, 그리고 저장된 프로세스 모델에 따라 인스턴스를 생성하여 실제 워크플로우를 실행하는 모듈과 클라이언트 모듈 등으로 구성된다.

비즈니스 프로세스를 정의하는 모델링 도구에서는 각각의 논리적인 단위 업무(activity)를 정의하고 각 단위 업무에 프로세스 참여자(participant)나 역할(role)을 할당한다. 또, 단위 업무간의 전이 정보를 주어 조건 분기(transition)가 가능한 흐름을 결정하고 각 단위 업무가 수행하는 애플리케이션(invoked application)과 연관 데이터(relevant data)를 정의한다. 워크플로우 엔진은 이러한 프로세스의 정의를 데이터베이스에 저장하고 런타임시에 워크플로우의 비

즈니스 프로세스와 단위 업무를 실제로 인스턴스화하여 프로세스의 진행 및 통제를 담당한다. 일반 업무 담당자가 접하게 되는 클라이언트 프로그램은 워크플로우 엔진이 생성한 프로세스 및 단위 업무의 인스턴스에서 사용자에게 해당되는 정보만을 취합하여 작업 목록(worklist)으로 전달하고 사용자는 적정한 작업을 선택하여 해당 업무용 프로그램을 기동하고 업무를 진행한다.

## 4. ERP와 WfMS의 비교

### 4.1 개요

WfMS는 기업들이 경쟁력, 고객 서비스, 생산성, 표준 적합성 등을 개선하기 위한 핵심 도구이다. 워크플로우 시스템은 워크플로우(비즈니스 프로세스)와 연관된 다양한 액티비티의 정의, 생성, 관리를 위한 애플리케이션과 도구들의 집합으로 구성된다. 워크플로우 인스턴스는 워크플로우 정의를 해석하고 워크플로우 참여자(participant)와 상호작용하며 필요할 경우 외부 도구와 애플리케이션 사용을 호출할 수 있다.

비즈니스 기능을 자동화하기 위해 등장한 ERP 시스템은 기업 인프라에 걸쳐 하나의 통합된 데이터 솔루션을 제공한다. 기존 시스템의 경우, 각 부서의 요구를 만족시키기 위해 개별적 애플리케이션을 개발하여 왔다. 이로 인해 기존 시스템은 일관성이 보장되지 않는 정보의 섬으로 발전했고 결국 시스템의 통합이 불가능하게 되었다. 그에 대한 해결책으로 ERP 시스템은 기업 전체에 걸쳐 부서들의 정보와 서비스를 통합하고 관리하기 위한 기능을 지원한다. 이것은 기업들의 자원을 보다 잘 관리하고 다양한 비즈니스 프로세스간의 정보 통합을 통해 비용 절감과 효율성을 높이도록 한다.

WfMS와 ERP 두 시스템의 개념적 차이는 표 2와 같다. 표 2는 도메인 영역, 기술 영역, 시스템 구현 세 측면으로 나누어 그 차이를 보여준다.

### 4.2 도메인 영역 측면

도메인 영역(domain scope)은 특정한 유형의 애

(표 2) WfMS와 ERP 시스템 비교

비교 측면	WfMS	ERP 시스템
도메인 영역	커스터마이징된 프로세스 도메인 독립적 즉흥적, 동적 도메인 국제적 설정 부재	내재된 프로세스 도메인 종속적(참조 모델의 사용)정적 도메인 국제적 설정 존재
기술 영역	프로세스 중심 사람,애플리케이션,처리 워크플로우를 포함한 워크플로우 자원 이질적이고 자동화된 환경	데이터 중심처리 프로세스 지원 공통 데이터 인 프라를 가진 동종 환경
시스템 구현	완성된 시스템으로 구입: 자동 생성된 코드 상향식 접근방식 데이터 변환이 요구되기도 함	미리 작성된 규격화된 컴포넌트 기반 하향식 접근 방식 데이터 변환이 요구됨

플리케이션이나 기업에 대한 시스템의 적합성을 의미한다. 기업마다 다양한 요구사항과 특성을 갖기 때문에 이러한 특성은 중요하다.

워크플로우 시스템은 폭넓은 기업 유형에 걸쳐 성공적으로 적용된다. 워크플로우 시스템은 워크플로우 프로세스 모델로부터 시스템 컴포넌트(애플리케이션과 데이터)사이의 흐름을 분리시켜 애플리케이션이 흐름-독립적(flow-independent)이 되도록 한다. 태스크와 액티비티 사이의 흐름의 분리는 워크플로우 기술이 수많은 도메인에 적합하도록 한다. WfMS는 다양한 종류의 데이터와 애플리케이션을 넓은 영역의 문맥에서 통합하기 위해 사용되는 도구이지만, 세계 시장에 배포하는데 있어 중요한 국제화, 다중-화폐, 다중-언어와 같은 필수적인 특성들이 아직 부족하다.

반면에, ERP 시스템은 최상의 비즈니스 사례를 구체화한 참조 모델(reference model) 혹은 프로세스 템플릿(template)을 적용하기 때문에 도메인-종속적(domain-dependent)이다. ERP 시스템은 다양한 업종을 위해 미리 정의된 프로세스 라이브러리로 구성된다. 참조 모델은 조직의 구조뿐만 아니라 하부의 데이터와 프로세스 모델을 포함한 표준화된 선진 비즈니스 모델을 반영한다.

ERP 다양한 산업별 최적 솔루션을 개발해왔고 특정 기업의 특성에 애플리케이션을 맞추기 위하여 수

천 개의 매개변수를 설정할 수 있도록 한다. 그러나, ERP 시스템과 기업 프로세스 사이의 불일치는 종종 상당량의 코드 구현과 모델의 수정을 요구한다.

즉흥적이고 이질적인 비즈니스 프로세스는 미리 정의된 참조 모델에 의존하지 않는 WfMS를 사용해서 더욱 잘 관리될 수 있다. 반면에, ERP 시스템은 다중-언어 지원, 다중-화폐 지원과 같은 특성을 지원하기 때문에 국제적 도메인에 더 적합하다.

#### 4.3 기술 영역 측면

두 시스템은 구조적 측면에서는 유사하게 모두 비즈니스 프로세스를 관리하지만 프로세스는 본질적으로 매우 다른 구조와 요구사항을 포함한다.

WfMS와 ERP는 명확히 구별되는 기술적 특성을 가지고 개발되며 WfMS는 ERP 시스템보다 더욱 넓은 기술 영역을 보여준다. WfMS의 목표는 인간 업무의 적절한 협력과 독립적, 이질적 때로는 분산되기 까지의 정보시스템, 데이터, 애플리케이션간의 통합이다. 특히, 워크플로우 기술은 분리되어 있던 통신, 정보, 데이터 흐름을 특정 시점에 하나의 작업 프로세스 안으로의 통합을 가능하게 한다. 동시에 워크플로우 기술은 분리되어 있던 정보시스템들의 통합을 지원하는 하나의 환경(environment)이기도 하다. WfMS 기술은 다음 세 가지 종류의 워크플로우 애플리케이션들을 지원한다[5].

- type1 : 인간이 연관된 작업 흐름
- type2 : 시스템과 애플리케이션이 연관된 작업 흐름
- type3 : 트랜잭션 작업 흐름

type1 작업 흐름의 경우, 업무 수행을 위해 사람의 개입(협력)을 포함하며 WfMS는 사람의 작업을 제어하고 조화시키며 사용자들 사이의 문서와 워크플로우 데이터의 일관성을 유지하는데 책임을 져야한다.

type2 작업 흐름의 경우, 거의 사람의 개입없이 업무 소프트웨어의 수행과 제어, 협력에 책임을 진다.

type3 작업 흐름의 경우, 트랜잭션에 기반한 사람

의 개입과 시스템이 포함된다. 이것은 사람의 협력 수행이 포함될 수도 있고 개별적 업무 혹은 전체 작업 흐름을 위한 트랜잭션 특성(원자성, 일관성, 고립성, 지속성)의 선택적 사용을 지원한다.

ERP 개념은 데이터 인프라가 기업에 걸쳐 동질적 임을 즉, 데이터가 상호운영 가능하거나 극단적인 경우 동일 벤더에 의한 데이터베이스들에 저장됨을 가정한다. 특정 ERP 시스템은 특정한 데이터베이스 시스템만을 지원하기도 한다. 이러한 가정은 기업으로 하여금 기존 시스템으로부터 표준화된 환경으로 이동하도록 강요한다. ERP 시스템은 구현과정에서 상호운영 가능한 데이터베이스들간의 데이터 통합만이 고려되는 데이터-중심(data-centric)의 특성을 갖기 때문에 트랜잭션 프로세스를 모델링하는데 적합하다.

한편, WfMS는 프로세스 중심(process-centric)이고 따라서 type1, type2, type3 3가지 유형에 모두 관련되는 반면 ERP 시스템은 type3만이 관련된다. WfMS는 인간과 소프트웨어 시스템(type1, type2)이 연관된 작업 흐름들을 모델링하는데 적합하고 특히, 시스템이 자동화되어있고 이질적인 경우에 더욱 그렇다. 반면에 ERP 시스템은 세 번째 트랜잭션 작업 흐름을 모델링하는데 더욱 적합하며 이것은 ERP 시스템이 잘 정의된 트랜잭션 모델을 지원하는 데이터베이스 인프라상에서 직접 작동하기 때문이다. 그러나, 트랜잭션 작업 흐름이 이질적인 시스템들을 포함할 경우 더욱 적절한 해결책은 WfMS를 적용하는 것이다. 이질적인 인프라를 갖고 있는 소규모 기업의 경우 시스템 통합을 위해 WfMS를 적용하는 것이 최선의 해결책이 될 수 있으며 이것은 ERP 구현과 관련한 시간과 비용 투자를 요구하지 않기 때문이다.

#### 4.4 시스템 구현 측면

흔한 질문중의 하나는 “WfMS의 구현과 ERP 시스템의 구현은 어떻게 다른가?”하는 것이다. 세 번째 분석 측면은 구현 단계과정에서 사용되는 전략과 방법론에 따른 시스템의 특성이다. 두 시스템 모두 소프트웨어 구현 측면에서는 생산성을 크게 향상시키고

하나의 비즈니스 프로세스를 실행하거나 지원하기 위해 사용될 수 있다.

WfMS와 ERP 시스템은 다음 3가지 특성에서 구별된다.

##### 4.4.1 미리작성된 코드 & 자동 생성된 코드

비즈니스 정보시스템은 맞춤형 애플리케이션을 직접 개발하거나 표준화된 완성형(off-the-shelf) 솔루션을 구입함에 의해 구축될 수 있다. 맞춤형 애플리케이션을 개발하는 방법은 일반적으로 많은 비용과 결과에 대한 불확실성 등으로 두 번째 방법이 일반적으로 선호된다.

ERP 시스템은 미리 작성된(prewritten) 기성품 형식의 소프트웨어 모듈들로 구성된다. 이것은 수천 개의 매개변수를 설정함에 의해 대부분의 기업 요구를 충족시키기 위한 적정 수준의 유연성을 제공한다.

반면에 WfMS는 모듈-중심(module-oriented)이 아니므로 특정 모듈을 구입하고 구축할 필요가 없다. 일단 워크플로우가 설계되면 애플리케이션의 전개는 별다른 프로그래밍 없이도 수행된다. 즉, 시스템은 각 애플리케이션을 위해 필요한 코드들 자동적으로 생성한다[6]. ERP의 경우처럼 하드코딩 또는 부분적으로 하드코딩된 애플리케이션 형태로 프로세스가 표현될 경우 유연성이 본질적으로 떨어지므로 이 특성은 특히 중요하다. ERP 시스템안의 유연성은 오직 설치될 때 매개변수로부터만 가능하다. WfMS는 비주얼한 도구를 통해 프로세스를 모델링할 수 있고, 그 뒤 소프트웨어의 행위를 설계할 수 있는 책임을 워크플로우 시스템에 위임할 수 있다. 자동 코드 생성은 기존 애플리케이션과 시스템의 통합에 매우 적합한 기술적 특성이다.

##### 4.4.2 상향식 전략 & 하향식 전략

WfMS 구현은 상향식으로 이루어진다. 첫 번째 단계는 기존 비즈니스 프로세스(로직, 제어흐름, 데이터 흐름)를 식별하는 것이고 두 번째 단계는 필요할 경우 기업으로 하여금 프로세스를 재공략하도록 하는 것이다. 상향식 전략은 기업들이 WfMS에 맞추는 것이 아니라 WfMS들을 그들의 비즈니스 프로세스

에 맞추도록 한다. 기존 비즈니스 프로세스는 워크플로우 모델 구축과 최종 구조를 유도한다.

ERP 구현은 하향식으로 이루어지는데 첫 번째 단계는 특정 부서를 위한 ERP 패키지(혹은 애플리케이션)를 구입하는 것이다. 패키지에는 필요로 하는 업무의 핵심 비즈니스 로직이 포함되어 있다. 두 번째 단계는 기업의 특성과 요구에 애플리케이션을 맞추기 위하여 매개변수를 설정하는 것이다. 구매한 애플리케이션은 매개변수 설정에 의해 기존 프로세스를 대신할 수 있는 새로운 비즈니스 프로세스를 생성한다. 불행히도 대부분의 경우, 새로운 프로세스가 기존 프로세스에 정확히 일치하지 않으며 결과적으로 기업에 따라 많은 수정이 추가적으로 요구된다. 하향식 전략은 기업으로 하여금 외부 애플리케이션과 정책을 따르도록 강요하게 되며 이러한 이유로 수정 부분에 대한 관리가 필수적이다.

#### 4.4.3 데이터 변환(conversion)

두 시스템은 모두 데이터 통합을 위한 해결책으로서 시스템 구현 시에 일정 정도의 데이터 조작과 변환이 발생하게 된다. WfMS의 경우, 통일되고 상호운용 가능한 데이터 인프라나 데이터베이스 시스템을 요구하지 않으며 따라서, 데이터 변환은 필수적이지 않다. 그럼에도 불구하고 최적화나 기업의 필요성에 의해 데이터가 변환될 수 있다. 반면에 ERP 시스템은 빈번하게 데이터 변환을 요구한다. ERP 플랫폼은 데이터 저장을 위한 구조를 한정하기 때문에 결과적으로 기존 데이터베이스는 ERP-호환 가능한 데이터베이스로 대체되어야 한다.

### 4.5 적용 방법

대부분의 WfMS는 4.2절의 3종류의 작업흐름 모두와 관련되는 반면에 ERP 시스템은 주로 type3의 트랜잭션 작업 흐름과 관련된다. 특히, 시스템이 자동화되고 이질화되어 있다면 WfMS는 인간과 소프트웨어 시스템(type1, type2)이 연관된 작업 흐름을 설계하는데 더욱 적합하다. 반면에, ERP 시스템은 트랜잭션 작업흐름의 설계에 더욱 적합하다. 그럼에도 불구하고

도 불구하고 거래 작업흐름이 이질적 시스템을 포함하고 있는 경우에는 WfMS의 선택이 보다 적절한 해결책이 될 수 있다. 이것은 ERP 시스템이 공동된 동종의 상호운용 가능한 인프라에 의존하고 있기 때문이다.

따라서, 기업들은 ERP 시스템 혹은 WfMS 중에서 하나를 혹은 둘 모두를 필요로 할 수 있다.

워크플로우 관리 시스템은 이질적이고 자동적이며 분산된 시스템의 애플리케이션과 데이터 통합을 포함하는 프로세스 관리를 위해 보다 적합하다. 대부분의 시스템은 어떤 비즈니스 영역에서도 구현될 수 있다는 의미에서 도메인 독립성(domain independence)을 보여준다. 이런 종류의 시스템은 특히 한 부서 단위 혹은 동적 작업 그룹 단위에 적합하며 그 구현은 일반적으로 상향식으로 수행된다.

ERP 시스템은 데이터-중심적(data-centric)이며 정보 관리와 데이터 통합에 더욱 집중한다. 이 시스템은 도메인 의존적(domain-dependent)으로 특정 분야와 시장에서 사용되어지는 비즈니스 템플릿(template)을 제공한다. ERP 시스템은 국내 혹은 국제적으로 이루어지는 부서내, 기업내, 기업간 업무에 관해 ERP 애플리케이션 안에 포함된 프로세스와 기업이 요구하는 프로세스가 잘 일치하는 경우에 매우 적합하다. 구현은 보통 블록화 된 애플리케이션으로부터 조립하는 하향식 전략으로 이루어진다.

ERP 시스템을 가지고 개발하는 개발자의 주요 문제 중의 하나는 애플리케이션 안에 포함된 프로세스 모델을 이해하는 것이다. 이 문제를 해결하기 위해 최근에는 워크플로우 컴포넌트를 기존 ERP 시스템 안에 포함시키는 경향이 있다. 몇몇 ERP 벤더들은 ERP 시스템과 통합되지 않은 독립적인 WfMS 제품들을 제공하기도 한다[7].

### 4.6 통합 방법

현재 또는 미래에 기업내(enterprise) 혹은 기업간(e-Commerce) 애플리케이션 통합을 위해 어떻게 WfMS와 ERP 시스템이 독립적으로 혹은 함께 사용될 수 있는지 기술한다.

4.6.1 기업내 통합(enterprise integration)

워크플로우 시스템은 type 2에서 기술된 것처럼 기존 애플리케이션[8]과 ERP 애플리케이션[9]들을 통제하는 일종의 통합 “미들웨어” 플랫폼으로 작용할 수 있다. 실무 측면에서는 이런 종류의 워크플로우 애플리케이션을 개발하기 위한 시스템을 EAI(Enterprise Application Integration) 도구로 부르기도 한다.[10]

한편, ERP 벤더들은 모든 기업들의 요구를 충족하는 통합된 전체 애플리케이션들의 집합을 제공하려고 노력하고 있다. 비록 기업들이 오직 한 벤더 제품만을 사용함으로써 이러한 이점을 얻을 수 있지만 실제로는 ERP 벤더들이 모든 분야의 요구를 충족할 수 없기 때문에 기업들은 보통 여러 벤더들의 ERP 컴포넌트를 구매하고 통합한다. 이 경우 EAI와 같은 워크플로우 시스템은 기업내 통합을 위한 중요한 역할을 제공한다.

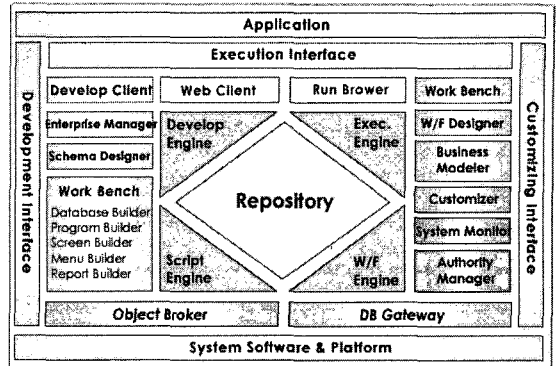
4.6.2 기업간 통합(e-Commerce)

공급망(supply chain)을 보다 잘 관리하기 위해 B2B를 구축하는 회사가 많으면 많을수록 상호운영성(interoperability)이 기업간 통합의 핵심 이슈가 되고 있다. 기존 ERP 벤더들이 공급망 관리 분야로 진출하고 있지만 기업들이 고정된 ERP 비즈니스 모델에 맞추어야만 하는 상황에서 동일한 벤더의 동일한 구조가 아닌 경우에는 통합에 많은 어려움이 존재한다.

그러나, 워크플로우 시스템을 사용하여 통합하는 경우에는 훨씬 간단하여 기업내 애플리케이션을 통합할 경우와 유사한 방식으로 둘 이상의 기업간 브릿지 역할을 할 수 있다. 또한, e-Commerce 워크플로우를 위해 특별히 설계된 워크플로우 시스템의 개발도 이루어지고 있다.

5. 시스템 예 : SAFE

비즈니스 프로세스의 통합을 위한 ERP 시스템과 워크플로우 시스템의 구현 예로써 다음 그림 4와 같은 프레임워크를 갖는 (주)유비넷에서 개발한 SAFE 솔루션을 소개한다.



(그림 4) SAFE 솔루션 프레임워크

5.1 SAFE-ERP

SAFE-ERP는 Standard Application For Enterprise ERP의 약자로서 차세대 e-Business를 지향하는 기업의 업무 처리를 위하여 자사 제품인 “SAFE Plus”로 개발된 컴포넌트 기반의 전자적 자원관리(ERP) 애플리케이션으로 기업의 영업, 물류, 생산, 자재, 회계등 기간 업무 프로세스를 통합적으로 연계 관리하여 주며, 기업 간 발생하는 정보들을 서로 공유하여 새로운 정보 생성 및 빠른 의사 결정을 도와 주는 기업통합 정보 시스템이다.

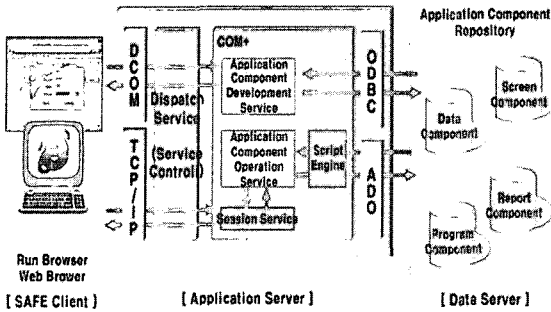
SAFE-ERP는 컴포넌트기반 ERP 패키지 시스템으로 기업의 표준 업무 처리를 위한 비즈니스 애플리케이션 컴포넌트 개념을 지원하고 독자적인 개발툴과 실행브라우저 등의 다양한 툴셋을 통한 애플리케이션 개발과 실행이 가능한 ERP 시스템으로 COM+ 분산 객체 미들웨어에 기반하고 있다.

특히, 다양한 업종 및 업무와 관련한 표준적인 비즈니스 컴포넌트들을 제공함으로써 커스터마이징을 통해 특정 기업에 맞는 ERP를 효과적으로 구축할 수 있는 솔루션을 제공한다.

SAFE-ERP의 전체적인 시스템 구성도는 그림 5와 같다.

그림 5에서 보는 것처럼 SAFE-ERP는 컴포넌트 리파지토리를 토대로 하여 다양한 개발, 실행 및 커스터마이징 툴들로 구성된 통합 개발 및 실행 환경을 제공한다.





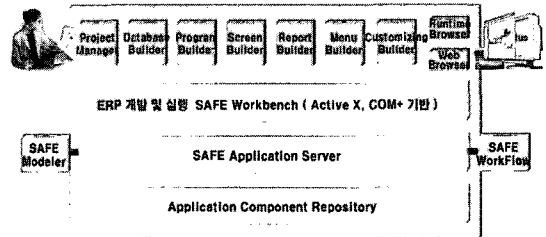
(그림 5) SAFE ERP 시스템 구성도

SAFE-ERP 시스템의 특징은 다음과 같다.

- e-Biz 업무 환경에 적합한 Web 지원 ERP 솔루션 : 기업전반의 업무 효율 증대와 비용 절감을 극대화 할 수 있는 Web 환경 지원의 차세대 ERP이며 사용이 쉽고 적용기간이 짧으며, 사용자 편리성을 최대한 구현한 중소, 중견 기업용 ERP 솔루션이다.
- 통합 및 Multi-Location 지원 : 생산/판매/회계 업무를 통합함으로써 정보의 단절 없이 복수 사업장을 일원화 관리가 가능하며 CCN(Control Center Number) 단위로 복수 사업부/사업장 관리가 가능하다.
- Globalization 대응 가능 구조 : 복수언어(영어, 일어, 중국어...)지원을 통하여 다 국적 기업의 업무 환경에 효율적 대응하고 다중 사업장 및 통화, 환율 관리를 통해 다국적 기업, 해외법인 등의 Global 경영 환경을 제공한다.
- 용이한 도입과 확장성 : 템플릿 기반의 통합 업무 모델을 활용하여 단기간의 시스템 구축이 가능하고 컴포넌트 기반의 3-Tier 구조로 기업 규모 및 업무 특성에 맞는 시스템 확장이 용이하다.

한편, SAFE-ERP는 SAFE Plus에 의해 개발된다. SAFE Plus는 최적의 ERP 시스템을 구축하기 위한 컴포넌트 기반의 ERP 시스템 저작 및 운영 아키텍처를 말한다. 리퍼지토리를 토대로 다양한 시스템 설계, 개발, 커스터마이징, 운영 및 유지 보수를 효과적으로 할 수 있는 통합된 개발 환경을 제공하며,

ERP 템플릿의 기반이 되는 컴포넌트의 생성, 소멸에서부터 트랜잭션 처리 기능까지 컴포넌트 운영에 있어서 최상의 성능과 기능을 제공한다.



(그림 6) SAFE Plus 구조

그림 6과 같은 SAFE Plus의 상세한 구성 내역은 다음과 같다. 먼저, 서버 툴이 제공하는 4가지 서비스는 다음과 같다.

- SAFE Development Service(Design Server) : 개발 툴로부터 편집된 컴포넌트 데이터를 리퍼지토리에 저장하는 서비스
- SAFE Dispatch Service : 부하 분산과 서버 모니터링을 위한 서비스 기능
- SAFE Session Service : 사용자의 접속과 Data 처리 정보를 유지, 관리하는 서비스 기능
- SAFE Operation Service(RunServer Service) : Data 처리 서버 기능

한편, 클라이언트 툴은 다음 5가지로 구성된다.

- SAFE Enterprise Manager : 전체 프로젝트의 컴포넌트를 관리 하는 관리도구
- SAFE Workbench : Screen Builder, Program Builder(Server Proc), Report Builder, Menu Builder, Webform Builder 등의 개발도구 집합
- SAFE Runtime : ActiveX Control로 작성된 실행 도구
- SAFE Connector : Session Service와 연결된 Client 연결 관리자
- SAFE Skin : SAFE Runtime의 외부 UI, SAFE Runtime이 ActiveX이므로 이를 실행할 수 있는 Web Browser도 SAFE Skin과 통신

(표 3) 프로세스 노드 정의 노테이션

구분	의미	비고
Activity	동작을 연결하는 노드	이 노드에 연결된 Data Object Method를 실행하고 그 결과를 프로세스 인스턴스 컨테이너에 받는다.
Condition	True/False 조건을 결정하는 노드	비교 구문을 기술하고, 분기점에 해당하는 Event에 따라 플로우가 흘러간다.
MultiCondition	여러 조건 중 한 가지 조건을 결정하는 노드	비교구문을 기술하고 분기점에 해당하는 Event에 따라 플로우가 흘러간다.
EventCreator	Global Event를 발생시키는 노드	다른 프로세스 인스턴스를 만들기 위해서 Global Event를 발생시킨다.
Until	조건 검사를 해서, 다시 실행할 지를 결정하는 노테이션	Until 블록에 포함되는 프로세스를 한번 수행 후에 조건 검사를 해서, 다시 실행할 지를 결정.
While	조건 검사를 먼저 수행한 후에 블록에 포함되는 프로세스를 수행할 것 인지를 결정하는 노테이션	
Fork	속성에서 지정하는 값 또는 프로세스 플로우 수행 중에 생성되는 값	동시에 수행해야 할 프로세스를 생성하는 하위 프로세스 생성 노테이션으로 Fork로 생성한 하위 프로세스가 몇 개가 완료 될 때까지 Wating 할 것인지를 지정할 수 있다.
WaitStep	특정시간 동안 프로세스 흐름을 중단시키기 위해 노테이션	
Event	프로세스 수행 도중에 발생 할 수 있는 Event를 정의 하는 노테이션	
Joint	프로세스의 흐름이 합쳐지거나 분기를 할 위치를 지정하고 합쳐지는 조건이나 분기 조건을 명시하는 노테이션	

## 5.2 SAFE W/F

(주)유비넷에서는 SAFE-ERP 모듈과의 원활한 연동과 비즈니스 프로세스의 통합 및 관리를 위하여 국내 기업 환경에 적합하고 WfMC에서 규정한 스펙에 따라 워크플로우 주요 핵심 기술인 워크플로우 엔진과 워크플로우 정의기, 워크플로우 클라이언트를 포함한 SAFE W/F를 개발하였다. 그림 7은 실행 시점에서의 SAFE W/F 시스템의 구성을 보여준다.

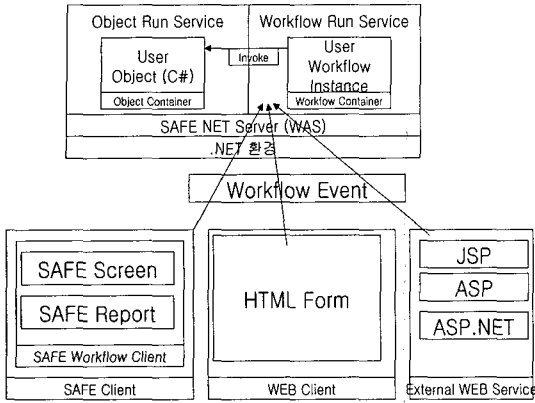
SAFE W/F에서 비즈니스 프로세스는 이벤트 기반으로 처리가 된다. 기존에 많이 사용되는 방식은 절차와 조건 분기로 흘러가며, 프로세스가 흐르는 중에 사용자의 처리가 필요하면 처리가 될 때까지 대기(wait)한다는 개념으로 외부 시스템이나 내부 애플리

케이션과 동기화된 연동을 통해 프로세스를 정의하게 된다.

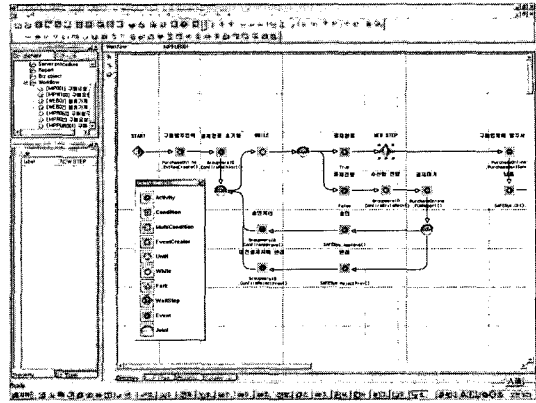
반면에, 이벤트 기반은 프로세스의 흐름의 단계에서 처리 후 사용자 처리나 외부 시스템 처리를 기다려야 할 경우, 대기하기 보다는 이벤트가 발생하기를 기다리는 구조로 정의가 된다. 이벤트는 프로세스 내부 이벤트와 전역 이벤트로 구분되어서, 프로세스가 진행되는 중에도 외부 이벤트를 통해서 외부 시스템과 비동기 연동을 할 수가 있다.

그림 8은 SAFE W/F에서 비즈니스 프로세스 정의하는 과정을 보여주며 표 3은 그림 8과 같은 프로세스 플로우 정의시에 사용되는 다양한 노테이션과 그 의미를 보여준다.

SAFE W/F는 비즈니스 프로세스를 하나하나 매번



(그림 7) SAFE W/F 시스템 구성



(그림 8) SAFE W/F 프로세스 정의

새로이 정의하지 않고 SAFE Plus를 통해 미리 개발된 ERP 컴포넌트들을 선택하거나 새로운 비즈니스 로직을 포함할 수 있는 범용 비즈니스 객체를 정의할 수 있게 함으로써 기존 ERP와의 손쉬운 연동과 독립된 비즈니스 플로우의 생성도 지원한다. SAFE W/F는 SAFE-ERP 모듈뿐만 아니라 타 ERP 패키지의 소프트웨어 모듈과의 연동도 자유롭게 지원하며 C#을 통한 범용적인 비즈니스 오브젝트를 생성함에 의해 비즈니스 프로세스의 생성과 확장이 용이하다는 이점이 있다.

## 6. 결 론

WfMS와 ERP 시스템은 모두 정보 관리를 위한 전략적 인프라를 생성하거나 비즈니스 프로세스를 지원하기 위해서 필요하다. WfMS와 ERP 시스템은 기업 정보와 프로세스를 통합을 위해 적용될 수 있는 명확히 구별되는 두 종류의 정보시스템이다. 두 시스템은 기업의 요구와 상황에 따라 선택적으로 혹은 함께 적용될 수 있는 특성들을 갖고 있다.

결론적으로 WfMS와 ERP 시스템의 명확한 차이점을 단정하기는 어렵지만, 이러한 차이를 인식하는 것은 특정 요구에 대한 적절한 시스템을 선택하기 위해 매우 중요하기 때문에 두 시스템의 차이점에 대해 개략적으로 살펴보았다. 두 시스템은 유사한 솔루션으로서 각각 다른 방식과 다른 효율성을 보여주며 여러 통합 환경에서 어떻게 작용할 수 있는지 고

찰하였다.

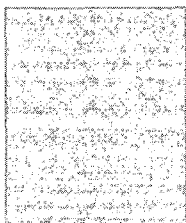
또한, 예로써 ERP와 워크플로우 시스템을 모두 제공하는 SAFE 솔루션에 대해서 설명하였다.

## 참 고 문 헌

- [1] D. Hollingsworth "Workflow Management Coalition Specification: The Workflow Reference Model", WFMC, 29 November, 1994.
- [2] D. Georgakopoulos, M. Hornick, A. Sheth, "An Overview of Workflow Management : From Modeling to Workflow Automation Infrastructure", Distributed and Parallel Database, April, 1995.
- [3] C. Mohan, "Tutorial : State of the Art in Workflow Management System Research and Products", IBM Almaden Research Center, Workflow Systems and Interoperability, August, 1997.
- [4] John A, Miler, Amit P. Sheth, Krys J. Kochut and Devanand Palaniswami, "The Future of Web-Based Workflows", Proc. of the Int'l. Workshop on Research Directions in Process Technology, July, 1997.
- [5] Georgakopoulos, D.,M.Hornick, and A.Sheth, "An Overview of Workflow Management: From Process Modeling to Infrastructure for

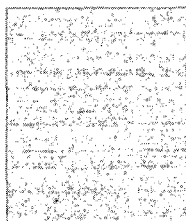
- Automation", Distributed and Parallel Databases, An International Journal, vol 3, no 2, 1995.
- [6] Kochut, K.J., A.P. Sheth, and J.A. Miller, "ORBWork: A CORBA-Based Fully Distributed, Scalable and Dynamic Workflow Enactment Service of METEOR", Large Scale Distributed Information System L류, Department of Computer Science, University of Georgia, Athens, GA. 1999.
- [7] SAP, SAP. <http://www.sap.com>, 2001.
- [8] Worah, D. and A.P.Sheth, "Transactions in Transactional Workflows", in advanced Transaction Models and Architectures, 1997.
- [9] Stohr, E.A. and J.L. Zhao, "Workflow Automation: Overview and Research Issues", Information Systems Frontiers, Vol 3, No 3, pp. 181-196, 2001.
- [10] Slater, D., Costly, "Painful and Worth It", CIO Magazine, <http://www.cio.com/archive/011502/costly.html>, 2002.

## ● 저 자 소 개 ●



### 백 종 명

1983년 고려대학교 산업공학과 (공학사)  
 1999년 고려대학교 일반대학원 전산학 (석사)  
 1982년~1991년 삼성전자 과장  
 1991년~1998년 시스템공학연구소 책임연구원  
 1998년~2001년 한국전자통신연구원 책임연구원/정보통합연구팀 팀장  
 2001년~현재 (주)유비넷 대표이사  
 관심분야 : ERP, Workflow, CRM, XML



### 박 성 진

1991년 고려대학교 전산학 (학사)  
 1993년 고려대학교 일반대학원 전산학 (석사)  
 1998년 고려대학교 일반대학원 전산학 (박사)  
 1998년~2000년 한국전자통신연구원 선임연구원  
 2000년~현재 한신대학교 정보시스템공학과 부교수  
 관심분야 : Data warehousing, Repository, Mobile Database