Fact constellation 스키마와 트리 기반 XML 모델을 적용한 실험실 레벨의 단백질 데이터 통합 기법

박 성 희[†] · 이 영 화[††] · 류 근 호[†††]

요 약

유전자 및 단백질간의 복잡한 상호작용에 의해 기능이 결정되는 생명정보 데이터의 특성으로 인하여 생명정보 데이터 분석을 위해서는 이질적인 데이터를 통합적으로 분석할 수 있는 통합시스템이 요구된다. 따라서 이 논문에서는 생물학 실험실 레벨에서 단백질 구조 관련 데이터를 통합할 수 있도록 XML 모델기반에 웨어하우스 미디에이터 통합시스템을 제안한다. 제안 시스템은 fact constellation 모델을 기반하여 이질적인 소스에 대한 통합 모델링을 진행하고 통합 스키마를 XML 스키마로 변환하여 유지한다. 또한 통합 데이터베이스에 포함된 소스 데이터의 변경 및 출처에 대한 추적 관리를 위해 데이터의 점진적 갱신방법과 서열에 대한 버전관리를 이용한다. 실제로 이 시스템을 단백질 구조(PDB), 서열(Swiss-Prot)과 도메인 분류데이터(CATH) 통합에 적용한 통합 모델링 과정을 보여준다.

An Approach for Integrated Modeling of Protein Data using a Fact Constellation Schema and a Tree based XML Model

Sung Hee Park[†] · Rong Hua Li[††] · Keun Ho Ryu[†††]

ABSTRACT

With the explosion of bioinformatics data such proteins and genes, biologists need a integrated system to analyze and organize large datasets that interact with heterogeneous types of biological data. In this paper, we propose a integration system based on a mediated data warehouse architecture using a XML model in order to combine protein related data at biology laboratories. A fact constellation model in this system is used as a common model for integration and an integrated schema is translated to a XML schema. In addition, to track source changes and provenance of data in an integrated database employ incremental update and management of sequence version. This paper shows modeling of integration for protein structures, sequences and classification of structures using the proposed system.

키워드 : 생명정보학(Bioinformatics), 통합 시스템(Integration System), 단백질 데이터 모델링(Protein data Modeling), XML, BSML

1. 서 론

HGP의 초안 발표 이후 유전자 및 단백질의 기능을 밝히기 위한 유전체 및 단백체학 연구가 활발해졌다. 이러한 생물학 연구를 위해서는 유전체 서열 및 유전자 발현, 단백질 서열 및 구조, 단백질 상호작용 및 대사 경로, 계통분류와 기능을 포함하는 문헌 정보 등을 통합적으로 분석할 수 있는 데이터베이스와 분석 도구가 요구된다. 특히 단백질 구조 정보를 분석하여 기능이 알려지지 않은 단백질의 기능을 예측하는 구조 유전체학(Structural Genomics)을 위해서는 데이터의 통합적 분석을 지원할 수 있는 통합 데이터 관리 시스템의 개발이 필수적이다.

생명정보 통합시스템에 대한 연구는 데이터베이스 통합 시스템 방식에 따라 데이터를 중앙 집중된 저장소에 통합하는 데이터 웨어하우스방식과 연합데이터베이스 시스템 구조를 이용한 형성 뷰(materialized view)를 이용한 방식으로 나누어 진다. 데이터 웨어하우스 방식[1-3]은 질의 실행시 소스 데이터베이스에 질의를 변환하지 않아 질의처리가 쉽다. 또한 데이터 마이닝에서 이용할 수 있는 데이터 분석 프로시저를 이용할 수 있는 장점이 있다. 반면에 소스 데이터 변경을 웨어하우스에 반영하여 소스 데이터와 일치성 유지를 위한 refresh 비용이 높다는 단점이 있다.

연합 데이터베이스 구조 기반 형성 뷰 통합 방식[4-6]은 소스 데이터베이스모델에 적합한 질의 변환 및 질의 결과의 통합에서 서로 다른 데이터 모델을 매핑할 수 있는 공통 모델이 요구된다. 이로인해 분산된 소스 데이터베이스로

※ 이 논문은 한국과학기술기획평가원의 국책 생물정보학연구개발사업 연구비 지원으로 수행되었음.
† 준 회 원 : 충북대학교 대학원 전자계산학과
†† 준 회 원 : 연변대학교 컴퓨터과학과
††† 종신회원 : 충북대학교 컴퓨터과학과 교수
논문접수 : 2003년 10월 10일, 심사완료 : 2004년 1월 10일

부터 질의 결과 통합이 지연될 수 있다.

단백질 구조 정보 통합을 위한 데이터들은 기존 데이터와 다른 다음과 같은 특성을 갖는다.

- 데이터 종류가 다양하고 이질적인 포맷을 갖으며 데이터가 분산되어짐
- 데이터 자체가 생물학적 변이를 포함하고 이러한 데이터가 실험을 통해 생산되기 때문에 데이터의 상태가 항상 변화하고 유동적임
- 데이터 사이의 연관관계가 많아 데이터 구조 및 관계가 복잡. 특히 이러한 복잡한 관계는 계층구조이거나 순환적 구조로 나타남
- 실험을 통해 얻어진 원시 데이터에 대한 해석을 위한 추가적인 주석 정보를 가지며 여러 단계의 분석과정을 통해 많은 유도된 데이터가 창출
- 다양한 생명정보 사용 및 분석을 위해 서로 다른 뷰와 포맷 요구

생명정보처럼 복잡한 스키마를 갖는 데이터에 대한 통합을 위한 통합 모델링에 관계형 데이터 모델은 적합하지 않다. 특히 전 페이지에 기술한 바와 같이 단백질 구조를 포함하는 생명정보 데이터는 일반적으로 반 구조적(Semi-structured data) 데이터의 특성과 비정형 데이터 특징을 보인다.

최근에 XML 기술을 적용하여 XML 미디에이터[7, 8] 구조를 이질적인 소스 데이터베이스의 통합에 이용하여 XML 모델을 통합을 위한 공통 모델로서 이용하고 있다. 특히, XML 모델 및 질의언어는 복잡한 생명정보 데이터 모델링과 질의에 적합한 계층적 구조와 패턴 매핑 질의를 지원한다. XML 미디에이터는 기존 관계형 데이터에 대한 XML 뷰를 제공하는 장점을 갖는다.

이 논문에서는 생물학 연구실 수준에서 단백질 구조 관련 데이터베이스를 통합하기 위해 트리 기반의 XML 모델을 통합 모델로서 이용하여 XML 미디에이터 웨어하우스 통합 시스템 구조를 제안하고 이 시스템을 단백질 구조(PDB), 서열 및 기능적 주석(Swiss-Prot)과 도메인 분류데이터(CATH) 통합에 적용을 보여준다. XML 미디에이터 웨어하우스 통합 시스템은 단백질 구조 관련 정보를 데이터 웨어하우스 기법으로 통합하고 분석 및 검색을 위한 계층 및 복합질의는 XML 미디에이터 시스템을 이용한다. 이를 위해 기존의 연구된 XML 미디에이터 시스템을 기반하고 관계형의 Fact constellation 스키마를 트리 구조에 기반한 XML 스키마로 변환을 수행한다.

단백질 구조 정보를 통합한 데이터웨어하우스의 소스 데이터베이스의 갱신 및 변경을 관리하기 위해 플랫파일을 정기적으로 다운받아 변경된 정보만을 점진적으로 갱신 및 변경한다. 데이터의 변경정보 및 상호 참조 정보를 관리하기 위해서는 변경정보에 대한 버전을 관리한다. 이렇게 함으로써 단백질 구조 데이터 관리를 위해 반 구조 데이터 저장관리 기법과 계층 및 순환적 질의를 포함하는 반 구조 데이터에 대한 질의를 활용할 수 있다.

이 논문의 구성은 다음과 같다. 2장에서는 생물정보 통합을 위한 기존 연구를소개한다. 3장에서는 제안하는 XML 미디에이터의 웨어하우스시스템의 구조에 대해 기술한다. 4장과 5장에서는 통합하려는 단백질 구조 소스 정보와 통합 모델링에 대해서 상세히 설명한다. 6장에서는 시스템의 구현결과 및 기존 시스템과 비교평가하고 7장에서 결론을 맺는다.

2. 관련 연구

이 절에서는 현재까지 생명정보학 데이터 통합을 위해 제시된 기존 연구로써 링크기반, 미디에이터기반과 데이터웨어하우즈기반 통합시스템을 살펴본다.

2.1 링크기반 통합기법

링크기반 통합기법은 구현하기 쉽고 현재 가장 많이 사용하고 있다. 이러한 통합기법을 사용한 시스템들은 플랫파일 데이터베이스들을 WWW 링크와 인덱스를 이용하여 연결하였다. 이러한 링크 솔루션들은 실질적으로 시스템들을 통합하지 않기 때문에 사용자에게 단일 엔트리의 액세스를 제공하고 구현하기 쉬운 장점이 있다. 단점으로는 ① 각 데이터소스의 데이터들에 대한 인덱스와 WWW 링크들은 유지되지 않으며 에러가 발생할 수 있다. ② 브라우징과 키워드 검색만 가능하고 ad-hoc 질의를 지원하지 않는다. ③ 또한, 진일보의 데이터의 분석을 하려면 사용자의 많은 수작업이 필요하다. 이런 시스템들의 예로는 SRS[9], DBGET/LinkDB, Entrez 등이 있다.

2.2 미디에이터기반 통합기법

미디에이터기반 통합기법은 데이터 소스에 대한 통합 뷰를 생성하고 뷰에 대한 통합 질의를 수행한다. 사용자의 뷰에 대한 통합 질의는 각 해당하는 데이터 소스들의 질의로 변환하여 결과를 반환하며 그 결과를 통합하여 사용자한테 통합 결과를 보여준다. 이러한 시스템들의 장점으로는 ① 사용자한태 하나의 질의 인터페이스를 제공한다. ② 투명하게 또 자동적으로 통합 데이터 소스들을 액세스한다. ③ 최신의 데이터로 구성된 질의 결과를 반환한다. 그러나 분산 질의들이 인터넷을 통하여 수행되기 때문에 응답시간이 길고 또 데이터의 재조직이 어렵다는 단점이 있다. 이러한 시스템의 예로는 OPM[4], CPL/Kleisli[5, 1], TAMBIS[10]등이 있다. 그 외에도 MIX[11], TSIMMIS[8]와 XPERANTO[7] 반구조 데이터 통합 시스템들이 있다. 이러한 시스템들은 통일 질의언어를 정의하고 통합 뷰를 설계하고 소스 뷰를

이용하여 질의를 분해하고 실행한다.

2.3 데이터웨어하우스기반 통합기법

데이터웨어하우스기반 통합기법은 소스 데이터를 물리적으로 저장하고 통합 스키마를 가지고 주기적으로 모든 소스 데이터를 데이터 웨어하우스에 로딩하고 관리하는 기법이다. 이러한 통합기법은 ① 물리적으로 존재하는 하나의 통일된 데이터베이스를 액세스하기 때문에 질의 최적화가 로컬에서 수행할 수 있다. ② 질의 처리시데이터 소스와 교류 대기시간이 필요 없고 네트워크 연결에 많이 의존하지 않기 때문에 시스템 신뢰도가 좋다. ③ 소스 데이터의 에러를 제거한 정제된 통합된 데이터를 가지므로 이 통합 데이터를 기반한 새로운 생물학적 지식을 얻을 수 있다. 반면에 다음과 같은 단점을 포함하고 있다. ① 많은 데이터들을 물리적으로 유지하는데 많은 저장공간 비용이 필요하다. ② 데이터의 업데이트를 웨어하우스에 반영을 위해 많은 비용이 소모되고 이에 대한 기법이 요구된다. 웨어하우스 통합 시스템으로는 GUS(Genomic Unified Schema)[1]와 IGD(Integrated Genome Databases) 뿐만 아니라 웨어하우스 미디에이터 시스템[12] 등이 있다.

3. XML 미디에이터 웨어하우스 통합 시스템 구조

이 장에서는 XML 미디에이터와 웨어하우스 통합 방식을 결합한 XML 미디에이터 구조를 설명한다. XML 미디에이터 웨어하우스 통합 시스템은 단백질 구조 관련 소스 데이터를 통합한 Fact constellation 스키마를 이용한 데이터 웨어하우스, 관계형 스키마를 XML 스키마로 변환하는 스키마 변환기, 사용자 질의를 분석하여 관계형질의와 XML 질의로 분류하는 통합 질의분석기, 질의 결과를 XML문서로 생성하는 결과생성기와 결과를 특정한 포맷으로 변환을 하기 위한 포맷 변환기로 구성된다. 각 구성에 대한 상세한 설명은 아래에서 설명한다.

3.1 접근 방식

단백질 구조 정보는 NMR이나 craystallograpy와 같은 실험에 의해 데이터가 획득되고 실험에 따라 동일한 시료라도 데이터가 조금씩 다르다. 또한 실험을 통해 얻은 3차원 좌표 데이터만으로 기능이나 구조에 대한 상세한 정보를 얻을 수 없다. 따라서 구조를 얻은 실험에 대한 상세한 주석정보와 참고문헌과 같은 정보가 추가되거나 다른 데이터베이스 링크정보가 추가된다. 뿐만 아니라 분석을 통해서 서열 패밀리, 구조의 유사성 및 도메인 정보가 새롭게 얻어진다.

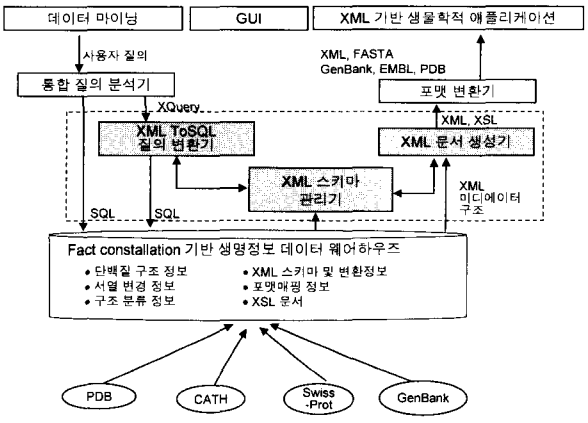
이로 인해 데이터베이스의 스키마 및 데이터의 변경이 자주 발생한다. 이러한 추가적인 정보를 반영하기 위해서는 통합 뷰에 대한 물리적 사본을 유지하지 않는 뷰 통합 방식보다 데이터 웨어하우스 기반 통합 방식이 새로운 데이터의 추가 저장에 더 적합하다. 또한 웨어하우스 방식을 적용하면 분석을 위해 데이터에 포함된 에러가 제거되고 정제되기 때문에 데이터 마이닝이나 분석 프로그램에서 활용이 쉽다.

XML은 기존 데이터와는 다르게 스키마가 고정되지 않고 빠른 변경이 요구되는 정보의 표현이 가능하며 XML 데이터 모델은 주로 트리와 그래프 구조를 갖는다. 또한 엘리먼트가 다른 엘리먼트에 중첩된 계층적 구조를 가지며 이러한 계층 구조의 표현과 질의를 위한 패스 표현과 패턴 질의를 포함하는 XML 질의 언어가 사용된다. 이러한 특징으로 기존의 관계형 및 객체지향 데이터 모델과 질의언어는 XML에 대한 데이터 모델과 질의 언어의 부분 집합의 포함관계를 갖는다. 따라서, 여기서는 단백질과 같은 생명데이터 통합을 위한 공통 모델과 질의언어로써 XML모델과 질의언어를 사용한다. 이를 위해 XML 미디에이터 구조[7]를 데이터웨어하우스와 결합하여 구축된 데이터웨어하우스에 대한 XML 가상 뷰를 제공한다. 이러한 XML 뷰에 대하여 XML질의를 수행하며 기존 관계형의 데이터웨어하우스에 저장된 통합 데이터를 XML문서로 생성하여 제공한다.

즉, 데이터를 저장하는 물리적인 저장소로써 관계형 데이터베이스를 이용하고 데이터를 관리하기 위한 모델과 질의언어로써 XML을 이용함으로써 XML의 장점을 수용한다.

3.2 시스템 구조

XML 미디에이터웨어하우스 시스템은 (그림 3.1) 같이 단백질 구조를 통합 스키마와 데이터를 통합한 데이터 웨어하우스, 통합 질의 처리기, 사용자 및 생명정보 응용프로그램의 3계층으로 크게 구분된다. 통합 질의 처리기는 XML 미디에이터의 질의 처리기를 확장한 구조로 XML 질의를 기존 관계형 질의로 변환을 위한 XMLTOSQL질의 변환기, 질의결과를 통합하고 XML문서를 생성하는 XML태그 생성기, 관계형 스키마를 XML스키마 변환하고 변경을 관리하는 XML 스키마 관리기와 포맷변환기가 추가된 형태이다.



(그림 3.1) XML 미디에이터 웨어하우스 통합 시스템 구조

- 데이터 웨어하우스
  단백질 구조 분석 관련 데이터를 관계형 Fact constellation 통합 스키마와 통합 데이터 외에 데이터의 변경 이력 및 데이터의 출처에 대한 추적 정보, 관계형 스키마를 변환한 XML 스키마, 질의 결과를 다른 포맷으로 변환을 위한 XML과 다른 포맷과의 매핑 정보를 포함한다.
  데이터 웨어하우스는 단백질 관련 소스 데이터베이스에서 플랫파일을 다운받고 이것으로부터 관련 데이터를 추출하여 원시 소스 데이터베이스를 구축한다. 이러한 원시 소스 데이터베이스에 대한 형성 뷰를 이용해서 Fact constalltion의 통합 스키마를 생성한다.
  스키마 통합 수준에 따라 엄격한 통합 데이터베이스를 구축하거나 로컬 스키마의 합집합을 통한 공통 스키마를 생성하는 느슨한 통합 데이터베이스 구축 방법이 있다. 생명정보 데이터는 스키마의 변경이 자주 발생하므로 엄격한 통합 데이터베이스를 구축할 경우 스키마 변경의 반영이 복잡해진다. 여기서는 두 가지 방법을 절충하여 공통 스키마로 변환하고 의미적 스키마 매칭과 스키마 정합단계까지 수행하여 통합 스키마를 생성한다.

- XML 스키마 관리기
  관계형 스키마에 대한 XML 뷰를 제공하고 질의를 수행하기 위해 관계형 스키마를 XML 스키마로 변환하기 위한 모듈이다. XML 스키마 관리기는 Cot[13] 알고리즘을 이용하여 관계형 스키마에서 테이블간의 의존성을 고려하여 외래키로 연결된 테이블을 결합하여 계층적인 부모-자식 구조의 XML 스키마로 생성한다. 이렇게 생성된 XML 스키마에 대한 트리 스키마 구조[14]를 생성한다. XML 문서의 스키마 구조 관리기는 트리 구조를 가지며 관계형의 스키마가 변경되면 XML 스키마의 구조를 변경한다. 여기서는 관계형의 스키마 중 주석 정보를 제외한 구조 정보만을 XML 트리 스키마 구조로 유지하는 것으로 제약한다.

- 통합 질의 분석기
  사용자 및 응용프로그램에서 요청된 질의 유형을 분석하여 관계형 질의 또는 XML질의 유형인지를 분리한다. XML 질의 유형으로는 관계형 데이터베이스에서 처리 시 비용이 많이 드는 특정 패스에 대한 패턴질의 및 계층적 질의가 해당된다. 예를 들면, 단백질 구조 분류에 대한 경로 정보에 대한 질의가 해당된다.
  - CATH 데이터베이스의 특정 단백질 패밀리에 속하는 GenBank 식별자 및 PDB code를 검색하시오.
  - CATH 데이터베이스의 특정한 단백질 폴드와 패밀리를 검색하시오.

- XMLTOSQL질의 변환기
  관계형 데이터에 대한 XML 뷰를 통한 XML 질의를 관계형 질의로 변환하기 위한 모듈이다. XML 질의 언어로 XQuery를 관계형 질의 언어인 SQL로 변환한다. 이 모듈은 XQuery 파서와 질의 그래프를 동일한 질의결과를 갖는 간략화된 질의 그래프로 변환하는 질의 그래프 변환기, 질의 그래프를SQL로 변환하는 SQL 변환기로 구성된다. 이러한 XML질의를 처리에 대한 상세한 내용은 XPERANT[7]와 [15]를 참고할 수 있다.

- XML태그생성기
  이 모듈은 사용자 질의 결과에 해당하는 엔트리를 XML 문서로 생성한다. XML에 대한 질의와 관계형 데이터웨어하우스에 대한 질의 결과 모두 통합되어 XML 문서로 생성된다. 질의 결과를 XML 문서로 생성하기 위해서는 XML 트리 스키마 구조를 참조하여 이 구조에 맞는 하위 트리를 생성하고 이 트리를 기반으로 XML 태그를 생성한다.

- 포맷 변환기
  이 모듈은 질의 결과로 생성된 XML문서를 최종적으로 질의에서 원하는 포맷으로 변환을 하는 과정이다. 이를 위해 우선 통합 XML 스키마에 대하여 변환하려는 대상 파일들과 매핑정보를 XSL로 작성하여 데이터베이스에 메타 데이터로 저장한다. 따라서 질의 결과로 생성된 XML파일에 매핑정보를 포함한 XSL을 적용하여 HTML 형식 혹은 생물학적 응용프로그램의 표준 입력형식으로 변환하거나 XML로 변환한다. 예를 들어 BLAST 검색 시스템의 FASTA형식, 혹은 BSML[16]과 BioML[17]등 생명정보 파일 포맷으로 변환하여 직접 응용프로그램에서 사용할 수 있다.

- 사용자 및 생명정보 응용프로그램
  이 계층은 통합시스템의 응용계층으로 사용자가 직접 통합 데이터에 대한 질의를 실행할 수도 있고 마이닝 툴을 이용하여 패턴 및 새로운 규칙을 탐사할 수 있다. 또한 유사성 검색, 구조 가시화, 구조 비교 등과 같은 다른 생물학적 애플리케이션에서 통합 데이터베이스 및 시스템을 활용할 수 있다.

이 시스템 중 기존의 XML 미디에이터에서 지원하는 질의 변환부분을 제외하고 4장과 5장에서 상세히 설명한다. 특히 단백질 구조 분석에 적용하기 위하여 단백질 구조에 대한 관계형 데이터 웨어하우스의 통합 모델링과 이에 상응한 XML 통합 스키마 생성에 대해 설명한다.

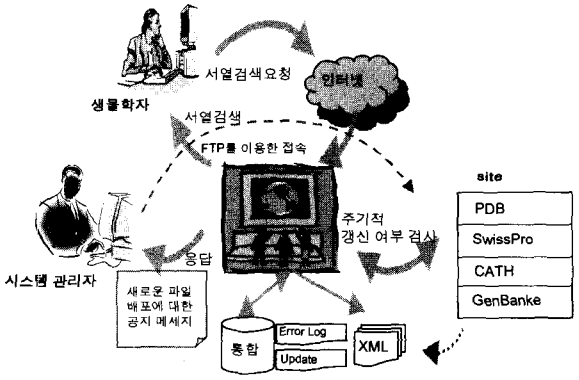
3.3 데이터웨어하우스의 갱신 처리 및 데이터 출처 관리

3.3.1 갱신 관리

데이터 소스의 변경 검출을 위해서는 push와 pull 기술을 이용할 수 있다. push 기술은 데이터 소스에 질의를 등록해 놓고 등록된 질의와 일치되는 변경 발생에 대한 외연적인 공지(notification)을 보낸다. 이 방식은 트리거를 이용하여

자동적으로 공지를 보낼 수 있다. pull 방식은 사용자가 주기적으로 데이터 소스에서 변경이 일어났는지를 확인해서 변경할 수 있다.

이 논문에서는 소스 데이터베이스의 서버와 협력작업이 요구되는 push 방식보다 Pull 방식을 이용한다. (그림 3.2) 같은 갱신 프로그램을 이용해서 사용자가 public domain 데이터베이스의 배포 주기에 따라 주기적으로 소스 데이터베이스의 갱신을 조사한다. 그리고 최신 버전의 플랫파일이 배포되면 이것을 다운받아 데이터베이스 갱신을 수행한다. 갱신은 소스 플랫파일에 포함된 엔트리들의 변경날짜 또는 버전 정보에 따라서 변경 없음, 변경, 새로운 엔트리로 분류된다. 새롭게 추가된 엔트리는 데이터베이스에 입력된다. 예를들면, SWISS-PROT은 DT 필드로서 가장 최신의 주석 갱신날짜를 나타내고 GenBank는 접근 식별자에 버전정보(Aceession.version)를 포함한다. PDB, Swiss-Prot, GenBank와 CATH 모두 플랫파일로서 XML파일을 지원하고 있다. 따라서 수정된 엔트리에 대해 변경된 정보를 검출하기 위해 차이점을 찾는 XMLTreeDiff 알고리즘을 이용한다.



(그림 3.2) 소스 데이터 변경에 대한 갱신 메커니즘

갱신은 서열 및 구조 정보와 주석 정보에 대해 서로 다르게 관리된다. 구조 정보는 변경 이력을 관리하기 위해 갱신이 발생할때 마다 서로 다른 버전을 생성하여 버전 테이블에 저장한다. 특히 단백질 서열 정보에 대해서는 서열의 변경정보를 저장 관리하는 서열 버전 테이블과 변경을 검출하는 알고리즘을 트리거를 이용하여 구현하였다. 이것에 대한 자세한 정보는 [18]를 참조하고 구조 정보에 대한 갱신 연구는 [19]에서 참조할 수 있다.

주석정보는 변경이 발생하면 갱신연산을 수행하고 갱신연산에 대한 변경 값을 로그테이블에 저장하여 갱신정보를 관리한다. 이를 통해서 데이터의 변경을 추적할 수 있다. 이 로그 테이블의 update date는 데이터베이스에서 속성 값이 변경된 시간을 나타내며 이 속성과 엔트리 식별자인 PID로 질의 결과를 정렬하면 엔트리가 데이터베이스에서 변경된 이력 정보를 검색할 수 있다. 예를 들어 다음 (그림 3.4)는 (그림 3.3) 같이 GenBank에 존재하는 U78433 서열의 변경 이력이 발생하였을 때 갱신 프로그램에 의해 통합 데이터베이스에 반영한 예이다.

Revision history for U78373

This ID replaces sequence(s)
Common Rev. history

1) U78433 (See Rev. history)

GI	Version	Update Date	Status	I	II
33578264	2	Aug 11 2003 1:53	Live		
3090813	1	Jun 20 2003 3:01	Dead		
3090813	1	Jun 20 2003 3:00	Dead		
3090813	1	Apr 29 1998 12:17	Dead		

Accession U78373 was first seen at NCBI on Apr 29 1998 12:17

Revision history for U78433

This ID was replaced by U78373 (See Rev. history)

GI	Version	Update Date	Status	I	II
3090814	1	Jun 20 2003 3:01	Dead		
3090814	1	Jun 20 2003 3:00	Dead		
3090814	1	Apr 29 1998 12:17	Dead		

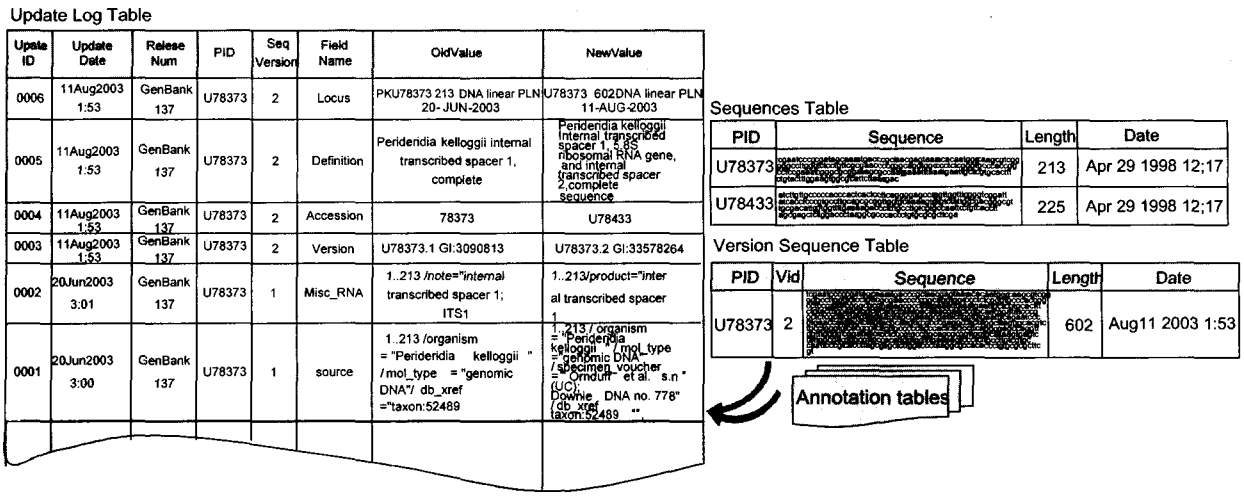
Accession U78433 was first seen at NCBI on Apr 29 1998 12:17

(그림 3.3) GenBank에서의 서열 변경 예

이 서열은 U78433과 U78373의 각각의 서열로 Apr 29 1998 12 : 17에 데이터베이스 입력된 후 각 서열에 대해 Jun 20 2003 3 : 00과 Jun 20 2003 3 : 01에 서열의 주석 정보가 변경이 발생하고 최종적으로 Aug 11 2003 1 : 53에 두 개의 서열이 결합되는 서열 변경이 발생하여 U78373 서열에 대한 두 번째 버전이 GenBank 데이터베이스에 존재하는 경우이다. 통합 데이터베이스에는 각 서열이 Apr 29 1998 12 : 17 시간에 입력된 이후 Jun 20 2003 3 : 00과 Jun 20 2003 3 : 01의 변경에 대해서는 해당되는 주석 테이블에 대한 갱신 연산을 수행하고 갱신 연산이 성공적으로 수행되면 능동데이터베이스의 트리거를 이용하여 갱신된 내용을 Update log테이블에 새롭게 입력한다. Aug 11 2003 1 : 53 시간에 U78433이 U78373서열과 통합되어 U78373서열에 대한 버전 서열이 생성되어 Version Sequence 테이블[18]에 입력된 것을 (그림 3.4)에서 확인할 수 있다. 또한 이러한 버전 서열에 대한 주석 정보의 변경은 원본 서열인 U78373 주석 정보를 갱신한다. Update Log테이블의 0006, 0005, 0004와 0003 튜플은 U78373의 버전 서열의 주석 정보 변경을 반영한 것이다.

3.3.2 데이터 소스 트랙킹

통합 질의는 서로 다른 데이터 소스로부터 통합 데이터에 대해 수행되어진다. 또한 다양한 데이터 마이닝 기반 분석 알고리즘에 의해서 새로운 데이터가 생성되고 유도된다. 예를 들면, 단백질의 기능은 유사성 검색을 통해서 정해진다. 이때 다른 단백질의 주석정보는 유사성 관계에 있는 단백질들에 상속되어 적용되기 때문에 소스 데이터베이스의 실험 데이터가 정확하지 않다고 결정되면 다른 모든 유사성 관계의 단백질에 대한 주석정보의 할당이 철회되어야 한다. 특히 HGP를 통해생산된 데이터는 실험적으로 검증되지 않은 주석정보를 포함한 서열데이터들이 많이 존재하



(그림 3.4) 서열 및 주석변경관리를 위한 서열 변경 로그 테이블과 버전 서열 테이블

기 때문에 주석정보의 외부 소스를 트래킹할 수 있는 정보는 매우 중요하다.

Fact테이블 중 주석정보를 포함하는 feature, reference, function은 update log 테이블을 이용하여 데이터의 변경을 추적한다. 또한 evidence와 algorithm 테이블을 이용하여 외부 소스 데이터 및 데이터의 상태 추적을 할 수 있다. 서열, helix, sheet, turn과 loop같은 구조 정보와 구조정보로부터 분석을 통해 유도된 구조 분류 정보 fact 테이블은 구조정보 버전 테이블, Status 테이블과 algorithms 테이블을 이용하여 외부 소스 데이터의 출처와 데이터의 상태 및 일치성과 변경을 트래킹 한다. Update log 테이블과 구조 정보 버전 테이블은 (그림 3.4) 같이 유지되고 Status 테이블과 algorithm 테이블에 대해서 아래에서 설명한다.

Algorithm Table : 유도되거나 예측된 주석 및 구조 정보를 생성하기 위해 이용된 알고리즘과 알고리즘의 구현정보(소프트웨어 버전, 언어), 알고리즘 실행 환경정보, 알고리즘에 호출시 사용된 파라메터, 알고리즘 실행시 생성된 데이터에 대한 정보이다. 이러한 정보는 각 튜플마다 기록된다.

Status Table : 주석 및 구조정보의 신뢰성 및 검증 정보를 포함한다. 즉, 상태 식별자, 외부 데이터 소스 이름, 외부 데이터베이스 식별자와 주석 삽입자에 대한 정보로 구성된다. 상태 식별자는 다음과 같은 6가지 경우로 나눌 수 있고 결합해서 사용할 수 있다.

- Experimental : 주석정보가 실험을 통해서 검증된 경우
- Predicated : 주석정보가 실험을 통해서 검증되지 않고 다른 서열과의 유사성 비교, 패턴 분석 및 예측기법에 의해서 유도된 경우
- Absent : 주석정보가 포함되지 않은 경우
- Atypical : 서로 다른 타입의 주석정보가 충돌되는 값을 가지므로 값을 확신할 수 없을 때
- Incomplete : 주석정보나 서열이 완벽한 정보를 포함하지 않은 경우
  (예를 들면, 서열 조각이나 2차 구조의 일부가 누락된 경우 등)
- Import : 외부 소스로부터 포함되고 주석정보가 아무런 검증 작업을 거치치 않은 데이터

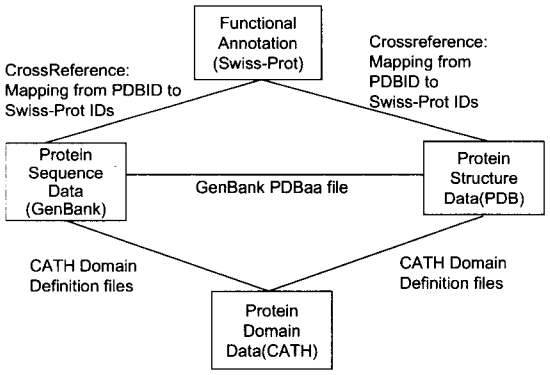
4. 단백질 구조 통합 모델링

4.1 단백질 구조관련 데이터 소스

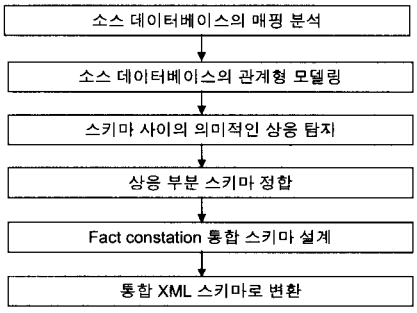
단백질 구조 관련 데이터로는 1차 구조, 2차 구조 및 3차 구조가 있다. 1차 구조는 단백질 서열 정보를 나타내고 2차 구조는 단백질 구조의 helix, sheet, turn과 loop정보를 포함하고 3차 구조 정보는 이러한 2차 구조 요소들이 3차원 공간상에서 배열된 형태이다. 3차 구조는 단백질 분자를 구성하는 원자의 3차원 좌표 값으로 표현된다. 여기서는 단백질 서열의 변경 정보를 포함한 데이터베이스로 GenBank[20-22], 단백질 서열의 고수준 주석을 제공하는 데이터베이스로서 Swiss-Prot[23], 단백질 구조 정보 데이터베이스로 PDB[24]를 선택하고 구조와 기능이 상동관계에 있는 단백질을 군집화한 구조 분류 데이터베이스로서 CATH[25]를 통합한다.

(그림 4.1)처럼 CATH 도메인 정의 파일을 파싱하여 도메인 분류 정보를 추출하여 도메인 정보가 명시된 PDBID를 이용하여 CATH와 PDB를 매핑한다. Swiss-Prot 데이터베이스는 DR필드에 PDB id정보로 상호참조 정보를 포함하고 있고 PDB 역시 DBREF필드에 Swiss-Prot의 엔트리에 대한 상호참조 정보를 포함한다. 따라서 Swiss-Prot과 PDB의 상호 참조 정보를 이용하여 각각의 식별자로 매핑하여 통합한다. 또한 GenBank의 단백질 서열 플랫 파일에

는 Swiss-Prot의 주석정보를 참조할 수 있는 dbsource 필드를 가지며 Siwss-Prot은 GenBank 단백질 서열을 참조할 수 있는 GenBank ID를 포함한다. 따라서 Swiss-Prot과 GenBank도 상호 참조를 이용하여 엔트리를 매핑한다.



(그림 4.1) 소스 데이터의 매핑 관계



(그림 4.2) 통합 모델링 과정

4.2 Fact constellation 스키마 기반 통합 모델링

관계형 웨어하우스에서 fact constellation 다차원 모델 기반 통합 모델링은 (그림 4.2)처럼 다음 같은 다섯 단계를 거친다.

- 관계형 모델로 변환

  이 단계는 통합의 첫 번째 단계로 통합하려는 데이터 소스를 관계형의 fact constellation 모델로 변환하는 단계이다. 4.1절의 소스 분석 내용을 통해서 통합할 부분의 범위를 결정하고 데이터 소스의 통합 순서를 결정한다.

  통합의 범위는 통합 수준에 따라 결정된다. 즉, tightly federated database를 구축하기 위해서는 공통 모델로 변환, 스키마의 의미적 매칭, 스키마 정합, 데이터의 변환과 데이터의 의미적 매칭의 통합 모델링 단계를 거친다. 그러나 loosely federated database는 스키마의 정합 단계까지를 수행하며 이러한 정합은 소스 스키마의 합집합 수준이다. 데이터의 변환이 빈번히 발생하여 스키마의 변경이 많은 단백질 같은 생물학 데이터베이스를 위해서는 tightly federated database 접근 방법보다 loosely federated database가 더 적합하다.

- 스키마 의미적인 상응 탐지

  이 단계는 스키마 매칭 단계로서 통합하려는 스키마들 간에 의미적으로 동일한 데이터를 나타내는 테이블과 속성의 상응관계를 탐지하는 것이다. (그림 4.3)은 PDB와 Swiss-Prot과 CATH 스키마간의 상응관계를 나타낸다. Swiss-Prot과 GenBank 스키마는 유사하여 상응관계를 생략한다. PDB와 Swiss-Prot은 각 식별자를 이용하여 상호 참조관계를 가지며 단백질 서열, 단백질의 소스 생명체, 참고문헌 정보, 참조 데이터베이스등에 해당되는 세부 항목들이 의미적으로 서로 대응된다.

  PDB와 CATH 데이터베이스간에는 PDB의 엔트리가 CATH의 도메인 정보를 나타내는 엔트리에 대응되며 식별자를 통해 상호 참조된다. CATH에서 PDB의 SOURCE와 COMPOUND에 해당하는 필드인 단백질의 소스 생명체, 단백질 분자 이름 및 유사한 이름 정보를 공유한다.

  GenBank와 EMBL은 DNA 정보를 세계적으로 공유하고 교환하는 협력관계를 맺고 있어서 GenBank의 스키마와 EMBL의 Swiss-Prot 포맷간에는 거의 일대일 매핑한다. GenBank의 DBSOURCE와 Swiss-Prot의 상호참조를 나타내는 DR필드가 서로 매핑된다. 따라서 GenBank 데이터베이스로부터 Swiss-Prot의 단백질 서열에 해당하는 서열의 갱신 정보만을 이용한다.

- 스키마 정합

  스키마 매핑 단계에서 스키마 통합을 위하여 상응부분이 있는 스키마를 정합하고 정렬한다. 스키마의 정합은 다음과 같은 정의와 통합 규칙[26]을 따른다.

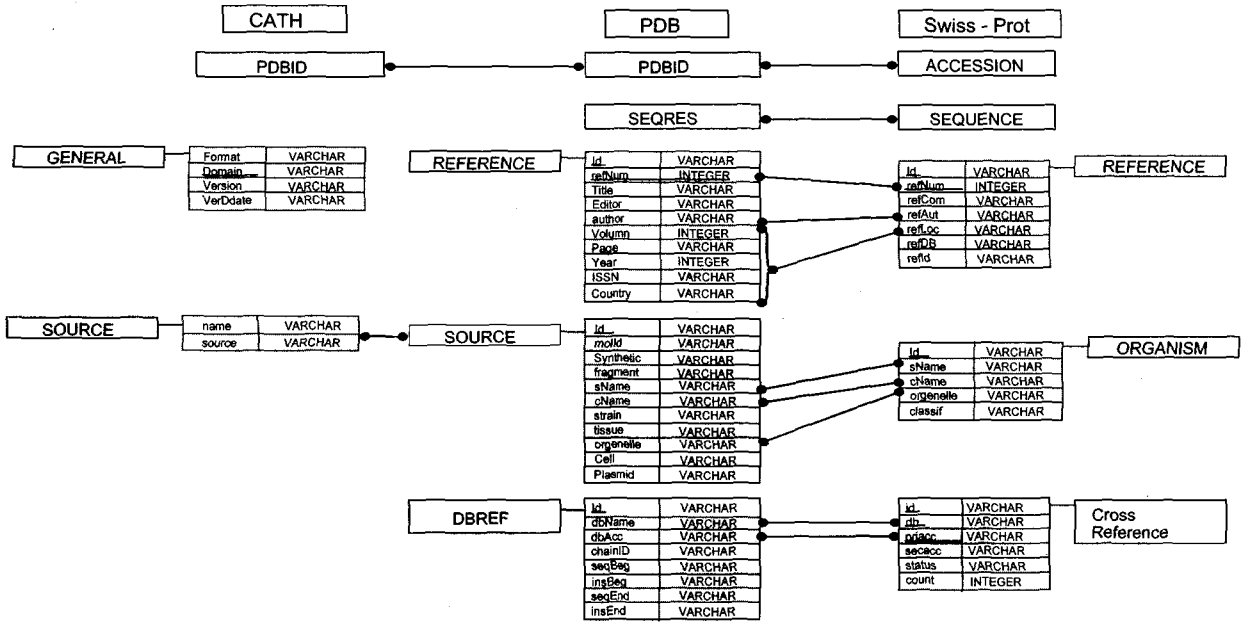
[정의 1] 두 스키마 S_1과 S_2의 상응요소 E_1, E_2가 있고, 값 속성 A_1, A_2가 주어진다면, A_1, A_2가 서로 상응되고 상응 속성 attcor(A_1, A_2)를 갖는 $E_1 \equiv E_2$로 표현한다. A_1과 A_2의 통합은 속성 A로 표현하고 A는 다음과 같이 정의된다.

① 속성 이름은 특정 이름으로 지정.

② 도메인 정의 : attcor(A_1, A_2)가 $A_1 = A_2$ 또는 $A_1 \supseteq A_2$이면 도메인은 domain(A_1)을 따르고, attcor(A_1, A_2)가 $A_1 \cap A_2$ 또는 $A_1 \neq A_2$이면 도메인은 domain(A_1)∪ domain(A_2)이다.

③ 기수성 : attcor(A_1, A_2)가 $A_1 = A_2$, $A_1 \supset A_2$ 또는 A_2=f(A_1)이면 기수성은 cardmin(A) = cardmin(A_1), cardmax(A) = cardmax(A_1)이다. attcor(A_1, A_2)가 $A_1 \cap A_2$이면 cardmin(A) = Max(cardmin(A_1), cardmin(A_2)), cardmax(A) = cardmax(A_1) + cardmax(A_2)이고 attcor(A_1, A_2)가 $A_1 \neq A_2$이면 cardmin(A) = cardmin(A_1) + cardmin(A_2)이다.

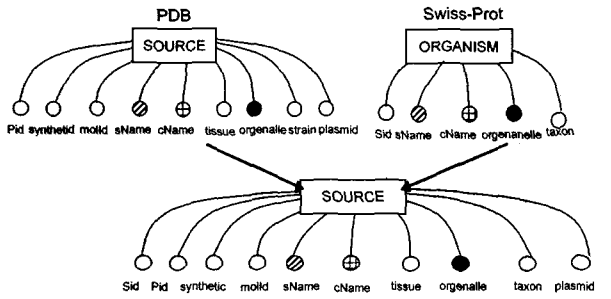
[정의 2] 소스 스키마 요소 통합 규칙 : 다른 스키마에 대응되는 것이 없는 스키마 S_1의 속성 X_1은 통합 스키마에 속성 X로써 더해진다. X의 타입은 X_1의 타입과 같다.



(그림 4.3) 스키마간의 상응 탐지

- 상응선언(correspondence assertion) : $X \equiv X_1$
- 사상(mapping) : $X = X_1$

(그림 4.4)는 PDB의 SOURCE 테이블과 Swiss-Prot의 ORGANISM 테이블의 상응 탐지 결과를 바탕으로 [정의 1]과 [정의 2]의 통합규칙을 적용하여 스키마 정합을 수행한 예를 보여준다. (그림 4.4)에서 테이블 스키마를 트리로 표현하고 사각형으로 표현된 루트 노드는 두 테이블을 나타내고 하위 노드는 테이블에 있는 속성이다. 두 테이블 스키마사이에 상응이 탐지된 노드를 같은 무늬로 표현한다.



(그림 4.4) 스키마 정합 예제

(그림 4.4)에서 이들의 상응관계는 SOURCE ≡ ORGANISM 이고 sName = sName, cName = cName, organelle = organelle이다. 따라서 통합 스키마는 SOURCE(Sid, Pid, synthetic, molID, sName, cName, tissue, organelle, taxon, plasmid)이다.

- 통합 스키마 설계

  하위 테이블 스키마의 정합 결과를 바탕으로 Fact constellation 모델 기반 통합 스키마를 설계하였다. 통합 스키마는 loosely federated database로 통합 접근 방식을 따르도록 통합 범위에 해당되는 소스 데이터베이스의 해당 스키마에 대해 통합규칙을 적용한다. 따라서 통합 스키마는 각 스키마 정합 결과의 합집합이다. Fact constellation 통합 스키마에서 사실 테이블은 단백질 1차, 2차, 3차 구조정보를 나타내는 Structure, 단백질 기능 및 유전적 진화정보를 포함하는 주석정보인 Annotation, 구조 분류 정보를 나타내는 Classfication, 버전 및 주석의 변경 정보를 보전하는 History, 참고 문헌정보를 포함하는 References와 단백질의 소스에 대한 정보를 나타내는 Protein으로 구성된다.

  - 사실 테이블 : Protein, Structure, Annotation, Classification, History
  - 공유 차원 테이블 : SSEs, Sequence, Domain, Source, Status, Update Log, Version Sequence, Super Family, Family, Reference
  - 차원 테이블 : Features, Function, Genetics, Accession, Class, Architecture, Topology, Compound, Authors

  공유 차원 테이블은 서로 다른 두 개의 사실테이블에 의해서 공유되는 차원 테이블이다. 위와 같은 스키마를 기반으로 서열정보 테이블에서 특정 서열을 셀력션하고 이 서열이 속한 도메인과 패밀리 정보를 셀력션할 수 있다. 또한 주석, 서열 및 구조 분류 정보가 통합됨으로써 특정 생명체의 조직에 발현된 단백질과 이 단백질이 속한 폴드 패밀리 및 진화적 조상에 대한 검색 등이 가능하다.

4.3 XML 스키마로의 변환

먼저 Fact constellation 통합 스키마를 XML 뷰를 제공하기 위해 XML 스키마로 변환한다. XML 스키마 생성을 위한 입력 모델[13]과 출력 모델[14]을 [정의 3]과 [정의 4]와 같이 기술한다. 먼저 $\hat{T}$는 테이블 명의 집합, $\hat{C}$는 속성 이름의 집합이고 $\hat{b}$는 표준 SQL에서 정의한 원자 타입이라고 가정한다. $C \in \hat{C}$인 속성 이름에 충돌이 생겼을 때 속성 명은 테이블 이름 $t \in \hat{T}$과 "[ ]"수식을 사용하여 표현한다 (예 : t[c]).

[정의 3] XML 스키마 변환을 위한 입력 모델로서 관계형 스키마를 다음과 같이 정의한다. 관계형 스키마 R은 4개의 튜플 $R=(T, C, P, \Delta)$로 정의한다.

- T는 테이블 이름의 유한 집합이다. C는 테이블 이름으로부터 그 테이블에 속하는 칼럼으로의 함수이다.
- P는 속성 이름에 대한 칼럼 타입을 정의하는 함수이다 : 예를 들어 $P(c)=\alpha$ 여기서 α는 (τ, u, n, d, f) 같은 다섯 튜플로 구성된다. 여기서 $\tau \in \hat{b}$, u는 유일성(Uniqueness)를 나타내고 n은 널값을 허용하는지를 나타낸다. d는 칼럼 c의 도메인 값이며 알려지지 않은 값이면 ε으로 나타낸다. f는 c의 디폴트 값이며 알 수 없으면 ε으로 나타낸다.
- Δ는 관계형 스키마에서의 일관성 제약사항을 나타낸다.

[정의 4]는 출력모델인 XML 스키마를 나타내고 있다. XML 스키마를 나타내는 엘리먼트의 집합 $\hat{E}$, 속성 집합 $\hat{A}$, 원자 데이터 타입 집합 $\hat{\tau}$라고 가정했을 때 [정의 4]는 다음과 같다.

[정의 4] XML 스키마는 다음과 같은 6개의 튜플로 나타낸다. $X=(E, A, M, P, \gamma, \Sigma)$

- E는 엘리먼트의 유한집합이며 A는 엘리먼트 $e, e \in \hat{E}$에 대한 속성 $a, a \in \hat{A}$ 함수이다.
- M은 엘리먼트 타입 정의 함수이다. 예를 들어 : $M(e)=\alpha, \alpha ::= \varepsilon|\tau|\alpha+\alpha|\alpha,\alpha|\alpha?|\alpha*|\alpha+$, ε은 빈 엘리먼트를 나타내고 $\tau \in \hat{\tau}$, "+"는 유니언을 나타내고 ","은 연결을 나타내고 "+"는 서브 엘리먼트가 한번 또는 여러 번 나타나는 것을 표현한다. "?"는 서브 엘리먼트가 나타나지 않거나 한번 나타나는 것을 표현한다, "*"는 서브 엘리먼트가 나타나지 않거나 여러 번 나타나는 것을 표현한다.
- P는 속성 α에 대한 속성 타입 정의 함수이다. 예를 들어, $P(\alpha)=\beta, \beta=(\tau, n, d, f), \tau \in \hat{\tau}$, n은 "?"(널값 허용) 혹은 "ㄱ?"(널값 허용 안함), d는 α의 유효한 도메인 집합, 알려지지 않았으면 ε, f는 α의 디폴트 값으로 표현한다.
- $\gamma \subseteq \hat{E}$ 루트 엘리먼트의 유한집합이며 Σ은 일관성을 표현하는 키들의 집합이다.

<표 4.1>에서는 사실테이블과 그에 속하는 차원 테이블간의 기수성에 따른 테이블 관계를 나타낸다. 사실 테이블과-차원 테이블은 각각 1 : N, 1 : 1의 기수성을 갖고 두 개의 사실 테이블들과 공유 차원 테이블간에는 각각 1 : N : 1과 1 : 1 : 1의 기수성을 갖는다.

〈표 4.1〉 통합 스키마의 4가지 경우

경우	테이블S	테이블T	기수성	예
1	사실 테이블	차원 테이블	1 : N	STRUCTURE : HELIX (SHEET, TURN, ATOM), ANNOTATION : FEATURES REFERENCE : AUTHORS
2			1 : 1	PROTEIN : ACCESSION CLASSIFICATION : CLASS (Architecutre, Topology, SuperFamily, Family)
3		공유 차원 테이블	1 : N : 1	PROTEIN : REFERENCE : STRUCTURE (Annotation) STRUCTURE : DOMAIN : CLASSIFICATION PROTEIN : VERSEQUENCE : HISTORY ANNOTATION : SEQUENCE : STRUCTURE ANNOTATION : STATUS(UpdateLog) : HISTORY
4			1 : 1 : 1	PRISTR : SOURCE : STRUCTURE

다음은 이러한 4가지 경우에 따라 Fact constellation 스키마를 XML 스키마로 변환하는 알고리즘이다. XML 스키마변환 알고리즘은 크게 사실테이블-차원 테이블을 변환하는 과정과 이들의 참조관계를 변환하는 단계로 나누어진다. (알고리즘 4.1)은 사실테이블과 차원 테이블의 기수성에 따른 XML 스키마의 엘리먼트로 변환을 나타낸다.

사실 테이블 f와 그에 속하는 차원 테이블 d가 있고 X와 Y는 f와 d의 모든 속성을 나타내며 $\alpha \subseteq X$가 f의 키이고 $\beta \subseteq Y$가 d의 키일 때, f와 d 차원 테이블간의 참조관계를 나타내는 외래키 제약사항은 $f[\alpha] \subseteq d[\beta]$이고 $k_d \subseteq X$는 차원 테이블 d의 키라고 가정한다. 이러한 가정하에 서로 다른 기수성과 외래키 제약 사항을 가지는 사실 테이블과 차원 테이블을 XML 스키마로의 변환은 다음과 같다.

차원 테이블을 참조할 수 있는 사실 테이블의 외래키(α)가 널 값이 아니면 외래키의 유일성 여부에 따라 두 테이블의 기수성이 1 : 1과 1 : N이 결정된다.

a. 두 테이블(fact-Dimension)에 1 : 1의 관계가 존재하면

XML 스키마는 다음과 같다.

$$M(f)=(X, d?), M(d)=(Y-\beta)$$
$$\sum=\left\{ k_d-\beta) \xrightarrow{key} d,\ \alpha \xrightarrow{key} f \right\}$$

b. 두 테이블 (fact-Diemnsion) 사이에 1 : N의 관계가 존재하면 XML 스키마는 다음과 같다.

$$M(f)=(X, d^*), M(d)=(Y-\beta)$$
$$\sum=\left\{ k_d-\beta) \xrightarrow{key} d,\ \alpha \xrightarrow{key} f \right\}$$

Input : d : share dimension table, f : fact table
Output : Element Types M(d), M(f)
Begin
  lists of columns for table f X
  lists of columns for table d Y
  primary key of talbe f α(f[α])
  primary key of table d β(d[β])

if (f[α] ⊆ d[β]) ∧ (α *is non null*) **then**
  if α is uniquie **then** // f와 d의 기수성이1 : 1인 경우
    M(f) = (X, d?), M(d) = (Y-β),

$$\sum=\left\{ k_d-\alpha) \xrightarrow{key} d,\ \beta \xrightarrow{key} f \right\}$$

  else
    M(f)=(X,d*), M(d) = (Y-β)
    // f와 d의 기수성이1 : n인 경우

$$\sum=\left\{ (k_d-\alpha) \xrightarrow{key} d,\ \beta \xrightarrow{key} f \right\}$$

  end-if
    A (d) = {ID_d}
    P(ID_d) = (*ID*, ㄱ?, ε, ε) // 외래키 참조관계를 XML의
    A(f) = {Ref_f} // ID-IDREF로 표현함
    P(Ref_f) = (*IDREF*, ㄱ?, ε, ε)
else // 참조하는 외래키가 없는 경우
    M(d)=(Y), M(f) =(X) // 이외의 경우는 플랫 스키마 변환

$$\sum=\left\{ f[\alpha] \subseteq d[\beta], k_d \xrightarrow{key} d,\ \beta \xrightarrow{key} f \right\}$$

end if
End Begin

(알고리즘 4.1) fact constallation 스키마의 XML 스키마 변환 알고리즘

사실 테이블과 차원 테이블이 1 : 1의 기수성을 가진 경우에는 사실 테이블을 하나의 엘리먼트 **M(f)**로 변환하고 이 엘리먼트의 하위 엘리먼트에로서 차원 테이블을 변환한 엘리먼트 **M(d)**를 하위 엘리먼트로 갖는다. 이때 **M(f)**는 사실테이블에서 차원 테이블을 참조할 수 있는 외래키 이를 제외한 속성들로 표현된다. 이는 차원 테이블을 참조하는 사실 테이블에 포함된 외래키는 하위 엘리먼트인 **M(d)**에 포함되므로 중복 기술을 피하기 위함이다. 하위 엘리먼트인 **M(d)**는 사실 테이블에 속한 외래키의 선택성에 따라 나타나지 않거나 한번 나타날 수 있다.

사실 테이블과 차원 테이블의 기수성이 **1 : N**을 가진 경우에도 마찬가지로 사실 테이블을 변환하여 **M(f)** 엘리먼트로 매핑하고 이것의 하위엘리먼트로써 차원 테이블에 해당되는 엘리먼트 **M(d)**가 1번이상 나타낸다.

사실테이블과 차원테이블에 대한 엘리먼트 매핑이 끝난 후에는 이들 간의 참조관계를 매핑할 수 있도록 M(d) 엘리먼트에는 ID 속성과 M(f) 엘리먼트에는 IDREF 속성을 추가한다. 즉, 관계형 데이터베이스의 주키/외래키간의 참조관계를 XML문서의 ID/IDREF 속성으로 표현한다. 따라서 A(d) = {ID_d}는 XML문서의 스키마에 속하는 M(d)엘리먼트에 ID속성을 추가하는 것이다. P(ID_d) = (*ID*, ㄱ?, ε, ε)는 ID 속성에 대한 타입을 정의하는 함수로써 널값을 허용하고 도메인 집합이 정해지지 않았고 디폴트 값을 갖지 않는다. A(f) = {Ref_f}도 M(f) 엘리먼트에 대한 Ref 속성을 정의하고 이 속성 타입 정의함수로써 P(Ref_f) = (*IDREF*, ㄱ?, ε, ε)를 정의하였다.

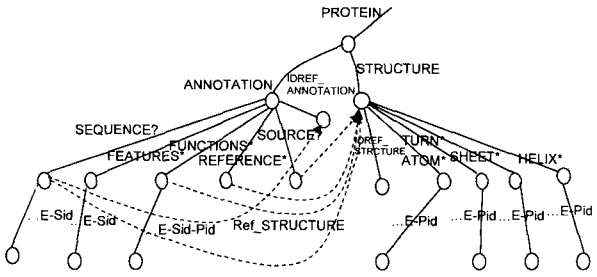
사실 테이블의 외래키(α) 값이 널값을 가지면 플랫 스키마 변환이 이루어진다. 따라서 사실테이블과 차원 테이블을 각각 하나의 엘리먼트로 변환한다.

사실테이블과 공유차원 테이블의 관계에 대한 매핑도 사실테이블과 차원 테이블의 매핑관계와 동일하게 이들간의 기수성에 따라서 변환된다. 사실테이블과 공유 차원 테이블의 매핑관계가 1 : 1 : 1인 경우는 사실테이블과 차원 테이블의 1 : 1기수성과 동일하게 변환하고 1 : N : 1인 경우에는 1 : N의 경우와 동일하게 변환한다.

예를 들어, 공유 차원 테이블 SEQUENCE와 사실 테이블 ANNOTATION과 STRUCTURE는 경우 3의 1 : N : 1에 해당된다. 먼저 공유 차원 테이블 SEQUENCE와 차원 테이블 ANNOTATION의 경우 b처럼 ANNOTATION 엘리먼트는 하나 이상의 SEQUENCE 서브 엘리먼트를 가진다. 그리고 두 번째 사실 테이블 STRUCTURE가 공유 차원 테이블 SEQUENCE의 ID속성을 참조하는 IDREF 속성을 가지므로 참조 무결성을 유지한다.

(그림 4.5)는 XML 스키마의 일부를 트리 모델로 표현하였다. 간선에 표시된 라벨은 엘리먼트나 속성들을 나타낸다. (그림 4.5)에서 A는 각 엘리먼트에 속하는 속성들을 나타내며 Protein엘리먼트의 서브 엘리먼트들은 외래 키인 속성을 제외하여 표현하였다. 또한 REFERENCE, SOURCE 공유 테이블은 외래 키인 Sid, Pid를 제외하고 Ref_STRUCTURE 속성을 포함하고 있으며 STRUCTURE 엘리먼트에는 ID_STRUCTURE 속성을 포함하여 STRUCTURE와 제약사항을 나타낸다.

XML 스키마 트리 생성 후에는 이를 기반하여 XML문서를 생성한다. 먼저 데이터베이스에 저장하고 있는 단백질 엔트리를 검색하여 검색한 단백질 엔트리의 속성 값들을 널 값이 아닌지를 체크하여 스키마 트리에 맞게 생성하여 XML문서를 출력한다.



(그림 4.5) XML 통합 스키마 트리 표현

5. 구현 및 평가

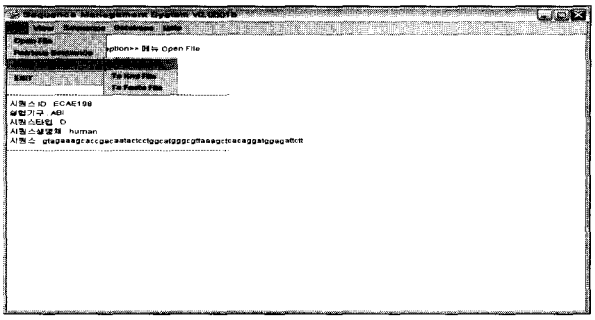
이 절에서는 단백질 구조 데이터에 대한 통합 시스템의 구현 환경과 결과를 보여주고 기존 통합 데이터베이스와 비교 평가하여 기술한다.

5.1 구현 환경

단백질 구조 데이터를 저장하기 위하여 Pentium PC 850 MHz 시스템에서 관계형 데이터베이스 시스템인 Oracle 8.1.7을 이용하였고 플랫폼 독립적으로 시스템을 실현하기 위해 JAVA 언어로 구현하였다. Oracle과의 연동을 위해 JDBC 드라이버를 사용하였다. 단백질 엔트리를 XML문서로 변환하고 변환된 XML문서에 XSL을 적용하여 최종파일을 생성하기 위하여 JAXP(Java API for XML Processing)를 이용하였다. JAXP는 XML에 문서에 대한 파싱을 위하여 DOM과 SAX 방식의 파서 API와 XSLT API 를 지원한다.

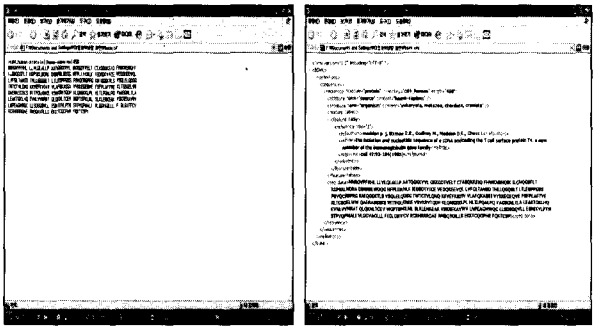
5.2 구현 결과

(그림 5.1)은 시스템 인터페이스이며 "File"메뉴의 "Save sequence"메뉴를 활성화한 상태이다. "File"메뉴의 "To Xml File"메뉴를 사용하여 필요한 엔트리의 id를 입력하여 상응한 XML 문서를 생성하고 "To Fasta File"메뉴를 이용하여 생성된 XML문서를 FASTA 형식의 문서로 변환할 수가 있다.



(그림 5.1) 시스템 인터페이스

(그림 5.2)는 생성된 XML 파일과 FASTA 파일을 웹 브라우저에서 디스플레이한 결과이다.



(그림 5.2) XML을 이용한 파일 포맷 변환 결과

5.3 평 가

지금까지 이질적인 생명정보학 데이터 통합에 대한 연구가 많이 이루어졌다. 이러한 연구들은 통합하려는 데이터의 범위에 따라 크게 두 가지로 분류할 수 있다. 하나는 전체적인 생물학적 데이터에 대한 통합을 목표로 하고 근간에는 실험실 레벨에서 특정 도메인에 속하는 데이터에 대한 통합을 목표로 하는 연구가 활발히 진행되고 있다. 이 절에서는 몇 개의 대표적인 시스템과 비교 평가를 진행하였다.

<표 5.1>에서 알 수 있듯이 전체적인 생물학적 데이터에 대한 통합을 시도한 시스템에는 웨어하우스기법의 대표적

〈표 5.1〉 기존의 시스템과의 비교

비교 항목 \ 비교 시스템	미디에이터기반 시스템	웨어하우스기반 시스템			제안 시스템
	TINET	GUS	PFDB	MetaFam	
질의 언어	OPM 멀티 데이터베이스 질의처리	CPL	SQL	SQL	SQL, XML질의
통합 자원	전체적인 생물학 데이터	전체적인 생물학 데이터	단백질 패밀리와 서열정보	서열 기반의 패밀리 정보	단백질 구조 정보
모델링 기법	객체-프로토콜 모델	센트롤 도그마 기반 관계형 모델	관계형 모델	관계형 모델	Fact Constellation 다차원 모델 ; XML 모델
특 징	특정 함수를 이용한 BLAST 정렬 결과를 이용한 데이터 분석	전체적인 생물학적 시각에서 데이터를 분석 가능	관계형 데이터베이스의 기본적인 기능 사용하여 데이터분석		XML 스키마로의 변환 ; 여러 표준 포맷으로의 데이터 표현이 쉬움.

인 시스템인 GUS와 미디에이터 기법의 대표적인 시스템인 TINET 시스템이 존재한다. GUS는 생물학의 기본 개념인 센트럴 도그마 개념을 이용하여 데이터를 모델링 하였으며 CPL이라는 질의언어를 별도로 정의하여 통합 질의를 실행한다. 미디에이터 기법을 기반한 TINET 시스템은 OPM 모델을 기반 하여 데이터를 모델링하고 BLAST을 소스 데이터에 수행하여 통합 유사성 검색을 할 수 있다. TINET은 통합 데이터를 XML로 표현은 가능하나 표준 형식으로 XML을 사용하지 않았다.

특정 개체에 대한 생물학 데이터 통합 시스템에는 CATH 단백질 분류 데이터베이스와 서열 기반의 단백질 패밀리 정보를 통합한 PFDB[12]와 서열 기반의 단백질 패밀리 정보를 통합한 MetaFam[3]이 대표적인 시스템이다. 두 개의 데이터베이스 모두 웨어하우스기반 기법을 기반으로 하였으며 관계형 모델을 기반으로 생물학적 데이터를 모델링 하였다. 데이터의 범위가 광범위하기 때문에 관계형 모델로 표현하기 쉽지 않은 생물 데이터지만 관계형 모델로 표현하였다. 따라서 관계형 데이터베이스의 여러 가지 기능들을 사용 가능하고 새로운 질의 언어가 아닌 SQL 질의언어를 사용할 수 있다는 장점이 있다.

이 논문에서 제안한 시스템은 데이터웨어 하우스를 기반하여 PFDB와 MetaFam 시스템의 장점을 소유하고 있으며 또한 관계형 스키마에 대한 XML 스키마의 생성을 통하여 관계형 데이터에 대한 XML 뷰를 제공한다는 장점을 갖는다. 또한 XML의 스타일 시트를 이용하여 포맷 변환을 진행하여 웹 상에서 표준 포맷으로 데이터 교환이 가능하다. 또한 특정 생명정보 응용 프로그램의 입력 포맷으로 많이 사용되는 FASTA 포맷으로 서열 데이터를 제공하여 사용자에게 응용 프로그램의 사용이 쉽게 이루어질 수 있다.

6. 결 론

1990년대 인간 게놈 프로젝트를 시작으로 생명정보와 이를 분석하기 위한 응용 프로그램들이 급증하고 있다. 급증하는 생명정보 데이터의 분석을 위해서는 생명정보 통합 관리 시스템이 요구되며 이러한 시스템에서는 생명정보 데이터의 고유 특성을 반영하여야 한다.

따라서 이 논문에서는 실험실 레벨에서 단백질 구조 정보를 통합할 수 있도록 XML 미디에이터 웨어하우스 통합 시스템 구조와 이를 기반한 Fact constellation 통합 스키마를 모델링하고 관계형 통합 스키마에 대한 XML 뷰를 제공하기 위한 XML 스키마로 변환 및 XML문서 생성 기법을 제시하였다. 제안한 통합 기법은 로컬 웨어하우스 통합 기법을 따르므로 통합 질의 최적화가 로컬에서 수행되고 질의 처리시 데이터 소스와 교류 대기시간이 필요 없고 네트워크 연결에 많이 의존하지 않기 때문에 시스템 신뢰도가 좋다. 소스 데이터의 에러를 제거한 정제된 통합된 데이터를 가지므로 이 통합 데이터를 기반하여 분석이 가능하다는 데이터웨어하우스 통합 기법의 장점을 갖는다. 반면, 소스 데이터 변경 문제는 점진적 갱신 방법을 이용하여 주석 정보에 대한 갱신 로그 유지와 서열 데이터에 대한 버전 관리를 통해서 소스 데이터 변경을 데이터 웨어하우스에 반영함으로써 개선하였다. 그러므로 데이터의 변경과 소스에 대한 트랙킹이 가능하다.

생명정보 데이터에 대한 XML 뷰 제공을 위해 관계형 데이터에 대한 XML 스키마로 변환하여 스키마 트리를 유지함으로써 생명정보 데이터 분석 응용프로그램에서 요구되는 데이터의 계층적 표현과 질의가 가능하다.

마지막으로 이 논문에서 제안한 시스템은 기존의 시스템과 비교하였을 때 표준 마크업 언어로 표현하여 웹 상에서의 데이터 교환이 가능하고 응용 프로그램을 이용한 분석이 쉽게 이루어지는 장점을 갖는다.

제안 시스템에 단백질간의 상호작용 및 대사 경로와 유전자 발현정보를 통합하는 DNA 칩 데이터를 추가하여 통합 모델링을 하고 이러한 데이터를 통합적으로 분석할 수 있는 데이터마이닝기법에 대한 연구를 향후 진행할 것이다.

참 고 문 헌

[1] S. B. Davison, J. Crabtree, B. Brunk, J. Schug, V. Tannen, C. Overton and C. Stoeckert "K2/Kleisli and GUS : Experiments in Integrated Access to Genomic Data Sources," IBM Systems Journal Deep computing for the life science, Vol.40, No.2, pp.512-535, 2001.

[2] A. J. Shepherd, N. J. Martin, R. G. Johnson, P. Kellam and C. A. Orengo "PFDB : a generic protein family database integrating the CATH domain structure database with sequence based protein family resources" Bioinformatics, Vol.18, No.12, pp.1666-1672, 2002.

[3] E. Shoop, K. A. T. Silverstein, J. E. Johnson and E. F. Retzel "MetaFam : a unified classification of protein families. II. Schema and query capabilities" Bioinformatics, Vol.17, No.3, pp.262-271, 2001.

[4] I. A. Chen and V. M. Markowitz, "An overview of the Object-Protocol Model and OPM Data Management Tools," Information system, Vol.20, No.5, pp.393-418, 1995.

[5] S. B. Davison, J. Crabtree, B. Brunk, J. Schug and V. Tannen "BioKleisli : A Digital Library for Biomedical Researchers," Journal of Digital Library, Vol.1, No.1, pp.36-53, 1996.

[6] B. A. Echman, A. S. Kosky and L. A. Laroco, "Extending traditional query-based integration approaches for func-

tional characterization of post-genomic data," Bioinformatics, Vol.17, No.7, pp.587-601, 2001.

[7] M. Carey, J. Kiernan, J. Shanmugasundaram, E. Shekita and S. Subramanian, "XPERANTO : A Middleware for ɔulishing Object-Relational Data as XML documents," VLDB, Vol.26, pp.646-648, 2000.

[8] S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman and J. Widom, "the TSIMMIS Project : Integration of Heterogeneous Information Sources," 16th Meeting of the Information Processing Society of Japan, pp.7-18, 1994.

[9] T. Etzold, A. Ulyanov and P. Argos, "SRS : Information Retrieval System for Molecular Biology Data Banks," Methods in Enzymology, Vol.266, pp.144-128, 1996.

[10] C. A. Goble, R. Stevens, G. Ng, S. Bechhofer, N. W. Paton, P. G. Baker, M. Peim, and A. Brass. Transparent n Access to Multiple Bioinformatics Information Sources. IBM Systems Journal, 40(2), pp.532-552, 2001.

[11] C. Baru, A. Gupta, B. Ludascher, R. Marciano, Y. Papakonstantinou and P. Velikhov, "XML-based information medination with MIX," In SIGMOD System Demonstration, 1999.

[12] T. Critchlow, M. Ganesh, R. Musick "Automatic Generation of Warehouse Mediators using an ontology engine," the 5th International workshop on Knowledge Representation meets Database, Vol.10, pp.8.1-8.8, 1998.

[13] D. W. Lee, M. Mani, F. Chiu, W. Chu, "Net&Cot : Translating Relational Schemas to XML Schemas using Semantic Constraints" 11th International Conference on Information and Knowledge Management, Vol.11, 2002.

[14] D. Lee and M. Mani, W. W. Chu, "Schema Conversion Methods between XML and Relations Models", Knowledge Transformation for the Semantic Web, Borys Omelayenko and Michel Klein editors, IOS Press, 2003.

[15] S. H. Park, E. S. Choi and K. H. Ryu, "Implementation of Algebra and Data Model based on a Directed Graph for XML," J. of Korean Information Processing Society, Vol.8-D, No.6, pp.799-812, 2001.

[16] J. Spitzner, "Bioinformatics Sequence Markup Language Manual," LabBook Inc., 1997.

[17] F. Achard, G. Vaysseix, E. Barillot "XML, bioinformatics and data integration" Bioinformatics Vol.17, No.2, pp 115-125, 2001.

[18] S. H. Park, K. H. Ryu and H. S. S. on, "A Protein Structural Information Management Based on Spatial Concepts and Active Trigger Rules," LNCS 2736 pp.413-422, 14th International Conference DEXA03, 2003.

[19] K. H. Ryu, "Building a Genome and Protein Sequence Information Management System," Korea Institute of Science and Technology Information Project Report, 2002.

[20] A. D Baxevanis and B. F. F. Ouellette, "Bioinformatics : A Practical Guide to the Analysis of Genes and Proteins," pp.45-59, Wiley-Liss, Inc, 2001.

[21] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, B. A. Rapp and D. L. Wheeler "GenBank," Nucl. Acids. Res., Vol.30, pp.17-20, 2002.

[22] J. Ostell, S. J. Wheelan and J. A. Kans, "The NCBI data model. Chapter 2 in Bioinformatics : A Practical Guide to the Analysis of Genes and Proteins," 2nd ed., New York : John Wiley & Sons, pp.19-43. 2001.

[23] A. Bairoch and R. Apweiler, "The Swiss-Prot protein sequence database and its new supplement TrEMBL," Nucleic Acids Res., Vol.26, pp.21-25, 1996.

[24] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, "The Protein Data Bank," Nucleic Acids Research, Vol.28, pp. 235-242, 2000.

[25] C. A. Orengo, A. D. Michie, D. T. Jones, M. B. Swindells and J. M Thornton, "CATH- A hierachic classification of protein domain structures," Structures, Vol.5, pp.1093-1108, 1997.

[26] K. H. Ryu, "A Study of Database Schema Integration for Logistics," Electronics and Telecommunications Research Institute Project Report, 1998.

[27] K. H. Ryu, "Development of Updating Protein 3-Dimensional Database and Similarity Search System," Korea Institute of Science and Technology Information Project Report, 2001.

[28] T. Critchlow, K. Fidelis, M. Ganesh, R. Musick and T. Slezak, "DataFoundry : Information Management for Scientific Data," IEEE Transactions on Information Technology in Biomedicine, Vol.4, No.1, pp.52-57, 2000.

[29] A. J. Mackey and William R. Pearson, "Relational databases for biologists," Intelligent Systems for Molecular Biology tutorial, 2002.

[30] P. M. Nadkarni, L. Marenco, R. Chen, E. Skoufos, G. Shepherd and P. Miller "Organization of heterogeneous scientific data using the EAV/CR representation," J. of Am Med Inform Assoc, Vol.6, No.6, pp.478-93, 1999.

[31] S. B. Davidson, C. Overton and P. Buneman "Challenge in Integrating Biological Data Sources," Technical Report, 1995.

[32] P. G. Barker, C. A. Goble, S. Bechhofer, N. W. Paton, R. Stevens and A. Brass, "An ontology for bioinformatics applications," Bioinformatics, Vol.15, No.6, pp.510-520, 1999.

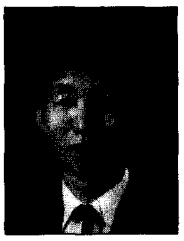
[33] R. H. Li, S. H. Park, B. J. Jeong and K. H. Ryu, "Transformation of heterogeneous data files for bioinformatics," Korean Society for Bioinformatics annual meeting, Vol.1, pp.118-124, 2002.

박 성 희

e-mail : shpark@dblab.chungbuk.ac.kr
1996년 충북대학교 도시공학과(공학사)
1998년 한국전자통신 연구원 컴퓨터소프트웨어 연구소 위촉 연구원
2001년 충북대학교 대학원 전자계산학과 석사(이학석사)
2001년~현재 충북대학교 대학원 전자계산학과 박사과정
관심분야 : Bioinformatics, XML 데이터베이스, 시공간데이터베이스, Bioinformatics 등

이 영 화

e-mail : shpark@dblab.chungbuk.ac.kr
2001년 중국 동북대학교 컴퓨터학과 (이학사)
2003년 충북대학교 대학원 전자계산학과 석사(이학석사)
2003년~현재 연변대학교 컴퓨터과학과
관심분야 : Bioinformatics, XML 데이터베이스, 공간 데이터베이스 등



류 근 호

e-mail : khryu@dblab.chungbuk.ac.kr
1976년 숭실대학교 전산학과(이학사)
1980년 연세대학교 공업대학원 전산전공 (공학석사)
1988년 연세대학교 대학원 전산전공(공학박사)
1976년~1986년 육군군수 지원사 전산실(ROTC 장교), 한국전자통신연구원(연구원), 한국방송통신대 전산학과(조교수) 근무
1989년~1991년 Univ. of Arizona Research Staff(TempIS 연구원, Temporal DB)
1986년~현재 충북대학교 전기전자컴퓨터공학부 교수
관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal GIS 및 지식기반 정보검색 시스템, 데이터 마이닝 및 데이터베이스 보안, 바이오 인포메틱스