(19)대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。Int. Cl.
G06F 9/00 (2006.01)
G06F 15/16 (2006.01)

(45) 공고일자 2007년06월14일
(11) 등록번호 10-0728612
(24) 등록일자 2007년06월08일

(21) 출원번호 10-2004-0059940
(22) 출원일자 2004년07월29일
심사청구일자 2005년06월01일

(65) 공개번호 10-2006-0011233
(43) 공개일자 2006년02월03일

(73) 특허권자 (주)위세아이텍
서울특별시 강남구 역삼동 708-33 파라다이스벤처타워 8층

(72) 발명자 이제동
서울시 관악구 신림동 1703-1번지 건영아파트 2동 606호

(74) 대리인 리앤목특허법인

(56) 선행기술조사문헌
KR1020000055986 A
KR1020010044398 A

심사관 : 노지명

전체 청구항 수 : 총 12 항

(54) 데이터이행에 있어서의 데이터 품질관리 방법

(57) 요약

데이터이행 기술에 있어서 데이터 품질을 관리하는 방법에 관한 발명. 본 발명은 델타(Delta)와 타겟(Target)의 2층(Layer) 구조를 갖는 데이터 이행에 있어서, 원천 데이터로부터 여과를 통해 데이터의 유효성을 검증한 후에 추출됨으로써 Delta 1 데이터가 생성되고, 이 Delta 1 데이터로부터 정제 과정을 거쳐 Delta 2 데이터를 생성하고, 이 Delta 2 데이터를, 검증 과정을 거쳐 Target으로 로딩하는 과정을 포함하는 것을 특징으로 함.

대표도

도 2

특허청구의 범위

청구항 1.

델타(Delta)와 타겟(Target)의 2층(Layer) 구조를 갖는 데이터 이행에 있어서,

원천 데이터로부터 여과를 통해 데이터의 유효성을 검증한 후에 Delta 1 데이터를 추출하고; 이 Delta 1 데이터를 정제하여 Delta 2 데이터를 생성하고; 이 Delta 2 데이터를 검증하여 Target으로 로딩하는 과정을 포함하되,

상기 원천 데이터의 유효성 검증은, 칼럼에 대한 의미성 여부를 평가하고 의미있는 칼럼에 대한 유효여부를 평가하는 단계; 각 데이터 요소가 명백하지 않고 의미없는 값을 갖고 있는지 확인하는 단계; 위의 값에 해당하지 않지만 유효한 영역 값 범위에서 벗어나는 값을 확인하는 단계를 포함하고,

상기 Delta 1 데이터의 정제는, 소정의 정제룰에 의하여 중복 데이터를 삭제하는 중복제거 단계; Null이 있는 모든 데이터를 삭제하는 Null값 제거 단계; 데이터 컬럼의 특정 자리의 패리티를 체크하는 Parity 체크 단계; 데이터의 형식(숫자형, 날짜형 등)을 체크하는 Data type 체크 단계; 데이터 컬럼의 내용 전체가 상기 형식을 따르는지 검사하는 Format 체크 단계; Sql 문에서 사용하는 특수문자가 있는 데이터의 경우에는 이를 필터링하는 제어문자 체크 단계; Dimension 타입의 테이블과 조인하여 Dimension에 없는 데이터는 새로운 값으로 업데이트하는 코드변환 단계를 포함하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 2.

삭제

청구항 3.

제1항에 있어서, 상기 데이터 요소의 의미없는 값은 Null값을 갖는 것, 공백을 갖는 것, 불명확한 기본값을 갖는 것을 포함하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 4.

삭제

청구항 5.

제1항에 있어서, 상기 정제룰은 아래 표와 같은 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

정제룰	설명
중복 제거	전체 삭제는 해당 중복 데이터를 모두 Delete 시킨다. 부분 삭제일 경우 조건으로 사용할 컬럼을 선택하고 Max 나 Min 의 함수를 이용해서 해당하는 데이터만 남기고 삭제한다.
Null값 제거	전체 제거는 Null이 있는 모든 데이터를 모두 Delete 시킨다. 대체 조건을 이용할 때에는 3가지의 옵션이 있다. Column 의 데이터 타입에 따라서 '빈문자열' , '숫자0' , '사용자 입력내용' 으로 Null Data를 Update 시킨다. 날짜형일 경우 Getdate() 함수를 사용할 수 있다. 빈문자열 제거 옵션은 빈문자열도 Null 값 처럼 제거 할 수 있게 한다.
Parity 체크	Parity 체크는 주민번호 뒷자리의 첫번째 자리처럼 컬럼의 특정자리의 Parity 를 체크 한다. 정제에는 모두 삭제하는 경우와 특정문자로 치환 할 수 있게 해주는 조건이 있다.
데이터 타입 체크	숫자형 , 날짜형식의 데이터인지 검사할 수 있다. 조건에 어긋날 경우 삭제하거나 'Getdate()' 같은 형식으로 치환할 수 있다.
Format 체크	Format 체크는 컬럼의 내용전체가 숫자 문자 등의 형식을 따르는지 검사한다.
제어문자 체크	Sql문에서 사용하는 특수문자가 있는 데이터의 경우에는 이를 필터링 한다. 다른 문자열로 치환하는 경우만 지원한다.
코드변환	Dimension 타입의 테이블과 조인하여 Dimension에 없는 데이터는 새로운 값으로 업데이트 한다.

청구항 6.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 데이터의 변환 처리 과정 검색과 적재 작업의 정확성을 검증하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 7.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 상기 타겟 데이터에서 사용하는 정규화에 따라서 데이터 중복을 제거하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 8.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 소스 데이터 중에서 무의미한 데이터를 식별하여 제거하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 9.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 두 개 이상의 소스시스템에서 하나 또는 그 이상의 타겟으로 추출하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 10.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 데이터가 변경되지 않고 소스에서 타겟으로 추출하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 11.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 매핑 테이블을 이용한 코드 변환인 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 12.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, If … Then … Else를 사용해서 추출하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 13.

제1항에 있어서, 상기 Delta 2 데이터에 대해 검증하는 단계는, 계산식에 의한 데이터 생성 과정을 거쳐 Target으로 로딩하는 과정을 포함하는 것을 특징으로 하는, 데이터이행에 있어서의 데이터 품질관리 방법.

청구항 14.

제1, 3, 5~13항 중 어느 한 항의 방법을 구현하는 컴퓨터 프로그램을 포함하고 있는 컴퓨터 기록매체.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 데이터이행에 관한 것으로서, 보다 구체적으로는 데이터이행 기술에 있어서 데이터 품질을 관리하는 방법에 관한 것이다.

데이터이행(data migration)이란 데이터 집합 간의 변환·이동을 의미하는 공지의 용어이다. 데이터이행 개념에는 데이터 추출 기술이 포함된다.
도1과 같은 일반적인 데이터 추출 기술(제품)은 데이터통합 및 데이터이행을 위한 제품으로 해외의 주요 IT 기업에 의해 주도되고 있다. 현재 해외의 추출기술(제품)은 구조적, 기능적, 사용적 측면에서 다음과 같은 수준을 보이고 있다.

- 구조적인 측면

일련의 작업들을 자동화하여 하나의 작업에서 일관된 사용자 화면으로 순차적으로 처리할 수 있음.

-기능적인 측면

수백개의 표준변환 함수들을 라이브러리 형태로 내장하고 있어서 대부분의 데이터 변환 룰 정의에 사용되고, 필요시 사용자가 새로운 함수를 작성/추가하여 변환기능을 쉽게 확장할 수 있음. 또한 체계적인 Repository관리 기능을 제공하는데, 서버의 한 구성 요소로서 Repository라는 장소를 할당하여, 정의한 JOB, 메타 데이터, 프로세스, 변환 룰, 함수 등을 그래픽하게 통합, 유지, 관리함으로써 언제라도 재사용 될 수 있도록 함. 이와 함께 JOB수행의 통제 및 감시 기능을 제공하는데, 생성한 JOB들의 수행절차(즉시/예약수행 등)를 정의하고 실행과정을 실시간으로 감시할 수 있으며 JOB수행 인자를 설정하여 사용할 수 있음

- 사용 측면

여러 명의 개발자가 동시에 서로 다른 작업을 수행할 수 있도록 하는 다중 사용자 환경을 제공하며, 새로운 기능의 추가나 개선시, Package Installer를 이용하여 운영에 지장을 초래하지 않고 해당 모듈만 간단히 업그레이드할 수 있게 함

이와 같이, 현재 추출기술(제품)을 위주로 한 시장이 형성되어 있다. 이 데이터추출 기술은 데이터이행 작업을 위한 기술에 포함되는 것이므로 이하에서는 보다 포괄적으로 데이터이행 기술에 대해 설명한다.

(1) 데이터이행시 고려 사항

- 업무 중단 최소화

업무 중단 최소화를 위한 데이터 전환시 서비스 방안을 마련하여야 함

- 데이터이행 소요기간

방대한 자료량, 데이터의 분포, 분리, 합병 및 로드의 절대적인 시간을 감안하여야 함

- 철저한 자료정비

대량의 자료 이동에 따른 장애를 방지하여야 하며, 자료의 정합성을 고려한 이행이어야 함

- 위험 최소화

단기간 내 이행에 따른 오류를 최소화하고, 장애 유형별 해결방안을 마련하여야 함

(2) 데이터이행 기술(제품)의 필요성

데이터이행에 있어서 전문기술(제품)없이 수행하는 경우 다음과 같은 위험요소가 존재한다.

- 단기간에 다수의 업무영역을 프로그램에 의해 동시에 통합하므로 중요한 데이터품질 점검이 누락될 수 있음

- 다수의 프로그래머가 작업함에 따라 개인능력 차에 따라 로직 오류 및 누락 가능성이 존재하며, 프로그램 디버깅, 검증에 많은 노력 소요됨

- 이행규칙 변경시 프로그램 로직을 직접 수정하므로 정확성과 신속성이 떨어지고 프로그램의 버전 관리가 어려워짐

- 대량의 데이터 이행시 에러위치와 에러원인을 신속하고 정확하게 파악하여 조치하는데 한계가 있음

이러한 위험요소를 해결하기 위하여 다음과 같은 대책이 요구된다.

- 데이터이행 전 과정을 관리

- 데이터 이행 전 소스 데이터의 품질 검증

- 이행 완료된 데이터의 품질 검증

- 이행 프로그램에 대한 효율적인 디버깅

- 데이터 이행 과정에서 오류 발생 시 신속한 원인파악

- 이행프로그램에 대한 버전관리

일반적인 추출 작업과 데이터이행 작업에는 다음 표1과 같은 차이점이 존재한다. 표1에서 ●는 ○보다 더 높은 수준이 요구되는 기능을 의미하며, ×는 필요없는 기능을 의미한다.

[표 1]

구분		추출	이행(Migration)	비고
속성		반복적	1회성	
주요 목적		데이터 통합	데이터 이행	
주요 분야		데이터웨어하우스 구축	애플리케이션 간 데이터 이관	
요구기능				
기본기능	추출	●	●	
	정제	●	○	
	변환	●	○	
	적재	●	●	
추가 기능	Job Scheduling	●	×	
	Meta 관리	●	○	
	품질 관리	○	●	
	이력 관리	○	○	
	DSS Tool과의 통합	○	×	

이와 같이 일반적인 추출기술(제품)과는 달리 데이터이행 기술(제품)은 요구되는 기능이 적은 반면에, 품질관리와 같은 일부 기능에 있어서는 더 높은 수준의 기술(제품)을 요구하고 있다.

또한 데이터이행 작업은 1회성으로 고객들이 기술(제품)자체에 많은 비용을 지불하지 않는 상황이다.

따라서 데이터이행을 위한 기술(제품)은 필수적인 기능과 데이터이행에 특별히 요구되는 기능만을 구현할 수 있는 저렴한 기술(제품)을 개발하는 것이 관건이다.

이와 같은 기술(제품)을 통해 기업이 보유한 다양하고 방대한 데이터를 활용 가능한 지적 자산으로 전환하여, 마케팅과 의사결정에 활용하고, 이를 통해 경쟁력을 확보하고 유지할 수 있게 도와줄 수 있을 것이다.

발명이 이루고자 하는 기술적 과제

본 발명의 목적은 데이터이행 기술이 갖추어야 할 기본적인 기능(추출/정제/변환/적재)을 포함하며 데이터이행 작업에 특화된 데이터 품질 관리 기능을 위주로, 실제 산업에 적용 가능한 데이터이행 제품화를 위한 방법에 관한 것이다.

발명의 구성

종래 기술과 본 발명의 차별성

본 기술(제품)은 데이터이행 기술(제품)이 갖추어야 할 기본적인 기능(추출/정제/변환/적재 등)을 포함하며 데이터이행 작업에 특화된 데이터 품질 관리 기능을 위주로, 실제 산업에 적용 가능한 데이터이행 기술(제품)로 개발되었다.

또한, 본 기술(제품)은 데이터이행에 있어서의 위험요소를 최소화하고 품질관리를 강화하기 위해 아래 표2와 같은 체계적인 검증 방법론을 바탕으로 개발되었다.

[표 2]

구분	검증방법	검증유형
도메인 분석	각 컬럼이 가질 수 있는 영역 값(domain value)를 확인하고 각 값의 발생 빈도수 확인 영역 값의 업무적 적합성 평가	코드 인스탄스 값의 Range 속성의 범위 값
데이터 유효성 검증	컬럼에 대한 의미성 여부 평가와 의미 있는 컬럼에 대한 유효여부 평가 각 데이터 요소가 명백하지 않고 의미 없는 값을 가지고 있는지 확인 - Null 값을 가지는 것 - 공백을 가지는 것 (수치 값일 때는 0) - 불명확한 기본값을 가지는 것 위의 값에 해당하지 않지만 유효한 영역 값 범위에서 벗어나는 값 확인	데이터의 유효성 속성의 Numeric 값
구조적 무결성 검증	데이터의 구조적인 무결성 보장 레코드의 구조와 레코드 간의 관계에 따른 데이터 품질을 검사 구조적인 형태 - 기본키(Primary Key)와 속성간의 1:1 관계 - 외부키(Foreign Keys)의 참조 무결성 - 관계의 기수성(1:N, then N) 기본키(Primary Key)의 유일성 테스트	Key 검증 검사의 정합성
업무규칙 반영	데이터의 논리적인 업무 규칙 검사 특정한 업무 규칙에 대한 품질 검사 업무 규칙의 4가지 형태 - 유용한 값의 조합 규칙(Valid-value combinations rules) - 연산 규칙(Computational rules (Equation, Set)) - 시간 규칙(Time rules (Range, Sequence)) - 조건 규칙(If … Then … Else rules)	속성의 exist 형 참조무결성 검증 속성의 count 형 무결성 검증 속성의 연산 형 무결성 검증 속성의 최소값 형 무결성 검증 속성의 최대값 형 무결성 검증

변환 로직 검증	데이터의 변환 처리 과정 검색과 적재 작업의 정확성을 검증 - 타겟 데이터에서 사용하는 정규화에 따라서 데이터 중복 제거 - 소스 데이터 중에서 무의미한 데이터를 식별 후 제거 - 두개 이상의 소스시스템에서 하나 또는 그 이상의 타겟으로 추출 - 데이터가 변경되지 않고 소스에서 타겟으로 추출 - 메핑 테이블을 이용한 코드 변환 - If … Then … Else를 사용해서 추출 - 계산식에 의한 데이터 생성	각 속성의 data exception 방지 속성들 간의 종속 관계

본 발명에 따른 데이터이행 방법의 개념은 다음과 같다.

-데이터이행의 전체 흐름(Work Flow)은 델타(Delta)와 타겟(Target)의 2 층(Layer) 구조를 갖고 있다.

-델타층에 있어서는, 원천 데이터(source)에서 추출(extraction)되어 Delta 1이 생성된다.

-추출된 Delta 1로부터 정제(Cleansing) 과정을 거쳐 Delta 2를 생성하게 된다.

-Delta 2에서 각종 변환 로직 검증 과정을 거쳐 Target으로 로딩(Loading)한다.

도2는 데이터 이행에 있어서의 특화된 데이터품질관리 기술의 작업 흐름도이다. 흐름도에서 보이는 바와 같이 데이터 추출에 대한 기본적인 기능인 ETL 단계를 가지고 있다.

즉, 다양한 소스시스템(Source System)으로부터 필요한 데이터를 추출(Extraction)하여 변환(Transformation) 작업을 거쳐 타겟 시스템(Target System)으로 전송 및 로딩(Loading)하는 모든 과정을 포함하고 있다.

그러나 이러한 일련의 작업만으로 완벽한 데이터 이행이 되었다고는 볼 수는 없다.

따라서 본 발명의 목적인 데이터 이행에 있어서의 데이터 품질을 검증하는 방편으로 소스 데이터를 정제함은 물론 이행 완료된 데이터의 품질 검증까지의 전 과정을 관리하도록 한다.

데이터는 다양한 형태의 소스로 존재하며, 이에 따라 데이터 이전 방법 또한 다양한 형태를 가지고 있다. 이러한 소스(원천 데이터)는 여과(Filtering)를 통해 데이터의 유효성을 검사하여 추출되게 된다. 이러한 데이터 유효성 검증은 다음과 같은 방법으로 이루어지게 된다.

- 칼럼에 대한 의미성 여부 평가와 의미있는 칼럼에 대한 유효여부 평가

- 각 데이터 요소가 명백하지 않고 의미 없는 값(즉, Null 값을 가지는 것, 공백을 가지는 것 (수치 값일 때는 0), 불명확한 기본값을 가지는 것)을 가지고 있는지 확인

삭제

삭제

삭제

- 위의 값에 해당하지 않지만 유효한 영역 값 범위에서 벗어나는 값 확인.

이러한 소스데이터가 유효성 검증과 같은 여과과정을 거쳐 추출되어 Delta 1이라는 데이터가 생성된다. 그러나 여과를 통해 추출된 데이터(Delta 1)들을 전적으로 신뢰할 수 있는 것은 아니다.

여과(Filtering)된 데이터소스라 할지라도 데이터의 정확도나 신뢰도는 떨어질 수 밖에 없다. 따라서 데이터 정제(cleansing) 과정을 통해 불필요한 데이터를 제거하게 된다.

Delta 1에서 정제 룰(rule)에 따라 정제(cleansing) 과정을 거치게 된다. 이러한 과정에는 데이터의 통합 및 코드 통합 등도 포함되며, 중복제거, Null 값 제거, Parity 체크, Data type 체크, Format 체크, 제어문자, 코드변환과 같은 7가지 작업을 통해 소스데이터를 정제하게 된다.
한편, 본 발명에서 상기 Cleansing(정제) 단계는 변형가능하다. 데이터 정제를 하기 위한 방법으로는 다음과 같은 7가지 작업을 통해 소스데이터를 정제하게 된다 - 중복제거, Null 값 제거, Parity 체크, Data type 체크, Format 체크, 제어문자 체크, 코드변환.
각 작업들은 소스 데이터의 성격에 따라 정제 룰을 추가/편집/삭제를 할 수 있다. 또한 기본적으로 제공되는 정제 룰 이외에도 향후 추가적인 룰이 적용될 수 있다. 각각의 작업에 대한 설명은 다음과 같다.
- 중복 제거
전체 삭제는 해당 중복 데이터를 모두 Delete 시킨다. 부분 삭제일 경우 조건으로 사용할 컬럼을 선택하고 Max 나 Min 의 함수를 이용해서 해당하는 데이터만 남기고 삭제한다.
- Null 값 제거
전체 제거는 Null이 있는 모든 데이터를 모두 Delete 시킨다. 대체 조건을 이용할 때에는 3가지의 옵션이 있다. Column 의 데이터 타입에 따라서 '빈문자열', '숫자0', '사용자 입력내용'으로 Null Data를 Update 시킨다. 날짜형일 경우 Getdate() 함수를 사용할 수 있다. 빈문자열 제거 옵션은 빈문자열도 Null 값처럼 제거할 수 있게 한다.
- Parity 체크
Parity 체크는 주민번호 뒷자리의 첫번째 자리처럼 컬럼의 특정자리의 Parity 를 체크 한다. 정제에는 모두 삭제하는 경우와 특정문자로 치환 할 수 있게 해주는 조건이 있다.
- 데이터 타입 체크
숫자형, 날짜형식의 데이터인지 검사할 수 있다. 조건에 어긋날 경우 삭제하거나 'Getdate()' 같은 형식으로 치환할 수 있다.
- Format 체크
Format 체크는 컬럼의 내용전체가 숫자 문자 등의 형식을 따르는지 검사한다.
- 제어문자 체크
Sql문에서 사용하는 특수문자가 있는 데이터의 경우에는 이를 필터링한다. 다른 문자열로 치환하는 경우만 지원한다.
- 코드변환
Dimension 타입의 테이블과 조인하여 Dimension에 없는 데이터는 새로운 값으로 업데이트 한다.

삭제

이처럼 Delta 1 데이터는 일련의 가공과정을 거쳐 분석 가능한 형태로 최종 변환되어 Delta 2 데이터를 생성하게 된다.

이러한 Delta 2 데이터는 최종 변환된 데이터이기 때문에 Target(목표) 시스템 등에 적재(Loading)된다. 그러나 Delta 2 데이터라도, 적재되는 과정에 데이터 중복 및 오류가 발생할 수 있기 때문에 다음과 같은 검증 방법을 통해 데이터를 적재(Loading)한다.

- 데이터의 변환 처리 과정 검색과 적재 작업의 정확성을 검증

- 타겟 데이터에서 사용하는 정규화에 따라서 데이터 중복 제거

- 소스 데이터 중에서 무의미한 데이터를 식별 후 제거

- 두 개 이상의 소스시스템에서 하나 또는 그 이상의 타겟으로 추출

- 데이터가 변경되지 않고 소스에서 타겟으로 추출

- 매핑 테이블을 이용한 코드 변환

- If … Then … Else를 사용해서 추출

- 계산식에 의한 데이터 생성

위에서 설명한 바와 같이 이러한 일련의 데이터 이행에 관한 작업을 거쳐 목표했던 완벽한 데이터이행을 이루게 된다.

이밖에도 도메인 분석을 통해 각 컬럼이 가질 수 있는 영역 값(domain value)을 확인하고 각 값의 발생 빈도수 확인 및 영역 값의 업무적 적합성을 평가하게 되며, 구조적 무결성 검증을 통해 데이터의 구조적인 무결성 보장 및 레코드의 구조와 레코드 간의 관계에 따른 데이터 품질 검사, 기본키(Primary Key)의 유일성 테스트를 하게 된다. 또한 업무규칙에 대한 반영을 통해 데이터의 논리적인 업무 규칙 검사와 특정한 업무 규칙에 대한 품질을 검사하게 된다.

이와 같은 일련의 차별화된 검증방안들을 통해서 데이터이행에 있어서의 데이터품질을 강화하게 된다.
이상에서 설명한 방법은 물론, 컴퓨터 프로그램으로 구현할 수 있다. 이렇게 본 발명의 사상을 구현하는 컴퓨터 프로그램을 기록한 컴퓨터 기록매체도 또한 본 발명의 권리범위에 포함된다.

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

삭제

발명의 효과

본 발명에 따른 데이터이행 방법을 통해 데이터를 빠른 속도로 옮기고 재편성하는 작업이 가능해진다. 따라서 데이터 이행 작업에 관계된 업무 중복 및 비용과 인력을 줄일 수 있게 된다. 또한 일반추출 기술(제품)의 고비용으로 인해 활성화되지 못했던 데이터이행 시장에 외국산보다 최고 5배 이상 저렴하게 기술(제품)을 공급함으로써 시장 창출효과를 가져올 수 있으며, 기업들은 본 발명에 따른 기술(제품)을 저비용으로 사용함으로써 방대한 정보자산의 효율적인 관리를 통해 경쟁력 향상에 도움이 될 것이다.

분야	파급효과
기술(제품)적 측면	컨설팅 지식의 IT 기술(제품)화
경제/산업적 측면	기업의 데이터이행 작업의 비용 절감(아래 참고) 일반추출 기술(제품)의 고비용으로 활성화되지 못한 데이터이행 시장의 창출 효과 외산 기술(제품)의 시장점유율을 국내 기술(제품)로 대체할 수 있는 수입대체 효과(아래 참고)

- 비용절감 사례

금융권 데이터이행 사례를 통해 볼 때, 일반 추출기술(제품)을 사용 시 60MM(Man/Month)에서 20MM 줄어든 40MM가 소요된다.

즉, 데이터이행 기술(제품)을 사용 할 때, 일반 추출기술(제품) 보다 더 높은 비용절감 효과와 함께 작업의 질(데이터 품질 보증)을 높일 수 있을 것이다.

공정	분석	설계	개발	테스트	데이터 인도	계
전부 수작업	12MM	14MM	21MM	7MM	6MM	60MM
일반 추출도구 이용	9MM	10MM	10MM	6MM	5MM	40MM
MM 절감효과	3MM	4MM	11MM	1MM	1MM	20MM

- 수입대체 및 수출 효과

세계최고 수준의 데이터이행 기술(제품)을 외국산보다 최고 5배 이상 저렴하게 공급하여, 국내 기업이 사용하는 데이터이행 기술(제품)의 국산화를 꾀할 수 있으며, 향후 기능 보강을 통하여 외국으로 기술(제품) 수출을 모색할 수 있을 것이다.

예) 외산 제품 : 최소 1억원 이상, 본 기술(제품) : Copy 당 2,000만원

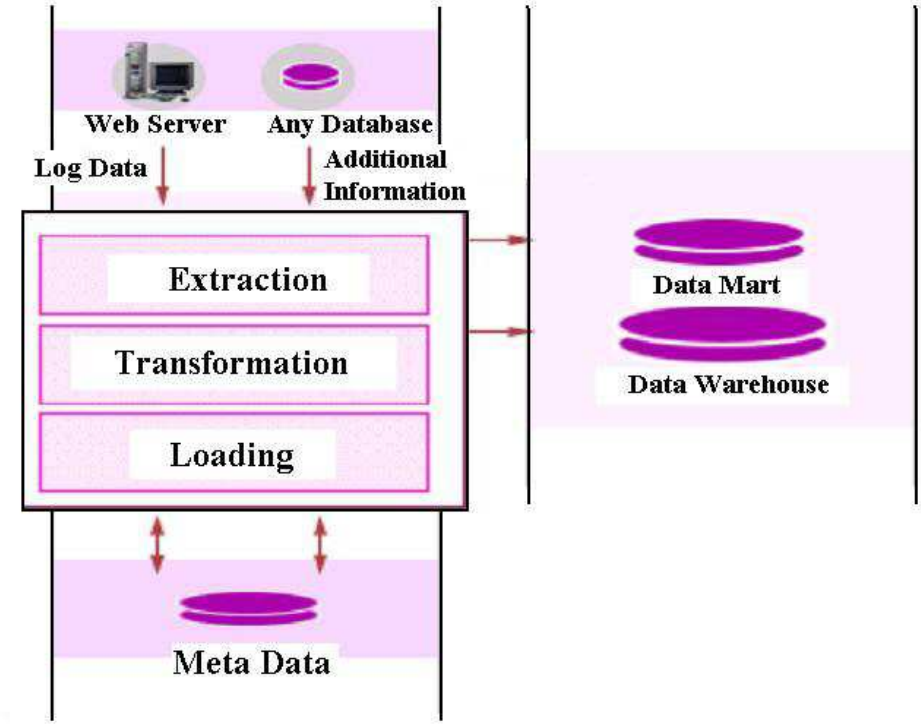
도면의 간단한 설명

도1은 종래의 데이터추출 기술의 개요도.

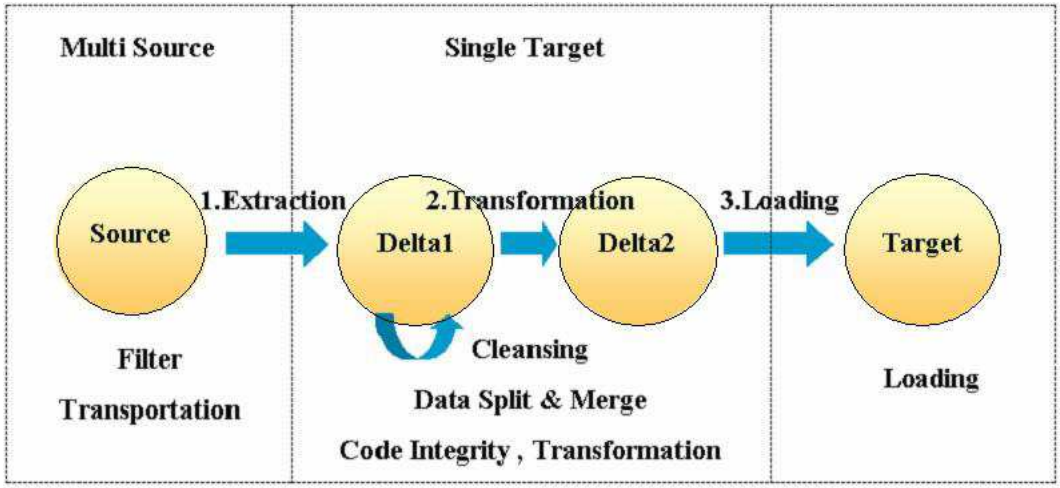
도2는 본 발명에 따른 데이터이행 기술의 개요도.

도면

도면1


Web Server
Any Database
Log Data
Additional Information
Extraction
Transformation
Loading
Data Mart
Data Warehouse
Meta Data

도면2


Multi Source
Single Target
Source
1.Extraction
Delta1
2.Transformation
Delta2
3.Loading
Target
Filter
Transportation
Cleansing
Data Split & Merge
Code Integrity , Transformation
Loading