

데이터웨어 하우스 구축 방법론에 대한 연구†

(A Study of Implementation Methodology for Data Warehouse)

이 병 수* 이 상 략** 장 근***
(Byong-Soo Lee) (Sang-Rak Lee) (Keun Chang)

요 약 기업들은 정보시스템을 활용하여 방대한 데이터를 신속하고 정확하게 처리를 함으로써 생산성을 높일 수 있는 기회를 갖게 되었다. 그러나 의사결정을 지원하는데 있어 정보시스템 기능의 한계가 자주 제기 되었다. 이에 따라 의사결정 지원시스템(DSS)의 중심을 이루는 수리적 모델의 응용 이외에 업무적 분석을 지원하기 위한 시스템으로서 데이터웨어하우스의 필요성이 제시되고 있다. 이를 위해 본 연구에서는 데이터웨어하우스의 개발방법론과 관련한 문제 논의, 특히 개발과정과 관련된 다양한 모형을 소개하고 데이터웨어하우스의 성공을 위한 핵심사항과의 관련성을 고찰하였다.

Abstract Using Information Systems to process massive data, quickly and exactly, organizations have chances to enhance their performance. The limitations of IS function to support decision-making, however, have been frequently mentioned. In this context, in addition to traditional mathematical model that is a kernel of DSS, the needs for Data Warehouse which is a system supporting business process analysis are emerging. In this study, for those needs, first, we introduce issues of implementation methodology for D/W, especially various models relating development process. Then, we investigate correlation between these models and key factors for success of D/W.

1. 서 론

오늘날 기업들은 정보시스템을 활용하여 방대한 데이터를 신속하고 정확하게 처리를 함으로써 생산성을 높일 수 있었다. 특히 단순 반복적 혹은 정형적인 업무에 적용은 전통적인 데이터처리 영역이라 할 수 있다. 그러나 의사결정을 지원하는데 있어 정보시스템 기능의 미흡한 점이 자주 언급되어왔다. 이에 따라 의사결정 지원시스템(DSS)의 중심을 이루는 수리적 모델의 응용 이외에 업무적 분석을 지원하기 위한 시스템으로서 데이터웨어하우스의 필요성이 제기되어 왔다. 이러한 개념은 새로이 나타난 것은 아니지만 그것을 구현할 수 있는 기술적 한계는 최근에 극복 가능하게 되었다고 할 수 있다.

모든 시스템은 목적을 가지고 있다. 데이터웨어하우스는 특정한 업무영역 또는 기능영역을 지원하는 기존의 시스템과는 다른 적용범위를 갖는다. 또한 설계의 문제에서조차 사용자를 중심으로 해야 한다는 특징을 갖는다. 즉, 기존의 정보시스템의 개발방법론에서는 수용하기 어려운 특성을 갖고 있는 것이다.

따라서 본 연구에서는 데이터웨어하우스의 개발방법론과 관련한 문제를 논의하고자 한다. 특히 개발과정과 관련된 다양한 모형을 소개하고 데이터웨어하우스의 성공을 위한 핵심 사항과의 관련성을 고찰해보고자 한다.

2. 데이터웨어하우스의 개관

2.1 데이터웨어하우스의 정의

* 시립 인천대학교 전자계산학과 교수

** 시립 인천대학교 전자계산학과 조교수

*** 시립 인천대학교 전자계산학과 강사

† 본 연구는 1997년 인천대학교 연구비에 의해 수행 되었음.

데이터웨어하우스는 '의사결정 지원을 위한 주제 지향적, 통합적, 시간변이적, 비휘발성의 특성을 갖는 데이터 집합'이라고 할 수 있다 [2]. 이는 운영시스템 및 운영용 데이터베이스와의 차이를 중심으로 정의한 것으로 가장 일반적으로 인용되고 있다.

Poe는 '의사결정 지원시스템의 기초로 사용되는 읽기 전용의 데이터베이스'라고 정의다. 한편, Kelly는 '기업 내에서 다양한 플랫폼 및 아키텍처 상에서 구현된 다양한 데이터 모델을 포함하고 있는 문제를 해결하기 위한 주제 지향적 전사적 데이터베이스'라고 언급하고 있다.

이를 구체적으로 살펴보면, 데이터웨어하우스는 '직접 조회와 분석이 가능하도록 다양한 데이터베이스 혹은 정보 원천(information sources)으로부터 채택된 데이터를 하나의 정보 저장소에 모으기 위한 아키텍처, 알고리즘, 툴들'을 포함하는 개념이라 할 수 있다. 또한 개발 과정을 포함하여 '데이터웨어하우징'이라 언급되기도 한다.

2.2 데이터웨어하우스의 유용성

데이터웨어하우스의 사용에 따라 기대되는 혜택은 의사결정에 있어서 정보의 고유의(intrinsic) 질과 가치를 기초로 하는 것이 아니라, 접근성을 기초로한 정보의 이용을 시간(stime span, and interval)과 사용자(user) 측면에서 확장하는 것이다. 즉, 접근 가능한 데이터를 크게 늘림으로서 나타나는 것이다. 방대한 분량의 정보를 수집하는데 관리자가 소요하는 시간을 줄일 수 있으며, 환경에 관한 정보를 상시적으로 검토하는 기능을 포함시킬 수 있다. 또한 의사결정자의 추가적인 정보요구를 메타 데이터를 통하여 추적함으로써 효과적인 의사결정지원을 할 뿐 아니라 기업의 전략적 변화(혁신)를 추진하는 보조적인 역할을 할 수 있을 것이다.

2.3 데이터웨어하우스의 시스템적 특징

데이터웨어하우스는 기존의 정보시스템과는 상이한 시스템 적인 특징을 갖는다. 첫째, 요구분석이 사전적으로 결정되지 못한 상태에서 개발을 하여야 한다. 따라서 사용자의 요구사항을 파악하는 것이 반복적(iterative)으로 반복되어야 한다. 둘째, 처리하는 데이터가 운영시스템에서와는 다른 속성을 갖는다. 특히 시간이 하나의 키(key) 또는 필드로 포함되어야 한다. 셋째, 시스템이 데이터의 처리보다도 접근과 적재를 중심으로 하는 것이다. 따라서 시스템에서의 입·출력이 처리시간 등을 고려한 설계에서 중요한 요인이 된다.

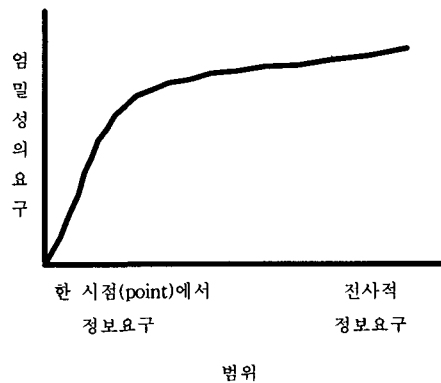
2.4 데이터베이스와 데이터웨어하우스

데이터웨어하우스는 사용자의 DB 종류(계층형, 관계형, 네트워크형)에 따라 상이한 솔루션을 가질 수 있다. 현재의 기술상태에서는 일반적으로 관계형 DB 솔루션이 적절하다. 조직의 업무특성에 따른 시간적 대응성의 필요 정도에 따라 온라인 분석업무의 요구를 결정하게 된다. 분석적 업무의 필요성 등에 따라 그 구축의 대상이 되는 데이터웨어하우스의 자체크기와 대상업무 범위를 결정하게 된다. 작은 규모 특히 특정 기능 부서를 지원하는 데이터웨어하우스를 데이터마트라고 부른다.

3. 개발 방법론과 데이터웨어하우스

3.1 SDLC 와 데이터웨어하우스

데이터웨어하우스의 개발은 다른 소프트웨어 프로젝트와 유사하다. 그러나 보통의 프로젝트와는 다른 특징을 가지고 있다. 데이터웨어하우스 프로젝트는 거의 항상 반복적이고 신속한 개발방법(rapid development mode)을 사용한다. 이러한 신속한 개발은 엄밀성(rigor)을 필요로 한다. 또한 [그림 1]에서 보이는 것처럼 전사적인 정보요구를 만족시키기 위해서는 더욱 엄밀성이 요구되는 과정으로 이해될 수 있다.



<그림 1> 범위와 엄밀성의 관계 [12]

3.2 시스템개발 수명주기와 데이터웨어하우스의 개발수명주기

전통적인 시스템개발수명주기를 데이터웨어하우스의 설계 및 구축에서 적용할 수 있는지를 검토하는 것은 의미가 있을 것이다. 요구사항분석이 데이터웨어하우스에서 차지하는 중요성은 전통적인 시스템 개발 방법론과 비교하여 매우 큰 차이가 나는 부분이 되고 있다. [2]은 시스

템 개발수명주기에 있어서 <표 1>에서와 같이 전통적인 시스템과 데이터웨어하우스는 역의 관계로 설명할 수 있다.

<표 1> 시스템개발수명주기의 비교

전통적인 시스템개발수명주기	DW 시스템개발주기
요구사항수집 (requirements gathering) 분석(analysis) 설계(design) 프로그래밍(programming) 시험(testing) 통합(integration) 구현(implementation)	DW 구현 데이터의 통합 편(bias)의 시험 데이터에 대한 프로그램 DSS 시스템의 설계 결과의 분석 요구사항의 이해

데이터웨어하우스의 개발을 위한 첫 번째 지침은 반복적(iterative)인 방법과 RAD(Rapid Application Development)기법을 사용하는 것이다. 따라서 각 프로세스는 다음의 특징을 갖는다.

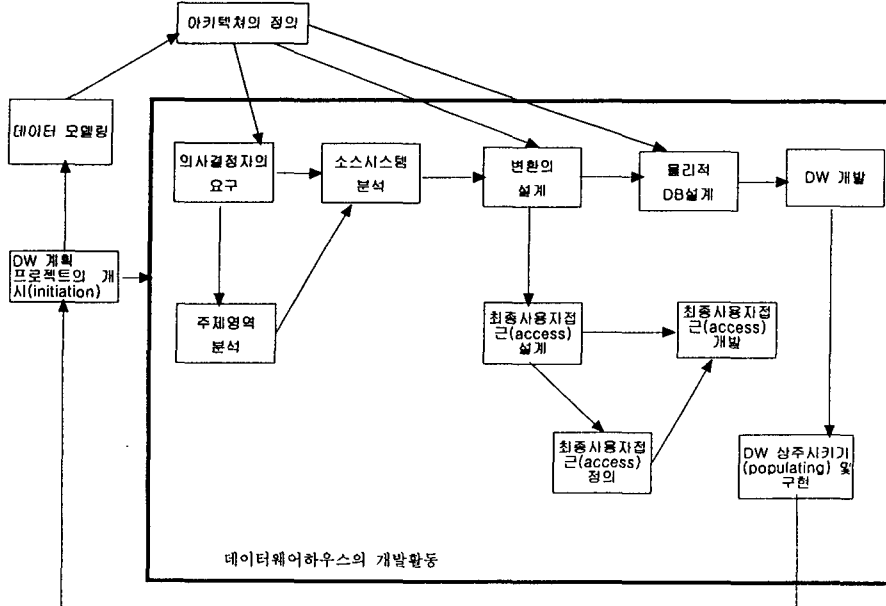
- ① 단기적, 실체적(tangible), 반복적인 산출물을 가지고 진도를 확인한다.
- ② 모든 단계(phases)와 활동(activities)은 시간적 제한이 있다.
- ③ 매우 높은 수준의 사용자 참여를 필요로 한다.

데이터웨어하우스의 설계에서 중요한 문제 중의 하나는 주기성이다. 여기서 주기성이란 운영환경에서 데이터의 변화가 데이터웨어하우스에 반영되는데 걸리는 시간을 의미한다.

데이터웨어하우스의 개발에서 가장 중요한 문제는 세밀정도(granularity)의 결정이다. 세밀정도가 적절하게 결정되면 그 밖의 설계 및 구현은 순탄하게 진행될 수 있다. 세밀정도의 결정은 데이터의 분량과 자세한 데이터를 사용할 수 있는 수준의 상충관계(trade-off)에 따라 결정된다.

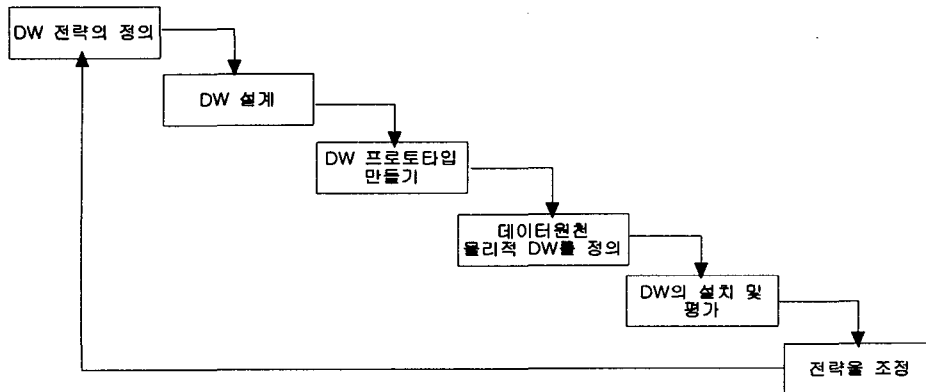
의사결정 지원 시스템의 개발을 위한 수명주기 단계는 다음과 같다.

- ① 계획
- ② 데이터 요구사항의 수집 및 모델링
- ③ 물리적 데이터베이스 설계 및 개발
- ④ 데이터 매핑과 변환
- ⑤ 데이터 추출과 적재
- ⑥ 데이터 관리 프로세스의 자동화
- ⑦ 애플리케이션 개발
- ⑧ 데이터 검증(validation)과 시험
- ⑨ 훈련
- ⑩ 배치(rollout)



[그림 2] 본원적 방법론(Generic Methodology)

(자료원 : John Ladley, "A Flexible Approach to Developing a Data Warehouse," Data Warehouse - Practical Advice From the Experts, Prentice Hall, 1997.)



<그림 3> 진화적/순환적 개발 접근법
(자료원 : Ken Orr Institute, 1995)

3.3 데이터마트와 관련한 데이터웨어하우스의 구축문제

Kimball은 9단계의 데이터웨어하우스의 설계방법을 제안하고 있다. 일반적으로 사용자들은 데이터웨어하우스의 설계문제에 대해 많은 관심을 가지고 있지 않으며, 시스템 설계는 정보시스템 부서의 업무로 인식하는 경향이 있다. 따라서 데이터웨어하우스 설계자는 하향 및 상향의 두 시각에서 다음의 9가지 단계를 수행하여야 한다.

① 1단계 : 프로세스의 선택

여기서 프로세스는 특정한 데이터마트의 주제를 의미한다. 처음에 고객 송장이나 월별 보고와 같이 가시적인 부분에서 출발하는 것이 좋다. 이러한 데이터원천은 접근성과 품질이 좋은 것이 일반적이다. 또한 초기에 데이터마트에 비용과 생산성 항목을 포함시키는 것은 프로젝트를 가로막는 결과를 가져올 수 있으므로 이를 피해야 한다.

② 2단계 : 세밀성의 선택.

세밀성의 선택은 기술적 세부사항처럼 보이지만, 설계를 위해서는 가장 우선적으로 이루어져야 하는 부분이다. 세밀성이 정해진 후에야 데이터마트에서 사실 테이블(fact table)의 차원이 무엇인가를 의미 있게 논의할 수 있다.

③ 3단계 : 차원을 식별하고 부합되게 만들.

차원은 데이터마트의 동인(driver)이다. 차원은 조건이 되는 값의 범위를 정하는 기초가 된다.

그러므로 사용자가 접하는 최종보고서에서 행의 표제가 되는 것이 바로 차원이다. 잘 설계된 차원의 집합은 데이

터마트를 이해하고 사용하기 쉽게 해준다. 표현을 잘못하거나 차원의 정의가 제대로 되어있지 않으면, 데이터마트의 유용성은 크게 떨어진다. 또한 차원은 장기적인 관점에서 선택되어야 한다. 두 개의 데이터마트에서 같은 차원이 포함되면, 이 둘은 정확히 일치하거나 하나가 다른 하나의 부분집합이 되어야 한다. 이 조건이 만족되어야 응용프로그램에서 차원을 공유할 수 있다. 이러한 차원을 부합되어 있다고 말한다. 가장 흔히 나타나는 예는 고객이나 제품차원이다.

④ 4단계 : 사실을 선택.

사실테이블의 세밀성은 데이터마트에서 사용할 수 있는 사실을 결정한다. 모든 사실은 세밀성으로 정해진 일정수준으로 표현되어야 한다. 사실테이블의 세밀성이 고객청구서의 각 개별항목을 나타낸다면, 모든 수치적인 사실은 특정항목에 해당하는 것이어야 한다. 또한 사실은 덧셈연산이 최대한 가능하도록 구성하여야 한다.

⑤ 5단계 : 사실테이블에 미리 연산된 값을 저장.

사실테이블이 손익계산서의 구성항목일 때 자주 나타나는 경우이다. 예를 들면, 고객청구서를 기초로 하는 사실 테이블이 있다.

⑥ 6단계 : 차원테이블을 확장(rounding out).

차원에서 문자형태(text-like)의 기술(descriptor)을 최대한으로 붙인다. 이들 문자형태의 속성은 사용자 인터페이스에 사용된다.

⑦ 7단계 : 데이터베이스의 지속기간을 선택.

지속(duration)기간은 사실테이블이 얼마나 오래되었는가를 측정하는 개념이다. 업무상 지난해의 같은 기간을 살펴야 할 필요가 생기는 것이 보통이다. 보험회사 등 외부보고가 중요한 기업에서는 장기간의 지속기간을 가져야 한다. 장기의 지속기간을 가지는 경우에는 데이터웨어하우스의 설계에서 중요한 두 가지 문제가 발생한다. 첫째, 오래된 데이터일수록 읽거나 해석하는 문제가 심각하다. 둘째, 중요한 차원에 대하여 예전의 것을 사용하여야 한다.

⑧ 8단계 : 차원의 지연된 변화를 추적.

⑨ 9단계 : 조회의 우선 순위와 방법(mode)을 결정.

이 단계는 물리적 설계에서 가장 중요하다. 물리적 저장매체에서 사실테이블의 정렬방법, 요약 또는 집계적 저장여부를 결정한다.

3.4 적재와 접근의 최적화를 위한 시스템

데이터웨어하우스는 적재와 접근을 최적화 하는 시스템으로 이해할 수 있다. 이는 데이터 모델의 설계문제를 도외시한 접근이지만, 기존의 정보시스템이 구체적인 업무적 목표에 대한 최적화 시스템이라는 것에 대비하여 그 특성을 시스템의 기술적 목표측면에서 잘 설명 해 준다.

3.5 프로토타이핑(Prototyping)과 데이터웨어하우스

1) 프로토타이핑 방법론

프로토타이핑은 사용자가 정보시스템의 특성을 파악하고 있지 못할 때 요구사항을 알아내는데 효과적이다. 특히 의사결정지원 시스템이나 임원정보 시스템에서 그런 특성이 나타난다.

프로토타이핑의 장점으로는 ① 더 짧은 개발시간, ② 정보시스템의 질 향상, ③ 경제성이 있다는 것이다. 또한 사용자 참여를 촉진하여 진정한 요구사항을 짧은 시간 내에 결정할 수 있으므로 개발시간을 단축시킬 수 있다.

프로토타이핑의 시행순서는 다음과 같다. ① 경영층과 사용자의 승인 및 지원을 얻는다. ② 될수록 많은 기반기술을 설치하여 사용한다. ③ 성공가능성이 높은 파일럿 프로젝트를 고른다. ④ 프로젝트에 참여할 진취적인 스템을 선정하여 교육하고 'Champion'을 삼는다.

2) 요구분석을 위한 프레임워크

DSS는 특성상 요구분석이 설계 및 개발단계에 앞서서 이루어질 수 없다. 사용자들이 원하는 정보를 스스로 알지 못할 뿐 아니라 데이터웨어하우스가 어떻게 도움이 될

것인지를 사용해 본 후에야 알 수 있기 때문이다. 요구사항 모델은 명확히 문서화된 업무모델이 없을 경우가 실제로 자주 있기 때문에 문제가 된다. 요구사항 모델은 새로운 시스템 개발의 좋은 프레임워크를 제공하고 리엔지니어링이나 재평가가 필요한 영역을 가려내는 프레임워크이다.

요구분석을 위한 프레임워크
주제중심의 프레임워크 : PIECES(Performance, Information, Economy, Control, Efficiency, and Services)(Wetherbe)
P : 처리능력(throughput) 및 반응시간, I : 데이터모델의 기초가 되는 입력 · 출력 · 저장 데이터를 취급, E : 프로젝트 개발 및 운영비용, 시스템 도입으로 생기는 비용절감, C : 입력 데이터의 편집과 시스템 보안, E : 방법의 적정성을 기업 · 부서 · 개인 중 최소한 한 수준으로 측정됨, S : 시스템의 기능, 프로세스의 기능

위의 프레임워크를 데이터웨어하우스에 적용하면, 성과(P) 차원에서의 처리능력보다는 확장성(scalability)이 적절할 것이다. 정보(I)차원에서는 데이터모델 자체의 구성이 더욱 강조된다. 경제성(E)에서는 타사의 유사모델을 파악하지 못하면 비용이나 혜택을 도입에 앞서 예측하기가 더욱 어렵게 된다. 따라서 시행착오(trial-and-error)를 통해서 평가기술이 축적될 것이다. 통제(C)는 데이터의 편집과정이 필요치 않고(메타 데이터를 통하여 변환과정에서 일괄처리) 시스템 보안문제가 높은 수준으로 이루어져야 할 필요가 있다(이는 사용자의 다양성 때문이다). 효율(E) 차원의 방법의 적정성은 기업 · 부서 · 개인보다는 상위관리층 · 중간관리층 · 하위관리층 · 고객 · 공급업자 · 이해관계자 등의 이용에 따른 방법을 사용의 용이성과 정보에 대한 만족 도를 하위차원으로 측정할 수 있을 것이다. 또한 요구분석 과정은 ① 개념적 설계, ② 논리적 설계, ③ 검증, ④ 공식적 명세화의 과정을 이루어진다고 볼 수 있다.

3) 반복순환 개발의 필요성

시스템을 개발하기 전에 미리 사용자의 요구사항을 알 수 없기 때문에 반복순환을 통한 사용자 요구분석의 정의와 확인이 필요하다. 따라서 방법론적 특징은 순환개발(iteration)을 하여야 한다는 것과 단계적 재발에 있어서 각 단계의 수행이 더 많이 중첩되고, 빠른 순환주기를 가져야 한다는 것이다. 사용자가 실제로 어떻게 사용하는가 하는 문제는 업무적 환경, 기술적 환경, 사용자의 수준, 산업의 동향 등 사업관련 변수들의 전반적인 변화에 따른 정보요구를 반영하는 것이다. 따라서 정보요구사항을 사전적으로 결정할 수 있다고 가정하면 안 된다. 데이터웨어하우스가 지원하는 의사결정은 조직과 사업의 모든 측

면을 반영하고 있기 때문이다. 그러므로 특정기술로스 사업을 미리 규정할 수 없다는 것을 고려하여야한다.

4. 데이터웨어하우스의 구축방법론

4.1 데이터웨어하우스의 설계를 위한 변수

1) 데이터모델

데이터모델이란 '데이터구조와 프로세스에 대하여 언어, 그림 숫자, 기타의 매체에 의해 추상화된 표현'이다. 데이터모델은 구조와 프로세스를 잘 이해할 수 있도록 해준다. 데이터모델은 데이터 설계에서 데이터 구조가 정확하도록 돕는다. 모델의 핵심은 효율적인 표현이다. 그러나 데이터 구조는 시간이 진행됨에 따라 확장된다. 이는 주제 영역에 대한 지식이 변화함에 따른 것이다. 데이터 모델링을 돕는 방법으로 잘 정립된 방법론과 CASE 툴이 있다.

데이터웨어하우스를 위한 데이터모델은 기업 데이터모델을 수정하여 사용한다. 기업 데이터모델은 기초적인 데이터로 구성되며, 운영시스템과 데이터웨어하우스의 구분 없이 적용할 수 있는 전체적인 데이터모델이다. 운영적 사용에서는 거의 변경되지 않지만, 데이터웨어하우스를 위해서는 상당부분 변경된다.

데이터웨어하우스의 데이터모델에서 핵심을 이루는 것은 바로 메타 데이터이다. 메타 데이터란 데이터에 관한 데이터이다. 데이터웨어하우스에서는 메타 데이터가 기존 정보시스템에서보다 더욱 중요하다. 메타 데이터가 데이터웨어하우스를 효과적으로 사용하는데 결정적이기 때문

이다. 메타 데이터는 사용자인 분석가들이 정보를 검색하는 경로를 관리하는 방법을 제공한다. 메타 데이터는 ① 데이터의 구조, ② 원천 데이터, ③ 데이터의 변환, ④ 데이터모델, ⑤ 데이터모델과 데이터웨어하우스의 관계, ⑥ 추출에 관한 기록 등을 추적하는 정보를 가지고 있다. 운영적 환경에서 데이터웨어하우스 환경으로 사상하는 정보를 메타 데이터가 관리한다. 정보요구는 조직의 계층 수준에 따라 그리고 기능에 따라 달라진다. 계층수준은 데이터의 요약정도로 설계되고 기능은 메타 데이터로 설계될 수 있다.

2) 세밀화정도(Granularity)

세밀화정도는 요약의 수준과 집계 수준으로 크게 나눌 수 있다. 요약이란 정보를 양적으로 감소시키고, 질적으로 풍부한 내용을 가진 정보로 만드는 것이다. 집계란 정보를 일정한 자원으로 합하여 총괄적인 정보로 만드는 것을 말한다.

① 요약(Summarization)의 수준 결정

요약된 데이터를 만들기 위한 세밀성의 정도를 정하기 위해서는 최종사용자에게 데이터를 보여주어야만 알 수 있다. 따라서 직접 사용할 시스템의 고객을 명확히 결정하여 이들의 요구를 통해서 파악하는 것이 필요하다. 따라서 빠른 프로토타이핑을 사용하는 것이 유용할 수 있다.

② 집계(Aggregation)의 수준 결정

데이터웨어하우스 설계에서 가장 중요한 측면은 집계수준이다. 집계수준은 데이터웨어하우스에서 단위 데이터에 적용되는 정밀도를 말한다. 특히 시간차원에서의 집계수준은 데이터의 분량결정에 매우 중요하다. 운영시스템에

<표 2> OLAP와 데이터웨어하우스의 보완적인 관계

목표		방대한 데이터지원	세밀 정도가 다른 다양한 수준의 지원	많은 데이터 분석요인(Analysis Factor)의 지원	많은 사이트에서 지원
적시성	사전 처리됨	DW	DW	DW	DW
	빠른 접근	OLAP	OLAP	OLAP	DW
	빠른 계산	OLAP	OLAP	OLAP	DW
정확성	정확한 원시 데이터	DW	DW	DW	DW
	연산의 풍부성	OLAP	OLAP	OLAP	OLAP
이해가능성	쉬운 인터페이스	OLAP	OLAP	OLAP	N/A
	유연한 보기 기능	OLAP	OLAP	OLAP	OLAP

서 집계수준은 가장 낮은 수준의 데이터로 저장되는 것이 당연하다. 집계수준은 데이터웨어하우스에 상조하는 데이터의 양에 큰 영향을 미치며, 응답될 수 있는 조회의 유형에도 큰 영향을 미친다. 데이터웨어하우스에서 데이터의 양은 조회에서의 자세한 정도와 상충 하는 관계를 가지고 있다.

3) 구현 방법의 결정 : 온라인 분석의 필요성

데이터웨어하우스 구축 시에 온라인 프로세싱을 어느 정도 수용할 것인가의 문제는 설계와 구축에서 유용성을 결정하는데 있어 매우 중요한 변수가 된다. 기업의 의사결정의 요구가 시간경쟁(time-based competition)의 중요성에 따라 달라질 것이지만, 분석업무의 적시성의 수준은 업무적 요구를 통한 별도의 분석을 필요로 한다. 결론적으로 OLAP의 필요성 문제는 데이터웨어하우스 구축의 유용성 문제와 보완적인 관계를 가진다. 다음의 <표 2>을 OLAP와 데이터웨어하우스의 보완적인 관계를 나타낸 것이다.

4) 적재의 간격 결정

데이터웨어하우스에서 어느 정도의 시간간격을 두고 데이터를 상주시킬 것인가의 문제는 설계의 중요한 변수가 된다. 재 적재기간은 데이터모델의 수정과 의사결정요구를 반영하는 주기를 이루게 된다. 따라서 재 적재는 의사결정 요구에 대한 시간적 주기와 밀접한 관련을 가지도록 설계되어야 한다. 이를 운영시스템과 독립적인 기능을 유지하는 적절한 기간으로 고려될 수도 있다. 특히 의

사결정의 지원을 위해 최적화된 시스템으로 정착되는 것이 성공의 지표라 할 때 다양한 목적을 충족시키려는 시도는 실패 위험을 매우 크게 만든다. 또한 시간 간격은 데이터웨어하우스에 옮기기 전에 데이터를 안정시킬 기회를 제공한다. 이는 데이터의 품질을 높이는 역할을 수행한다.

이를 데이터의 주기성으로 파악할 수도 있다. 데이터의 주기성이란 운영환경에서 일어난 데이터의 변화가 데이터웨어하우스 환경에 반영되기까지의 시간 길이를 말한다. 예를 들어 24시간 의 간격이 있으면 데이터웨어하우스에서 운영프로세스를 수행하거나 운영프로세스에서 데이터웨어하우스 프로세스를 수행하려는 유혹을 미연에 방지할 수 있다.

운영환경과 데이터웨어하우스 환경을 결합시키는 정도가 강할수록 기술적으로 복잡해지고 비용이 많이 든다. 시간적 지연이 있는 경우 환경에의 특정한 규율을 적용 가능하게 한다.

4.2 데이터마트

1) 데이터마트와 데이터웨어하우스

데이터웨어하우스와 데이터마트는 개념이 명확히 구분되지 않을 정도로 혼용되어 사용되기도 한다. 보편적으로 전사적 의사결정의 지원과 부서별 지원이라는 점에서 구분되어진다. 특히 동시에 설계할 경우에 확장성의 문제와 관련하여 논의가 활발하다. 다음 <표 3> 는 데이터웨어하우스와 데이터마트를 구분하는 가트너 그룹(Gartner

<표 3> 데이터웨어하우스와 데이터마트의 비교

구분	데이터웨어하우스	데이터마트
범위	애플리케이션에 독립 중앙집중형, 공유됨 여러 업무에 걸침/ 기업 의도적으로 설계됨	특정의 애플리케이션 요구 여러 업무에 걸치거나, 부서, 사용자 영역 업무 - 프로세스 지향 중복 데이터를 가진 여러 개의 데이터베이스
데이터 관점	역사적 - 세부 데이터 약간의 요약 약간 비정규 화됨	세부적(일부 역사적) 요약됨 고도로 비정규 화됨
주제	다양한 주제 영역	단일 주제 여러 개의 부분 주제 영역 운영적 원천의 스냅샷
데이터의 원천	많은 운영적, 외부 데이터	적음 운영적, 외부 데이터 OLTP 데이터베이스 스냅샷 '부트 레그' 데이터 추출
구현 시간대	첫단계에서 9~18개월 (2~3개의 주제 영역) 다 단계 구현	4~12개월
특성	유연성 장기적/ 전략적 데이터 지향	제한적 단기적 수명/ 전술적 프로젝트 지향

Group, Inc.)이 제시한 기준이다.

2) 방대한 양

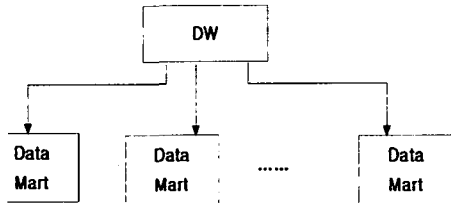
데이터웨어하우스 구축을 위해 대상이 되는 방대한 양의 데이터를 관리하도록 프로파일 레코드(profile record)를 만드는 것이 효과적이다. 여기서 데이터웨어하우스가 다루어야 할 데이터의 양이 방대하거나 내용이 자주 바뀌는 경우에는 특별한 종류의 레코드를 필요로 한다.

여기서 다수의 운영데이터를 묶어주는 하나의 레코드를 프로파일 레코드 또는 집계 레코드(aggregate record)라고 부른다. 이것은 모든 데이터 구조에 들어가게 된다. 특히 양적인 관리에서는 프로파일 또는 종합 레코드를 만드는 것이 가장 중요한 기법이 된다. 반면에 이에 따른 데이터웨어하우스의 능력과 가능성이 손상되는 단점도 갖는다. 그러므로 DSS 분석가에게 세밀함이 얼마나 중요한지를 먼저 평가하도록 하여 프로파일 레코드를 설계하여야 한다.

프로파일 레코드를 주기적, 반복적으로 만든다면 매우 중요한 요소인 세밀함을 놓치는 오류를 방지할 수 있다. 데이터웨어하우스의 주기적인 반복활동이 짧고, 신속하면 변경에 있어 조율성을 높일 수 있다. 여기서 중요한 세밀성을 보장하는 또다른 방법으로는 세밀성이 다른 수준에서도 레코드를 만드는 것이다.

3) 하향식 설계(Top-down) 모델

Top-Down Development of Data Warehouse



<그림 4> 하향식 설계

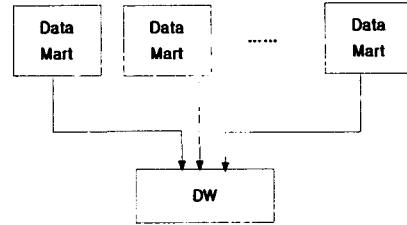
켄 오어는 데이터마트의 설계와 데이터웨어하우스의 설계를 연관하여 하향식 설계를 <그림 4>와 같이 정의한다.

4) 상향식 설계(Bottom-up)

또한 데이터마트의 설계와 데이터웨어하우스의 설계를 연관하여 상향식 설계를 <그림 5>와 같이 정의한다.

정보시스템 설계 접근 방법은 위의 <그림 4>, <그림 5>와 같이 크게 하향식(top-down) 접근법과 상향식

Bottom-Up Development of Data Warehouse



<그림 5> 상향식 설계

(bottom-up) 접근법으로 나눌 수 있다. 하향식 접근법은 최상위 목표로부터 시작하여 점차 하위 목표에 접근하면서 요구되는 정보시스템의 하부구조를 파악하여 정보시스템을 구축하는 것을 의미하며, 상향식 접근법은 최하위 시스템으로부터 점차 전체적인 최상위 시스템을 정의하고 구현하는 것을 의미한다.

정보시스템 계획(planning)은 하향식 접근법을, 실행(implementation)은 상향식 접근법을 활용하는 경우 정보시스템의 응용과 관련된 조직 부문의 폭 넓은 참여를 유도할 수 있음과 동시에 조직의 최상위 목표를 효과적으로 파악하고 분석할 수 있을 것이다. 다만, 이 경우 일반적으로 많은 시간이 소요된다는 단점이 있다.

5. 결론

데이터웨어하우스의 구축을 위한 문제영역은 첫째, 데이터모델의 필요성, 둘째, 모델에 데이터를 상주시키기 위한 전략, 셋째, 사용자가 유용한 정보를 조회하는 방법의 3가지 영역으로 크게 구분할 수 있다. 여기서는 데이터웨어하우스의 구축문제로서 ① 요구사항의 분석, ② 순환개발, ③ 데이터마트의 확장성 문제를 다루었고, 설계변수로는 세밀화 정도(Granularity), 온라인 질의의 필요성, 데이터 주기성을 다루었다. 이외에 의사결정모델, 분석 업무의 요구를 별도로 분석하고, 정보의 시간차원에 대한 보다 깊은 연구가 필요할 것이다.

데이터웨어하우스 설계 및 구축에서는 사용자의 요구사항의 분석, 개발기간, 외부 데이터의 고려, 다목적으로 설계하는 문제 등이 성과와 관련된다. 특히 기존의 사례 연구결과에서 데이터웨어하우스를 구축하는 기업의 특징으로 첫째, 경영관리를 정보중심으로 접근, 둘째, 매우 경쟁

적이고 급변하는 시장환경. 셋째, 많고 다양한 고객기반. 넷째, 데이터가 저장된 시스템의 다양성. 다섯째, 동일한 데이터의 시스템간에 상이하게 표현됨. 여섯째, 포맷의 해석과 정돈의 어려움 등의 특징을 가진다. 따라서 데이터 웨어하우스 구축에서 산업의 특성과 고객의 특성에 따라 성패에 미치는 영향이 클 수 있다.

[11] Ralph Kimball, *Data Warehouse Architect*, DBMS magazine, Dec., 1996.

[12] *Data Warehousing in Action*, Sean Kelly 1997.

참 고 문 헌

[1] Widom J. "Research Problem in Data Warehousing," *Proceedings of 4th International Conference and Knowledge Management(CIKM)*, Nov. 1995.

[2] Immon, B. *Building the Data Warehouse*, New York: John Wiley & Sons, 1996.

[3] Strange, K. "Can Data Marts Grow?, " *CIO magazine*, July 1, 1997.

[4] Browne, M. *Organizational Decision Making and Information*, New York: Ablex Corp., 1993.

[5] Thomsen, E. *OLAP Solutions*, New York: John Wiley & Sons, 1997.

[6] Norman, R. J. *Object-Oriented Systems Analysis and Design*, New Jersey: Prentice Hall, 1996.

[7] Sprague, R. H. Jr. "A Framework for the Development of Decision Support Systems," *MIS Quarterly*, Dec, 1980.

[8] Epstein, B. J. *The Multidimensional Value of Information*, Michigan: UMI, 1980.

[9] Drucker, P. F. "The Information Executives Truly Need," *HBR*, Jan-Feb, 1995.

[10] John Ladley, "A Flexible Approach to Developing a Data Warehouse," *Data Warehouse - Practical Advice From the Experts*, Prentice Hall, 1997, pp.100-119.