데이터웨어하우스 개발 프로세스를 위한 측정지표의 설계

Design of Metrics for Datawarehouse Process

박 종 모[†], 안 효 범[††], 김 홍 준[†††]

[†] 단국대학교 정보컴퓨터학부 jmpark@dankook.ac.kr
[††] 공주대학교 정보통신공학부 hbahn@kongju.ac.kr
[†††] 진주산업대학교 컴퓨터공학부 thinkthe@jinju.ac.kr

요 약

최근의 기업환경에서는 데이터 분석을 통한 마케팅이 기업의 경쟁력에서 중요한 부분을 차지하며, 이를 지원하기 위하여 분석 정보를 추출하여 저장하는 데이터웨어하우스가 사용된다. 그러나 데이터웨어하우스는 다양한 종류의 업무 시스템으로부터 대규모의 데이터를 처리하기 때문에 많은 시간과 비용이 소요된다. 이러한 문제의 해결방안으로 지속적인 데이터웨어하우스의 프로세스 개선을 통해 시간과 비용의 절검이 가능하다. 프로세스 개선을 위해 본 연구에서는 생산성, 프로세스 품질, 데이터 품질의 영역에서 측정지표를 제안한다. 측정 지표를 통해 프로세스 개선과 제어를 위한 기반을 제공한다.

키워드: 데이터웨어하우스, 측정지표

1. 서론

정보시스템의 기술혁신에 따라 소프트웨어 개발 프로세스는 규모가 거대해지고 처리되는 데이터의 크기도 증가하는 반면에 짧은 시간 내에 소프트웨어를 개발해야 하는 등으로 환경이 변하고 있다. 이에 따라 소프트웨어 개발만을 목표로 하지 않고, 소프트웨어 프로세스 개선을 위한 프로세스 측정의 필요성이 제시되었다. 프로세스 측정은 프로세스의 수행 과정에서 계획과의 편차를 발견하는 기초가 되며, 개선을 위한 기회를 제공하고 감시 및 제어 활동의 기준이 된다[1]. 프로세스 측정은 다음의 네 과정으로 수행된다. 프로세스의 성과를 측정하기 위해 데이터를 수집하고, 각 프로세스의 성과를 분석한다. 다음으로 프로세스의 안정성과 분석결과를 해석하고, 마지막으로 미래의 비용 및 성과를 예상하기 위해 데이터를 사용하고 유지한다.

본 연구에서는 데이터웨어하우스의 개발 프로세스를 개선하기 위해 프로세스 측정에 필요한 지표를 설계하여 프로세스 개선의 기반을 제공한다. 프로세스 개선을 위한 생산성, 프로세스 품질, 데이터에 관한 측정지표를 제안한다. 즉, 데이터웨어하우스의 개발프로세스를 대상으로 프로세스 측정 지표의 설계를 통해 계획과의 편차를 측정함으로 부족한 부분을 보완하여 프로세스 개선의 기반을 마련한다.

본 논문의 구성은 다음과 같다. 2장에서는 데이터웨어하우스와 프로세스 측정에 관련된 연구를 제시한다. 3장에서는 데이터웨어하우스 개발 프로세스의 개선에 활용할 측정 지표를 제안하고, 4장의 결론 및 향후 연구로 본 연구를 마무리한다.

2. 관련연구

데이터웨어하우스는 운영업무 지원시스템인 OLTP(OnLine Transaction Process) 시스템에

서 생성된 데이터로부터 다양한 분석 정보를 추출하여 의사결정 지원시스템인 OLAP(OnLine Analysis Process)에 사용하기 위한 데이터의 저장고이다[2,3]. 데이터웨어하우스를 이용하면 사용자가 대화식으로 정보를 분석할 수 있는 의사결정 지원시스템인 OLAP 질의에 빠른 응답을 할 수 있다. 데이터웨어하우스는 대규모의 데이터를 처리하기 때문에 지속적인 품질과 생산성을 향상시키기 위해 프로세스의 개선이 필요하다.

본 연구에서 측정을 위해 제안하려는 첫 번째 지표의 대상은 프로세스이다. [4]에서는 <표 1>과 같이 CMMI의 측정 프로세스들에 대하여 프로세스 개선에 대한 객관적인 증거를 제시하기 위해 측정을 위한 정량적인 지표들을 정의하였다. 이 지표를 통해 프로세스 능력 수준이 결정되고 지속적인 측정값 수집을 통한 조직의 프로세스 개선 예측이 가능하며, 자원의 효과적인 분배와 문제발생에 대한 조기 대응하여 프로젝트 성공확률을 증대시킬 수 있다.

<표 1> 프로세스의 측정지표

구분	지표명	산출 공식 (단위: %)
일정	계획공정 준수율	실행 공정수/계획 공정수
	공정 진도율	실제 진도/계획 진도
	계획 투입공수 준수율	실제 투입공수/계획 투입공수
비용	계획 예산 준수율	집행예산/계획예산
생산성	공정별 생산성	분석: 요구사항 수/투입공수 설계: 설계항목 수/투입공수
프로세스 품질	요구사항 변경율	변경된 요구사항수/최초 요구사항수
	위험발생 비율	실현된 위험 수/파악된 위험수
	발견대비 결함 제거비율	제거된 결함 수/발견된 결함수

측정을 위해 제안되는 두 번째 지표의 대상은 데이터이다. 원천 소스의 데이터에 따라 분석의 정확성과 신뢰성이 보장되기 때문에 데이터웨어하우스의 개선을 위해서는 원천 소스의 데이터 품질이 매우 중요하다. 데이터의 품질을 측정하기 위한 지표는 아래의 <표 2>와 같이 완전성, 일관성, 최신성, 정확성의 4가지 기준에 따라 제시되었다[5]. 완전성은 정해진 데이터 구조 안에서 정보가 누락됨 없이 모두 포함되어 있는지를 판단하며, 일관성은 데이터가 상충되지 않고 일관된 상태를 이루고 있는지를 관리한다. 최신성은 최근의 정보를 제공하여 지속적인 갱신이 이루어지는가에 대한 관점이며, 정확성은 데이터 값과 데이터 표현이 실제 값과 동일한지를 측정하는 기준이다.

<표 2> 데이터 품질의 측정지표

측정 기준	데이터 값	데이터 구조	데이터 흐름
완전성	데이터 크기 데이터 범위 데이터 값 누락	중요 속성 누락 필수 속성 설계	데이터 생성 가공 시 누락
일관성	데이터 속성과 값의 일치 데이터 제약 조건의 일치 테이블 정의와 레코드 일치 동일 데이터의 상호일관성	데이터 표준 정의의 적절성 도메인 정의의 적절성 코드 정의의 적절성	데이터 생성 가공 시 데이터 적용
최신성	최신 데이터의 제공	-	데이터 갱신 주기
정확성	데이터 오탈자 실제 사실과의 일치 레코드 중복	참조 무결성 속성 중복 및 유일성 보장	원천 데이터의 신뢰성 데이터 생성 가공 시 오류 및 중복

3. 데이터웨어하우스 개발 프로세스를 위한 측정지표의 제안

본 장에서는 데이터웨어하우스 개발 프로세스를 측정하기 위한 지표를 제안한다. 측정지표를 통해 프로세스의 목적달성이나 활동의 수행 정도를 측정하여 계획과의 편차를 분석함으로 프로세스 개선을 진행할 수 있기 때문이다. 프로세스 개선을 위한 <표 3>의 측정항목은 앞 장의 <표 1>과 <표 2>에서 제시된 프로세스와 데이터 품질 측정에 사용된 항목을 데이터웨어하우스의 특성에 적합하도록 보완한 것이다. [6]에서 제시된 데이터웨어하우스 개발에 가장 중요한 특성에 따라 측정항목을 첫째, 데이터웨어하우스의 대규모 데이터로 인한 프로세스 처리를 측정하기 위한 생산성 항목 둘째, 요구사항 변경 및 오류 등을 처리를 측정하는 프로세스 품질 항목, 마지막으로 원천 소스의 데이터로 인해 발생하는 데이터 자체의 품질을 분석하기 위한 데이터 품질 항목으로 구분하였다.

<표 3> 데이터웨어하우스의 측정항목

구분	측정항목		설 명
생산성	S		소프트웨어의 크기
	E		프로젝트에 투입된 공수(Man/Month) (시스템 구축의 기간과 투입비용)
	D	D1	프로젝트 계획일수
		D2	프로젝트 지연일수
프로세스 품질	R	R1	개발 기간 동안 제거된 결함개수
		R2	고객에게 인도 후 제거된 결함개수
		R	제거된 결함 수 (R = R1 + R2)
	A	A1	요구사항 변경건수
		A2	요구사항 구현건수
		A	요구사항 수
	F	F1	개발 기간 동안 발견된 결함개수
		F2	고객에게 인도 후 발견된 결함개수
		F	프로젝트 전체 결함개수(F=F1+F2)
데이터 품질	데이터		데이터 자체의 품질
	데이터관리		데이터 관리 정책
	데이터구조		데이터 간의 구조

프로세스 품질을 측정하기 위한 측정 지표는 <표 3>의 항목들을 기반으로 하여 다음의 <표 4>와 같이 제안한다.

<표 4> 프로세스 측정지표

구분	평가대상	측정 지표
생산성	투입공수	투입공수율 = E / S
	지연일수	계획공수 지연율 = D2 / D1 투입공수 지연율 = D2 * E
프로세스 품질	결 함	결함 주입률 = F / S 결함 제거율 = R / F 개발기간 동안 결함제거율 = R1 / F1 인도 후 결함 제거율 = R2 / F2
	위 험	재작업 비율 = (F1 - R1) / S 요구사항 변경율 = A1 / A 요구사항 구현율 = A2 / A
데이터 품질	데이터	데이터 값 및 항목의 누락여부 데이터 정확성에 대한 신뢰도 데이터 표현형식의 적절성 데이터의 유일성, 제약조건(유효범위) 데이터의 최신성
	데이터 관리	데이터 표준 관리 (용어사전, 명명규칙, 코드표준) 데이터 관리 정책 (데이터 권한 및 보안)
	데이터 구조	참조무결성을 위한 테이블 간의 관계 데이터 구조 변경에 따른 데이터모델관리 시스템간의 데이터 중복성

<표 4>의 프로세스의 측정지표에서 생산성은 투입공수 지연율의 새로운 지표를 제안하였고, 프로세스 품질은 데이터웨어하우스에 적합하도록 수정하였다. 데이터 품질의 영역에서는 <표 2>의 데이터 품질측정 지표 중에 데이터웨어하우스에 적합한 지표를 선택하였다.

생산성은 투입공수와 지연일수로 측정한다. 투입공수에서 소프트웨어의 크기는 데이터웨어하우스의 경우 일반적인 프로그램 규모 산정에 사용되는 LOC(Line Of Code)가 아니라 축적된 데이터베이스의 크기이다. 왜냐하면 데이터웨어하우스가 분석을 위한 데이터의 저장소로서 사용되기 때문에 데이터베이스의 크기가 중요 관점이다. 생산성과 밀접한 관련이 있는 계획공수 지연율은 프로세스 관리에 초점이 맞추어 있으며, 프로세스 지연에 따른 계획일수와 실제일수의 차이를 분석한다. 지연일수가 발생하였을 때 동일한 일정의 지연이라 할지라도 비용이 더 투입되는 곳을 집중 관리하기 위해, 본 연구에서는 계획공정 지연일수와 비용의 크기인투입 공수를 함께 고려하였다. 즉, 지연된 일수에 투입공수를 비용으로 환산한 지연된 비용의 가치인[M/M]를 곱하여 투입공수-지연율(M/M-Days)을 측정 기준으로 제안한다.

프로세스 품질영역에서 결함은 결함 제거 측면에서 결함이 발생할 여지가 있는 결함 주입율과 결함 데이터의 제거비율로 측정한다. 위험은 재작업 비율과 요구사항의 변경 및 구현율로 측정한다. 데이터 영역은 데이터 품질 수준에서 데이터, 데이터 관리, 데이터 구조의 측정지표를 제시하고 분석한다.

4. 결론

최근의 기업환경에서는 데이터 분석을 통한 마케팅이 기업의 경쟁력에서 중요한 역할을 담당한다. 마케팅 및 기업의 의사결정 지원을 위하여 업무 시스템으로부터 생성된 데이터에서 다양한 분석 정보를 추출하여 저장하는 정보의 저장고로 데이터웨어하우스가 사용된다. 그러나 데이터웨어하우스는 다양한 종류의 업무 시스템으로부터 대규모의 데이터를 처리하기 때문에 많은 시간과 비용이 소요되며, 오류로 인해 발생되

는 위험과 재작업의 비율이 높다. 또한 원천 소스에서 발생하는 데이터의 품질문제로 인해 분석의 신뢰도가 떨어지게 된다. 이와 같이 다양화된 개발 환경과 개발 생산성의 향상 및 비용을 절감하기 위해서는 지속적인 데이터웨어하우스의 프로세스 개선이 필요하다. 프로세스 개선을 위해 본 연구에서는 생산성, 프로세스 품질, 데이터 품질의 영역에서 측정 지표를 제안하였다. 프로세스의 측정은 프로세스의 수행 과정에서 계획과의 편차를 발견하는 기초가 되며 프로세스 개선과 제어를 위한 기회를 제공하기 때문이다.

향후 연구과제는 제시된 측정지표를 기반으로 실제 개발 프로세스에 대해 정량적인 측정의 수행이다.

참 고 문 헌

[1] KliWon S., "Research about confidence verification of KPA question item through SEI Maturity Questionnaire's calibration and SPICE Level metathesis modeling," SERA03, 2003

[2] Immon W., Building the Data Warehouse, 3nd Ed., John Wiley & Sons Inc., 2002

[3] Kimball R., Reeves L., Ross M., and Thornthwaite W., The Data Warehouse LifeCycle Toolkit, John Wiley & Sons Inc., 1998

[4] 황선명, 엄희균, "소프트웨어 프로세스 측정을 위한 척도 설계 및 활용," 한국정보처리학회, 제12권 7호, p937~946, 2005

[5] 김찬수, "데이터의 구조적 품질관리 성숙도 모델 개발," 경희대학교, 박사학위논문, 2004

[6] 박종모, 조경산, "CMMI의 형상관리를 적용한 데이터웨어하우스 개발 프로세스의 개선," 한국정보처리학회, 제13권 4호, p625~632, 2006