이화여자대학교 과학기술대학원
2003 학년도
석사학위 청구논문

웹 서비스를 이용한
바이오 서열 정보 데이터베이스 및
통합 검색 시스템 개발

컴 퓨 터 학 과

이 수 정

2004

웹 서비스를 이용한
바이오 서열 정보 데이터베이스 및
통합 검색 시스템 개발

이 論文을 碩士學位 論文으로 提出 함

2004 年 **1** 月

梨花女子大學校 科學技術大學院

컴퓨터학과 李 受 庭

이 수 정의 碩士學位 論文 을 認准함

指導敎授　　용　환　승

審査委員　　박　승　수　　_____

김　　명　　_____

용　환　승　　_____

梨花女子大學校　科學技術大學院

목 차

그 림 목 차

표 목 차

論 文 槪 要

최근 게놈 프로젝트와 함께 화두가 되고 있는 바이오인포메틱스 (Bioinformatics)란 학문은 생물체와 관련된 정적 및 동적 데이터를 컴퓨터란 도구를 이용해 모델링, 분석, 제어 및 예측 등을 수행하는 분야로써, 바이오 관련 실험 장비들의 자동화가 진행됨에 따라 생명 현상의 신비를 밝히는 데 기본이 되는 바이오 관련 데이터들이 상상을 초월할 정도로 생성이 되고 있다. 이러한 데이터들은 개발 커뮤니티의 목적이나 데이터의 내용만큼이나 다양하게 개발되고 있어 연구자들이 의미 있는 정보를 추출해 내기란 그리 쉬운 일이 아니다. 최근, 개별 데이터베이스를 접근하여 그 결과를 통합, 분석해야 하는, 연구자들의 불필요한 작업을 줄이고자, 바이오 정보를 통합하고 분석하려는 환경을 제공하려는 많은 연구가 진행되고 있다.

본 논문에서는 기존의 통합 검색 시스템에서 사용하는 기법과는 다르게 SOAP 기반 웹 서비스(Web Service)기술을 사용하여 바이오 서열 데이터 통합 검색 시스템을 설계하였다. 우선, 생물 종을 벼와 돼지로 한정하고, 핵산 서열, 단백질 서열 데이터를 BSML 을 포함한 다양한 포맷으로 제공할 수 있는 데이터베이스 서버를 구축하였다. 구축된 데이터베이스에 없는 데이터는 외부 서열 데이터베이스에서 검색하여 결과값을 얻고 그 데이터는 구축한 데이터베이스로 저장한다. 바이오인포메틱스 분야에 웹 서비스를 적용하는 연구는 최근 들어 활발히 진행되고 있는 실정이어서 실제 사용 가능한 서비스는 그리 많다. 본 논문에서는 통합할 외부 서열 데이터베이스로 일본 DDBJ 의 XML Central of DDBJ 와 유럽 EBI 의 XEMBL 서버 두 개로 정하였다.

개발한 시스템은 웹 서비스 구조 3 요소 중, 외부 서열 데이터베이스를 검색 하는 클라이언트 역할을 하는 동시에, 서비스 제공자로써 다양한 데이터베이스 검색 모듈을 작성하여 그것을 웹 서비스하고 WSDL 문서로 공개한다.

I 서 론

1.1 연구 배경 및 목적

80 년대 중반 Human Genome 프로젝트가 시작된 이후로, 분자 생물학 분야의 데이터가 기하 급수적으로 증가하고 있으며, 그 정보들은 개발 커뮤니티의 특징에 따라 다수의 분산되고 이질적인 데이터 저장소에 저장된다. 이로 인해, 연구자들은 의미 있는 바이오 정보를 얻기 위하여, 여러 독립된 데이터베이스를 검색하고, 그 결과를 수작업으로 분석, 변형하며 통합한다. 그러나 현재와 같이 다양한 데이터베이스가 독립적으로 존재하는 경우, 통합적인 정보검색과 새로운 지식 추출은 쉽지가 않다. 이러한 이유로 상호 이질적인 데이터베이스 간의 정보 교환을 위한 데이터 표준화 작업의 필요성이 대두되고 있으며, 바이오 데이터 정보를 통합하고 분석하는데 용이한 환경을 제공하려는 문제 또한 이슈화되고 있다[1, 2, 3]. 기존의 통합 데이터 검색 시스템을 살펴보면, 자체적으로 운영하는 데이터베이스를 여러 개 두어 링크로 연결하거나, 생물학적인 개념을 포괄적으로 표현하기 위하여 거대한 온톨로지 기반으로 데이터를 모델링 하는 방법이 있다. 앞서 설명한 이러한 방법들은 링크 개념의, 공개된 대부분의 생물학 데이터베이스에서 많이 제공하는 상호 참조 방법으로써, 보통 한 방향으로만 이루어지고, 역 방향으로는 이루지지 않는다는 단점이 있다. 그리고 많은 경우 데이터 연결이 직접적으로 연결되는 것이 아니라, 여러 중간 단계를 거쳐 이루어진다. 이러한 점을 살펴볼 때, 비용, 효율 면에서 통합의 문제는 그리 쉬운 일이 아니다.

최근에 들어, 화두가 되고 있는 XML 과 웹 서비스 기술은, 위에서 설명한 바이오 데이터 교환과 통합 문제를 큰 어려움 없이 해결해줄 수 있는 실마리를 제공하고 있으며, 세계적으로 몇몇 연구 기관에서도 현재 연구가 진행 중이다.

XML 이 바이오인포매틱스 분야에서도 이슈가 되고 있는 이유는, 현재 공개된 생물학 데이터베이스들 대부분이 그들의 자료를 평범한 플랫 파일 형태로 배포하기 때문이다. 이는 서로 다른 데이터베이스 들이 똑같은 생물학적 현상을 다른 측면에서 묘사하는 등 플랫 파일이기 때문에 갖는 문제점이 많다. 이러한 문제점을 해결하기 위해서 많은 조직에서 W3C 에서 제안한 XML (eXtensible Markup Language)을 화학, 수학 분야를 포함한 생물학 분야에서도 데이터 관리를 위한 표준으로 채택하고 있는 것이다. 최근에 생물학 분야에서 서열 데이터를 표현하기 위한 목적으로 다양한 표준들이 제안되고 있다[4]. 본 연구의 또 다른 핵심 기술이라 할 수 있는 웹 서비스[5, 6]는 상이한 플랫폼에 상관없이 인터넷이 가능한 곳이라면 쉽게 시스템에 접근 할 수 있도록 하는 기술로, 바이오 데이터베이스 분야에서는 DB 상호간의 연동성을 가지게 하여 의미 있는 정보 추출의 효율성을 높일 수 있게 한다.

본 논문에서는 앞에서 설명한 XML 과 웹 서비스 기술을 이용하여 생물 종을 벼와 돼지 분야로 한정한, 유전체 서열 데이터베이스를 구축하여, 외부 데이터베이스들을 통합 검색하는 시스템을 개발하고, 그 검색 모듈을 웹 서비스로 공개하였다.

1.2 연구 내용

바이오인포매틱스 분야의 데이터 통합, 데이터베이스 시스템 통합의 문제는 데이터의 종류, 데이터의 양만큼이나 어려운 일이다. 본 논문에서는 기존의 통합 검색 시스템과는 차별적으로, 최근 이슈가 되고 있는 웹 서비스 기술을 이용한, 통합 검색 시스템을 제안한다. 시스템을 전체적을 설명하기 전에 우선, 웹 서비스의 3 요소인 서비스 사용자, 서비스 제공자, 서비스 레지스트리 이 세가지 관점에서, 제안한 시스템이 어떠한 내용으로 설계되고 구현되었는지 살펴보고자 한다.

우선, 서비스 사용자로서 본 연구와 같은 목적으로 웹 서비스를 이용하여 자체적으로 개발한 외부 데이터베이스를 조사하여, 그들의 서버로부터 필요한 정보를 얻기 위해 검색 프로그램을 작성하여, 새로운 애플리케이션을 만들었다. 그리고 서비스 제공자로써의 역할을 하기 위해 벼와 돼지 분야 유전체 서열 데이터베이스를 새롭게 구축하였으며, 이 데이터베이스를 검색하는 모듈을 작성하고, 공개 웹 서비스 레지스트리에 등록하였다. 마지막으로 서비스 레지스트리는 관점에서는, 직접 레지스트리를 개발한 것이 아니라, 바이오인포매틱스 분야, 공개 사설 레지스트리인 BioMoby 레지스트리를 UDDI4J 기술을 이용하여 레지스트리에 등록된 서비스를 검색하고, WSDL 파일을 생성시킬 수 있도록 레지스트리 검색 클라이언트를 개발하였다.

또한 제안한 시스템은 웹 서비스 기술을 충분히 반영하는 동시에, 데이터 상호 교환을 편하게 하기 위한 방법으로 데이터를 XML 형태로 표현할 수 있도록 하였다. 생물학 데이터들은 많은 데이터베이스가 생겨나면서 매우 다양한 형식들을 만들어지고 있어, 데이터를 구성하는 통일된 언어나, 이것이 불가능 하면 최소한 데이터베이스 상호간 용어의 호환성을 가질 필요성이 대두되고 있는 실정이다. 구조화된 정보와, 엘리먼트의 의미 표현까지 가능한 XML 은 이러한 문제점을 해결해줄 수 있다. 이러한 의도로 최근 들어 전 세계적으로 바이오인포메틱스 분야에서 서열 정보, 주석 정보 등 다양한 데이터 표현을 위한 XML 표준안(http://www.visualgenomics.ca/gordonp/xml/)이 많이 제안되고 있는데, 본 연구에서는 핵산 서열을 표현하기 위해 BSML 을 채택하고, 데이터베이스 스키마 역시 BSML 의 DTD[7]를 고려하여 작성하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 우선, 바이오 서열 데이터를 플랫 파일과 XML 형태로 구분하여 살펴본다. 그리고 본 논문의 핵심 기술이라 할 수 있는 웹 서비스 기술이 무엇인지 기술하고, 이러한 웹 서비스 기술을 이용하여 최근 바이오 정보

시스템을 통합하려는 다양한 시도들을 간단히 살펴본다. 3 장은 기존의 통합 검색 시스템을 간단히 살펴보고, 개발할 바이오 데이터 통합 검색 시스템 모델을 설계하고 데이터를 어떻게 전처리 하여 데이터베이스화 했는지를 테이블 스키마와 함께 제시하며, 웹 서비스에 기반한 통합 검색 모듈을 웹 서비스 구조에 비추어 설명한다. 4 장에서는 구현된 시스템의 결과로 사용자 질의에 따른 프로세스를 사용자 인터페이스 중심으로 살펴본다. 마지막으로 5 장에서는 본 연구의 결론으로 연구의 의의 및 향후 연구 과제를 제시한다.

II 관련 기술 및 연구 동향

이번 장에서는 유전체 서열 데이터베이스 구축을 위한 데이터와 그 포맷, 그리고 본 연구의 핵심이라고 할 수 있는 웹 서비스 기술에 대해서 소개를 하고, 최근에 웹 서비스를 기반하여 바이오 데이터를 통합하려는 사례들을 살펴보고자 한다.

2.1 바이오 서열 데이터와 그 형식

2.1.1 텍스트 파일 형태의 GenBank 와 FASTA

생물학적 서열 데이터는 GenBank[8, 9], EMBL, PIR과 같은 표준 형식으로 정의된 텍스트 파일이다. 따라서 서열 데이터베이스를 만들 때 이러한 텍스트 파일을 이용해 만들 수 있다. 유전체 서열에 대한 정보가 저장되어 있는 GenBank 텍스트 파일은 미국의 National Center for Biotechnology Information (NCBI)사이트의 GenBank에서 구할 수 있다(ftp://ncbi.nlm.nih.gov/GenBank/genomes/). GenBank의 자료는 DNA DataBank of Japan (DDBJ)과 the European Molecular Biology Laboratory (EMBL)의 자료와 연동되어 매일 자료를 교환하기 때문에 이 세 가지의 데이터 내용은 거의 똑같다고 말 할 수 있다.

데이터 포맷부분을 보면, GenBank 텍스트 파일은 크게 두 부분으로 나눌 수 있다. 하나는 해당되는 생물체의 서열이 언제 밝혀졌으며, 이 서열이 어떤 기능을 하며, 또 어떤 논문에 실렸었는가와 같은 일반적인 내용이다. 그리고 다른 하나는 그 서열의 어느 부분이 유전자고 어느 부분이 단백질로 변하는 부분이며, 그 변한 단백질의 이름은 무엇이고, 어떠한 기능을 하는가와 같은 서열의 모습에 관한 내용이다. 각 내용은 레코드로 기록돼 있다. 이러한 레코드는 각 라인의 몇 번째 열에는 어떠한 내용이 와야 하는지 정의되어 있는데 1~10열에는 키워드가 존재하고, 13열부터 80열까지는 그

키워드에 대한 내용으로 구성되어 있다. 핵산 또는 아미노산 서열 자료는10개 단위로 60개가 11열부터 75열까지 표시된다. [표 2.1]은 1~10열까지 올 수 있는 키워드를 정리한 것이다.

키워드	내 용
① **LOCUS**	짧고 기억하기 좋은 entry의 고유한 명칭. 각 entry에 1개가 꼭 필요하다. LOCUS 이름은 영문자 대문자와 숫자로 구성된다. 첫 자는 영문자이고 10자를 초과할 수 없다. LOCUS 이름 뒤에는 서열의 크기(50 - 350000 bp), 서열이 확인된 분자의 형태(DNA, RNA등), 분류하기 위한 GenBank division code, 자료가 공개된 날짜가 표시된다
② **DEFINITION**	Sequence에 대한 간단한 설명. 각 entry에 1개가 꼭 필요하다. 속명, 종명, 생산물 또는 유전자 이름, 전체 또는 부분 서열 등을 명시한다.
③ **ACCESSION**	GenBank에 보관된 sequence의 고유번호. 자료를 인용할 때는 이 번호를 사용한다. 각 entry마다 꼭 필요하며, secondary accession number가 동시에 표시되는 경우도 있다.
④ **VERSION**	sequence의 accession number와 version number로 구성된 고유번호. gi(geninfo) number라고도 하며 NCBI가 부여한다.
⑤ **NID**	4번의 VERSION과 동일함. 현재 하위 호환성 때문에 있음.
⑥ **KEYWORDS**	entry의 gene products 또는 다른 정보에 대한 짧은 설명. 각 entry마다 꼭 필요하다.
⑦ **SEGMENT**	같은 분자에서 불연속적으로 위치하는 sequence들의 순서에 대한 정보.

⑧ **SOURCE**	생물의 일반이름. 각 entry마다 꼭 필요하며 ORGANISM의 subkeyword가 있다. **ORGANISM** 생물의 공식적 학명(첫줄)과 분류단계별 분류군 이름목록(두 번째 줄부터). 각 entry마다 꼭 필요하다.
⑨ **REFERENCE**	entry에 포함된 자료의 인용문헌. 각 entry마다 꼭 필요하며 복수로 기재 가능하다. AUTHORS, TITLE, JOURNAL, MEDLINE, REMARK의 subkeyword가 있다. **AUTHORS** 인용문헌의 저자들 이름. **TITLE** 인용문헌의 제목. **JOURNAL** 인용문헌이 게재된 학술지명, 권, 연도, 페이지. **MEDLINE** 인용문헌의 Medline 고유번호. **REMARK** 인용문헌에 대한 기타사항.
⑩ **COMMENT**	다른 sequence entry에 대한 상호참조, 다른 collection과의 비교, LOCUS 이름의 변경 및 기타 주석내용.
⑪ **FEATURES**	단백질 또는 RNA를 암호화하는 부분에 대한 정보. 실험으로 증명된 생물학적으로 중요한 부분에 대한 정보. source, gene, CDS 등과 같은 하부 code가 있다.
⑫**BASE COUNT**	핵산서열을 구성하는 각 염기의 빈도수에 대한 요약. 각 entry마다 꼭 필요하다.
⑬ **ORIGIN**	다음 줄부터 표시되는 핵산서열의 첫 번째 염기의 genome내 위치에 대한 정보. Sequence의 시작을 알리는 keyword이므로 내용이 없더라도 반드시 필요하다.
⑭ //	Entry의 끝을 표시하는 기호. 반드시 필요하다.

[표 2.1] GenBank 데이터 키워드

```
LOCUS       AAURRA                   118 bp    rRNA    linear   PLN 13-DEC-1995
DEFINITION  Auricula auricula-judae 5S ribosomal RNA.
ACCESSION   K03160
VERSION     K03160.1  GI:173593
KEYWORDS    5S ribosomal RNA; ribosomal RNA.
SOURCE      Auricularia auricula-judae (ear fungus)
  ORGANISM  Auricularia auricula-judae
            Eukaryota; Fungi; Basidiomycota; Hymenomycetes;
            Heterobasidiomycetes; Heterobasidiomycetidae; Auriculariales;
            Auriculariaceae; Auricularia.
REFERENCE   1  (bases 1 to 118)
  AUTHORS   Huysmans,E., Dams,E., Vandenberghe,A. and De Wachter,R.
  TITLE     The nucleotide sequences of the 5S rRNAs of four mushrooms and
            their use in studying the phylogenetic position of basidiomycetes
            among the eukaryotes
  JOURNAL   Nucleic Acids Res. 11 (9), 2871-2880 (1983)
  MEDLINE   83220825
   PUBMED   6856478
COMMENT     Original source text: Auricularia auricula-judae rRNA.
FEATURES             Location/Qualifiers
     source          1..118
                     /organism="Auricularia auricula-judae"
                     /mol_type="rRNA"
                     /db_xref="taxon:29892"
     rRNA            1..>118
                     /product="5S ribosomal RNA"
ORIGIN
        1 atccacggcc ataggactct gaaagcactg catcccgtcc gatctgcaaa gttaaccaga
       61 gtaccgccca gttagtacca cggtgggggа ccacgcggga atcctgggtg ctgtggtt
//
```

[그림 2.1] GenBank 데이터 예시

FASTA 포맷 서열 파일은 반드시 첫 행을 ">"로 시작하며, 첫 행의 ">"뒤에는 행 바꿈 없이 해당 서열에 대한 어떠한 정보가 쓰여질 수 있다. 즉 ">" 뒤에 파일이름, 서열이름, 설명 등이나 사용자가 임의로 정하는 모든 문자열이 동반 될 수 있으며 이러한 정보들은 반드시 첫 행에서 행 바꿈 없이 존재하여야 한다. 그러나 일반적으로 80 글자를 넘지 않는 것이 좋으며, 그런 다음 두 번째 행부터 AGCT나 ALRC 등과 같은 핵산과 아미노산 정보가 배열되며 이들에 대한 행 바꿈 등에는 사실상 제한이 없다.

다음은 FASTA 형식의 단백질 서열의 예이다 :

```
>gi|532319|pir|TVFV2E|TVFV2E envelope protein
ELRLRYCAPAGFALLKCNDADYDGFKTNCSNVSVVHCTNLMNTTVTTGLLLNGSYSENRT
QIWQKHRTSNDSALILLNKHYNLTVTCKRPGNKTVLPVTIMAGLVFHSQKYNLRLRQAWC
HFPSNWKGAWKEVKEEIVNLPKERYRGTNDPKRIFFQRQWGDPETANLWFNCHGEFFYCK
MDWFLNYLNNLTVDADHNECKNTSGTKSGNKRAPGPCVQRTYVACHIRSVIIWLETISKK
TYAPPREGHLECTSTVTGMTVELNYIPKNRTNVTLSPQIESIWAAELDRYKLVEITPIGF
APTEVRRYTGGHERQKRVPFVXXXXXXXXXXXXXXXXXXXXXXVQSQHLLAGILQQQKNL
LAAVEAQQQMLKLTIWGVK
```

[그림 2.2] FASTA 데이터 예시

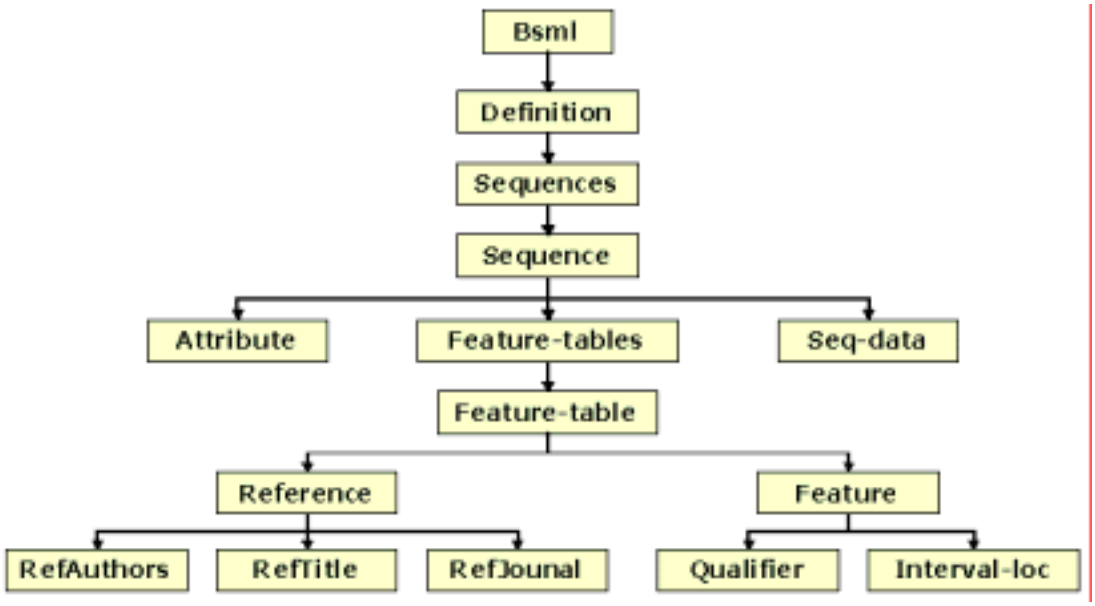
2.1.2 XML형태의 BSML

BSML(Bioinformatic Sequence Markup Language)[7]은 바이오 정보 시스템 간 농생물 데이터의 상호 교환을 편하게 하기 위하여 만들어진 XML공개 표준으로써, 1997년 National Human Genome Research Institute(NHGRI)에 의해 유전체 실험 정보를 교환하기 위한 방안으로 처음 고안되었다. BSML은 유전체 서열 정보를 담은 또 다른 XML 표준들과 다르게 서열 정보를 표현함은 물론, 서열의 직접적인 표현(data tables, visualization 등)의 범위 밖에서 다른 형식들에서 이용할 수 없는 많은 서열들의 특징을 지원 할 뿐만 아니라, 그것의 생체 물리학적인 특성 부분까지 표현할 수 있다. 현재 바이오 인포매틱스의 대표적인 회사인 Double Twist뿐만 아니라, Fujitsu, IBM, LabBook, Inc, EBI등의 업계에서 새로운 차세대 표준으로 BSML을 채택하고 있다. 이런 흐름에 맞추어, 본 연구에서도 유전체 서열 정보 표현을 위한 표준으로 BSML을 채택하였다.

다음은 BSML이 표현하는 서열 정보의 세가지 타입이다.

1. Definition 부분 : DNA, RNA, 단백질 서열과 같은 생물학적 분자 서열과 덧붙여 features라 불리는 서열 주석정보, 또한 분석 결과 등을 표현함.
2. Research 부분 : 쿼리, 분석, 실험적 프로토콜
3. Display 부분 : 바이오 정보를 가시화 하기 위한 그래픽 메타포로써, 서열 데이터의

특성을 어떻게 표현할 것인가 하는 메커니즘을 제공.



[그림 2.3] GenBank 를 고려한 BSML DTD 의 일부

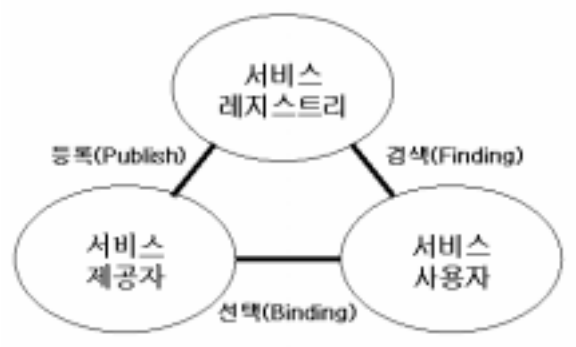
참고

	AGAVE	GAME	BSML	DAS	BIOML
Creator	DoubleTwist,Inc.	BDGP	Labbook,Inc	Lincoln Stein et al. at CSHL	
Version	Version 3.0, released in September 2001	Version 1.1, released in March 2000	Version 3.1, released in Jan 2002	Version 1.5, released in Feb 2002	released in Mar 1999
Tool Support	• AGAVE Java Library • Genomic Viewer • Conversion Tool	• Bioperl libraries	• Genomic Viewer • Genomic Browser	• Web-based protocol	
Data	• DNA • RNA • Protein	• DNA • RNA • Protein	• DNA • RNA • Protein	• DNA • RNA • Protein	• DNA • RNA • Protein
Goal	• 유전체 주석 정보표현	• 유전체 주석정보의 상호 교환을 수월히 하기 위함	• biological sequences와 그것의 특징, 기능 들을 종합적으로 기술하기 위함 • 시퀀스 정보의 그래픽 표현을 가능하게 함.	• 다수의 서버 사이의 정보를 통합하려는 목적	• bioPolymer 시퀀스 정보를 표현

[표 2.2] 바이오 서열 데이터를 표현하기 위한 다양한 XML 표준들

2.2 웹 서비스

웹 서비스는 XML 메세징을 통하여 인터넷을 통하여 접근 가능한 네트워크 명령들의 집합을 기술한 인터페이스로서, 단일 또는 서로 다른 플랫폼과의 상관 관계없이 상호운영(Interoperating)이 가능하도록 해주는 표준화된 기술이다. 웹 서비스는 [그림 2.4]에 나타난 것과 같이 크게 세 가지로 구성된다.



[그림 2.4] 웹 서비스 구조

- 서비스 제공자 (Service Provider) : 웹 서비스를 개발하는 역할을 함. 공개적으로 접근할 수 있는 중앙 레지스트리에 웹 서비스의 세부사항을 공개(Publish)함.

- 서비스 사용자 (Service Consumer : Client) : 웹 서비스를 사용하는 애플리케이션을 의미함. 레지스트리를 검색(Find)하여 서비스의 세부사항을 알아내고. 그 후 원하는 서비스에 binding하여 실제로 해당서비스의 기능을 이용함.

- 서비스 레지스트리 (Service Registry) : 서비스 제공자가 자사의 웹 서비스 상세내역을 올려두고 Client가 발견할 수 있도록 하는 중앙 저장소를 의미함.

[그림 2.4]에 보이는 세 가지 오퍼레이션[검색, 선택, 등록(공개)]에 관한 표준안은 WSDL (Web Service Description Language), SOAP(Simple Object Access Protocol), UDDI(Universal Description, Discovery, and Integration)로 구성된다. 각각에 대해서 살펴보면 다음과 같다.

□ **WSDL** : 웹 서비스 기술 (description)에 관한 표준안

WSDL은 서비스에 대한 자세한 정보를 명시하는 데 사용하는 언어로 애플리케이션이 서비스를 어떻게 호출하는지와 어떤 매개변수들이 필요한지를 알게 해준다. 실제 WSDL 문서는 복잡하며 import, type, schema, message, port, binding, service와 같은 엘리먼트들을 포함하고 있다.

□ **SOAP** : 웹 서비스 선택(Binding)에 관한 프로토콜

SOAP은 XML포맷을 사용하여 애플리케이션이 웹서비스의 메소드를 호출할 수 있도록 하는 인터넷 프로토콜로서, 서비스가 요구하는 정보를 포함하고 잇는 문서를 애플리케이션에서 서비스로 건네주고 응답을 받는다. SOAP역시 XML에 기반하기 때문에 애플리케이션이 동작하는 운영체제, 언어 그리고 객체 모델에 관계없이 애플리케이션 사이의 통신을 할 수 있게 된다.

□ **UDDI** : 웹 서비스 등록(publish)와 검색(find)에 관한 표준안

UDDI는 클라이언트가 동적으로 웹 서비스를 찾을 수 있게 하는 방법을 제공한다. UDDI 인터페이스를 이용하면 비즈니스는 외부 비즈니스 파트너가 제공하는 서비스에 동적으로 연결 할 수 있다. UDDI 레지스트리는 두 가지 종류의 클라이언트를 가지고 있다. 첫째는 서비스를 공개(Publish)하려는 비즈니스이고, 둘째는 서비스를 얻으려는 클라이언트이다.

2.3 바이오 데이터 통합 시도 연구 동향

2.3.1 기존의 바이오 데이터 통합 접근 방법

최근 몇 년을 거치면서, 몇 종류의 고도로 정교한 생물학적 검색 엔진이 만들어졌다. 미국 국립 생물 정보 센터(National Center for Biotechnology Information, NCBI)에서 보유하고 있는 데이터에 대한 검색 시스템으로 Entrez[10]를 예로 들 수 있는데, 이는 핵산 염기 서열, 단백질 아미노산 서열, 단백질 3차구조, 유전차 GENOME 정보, 분류학적 정보, 문헌 데이터베이스인 Pubmed를 상호 참조(Cross-Reference) 방법을 통하여 검색, 접근 할 수 있도록 한다. Entrez는 통합 시스템의 전역적인 데이터 모델이 존재하지 않고, 자제적으로 운영되는 내부 데이터베이스들을 하이퍼 링크방식으로 통합한다는 점과, 여러 데이터베이스를 동시에 검색할 수 없다는 점을 단점이라 할 수 있다.

또 다른 예로, LionBioscience에서 상업적으로 판매중인 SRS(Sequence Retrieval System)[11] 시스템은 전 세계에 각각 따로 설치되어 있는 databank간의 연결을 최대한 이용하여, 사용자가 사실상 모든 분자 생물학 databank들을 한번에 검색 할 수 있게 한다. 이는 SRS가 databank의 구조와 데이터 field의 syntax를 묘사하는 특별히 디자인 된 컴퓨터 언어를 사용함으로써 가능한 일이다. SRS는 설치, 유지가 쉬워 웹 사이트로 운영하기 적당하며, 새롭게 만들어진 데이터베이스를 자체적으로 제공하는 Wrapper 제작 언어를 이용하여 적당한 서술과 색인 작업으로 SRS에 첨가할 수 있다는 장점이 있다.

이러한 Entrez와 SRS같은 시스템은 링크 기반의 통합 방법이며, 이 외의 방법으로는 뷰 기반 통합, 데이터웨어 하우징(DW) 구축을 통합이 있다.[12] 이들 세 가지 방법의 특징을 살펴보면 다음 [표 2.3]과 같다.

구 분	설명	
링크 기반의 통합 (link Integration)	특징	□ 가장 친숙하고 접근하기 쉬운 방법 □ 웹 기반의 하이퍼링크를 통해 데이터 접근
	단점	□ 상호 링크된 데이터베이스 간, 데이터의 모호성 □ 잘못된 링크를 따라가기 쉬움 - 업데이트 된 페이지에서 그렇지 않은 페이지로의 이동 □ 통합과 해석의 문제가 연구자의 책임
	예	□ SRS, Entrez 대부분의 통합 검색 시스템에서 채택
뷰 기반 통합 (View Integraion)	특징	□ 정보는 소스데이터베이스에 두면서, 하나의 큰 데이터베이스 환경을 구축하여 각각의 시스템을 전체 시스템의 한 부분으로 구성하는 방법 □ 쿼리 언어 분석하여 서브 쿼리로 나눔 - 얻고자 하는 정보를 가지고 있는 소스 데이터베이스로 서브 쿼리를 보냄 - 결과를 변환, 통합하는 소프트웨어 필요
	단점	□ 쿼리 처리 기능이, 가장 늦은 데이터 소스에 의해 제한됨
	예	□ Klesli/K2 - 펜실베니아 대학
데이터웨어하우징 (DW) 구축 (Data warehousing)	특징	□ 단일 데이터베이스, 즉 DW로 구축 □ 소스 데이터베이스들의 데이터 의미를 통합된 데이터 모델 사용 □ 다양한 SW 작성 - 소스 데이터베이스로부터 데이터를 가지고 오는 기능 - 가지고 온 데이터를 통합 데이터 모델로 변환 시키는 기능 - 통합 모델로 변환된 데이터를 DW로 로딩하는 기능
	단점	□ 소스데이터베이스에 새롭게 추가된 데이터의 실시간 업로드 문제 □ 소스 데이터베이스의 스키마 변경시 DW의 업그레이드 문제 - 소프트웨어의 기능 추가(버전 업 된 새로운 SW 개발 필요
	예	□ IGD Project(Integrated Genome Database)

[표 2.3] 기존의 바이오 데이터 통합 방법 비교

2.3.2 웹 서비스를 이용한 통합 시도

본 연구에서는 앞 절에서 소개한 두 시스템을 포함한, 공개된 생물학 데이터베이스에서 많이 사용하고 있는 Cross-Reference방법이 아닌, 새로운 접근방법으로, 최근 바이오 정보 통합 시스템분야에서 화두가 되고 있는 웹 서비스 기술을 통한 통합 검색 시스템을 구축하고자 한다. 아래(가)~ (라)에서는, 현재 웹 서비스를 이용하여 바이오 데이터를 제공하려는 연구를 소개한다.

(가) BioMoby

BioMoby[13]는 웹 서비스를 통하여 바이오 데이터를 발견하고 배포하는 구조 개발을 목표로 하고 있는 Open Source Research 프로젝트로써 시스템 내부의 포맷이나 스키마에 상관없이 클라이언트가 바이오 데이터를 가지고 있는 많은 수의 Source들과 상호 작용할 수 있도록 해주는 시스템을 개발하고 있다.

기존의 많은 수의 바이오 정보나 분석 툴을 서비스 하고 있는 호스트들은 대부분이 이질적인 CGI 기술기반의 인터페이스를 통하여 사용자들에게 서비스를 제공하고 있으나, BioMoby는 웹 서비스를 통하여 사용자들에게 이질적으로 분산된 데이터베이스에 손쉽게 접근 가능토록 해주고 잇는 점에서 의의가 있다.

(나) BioDAS

BioDAS(Bio Distributed Annotation System)는 유전체 서열 데이터의 주석 정보를 상호교환하기 위한 오픈 소스 시스템으로, 지리적으로 분산된 여러 주석 데이터 베이스들에서부터 정보를 모아서 하나의 통합된 관점으로 데이터를 표현할 수 있도록 한다. 현재, TIGR, Wormbase, Ensembl, Flybase서버가 DAS/2 프로토콜로 데이터를 제공하고 있다.

(다) XML Central of DDBJ

DDBJ(DNA Data Bank of Japan)[14]는 동양에서 가장 활발한 활동을 하고 있는 DNA 유전자 은행으로써 1986년 일본 National Institute of Genetics(NIG)에서 시작하였다. DDBJ는 10여년 간을 GenBank/NCBI, EMBL/EBI 와 함께 사용자들에게 Primary database를 제공하기 위해서 협력해왔고, 그 결과 데이터의 양은 기하 급수적으로 늘어났다. 그리하여 사용자들은 이러한 데이터들을 로컬 사이트에 다운로드 받기가 어려워 짐에 따라 , Center for Information Biology and DDBJ에서는 사용자에게 데이터들을 직접 다운로드 받지 않고 데이터에 접근 할 수 있는 컴퓨터 환경을 만들고자 하였는데 이것이 바로 웹 서비스를 이용한 시스템이다. DDBJ SOAP 서버와 WSDL의 조합으로 , 사용자로 하여금 데이터베이스나 툴의 개발 없이도 자신만의 Computational 환경을 구축 할 수 있도록 하였다. 아래 [표 2.4]은 XML Central of DDBJ에서 제공하는 서비스 목록을 나타내며, 음영으로 나타낸 부분은 본 연구에서 통합하는 서열 검색 관련 서비스이다.

서비스 이름	Web Service Description language
Blast Demo	http://xml.nig.ac.jp/wsdl/BlastDemo.wsdl
Blast	http://xml.nig.ac.jp/wsdl/Blast.wsdl
ClustralW	http://xml.nig.ac.jp/wsdl/ClustalW.wsdl
DDBJ	http://xml.nig.ac.jp/wsdl/DDBJ.wsdl
ExClustalW	http://xml.nig.ac.jp/wsdl/ExClustalW.wsdl
Fasta	http://xml.nig.ac.jp/wsdl/Fasta.wsdl
GetEntry	http://xml.nig.ac.jp/wsdl/GetEntry.wsdl
Gtop	http://xml.nig.ac.jp/wsdl/Gtop.wsdl
SRS	http://xml.nig.ac.jp/wsdl/SRS.wsdl
TxSearch	http://xml.nig.ac.jp/wsdl/TxSearch.wsdl

[표 2.4] XML Central of DDBJ 가 제공하는 서비스 목록

(라) XEMBL

영국의 EBI(European Bioinformatics Institute)의 XEMBL서비스[15, 16]는 EBI의 EMBL 핵산 서열 데이터베이스를 다양한 XML형태로 제공하고자 하는 목적으로 만들어진 것으로, 웹 서비스 기술을 이용하여 EMBL데이터의 XML 형태의 접근을 쉽고 간단하게 해줄 수 있는 인터페이스를 제공한다. 개발자는 XEMBL사이트에서 XEMBL서비스의 WSDL 파일을(http://www.ebi.ac.uk/xembl/XEMBL.wsdl) 다운로드 받은 후, 웹 서비스 툴킷을 이용하여 Stub과 Skelleton 파일을 생성함으로써, XEMBL사이트의 내부 시스템의 플랫폼을 몰라도 EBI의 EMBL데이터베이스에 접근 가능한 독자적인 애플리케이션을 작성할 수 있다. XEMBL이 현재 지원하고 있는 XML표준으로는 BSML과 AGAVE 두 가지이며, 앞으로 BIOML과 GAME을 지원하겠다고 말하고 있다.

[그림 2.5] XEMBL 서비스 - 사용자 인터페이스

본 절에서는 다양하게 존재하는 바이오 정보 데이터들을 통합하려는 시도[17]들에 대해서 간단히 살펴보았다. 이러한 조사를 토대로, 본 논문에서 제시하고자 하는 시스템은 생물 종을 몇 가지 한정한 자체 데이터베이스를 구축하면서, 그 데이터베이스를 포함한 외부 데이터베이스까지 검색하는 모듈을 사용자가 이용하기 쉬운 웹 인터페이스로 JSP 프로그래밍을 하고, 검색 모듈을 웹 서비스로 공개하여 외부 사용자로 하여금 본 연구에서 구축한 데이터베이스를 검색 할 수 있도록 하였다. 또한 XEMBL 서비스를 벤치마킹하여,

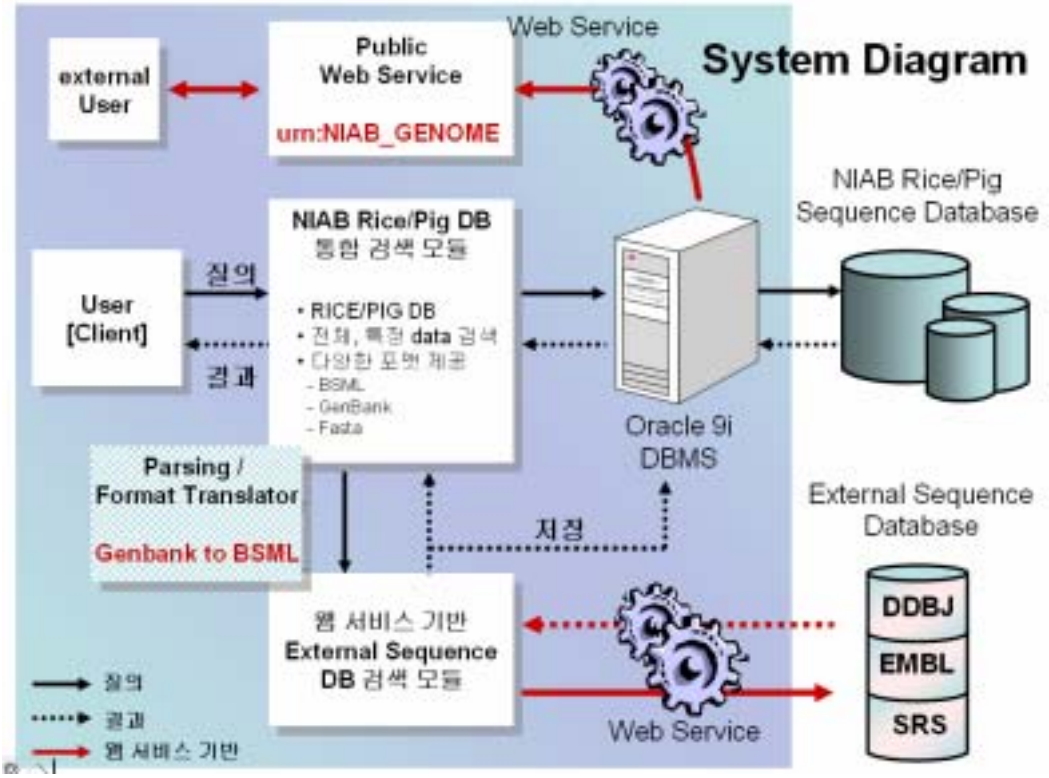
바이오 서열 정보를 BSML 을 포함한 다양한 형식으로 사용자에게 제공할 수 있도록 설계하였다. 아래 [표 2.5]는 앞에서 설명한 여러 통합 시도들을 간단하게 정리 하여 본 것이다.

	웹서비스 지원	자체 DB 구축	검색 범위	특징
BioMoby	○	X	레지스트리에 등록된 서버	- 공개 소스 프로젝트 - 바이오 데이터를 발견하고 공유하고자 서버들을 등록하는 자체 레지스트리 운영
BioDAS	○	X	DAS 프로젝트에 참여하는 서버	- 공개 소스 시스템 - Client-Server system
XML Central of DDBJ	○	○	DDBJ 서버	- 현재, 10 개의 웹 서비스 서비스를 제공함. - 웹서비스 검색 인터페이스 미약
XEMBL	○	○	EBI 서버	- EBI 에 있는 서열 정보 데이터들을 BSML, AGAVE 등 표준화된 XML 로 제공함.
SRS (통합 검색)		○ DB 추가 기능		- 상용 프로그램(LionBioscience) - 확장성에 중심을 둠 - 자체 Wrapper 언어 이용하여 DB 통합 가능
Entrez (통합 검색)		○ NCBI DB		- 통합적인 database 검색 시스템 - DNA 와 protein sequence data 뿐만 아니라 연관된 MEDLINE reference 와 GenBank 의 genome data, taxonomy, protein 3 차 구조 database 인 MMDB 의 검색을 제공하는 시스템이다

[표 2.5] 연구 동향 비교

III. 바이오 서열 정보 통합 검색 시스템 설계

본 장에서는 바이오 서열 정보에 대한 통합 검색 시스템을 설계하고자 한다. 우선 [그림 3.1]은 유전체 서열 웹 서비스 DB 통합 검색 시스템의 전체 구성도이다. 사용자의 입장에서 검색 할 수 있는 데이터베이스는 크게 NIAB <개발한 유전체 서열 DB을 이하 NIAB 라 칭함 > 데이터베이스와 외부 서열 데이터베이스 등 두 부분으로 나눌 수 있다. 우선 사용자는 NIAB DB를 검색하고, 만약 NIAB DB에 존재하지 않는 아이템에 대한 검색을 할 경우, 웹 서비스를 통한 다른 외부 서열 데이터베이스로의 검색을 가능하도록 한다. 또한 NIAB 데이터베이스 검색 모듈을 웹 서비스 제공자의 입장에서 공개를 한다. 이에 관련된 내용은 아래에서 자세히 설명을 하겠다.



[그림 3.1] 시스템 구성도

3.1 바이오 서열 정보 DB구축

현재 NCBI GenBank 데이터베이스에는 핵산 서열이 벼 유전자 595567개, 돼지의 유전자 169416개가 등록되어 있으며, 단백질 서열이 벼 64937개, 돼지 9335개가 등록되어 있다. 이중에서 핵산 서열은 EST서열만을 우선적으로 Bulk로 다운 받았고, 단백질 서열은 벼, 돼지 두 가지 모두 등록되어 있는 것을 모두 다운로드 받았다.

다운로드 받은 데이터는 사람이 읽고 해석하기 편한 텍스트파일의 형태로 ASCII 텍스트 파일이며, 그 크기는 모두 합쳐 1GB가 훨씬 넘는다. 이렇게 큰 텍스트 형태의 데이터를 데이터베이스로 구축하기 위해서는 데이터 전처리 과정이 필요하다. 다운로드 받은 데이터의 형식은 크게 GenBank 포맷과 FASTA 포맷 두 가지이고, 이러한 형태의 파일을 처리하기 위해 BioJava라이브러리[18]를 이용하여 자바 전처리 프로그램을 작성하여 구동하였다.

3.1.1 BioJava 라이브러리를 이용한 데이터 전처리

NCBI에서 다운로드 받은 서열 데이터의 형태는 *.gbk란 이름의 텍스트 파일이며, 데이터베이스를 구축하기 위해서는 이러한 텍스트 파일의 데이터를 테이블의 형식에 맞게 넣어야 한다. 테이블에 데이터를 입력하는 방법은 크게 두 가지로 나눠 볼 수 있다. 첫 번째는 사용자가 다운 받은 텍스트 파일을 하나하나 열어서 눈으로 보면서 SQL문으로 직접 하나씩 입력하는 것이다. 하지만 GenBank 텍스트 파일의 크기를 보면 보통 수십 MB에서 수백 MB까지 되는 것도 있으므로 눈으로 하나하나 보면서 입력한다는 것은 불가능하다. 이러한 어려움을 해결하기 위해 본 연구에서는 텍스트 파일을 분석해 파싱한 다음 그 결과로 나온 데이터를 자동으로 입력하는 프로그램을 작성하였다. GenBank텍스트 파일의 형식은 관련 연구 부분에서 살펴봤듯이 각

키워드들이 잘 정리되어 있고, 일정한 형식으로 지정되어 있기 때문에 파싱 작업을 하는데 커다란 무리는 없었다.

현재 생물학 데이터를 처리하기 위해 BioPerl, BioXML, BioPython, BioJava 등의 공개 프로젝트가 진행되고 있다, 이 중, 본 연구에서는 GenBank 텍스트 파일 파싱 프로그램을 만들기 위해 BioJava를 사용하였다. BioJava를 사용하기 위해서는 BioJava를 설치해야 한다. BioJava는 http://www.biojava.org/download/binaries에서 구할 수 있으며, 바이너리 파일을 다운 받으면 된다. 현재 BioJava는 버전 1.3이 배포되고 있으며. 본 연구에서는 1.2.2 버전을 사용하였다.

GenBank 텍스트 파일의 데이터 포맷은 키워드와 내용으로 구성되어 있다. 주어진 키워드에 대해 관련 내용이 기술되는 형식을 취하고 있다. 키워드는 반복되어 나타나는 키워드와 반복이 없는 키워드로 나누어진다. 반복 없는 키워드는 어떠한 GenBank 파일에서도 오직 한 번씩만 나타나는 것으로서, 이는 BioJava 라이브러리 내에서 제공되는 GenBank 텍스트 파일 지원 클래스를 이용하여 파싱할 수 있다. 그러나 반복되는 키워드의 경우는 각 GenBank 파일마다 각기 다른 형태를 가지며, 반복되는 횟수에도 차이가 있다. 반복되는 키워드 부분은 GenBank데이터의 상당 부분을 차지하고 있으며, 이로 인해 반복되는 키워드가 너무 많은 GenBank 파일의 경우는 파싱 작업 시 메모리 과부하가 일어난다. 이 경우 BioJava 라이브러리만으로는 파싱이 불가능하며, 따라서 이를 위해 클래스를 수정, 개선하였다. 본 논문에서는 새롭게 작성한 함수를 이용하여 반복 없는 키워드와 마찬가지로 반복되는 키워드에 해당하는 내용을 파싱할 수 있도록 하였다. FASTA 포맷의 파일은 구성자체가 서열 Description부분 한 줄과 나머지 서열로써 구성되어 있어 파싱 작업을 GenBank 파싱에 비해 훨씬 수월하게 할 수 있었다.

BioJava와 함께 연구자들이 많이 사용하고 있는 대표적인 공개 프로젝트로는 BioPerl, BioPython 등이 있으며, 아래에 간단히 그 둘의 특징을 설명하겠다.

(가) BioPerl

BioPerl[19]프로젝트는 생명 정보학, 생명과학 연구에 필요한 프로그램을 작성할 때 도움이 되는, 모듈들을 perl로 작성해 놓은 소스들을 모으는 공개 프로젝트이다. 1998년 ISMB 뭉 OIB Conferences에 논문으로 실렸으며, 현재 BITS(Bio Informatics Technology & System)저널 사이트에 그 논문 수정 본이 나와 있다. BioPerl 핵심 모듈들은 sequence, sequence alignments, BLAST sequence comparisons 그리고 3차원 분자 구조에 관한 것이고 , perl의 object-oriented features로 구현되어 있다. 그리고 각 perl모듈들의 이름은 그것이 존재하는 디렉토리를 포함해 지어져 있다.

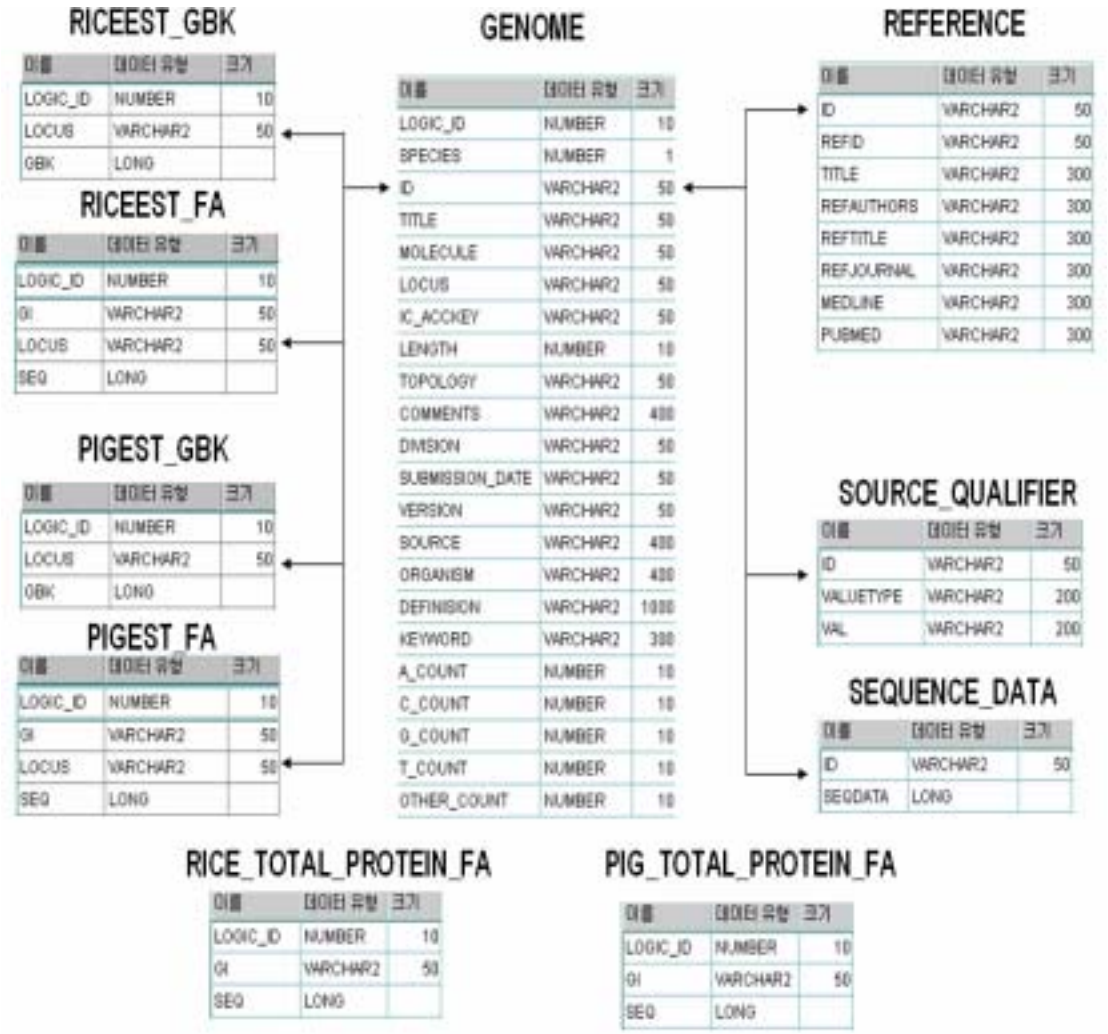
(나) BioPython

BioPython[20]은 생물정보학에 자주 활용되는 기능과 라이브러리들을 python으로 구축해 놓은 패키지이다. python의 명료한 문법 체계와 객체 지향 기법이 적절하게 생물학적 요구와 맞물려 구현했으며, BioPerl, BioJava와 마찬가지로 오픈 소스 프로젝트이다. BioPython에는 각종 서열들의 텍스트 파일 형식을 파싱해 python에 이용할 수 있는 데이터 형태로 만들거나, 온라인 바이오인포매틱스 자원들에 대해 python으로 접근하고 이용할 수 있는 기능, 유용한 바이오인포매틱스 관련 프로그램들(blast, Clustalw)을 사용할 수 있는 기능 등이 있다.

이외에도 BioXML, BioDAS, BioCorba가 있다.

3.1.2 데이터베이스 스키마

데이터베이스 내에 있는 서열 정보를 BSML포맷에 맞추어 제공할 수 있어야 한다는 부분이 스키마를 설계하는데 있어 가장 중요하게 고려되었던 점이다. 서열 정보의 일반적인 정보를 담은 GENOME 테이블을 포함하여 모두 10개의 테이블로, 핵산과 단백질 서열 정보를 충분히 표현할 수 있도록 했다.



[그림 3.2] 데이터베이스 스키마

Table name	Table 설명
GENOME	해당되는 생물체의 서열이 언제 밝혀졌는지, 이 서열이 어떤 기능을 하고, sequence의 핵산의 구성이 어떻게 되는지에 대한 일반적 정보를 저장한다.
REFERENCE	관련된 논문에 관한 내용, 논문의 저자, 제목, 저널의 이름과 같은 정보저장.
SOURCE_QUALIFIER	서열의 길이, source organism의 이름, Taxon ID 번호 와 같은 각 레코드에 있는 Feature의 설명
SEQUENCE_DATA	Sequence 서열을 저장함.
RICEEST_GBK	종이 RICE이고 GenBank형식의 유전자 서열파일을 저장함
RICEEST_FA	종이 RICE이고 FASTA형식의 유전자 서열 파일을 저장함
RICE_TOTAL_PROTEIN_FA	종이 RICE이고 FASTA형식의 단백질 서열을 저장함
PIGEST_GBK	종이 PIG이고 GenBank형식의 유전자 서열파일을 저장함
PIGEST_FA	종이 PIG이고 FASTA형식의 유전자 서열 파일을 저장함
PIG_TOTAL_PROTEIN_FA	종이 PIG이고 FASTA형식의 단백질 서열을 저장함

[표 3.1] 데이터베이스 테이블 설명

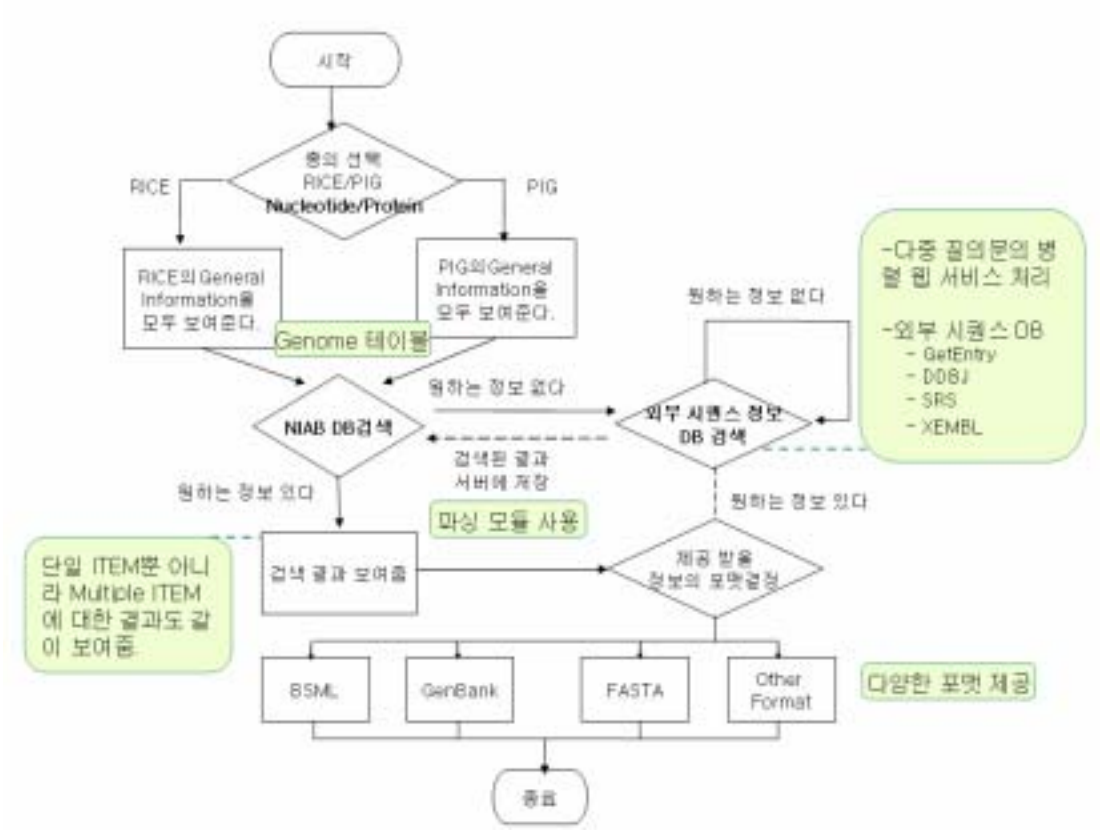
3.2 웹 서비스 기반 통합 DB 검색 및 검색 모듈 공개

3.2.1 웹 서비스 기반 바이오 서열 DB 통합 검색

메타 검색이란 여러 데이터베이스를 일일이 찾아 다니면서 검색하지 않고, 한 번에 검색 결과를 확인할 수 있는 장점이 있는 검색 방법을 말한다. 본 연구에서도 이러한 메타 검색 개념을 적용하여 NIAB 데이터베이스에 없는 정보에 대해서는 웹 서비스를 통한 외부 서열 데이터베이스로 검색을 할 수 있도록 했다. 사용자가 입력한 키워드로 검색된 결과가 벼나 돼지에 관련된 정보라면 사용자에게 검색 결과를 보여주는 동시에 내부적으로 NIAB 데이터베이스에 파싱되어 저장되도록 하였고, 그렇지 않다면

사용자에게 검색된 결과만을 보여주게 하였다. 다시 말해서 이 시스템은 시간이 지남에 따라, 즉 벼나 돼지에 관련된 사용자의 질의가 많아질수록, 그 만큼의 데이터가 데이터베이스에 저장되어 정보가 축적되게 된다.

아래 [그림 3.3]은 사용자가 질의를 입력했을 때의 검색 프로세스를 나타낸 것으로, 사용자는 우선 얻고자 하는 정보가 핵산 서열인지, 단백질 서열인지를 선택을 하고, 그 후에 벼와 돼지와 같은 종을 선택한다. 선택 파라티터 값에 의하여 우선적으로 NIAB 데이터베이스에 있는 전체 서열 아이템의 일반적인 정보를 보여주게 되며, 사용자가 Accession Number와 같은 서열의 키워드 정보를 추가로 입력함으로써 해당 아이템의 정보가 NIAB데이터베이스에 존재하는지 확인 할 수 있다. 키워드로 입력한 정보에 해당하는 정보가 NIAB 데이터베이스에 존재 하는 경우에는 검색 결과를 사용자 인터페이스를 통해 보여주고 다양한 포맷으로 데이터를 저장할 수 있도록 사용자에게 물으며, 만약 얻고자 하는 정보가 NIAB데이터베이스에 존재 하지 않을 경우에는 자동적으로 외부 서열 데이터베이스를 검색할 수 있도록 사용자 인터페이스를 구성한다. 사용자는 그 인터페이스를 통하여 외부의 데이터베이스를 선택적으로 접근 할 수 있다.



[그림 3.3] 시스템 작업 흐름도

여기서 외부 시스템에 접근 할 수 있도록 하는 메커니즘이 바로 웹 서비스 기술이며, 본 연구에서는 뒤에서 설명할 서비스를 제공자로써 DB검색 모듈을 공개함은 물론, 외부 시스템을 접근 하는 클라이언트 측 프로그램으로써의 역할을 할 수 있도록 시스템을 설계하였다. 외부 서열 데이터베이스가 공개하는 해당 서비스의 WSDL(Web Service Description Language)파일을 다운로드 받은 후, 웹 서비스 툴킷에 포함되어 있는 JAVA2WSDL 툴을 사용하여 클라이언트 측 스텁 코드를 생성했다. 클라이언트-사이드 스텁은 마치 메소드가 로컬에 있는 것처럼 애플리케이션이 원격 웹 서비스의 메소드를 호출할 수 있게 해준다.

아래 [표 3.2]는 본 시스템에서 접근하는 외부 서열 데이터베이스의 목록이다. 서버는 크게 일본의 XML Central of DDBJ와 유럽의 EBI 두 개이며, 서비스 개수는 모두 GetEntry, SRS, DDBJ, XEMBL 등 모두 4개이다. <자세한 함수 설명은 부록 표를 참조>

서비스이름	제공기관	함수 개수	역 할
GetEntry	XML Central for DDBJ	27개	Accession Number등등의 서열 해당정보로 결과 추출
SRS	XML Central for DDBJ	2개	원하는 데이터베이스에 SRS쿼리 폼으로 질의 가능
DDBJ	XML Central for DDBJ	6개	Locus, Gene, Project 정보를 입력으로 하여 DDBJ 엔트리 추출
XEMBL	EBI	1개	Accession Number를 받아 BSML, Agave형태의 XML포맷으로 데이터 추출

[표 3.2] 외부 서열 데이터베이스 검색 서비스

3.2.2 웹 서비스 제공자로서 DB 검색 모듈 공개

앞에서는 유전체 서열 정보를 담아 구축한 데이터베이스의 검색과 외부 데이터베이스를 어떻게 통합하여 검색할 것인가 하는 문제에 대해서 설명을 하였다. 본 시스템은 외부 데이터베이스를 검색하는 웹 서비스 아키텍쳐의 서비스 클라이언트 역할을 함은 물론, 또 다른 외부 사용자가 구축한 NIAB 유전체 서열 DB에 접근 가능토록 즉, 서비스 제공자의 역할 도 할 수 있도록 앞에서 개발한 다양한 검색 서비스를 웹 서비스로 공개 하였다. 서비스 이름은 urn:NIAB_GENOME이라 하였으며 그 서비스 내에 이용할 수 있는 함수는 모두 8가지로 구성하여 개발하였다. 8개 함수는 아래에 간단히 설명 하며, 이에 대한 자세한 설명은 5장 구현부분에서 설명하겠다.

함수 명	함수 설명
getNIAB_GENOME_Info	서열 데이터의 일반적인 정보 검색
getNIAB_GENOME_Sequence	서열 데이터만을 검색
getNIAB_REFERENCE_List	서열에 관련된 논문에 관한 내용, 저자 등 검색
getNIAB_GENBANKEntry	해당서열의 GenBank 파일 검색
getNIAB_GI_Number	해당 서열의 GI Number 검색
getNIAB_Nuc_FASTAEntry	해당 서열의 FASTA 파일 검색
getNIAB_Protein_FASTAEntry	해당 단백질 서열의 FASTA 파일 검색
getNIAB_BSMLEntry	핵산 서열의 BSML파일 검색

[표 3.3] 8 개의 NIAB 데이터베이스 검색 모듈

3.2.3 웹 서비스 레지스트리 검색 모듈

웹 서비스 기술은 사용자의 직접적인 제약 없이, 서비스 개발자가 서비스를 가능한 많은 클라이언트에게 제공할 수 있어야 하고, 클라이언트 역시 많은 서비스 제공자들로부터 원하는 서비스를 찾을 수 있어야 한다. 이러한 문제를 다루고 있는 기술이 UDDI로써, UDDI는 비즈니스와 해당 서비스에 대한 정보를 구조화된 방법으로 수용하기 위해 디자인된 공용 레지스트리로, UDDI를 통해 개발 커뮤니티와 커뮤니티가 제공하는 웹 서비스에 대한 정보를 게시하고 검색할 수 있다. 현재 제품화 되어 서비스 중인 공개 UDDI레지스트리는 세가지로 나누어 볼 수 있다. (IBM 레지스트리, Microsoft 레지스트리, HP레지스트리, 기타 사설 레지스트리). 본 논문에서는 바이오 분야의 공개된 사설 레지스트리인 BioMoby 레지스트리를 접근하는 애플리케이션을 개발했다.

3.3 다중 질의문의 병렬 웹 서비스 검색

본 논문에서는 바이오 서열 정보 검색시 질의를 효율적으로 처리하기 위하여 다중 질의문의 병렬 웹 서비스 검색 기능을 제안하고자 한다. 최근에 웹 서비스 기술을 통하여 바이오 정보를 통합하고 분석하려는 환경을 제공하려는 많은 연구가 진행되고 있으나, 동일 서비스를 하는 서버는 그리 많지 않다. 그래서 우선 구현할 시스템에서는 동일한 파라미터 값을 갖는 여러 질의문을 동시에 각각 서버에 보낼 경우를 고려하여, 자바 쓰레드 기법을 통한, 웹 서비스 검색을 병렬화 할 수 있도록 했다. 이에 대한 성능 검증은 4장에서 설명하겠다. ([그림 4.8] 참고)

본 연구를 통하여 얻은 결과는 향후 동일 서비스를 하는 서버가 여러 개 존재 할 경우, 질의문에 대한 서버별 검색을 병렬화 하여 검색기능의 향상을 기대해 볼 수 있다.

IV. 구현 및 결과

본 장에서는 3장에서 밝힌 바이오 데이터 통합 검색 시스템 설계를 토대로 구현한 결과를 제시한다. 우선 웹 서비스를 통한 검색 서비스를 하는 서버의 역할을 할 수 있도록 시스템 환경을 설정 부분을 살펴 본 후, 개발한 시스템을 사용자 관점에서 검색 시나리오를 가정하고 그 시나리오에 따른 사용자 인터페이스 중심으로 소개한다.

4.1 구현 환경

운영체제	Windows 2000 Advanced Server
서블릿 엔진(웹 서버)	Tomcat v4.0.6
DBMS	Oracle 9i
웹 서비스 개발 도구	IBM WSTK v3.3.2, Axis v1.1, IBM UDDI Registry
개발 언어	JAVA, JSP, JAVA Bean
라이브러리	BioJava library v1.2.1 , Xerces v2.2.0(XML Parser)
데이터 연결	ODBC:JDBC

[표 4.1] 시스템 개발환경

본 논문에서 시스템을 구현하고 테스트한 환경은 [표 4.1]과 같다. DBMS는 오라클 9i release2를 사용했고, 데이터 처리를 위해서 BioJava 라이브러리를 이용하였다. 웹 인터페이스로 검색 서비스를 구현하기 위해서 IBM 웹 서비스 툴 킷을 이용하여 생성한 각 서비스의 스텁 코드는 JAVA Bean으로 서블릿 엔진에 설치하고, 그를 이용하기 위한 GUI는 JSP로 작성하였다.

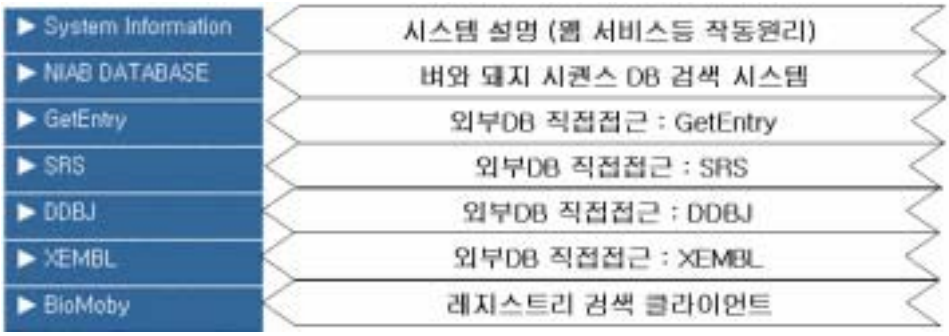
< 서버 시스템 환경 설정 >

외부 데이터베이스 검색과 구축한 데이터베이스 검색 모듈을 웹 서비스화 하기 위해서는 서버 자체로는 부족하다. 웹 서비스 애플리케이션 개발에 필요한 API를 서버에 설치해야 하는데 본 연구에서는 아파치 SOAP 3.0 버전 프로젝트인 AXIS1.0을 사용하였다. 다운로드 받은 AXIS를 서블릿 컨테이너로 사용한 톰캣 서버 4.0.6 홈 아래 /wepapps 디렉토리에 모두 복사한 후, 톰캣을 시작하고 http://localhost:8080/axis/을 열어보았을 때 아래 [그림 4.1]과 같은 화면이 나오면 AXIS설치는 완료된 것이다. AXIS가 올바르게 작동하게 하기 위해 AXIS라이브러리 이외에도 네임스페이스를 인식하는 XML 파서가 필요하다. 본 연구에서는 XML파서로 Xerces 2.2.0 을 사용하였다. 다운로드 받은 Xerces는 Axis 디렉토리 밑에 /webapps/axis/WEB-INF/lib아래 두고 클래스패스를 잡아주어야 한다.

[그림 4.1] AXIS 설치 완료후 초기화면

4.2 구현 결과

아래 그림 4.2는 전체 시스템 GUI의 왼쪽 주 메뉴를 설명한 것이다.

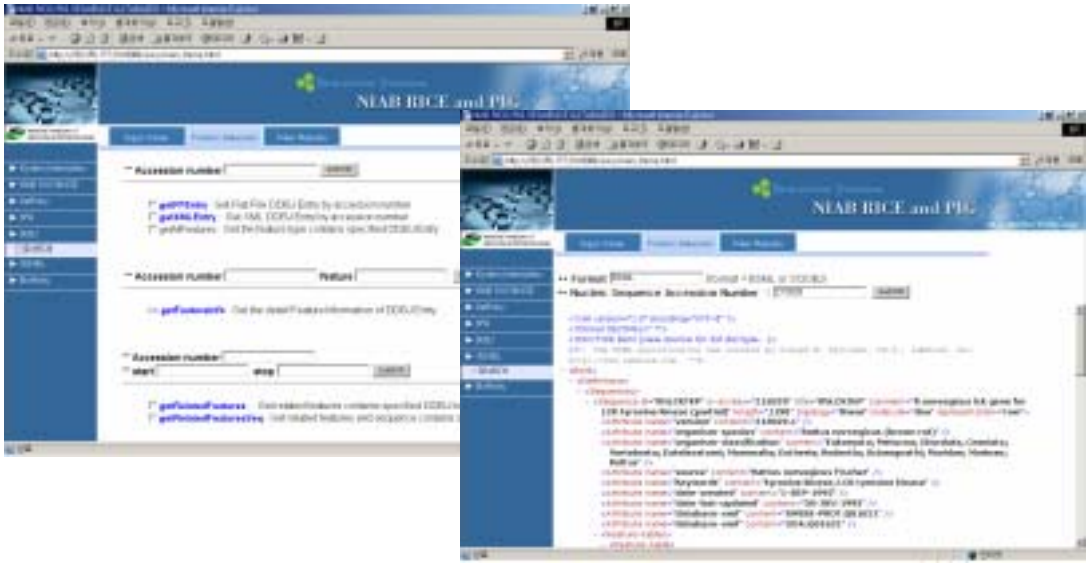


[그림 4.2] 메뉴 설명

4.2.1 서비스 사용자측 모듈 - 통합 검색 클라이언트

본 시스템에서 접근하고자 하는 외부 서열 데이터베이스는 크게 일본의 XML Central of DDBJ, 유럽의 EBI서버 두 가지로 나눌 수 있다. 이 두 서버로부터, 유전체 서열 분야에 해당하는 서비스의 종류를 4가지(GetEntry, SRS, DDBJ, XEMBL)로 한정하여 통합 검색 클라이언트를 개발하였다.

아래 [그림 4.2]의 <NIAB DATABASE>메뉴에서 본 연구에서 구축한 데이터베이스를 검색한 후, 검색 결과가 없을 경우, 외부로 자동 접근 할 수 있도록 하는 동시에, 생물종을 벼와 돼지로 한정하지 않은 경우, [그림 4.2]에 보이듯이 GetEntry, SRS, DDBJ, XEMBL 메뉴를 따로 두어 외부데이터베이스를 바로 접근할 수 있도록 하였다. [그림 4.3]의 (가)는 DDBJ서비스의 함수 선택 인터페이스이며, (나)는 XEMBL서비스 인터페이스로 Accession Number Z15029의 BSML형식의 정보를 검색한 화면이다.



(가) (나)

[그림 4.3] 통합 검색 클라이언트

4.2.2 서비스 제공자측 모듈 – 검색 서비스 공개

(가) 데이터베이스 구축

NCBI에서 다운로드 받은 텍스트 파일 형태의 대용량의 GenBank 파일을 BioJava 라이브러리를 이용한 데이터 전처리 과정을 통하여, 3장에서 제시한 스키마에 알맞게 정보를 추출한 후, 데이터베이스를 구축하였다.

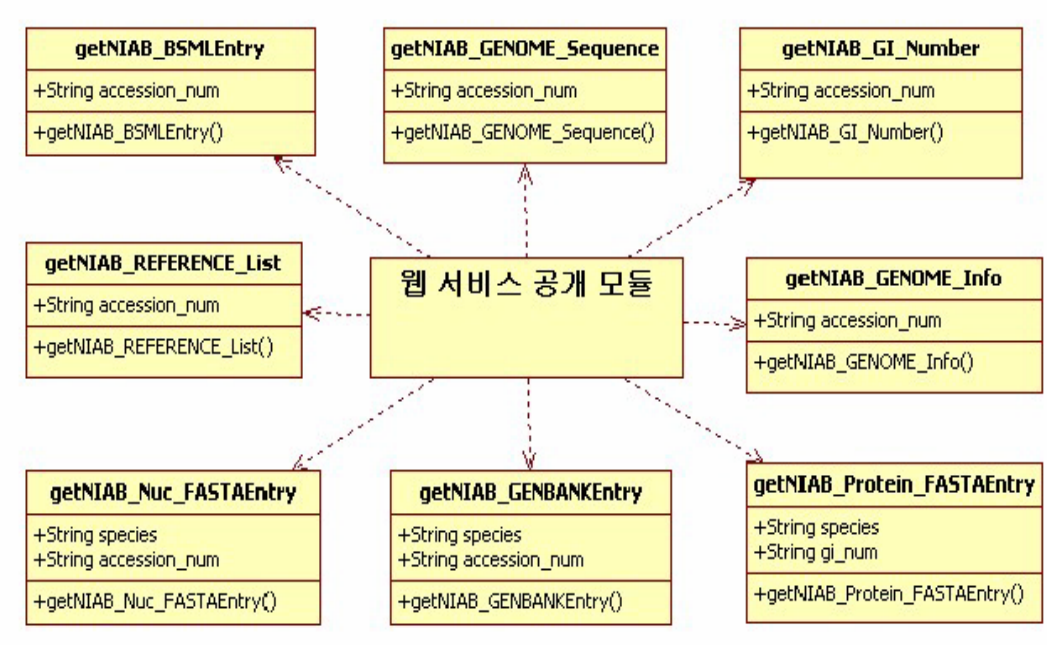
(나) 데이터베이스 검색 서비스 공개

AXIS를 이용하여 웹 서비스를 공개하는 방법은 JAVA코드를 컴파일 하지 않고, 확장자만 JWS로 바꾼 후 서블릿 컨테이너에 설치하는 경우와 WSDD파일을 이용하는 경우 두 가지가 있다. 앞의 경우 WSDD를 이용하는 것보다 간편하긴 하지만, 소스코드가 필요하다는 단점이 있다.

함수 이름	입력 값	출력 값
getNIAB_GENOME_Info	Accession Number	GENOME 테이블 안의 일반 정보
getNIAB_GENOME_Sequence	Accession Number	Accession num에 해당하는 서열
getNIAB_REFERENCE_List	Accession Number	Accession num에 해당하는 레퍼런스
getNIAB_GENBANKEntry	1. 종 구별값, 2. Accession Number	Accession num에 해당하는 GenBank 파일
getNIAB_GI_Number	Accession Number	GI number
getNIAB_Nuc_FASTAEntry	1. 종 구별값, 2. GI Number	Accession num에 해당하는 FASTA 파일
getNIAB_Protein_FASTAEntry	1. 종 구별값, 2. GI Number	Accession num에 해당하는 GenBank 파일
getNIAB_BSMLEntry	Accession Number	BSML 파일

[표 4.2] 웹 서비스로 공개할 검색 모듈

위의 [표 4.2]는 앞에서 구축한 벼와 돼지 종의 핵산서열과 단백질 서열 데이터베이스를 검색하는 8개의 함수에 대한 입력 값, 출력 값을 정리한 것이며 아래 [그림 4.4]는 이들 함수를 클래스 다이어그램으로 표현해 본 것이다.



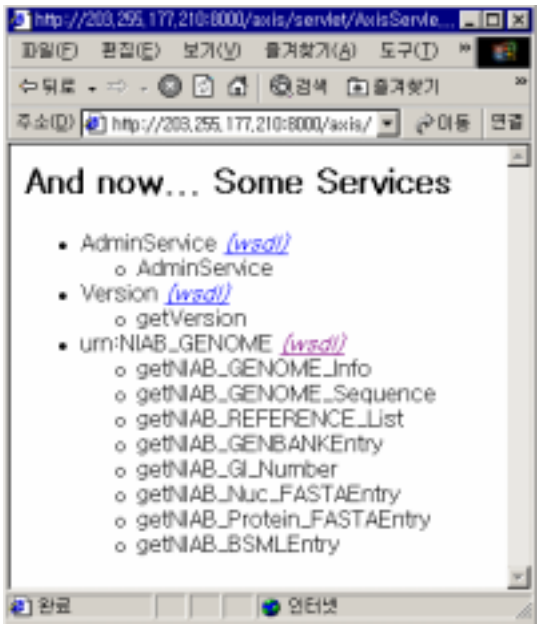
[그림 4.4] 웹 서비스로 공개할 검색 모듈의 클래스 다이어그램

이제는 개발된 함수를 이제 웹 서비스로 공개하는 일이 남았다. 서비스 이름은 urn:NIAB_GENOME이라 하고, AXIS서버에 등록하기 위한 WSDD(Web Service Deployment Descriptor)를 작성한 후, 커맨드 창에서 %java org.apache.axis.client.AdminClient deploy.wsdd 라 명령을 주면 정상적으로 등록되었다는 메시지와 함께 등록이 완료된다.

```
<deployment xmlns="http://xml.apache.org/axis/wsdd/"
                xmlns:java="http://xml.apache.org/axis/wsdd/providers/java">
<service name="urn:NIAB_GENOME" provider="java:RPC">
   <parameter name="className" value="server.niab.getNIAB_GENOME"/>
   <parameter name="allowedMethods" value="*"/>
</service>
</deployment>
```

[표 4.3] WSDD 파일

[그림 4.5] 정상적으로 서비스 공개 후 확인



AXIS 서버에 등록이 잘되었는지 확인은 익스플로러에서 서버 홈 밑의 axis/servlet/AxisServlet 경로로 들어갔을 때, 옆 [그림 4.5]과 같이 서비스 이름과 개발된 함수명이 제대로 보여지면 정상적으로 등록이 된 것이다

이렇게 웹 서비스 서버로 등록이 완료 되면, 이제 외부 사용자 누구나, 아래 [그림 4.6]의 개발된 데이터베이스로의 검색 기능이 명시된, 공개된 WSDD의 파일만 있으면, 본 연구에서 개발한 데이터베이스를 검색하는 자체적인 애플리케이션을 개발할 수 있다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<wsdl:definitions targetNamespace="http://203.255.177.210:8000/axis/services/urn:NIAB_GENOME"
  xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:apachesoap="http://xml.apache.org/xml-soap"
  xmlns:impl="http://203.255.177.210:8000/axis/services/urn:NIAB_GENOME"
  xmlns:intf="http://203.255.177.210:8000/axis/services/urn:NIAB_GENOME"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types />
+ <wsdl:message name="getNIAB_BSMLEntryResponse">
+ <wsdl:message name="getNIAB_GENBANKEntryResponse">
+ <wsdl:message name="getNIAB_REFERENCE_ListRequest">
+ <wsdl:message name="getNIAB_Protein_FASTAEntryRequest">
+ <wsdl:message name="getNIAB_GI_NumberResponse">
+ <wsdl:message name="getNIAB_GENOME_InfoRequest">
+ <wsdl:message name="getNIAB_GENOME_InfoResponse">
+ <wsdl:message name="getNIAB_REFERENCE_ListResponse">
+ <wsdl:message name="getNIAB_Protein_FASTAEntryResponse">
+ <wsdl:message name="getNIAB_Nuc_FASTAEntryResponse">
+ <wsdl:message name="getNIAB_GENBANKEntryRequest">
+ <wsdl:message name="getNIAB_BSMLEntryRequest">
+ <wsdl:message name="getNIAB_GENOME_SequenceRequest">
+ <wsdl:message name="getNIAB_GI_NumberRequest">
+ <wsdl:message name="getNIAB_Nuc_FASTAEntryRequest">
+ <wsdl:message name="getNIAB_GENOME_SequenceResponse">
+ <wsdl:portType name="getNIAB_GENOME">
+ <wsdl:binding name="urn:NIAB_GENOMESoapBinding" type="impl:getNIAB_GENOME">
- <wsdl:service name="getNIAB_GENOMEService">
  - <wsdl:port binding="impl:urn:NIAB_GENOMESoapBinding" name="urn:NIAB_GENOME">
      <wsdlsoap:address
        location="http://203.255.177.210:8000/axis/services/urn:NIAB_GENOME" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

[그림 4.6] 웹 서비스로 공개하기 위한 WSDL 파일

4.2.3 서비스 레지스트리 검색 모듈

본 절에서는 바이오 분야의 공개된 사설 레지스트리인 BioMoby 레지스트리를 접근하는 애플리케이션을 개발했으며 이를 공개한다.

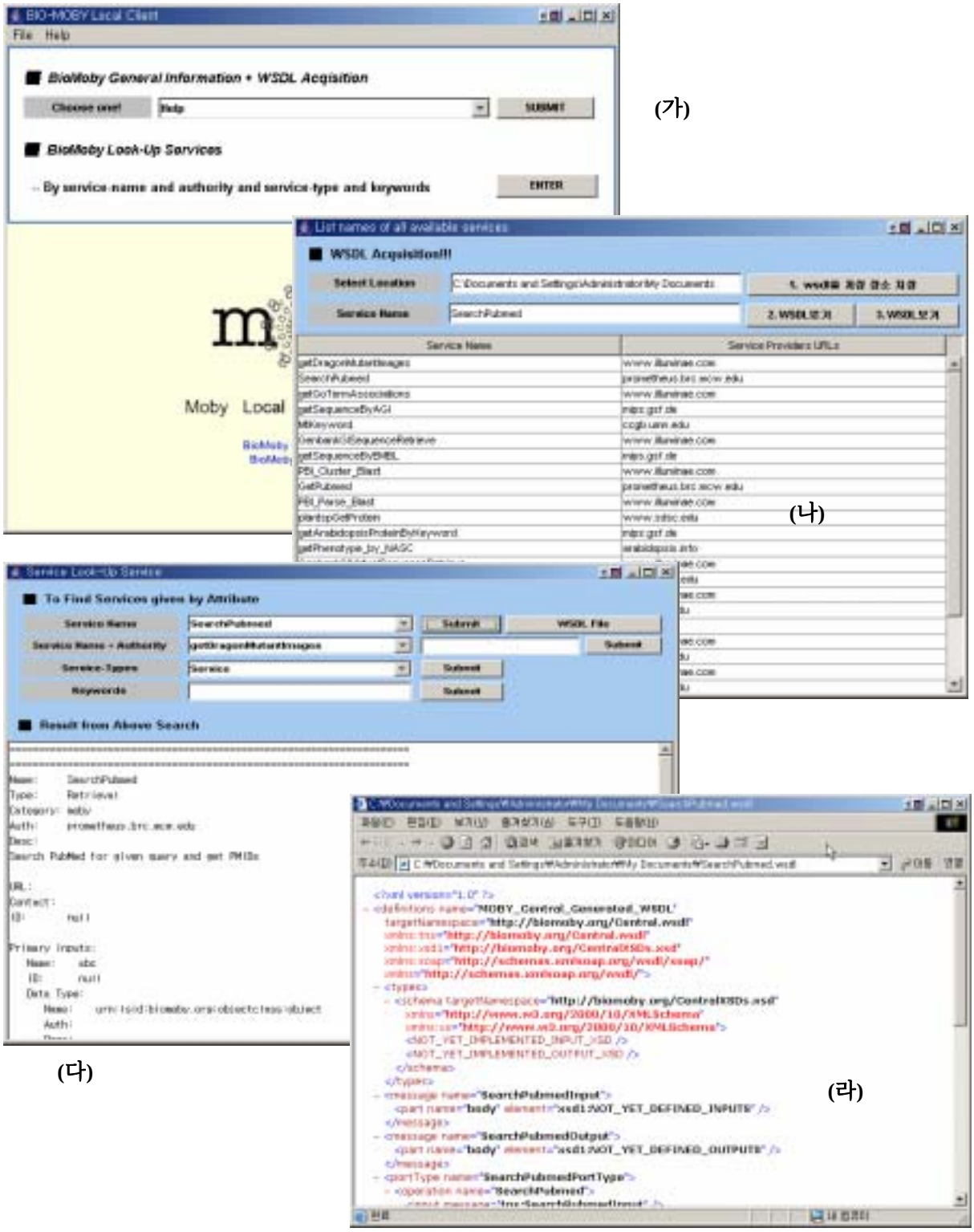
[그림 4.2]의 왼쪽 주 메뉴 중 BioMoby를 클릭하면, BioMoby 레지스트리 검색

애플리케이션을 JAVA WEB START(따로 설치 없이 바로 다운로드 받아 사용 가능)를 사용하여 바로 실행시킬 수 있다. BioMoby 레지스트리 검색 클라이언트 애플리케이션을 이용하여 BioMoby 레지스트리에 등록되어 있는 서비스들의 목록을 조회하고, 해당서비스의 WSDL파일을 생성시켜 사용자의 PC에 저장 할 수 있도록 프로그래밍 하였다.

[그림 4.7] 의 (가)는 BioMoby 레지스트리 검색 클라이언트의 초기 화면이며, (나)는 BioMoby 레지스트리에 등록되어 있는 모든 서비스의 목록을 표로 나타내어 조회 가능토록 했으며, (다)는 레지스트리에 등록되어 있는 서비스들을 여러 파라미터를 주어 검색 가능하도록 했다. (나) (다) 에서 WSDL생성 버튼을 누르면, 사용자의 컴퓨터에 WSDL파일이 저장되며, 필요할 경우, 익스플로러로 WSDL파일을 열어볼 수 있다.

□ **서비스 검색 파라미터**

- ■ 서비스 이름 (현재 등록되어 있는 서비스는 27개)
- ■ 서비스 이름 + 권한
- ■ 서비스 타입 – (EX. retrieval, parsing, analysis 등)
- ■ 서비스 관련 키워드



(가)

(나)

(다)

(라)

[그림 4.7] BioMoby 레지스트리 검색 클라이언트

4.2.4 검색 시나리오에 따른 시스템 흐름

본 절에서는 사용자 입장에서 통합 검색 시스템을 사용하는 프로세스를 가정하고 그 과정을 따라 사용자 인터페이스 중심으로 설명을 하고자 한다.

< 검색 시나리오와 그림 설명 >

A. 사용자는 벼의 핵산 서열 정보를 원한다.

□ 데이터베이스 안에 저장되어 있는 벼와 관련된 모든 핵산 서열을 검색

B. [그림 4.9] 데이터베이스에 등록되어 있는 벼의 핵산 서열을 모두 보여준다

C. [그림 4.10] 선택한 엔트리에 대한 정보를 다양한 포맷 제공

□ 1번부터 5번까지의 서열 정보를 GenBank 파일로 다운로드 받는 인터페이스

D. [그림 4.11] 체크 박스를 이용하여 선택한 항목의 세부 정보를 살펴본다.

□ 선택 항목 보기 버튼을 클릭하면 [그림 4.11]에 보이는 것처럼 나타나며, 체크 박스 앞에 '+' 기호를 클릭하면 해당서열의 세부정보를 펼쳐 보인다.

E. [그림 4.12] Acceesion number가 AG207700인 서열 정보가 있을 경우

□ 데이터베이스에서 Accession Number가 AG207700인 서열 정보가 있는지 검색하고, 검색 결과를 보여준다

□ [그림 4.13]는 AG207700에 해당하는 GenBank파일을 파싱한 후의 결과를 다운로드 받은 BSML파일이다

F. [그림 4.14] Acceesion number가 AG207701인 서열 정보가 없을 때

□ Accession Number가 AG207701인 정보를 검색했을 때 해당정보가 데이터베이스 내에 없는 경우.

□ 외부 데이터베이스로의 자동 검색 인터페이스 보임

G. [그림 4.15] 외부 서열 데이터베이스 검색 결과 조회

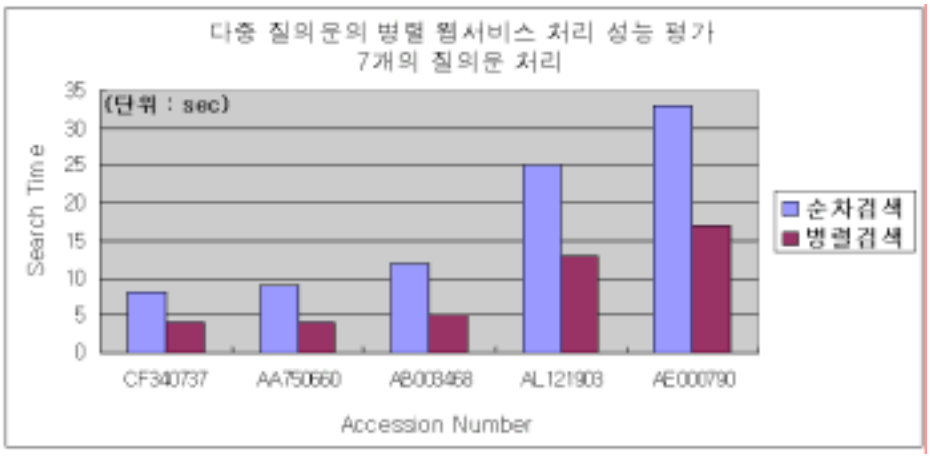
H. [그림 4.16] [그림 4.17] 구축한 데이터베이스로 저장되었는지 확인

□ 외부 서열 데이터베이스로 받은 결과를 내부적으로 데이터 처리 한 후 설계한 스키마에 맞도록 저장 했다.

□ [그림 4.16] : KEYWORD 검색으로 데이터베이스내에 있는지 확인

□ [그림 4.17] : 전체 목록에서도 확인 가능

시나리오 F의 외부 데이터베이스로의 접근 인터페이스는 동일 파라미터 값인 AG207701에 대한 여러 질의 문을 한꺼번에 외부 데이터베이스 서버에 보내어 결과 값을 얻을 수 있도록 설계하였다.

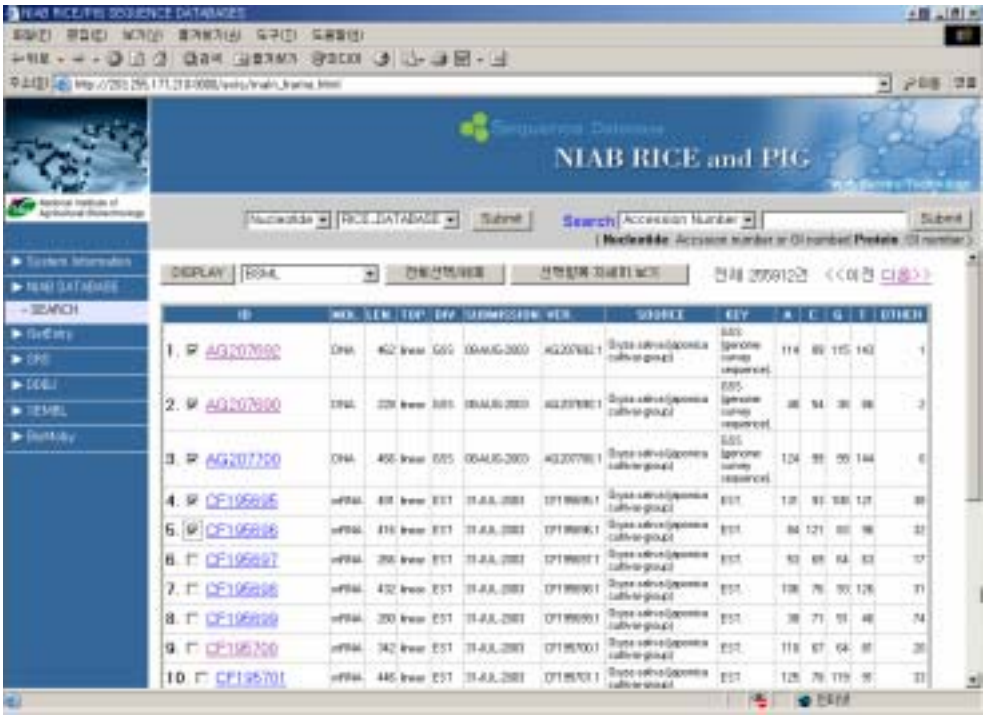


[그림 4.8] 다중질의문의 병렬 웹 서비스 처리 성능 평가

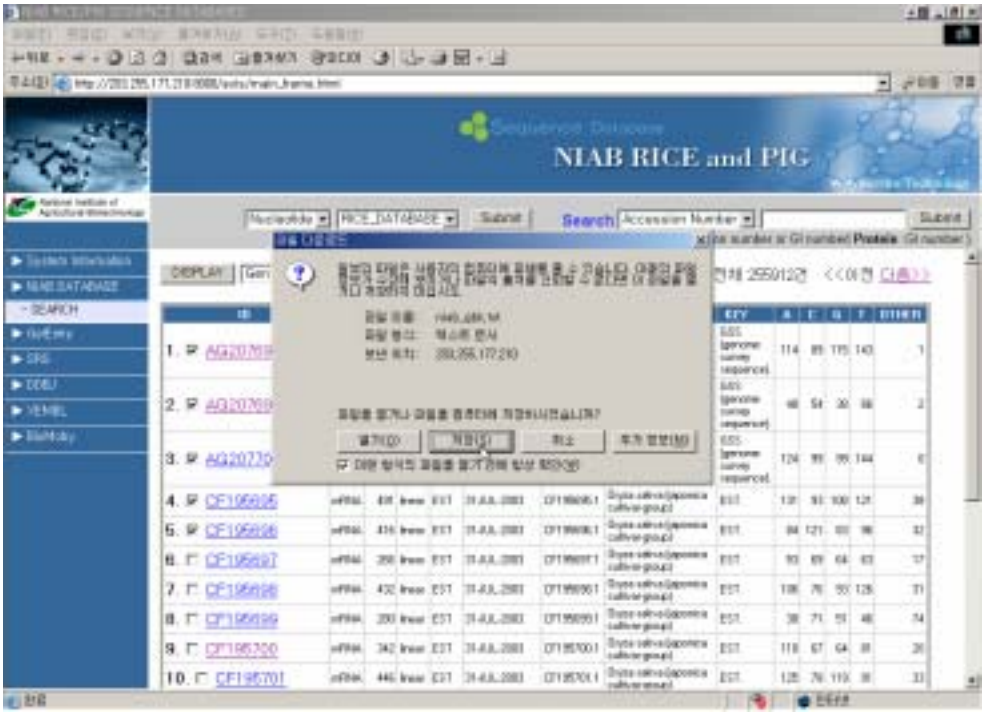
[그림 4.8]은 동일한 파라미터를 가지고 DDBJ, FASTA, EMBL, XML포맷 엔트리 외에 서열의 Config정보, 텍스트 파일 포맷의 DAD엔트리 정보를 추출하는 7개의 질의문을 하나의 서버에 각각 병렬로 보낼 경우, 전체 7개의 질의를 모두 처리하는데 걸린 시간을 측정한 것이다. 서열 데이터 정보의 크기가 다른 5개의 서열 Accession number를 파라미터 값으로 실험을 해본 결과 병렬 검색이 순차 검색에 비해 두 배 정도의 성능이 우수하다는 것을 알 수 있었다.

이는 앞으로 동일 서비스를 하는 서버가 여러 개 존재 할 경우, 서버 별 웹 서비스 검색을 병렬화 하여, 서버 별 검색 시간을 고려한 제어를 통하여 검색의 효율을 높일 수 있을 것으로 기대한다.

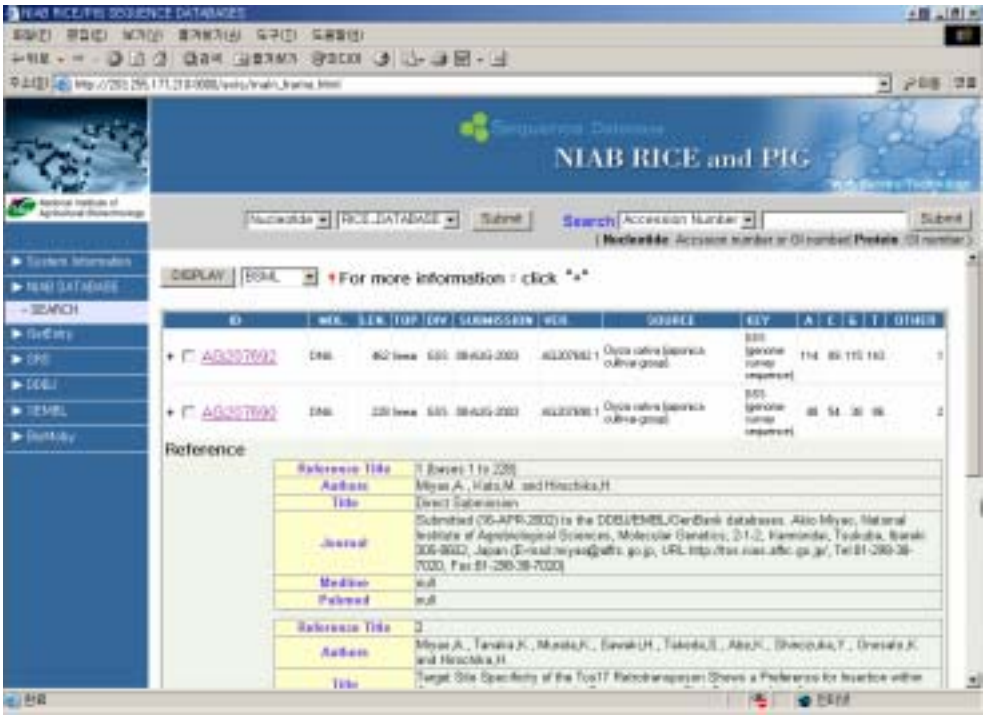
다음 페이지부터는 앞에서 가정한 검색 시나리오에 따른 시스템 흐름을 사용자 인터페이스를 따라 그림으로 나타내어 보았다.



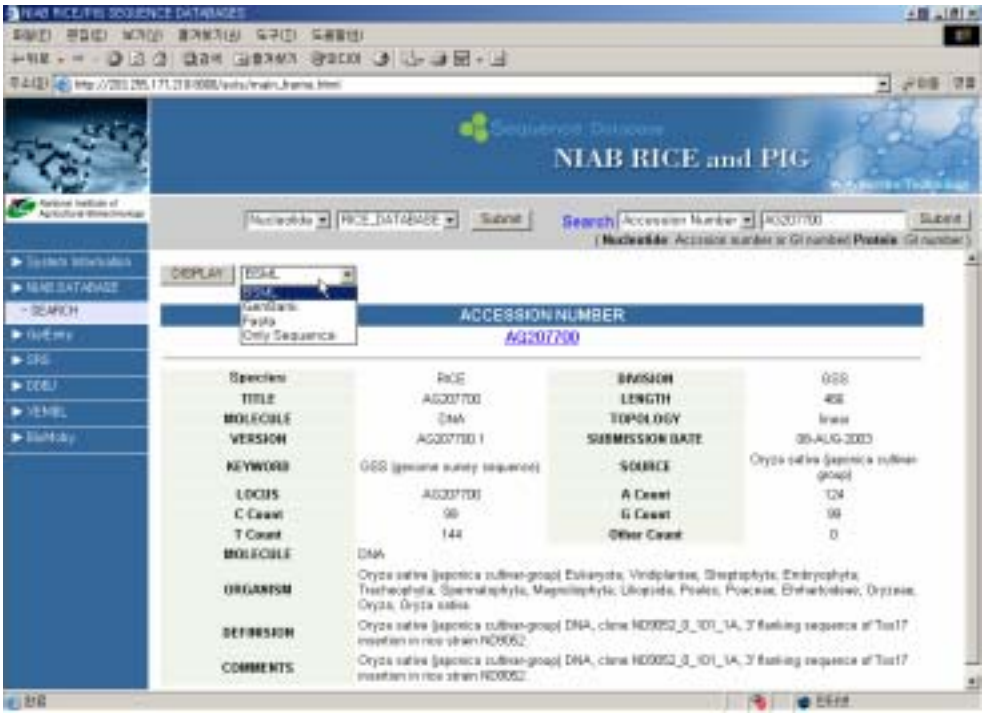
[그림 4.9] 검색 시나리오 A, B : 벼의 핵산정보 조회



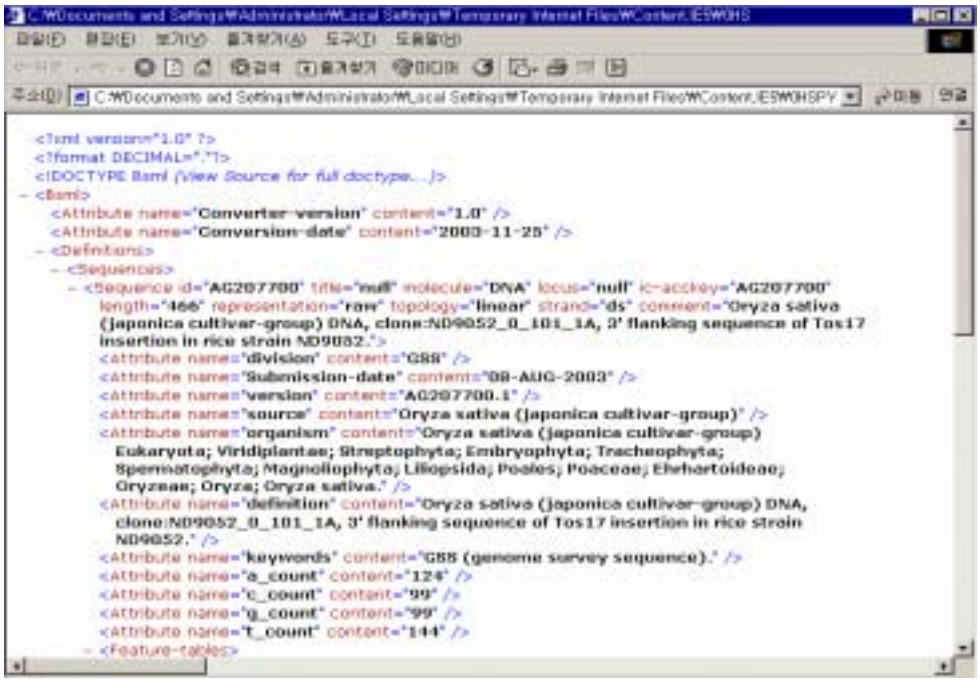
[그림 4.10] 검색 시나리오 C : 선택항목을 다양한 포맷으로 제공



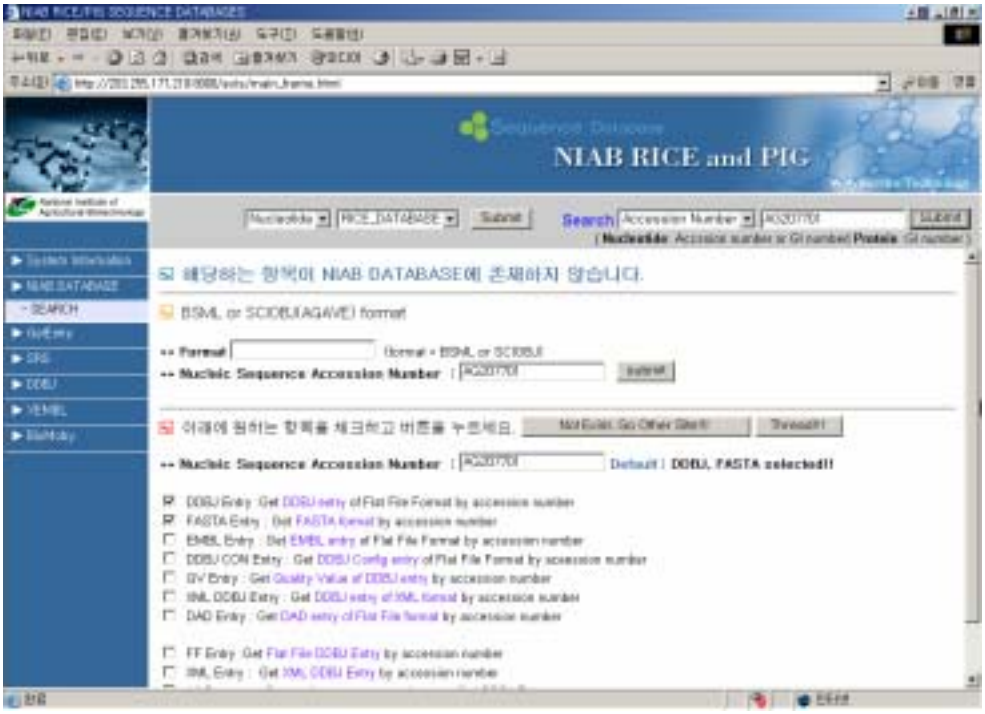
[그림 4.11] 검색 시나리오 D : 선택한 아이템에 대해서 세부정보 펼침 (+기호 클릭)



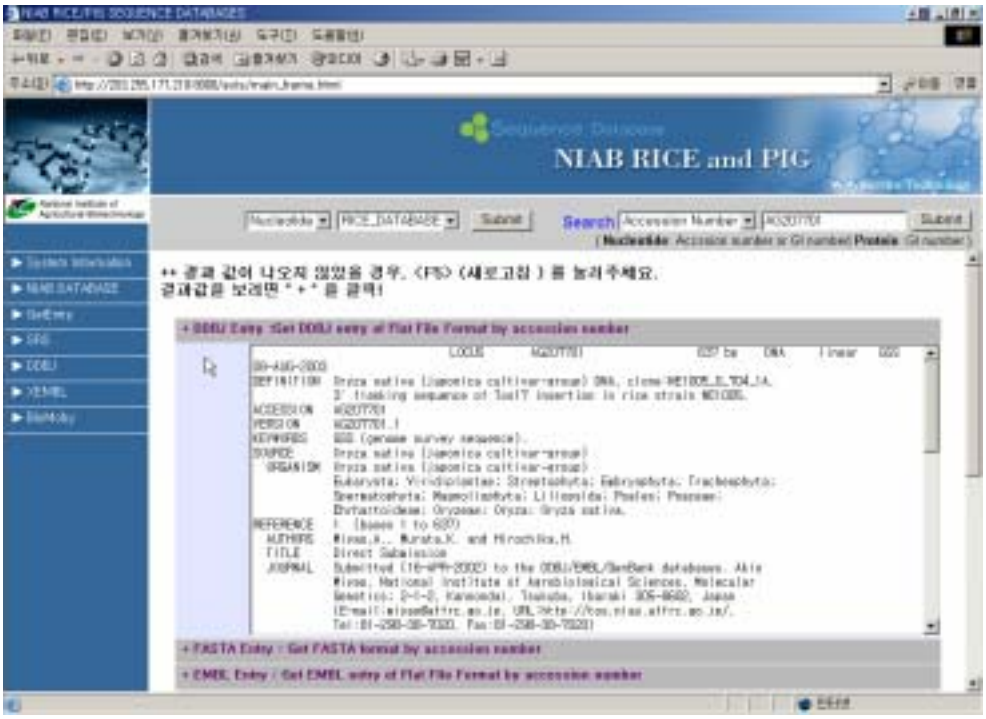
[그림 4.12] 검색 시나리오 E : AG207700 에 해당하는 정보가 있을 경우



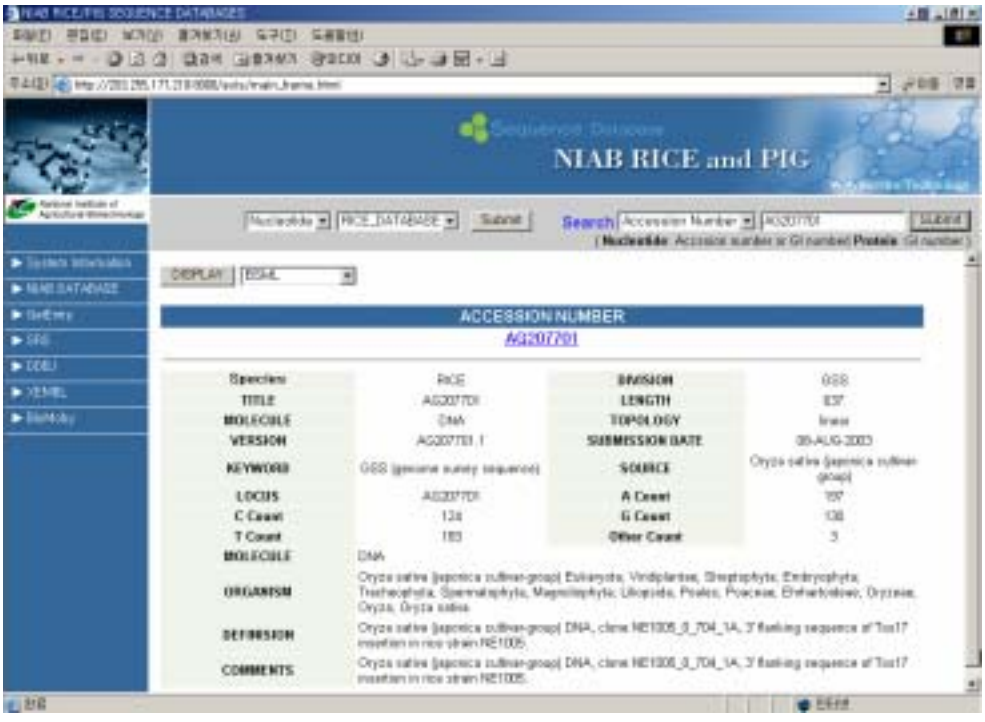
[그림 4.13] 다운로드 받은 AG207700 의 BSML 파일



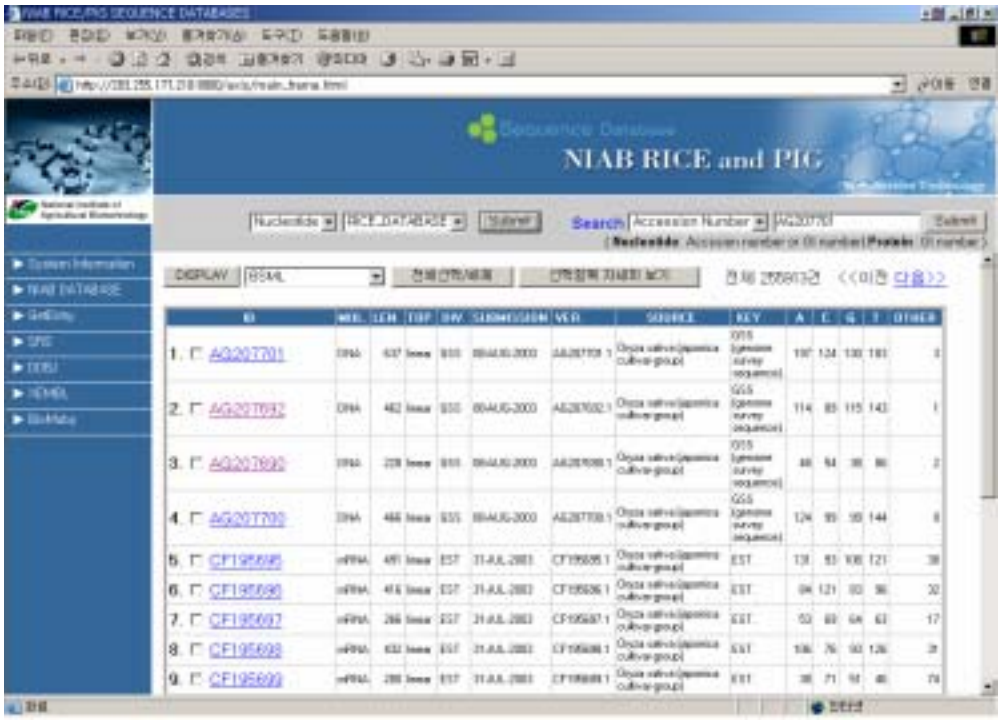
[그림 4.14] 검색 시나리오 F : AG207701 에 해당하는 정보는 존재하지 않음



[그림 4.15] 검색 시나리오 G : 외부 서열 데이터베이스 검색 결과 조회



[그림 4.16] 검색 시나리오 H : AG207701 을 직접 키워드로 넣어 확인



[그림 4.17] 검색 시나리오 H : 전체 목록에서 확인 AG207701 이 목록에 보임

[그림 4.18] 돼지 종의 단백질 서열 조회

V. 결론 및 향후 과제

오늘날의 생물학 데이터베이스 시스템은 끊임없이 증가하고 복잡하게 연결되는 데이터들을 처리해야 할 필요성과 그 데이터만큼이나 빨리 성장하는 사용자들의 요구에 부응해야 하는 필요성에 직면해 있다고 할 수 있다. 이는 통합된 형태의 정보 검색 시스템의 필요성을 더욱 부각시켜준다.

본 논문에서는 기존의 바이오 정보 통합 시스템의 특징을 간단히 살펴본 후, 최근 세계적인 많은 바이오 정보 시스템에서 채택하고 있는 웹 서비스 기술을 기반으로 한 새로운 정보 검색 시스템을 제시하였다. 우선 대용량 텍스트 파일로 받은 벼와 돼지의 핵산 서열과 단백질 서열 데이터를 바이오 자바를 이용한 전처리 과정을 통하여 데이터베이스로 구축하였으며, 이를 검색 할 수 있는 검색 모듈을 자바언어를 이용하여 개발하였다. 이 시스템이 기존의 다른 통합 정보 시스템과 차별성을 갖는다면 그것은 웹 서비스를 통한 외부 데이터베이스 시스템으로의 접근을 시도한다는 점이다. 다시 말해서, 데이터베이스에 없는 정보에 대해서는 웹 서비스를 통한 여러 외부 서열 데이터베이스로의 검색을 메타 검색 개념을 적용하여 병렬로 처리 할 수 있다는 것이다. 여기서 검색으로만 끝나는 것이 아니라, 검색한 정보가 벼와 돼지분야의 것이면, 그 데이터를 데이터 파싱 작업을 통해 새롭게 구축된 데이터베이스에 축적시킨다. 또한 다양한 검색 기능을 통해, 다양한 포맷으로 서열 정보를 얻을 수 있도록 한 본 시스템이 웹 서비스 아키텍쳐의 서비스 제공자로써의 역할을 할 수 있도록 검색 기능을 웹 서비스로 공개하고, UDDI 레지스트리에 등록하여 사용자로 하여금 접근 가능토록 했다.

흔히, 웹 서비스 기술은 SOAP과 같은 인터넷 프로토콜을 통하여 다른 소프트웨어가 이용할 수 있도록 한 설계된 소프트웨어라 정의한다. 이는 웹 서비스 기술을 통하면 큰

어려움 없이, 외부 시스템의 어플리케이션을 이용할 수 있다는 뜻으로, 최근 웹 서비스가 최근 각광을 받은 핵심 이유라 할 수 있다. 하지만, 화두가 되는 웹 서비스 기술도 보안, 프라이버시의 문제나, 다양한 서비스를 요청하고 수신하기 위해 많은 메시지를 이리저리 보내야 함으로 효율적인 메세징, 라우팅 기술이 요구되는 문제점을 가지고 있다. 본 연구에서 개발한 시스템도 예외가 아니다. 내용적인 면을 살펴보면, 지금 상태의 시스템은 바이오 데이터의 일차적인 정보만을 제공하는 단순 데이터 검색이라 할 수 있다. 이는 향후 데이터 정보간의 상호 연결성을 고려하여 보다 의미 있는 정보를 제공할 수 있도록 하는 연구가 필요할 것으로 생각된다.

참고 문헌

[1] Leser, U. 1999. Designing a Global Information Resource for Molecular Biology (Short paper), Datenbanken in Buro, Technik und Wissenschaft (BTW), Freiburg, Germany, pp. 362-369, Springer Verlag.

[2] Davies, K. Combating Creative Chaos in Bioinformatics. Bio-IT World, 2002

[3] Stein, L. Creating a Bioinformatics Nation. Nature 417, pp. 119-120, 9 May 2002 .

[4] Achard, F., G. Vaysseix and E. Barillot. 2001. XML, Bioinformatics and Data Integration. Bioinformatics Review 17(2), pp. 115-125.

[5] 정보문화사 Professional Java Web Services

[6] Oracle9 i Application Server http://otn.oracle.co.kr/tech/webservices/pdf/webservices_twp.pdf 웹 서비스 기술 백서

[7] Labbook, Inc., BSML DTD and Genomic XML Viewer. http://www.labbook.com

[8] GenBank : http://www.ncbi.nlm.nih.gov/Genbank/GenbankOverview.html

[9] Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL. GenBank. Nucleic Acids Res.;31(1):23-7, 2003 Jan 1

[10] Entrez online documentation : http://www.ncbi.nlm.nih.gov/Database/index.html

[11] Etzold,T., Ulyanov,A. and Argos,P. SRS: information retrieval system for molecular biology data banks. SRS: information retrieval system for molecular biology data banks. Methods Enzymol, 1996

[12] Stein, L. Integrating Biological Databases. Nature Reviews-Genetics Vol4. 337-345. 2003

[13] Wilkinson MD, Links M. BioMOBY: an open source biological web services proposal. Brief Bioinform.;3(4):331-41, 2002 Dec

[14] DDBJ Project : http://xml.nig.ac.jp/index.html

[15] Wang L, Riethoven JJ, Robinson A. XEMBL: distributing EMBL data in XML format. Bioinformatics. (8):1147-8, 2002 Aug;18

[16] Wang, L., P. Rodriguez-Tome, N. Redaschi, P. McNeil, A.J. Robinson, and P. Lijnzaad., Accessing and distributing EMBL data using CORBA (common object request broker architecture), Genome Biology 1(5), 2002 Aug;18

[17] Frishman D, Heumann K, Lesk A, Mewes HW. Comprehensive, comprehensible, distributed and intelligent databases: current status. Bioinformatics.;14(7):551-61. Review, 1998

[18] BioJava Tutorial : http://www.biojava.org/tutorials/index.html

[19] BioPerl : http://www.bioperl.org/About/

[20] BioPython : http://biopython.org/

ABSTRACT

Development of Integration Search System of the Biology Sequence Database Using Web Service

Department of Computer Science & Engineering

Ewha Institute of Science and Technology

Lee Su Jung

Molecular Biology databases are highly distributed and heterogeneous, reflecting the distribution and heterogeneity of the Molecular Biology research community. Moreover, a vast number of biological data host and analytical service have arisen in the wake of the explosion of available genome sequence information over the past decade. As a consequence, the integration and interoperability of Molecular Biology databases are issue of considerable importance. With few exceptions, these disparate hosts and services distribute their data in an uncoordinated manner via distinct CGI-based interface. Moreover they have neither a common schema, nor a few widely accepted schemas although querying different databases is a common practice in Computational Biology.

In this paper, we built the genome sequence database on rices and pigs and developed an integrated

search module and public web service module. First, we designed the relational database schema and implemented Nucleotide and Protein sequence search modules of the Biological data such as BSML, GenBank, Fasta format. Then, as the service provider, we published public SOAP service for 8 search modules. With the published WSDL file of the developed system, end-clients may connect to the our SOAP server making use of a simple interface defined in WSDL.

The developed system will allow users to traverse expansive and disparate data resources through e Web Service technology and provide integrated easy-to-use user interfaces.

부 록

아래 표는 시퀀스 정보를 웹 서비스로 제공하는 외부 데이터베이스 접근 함수를 정리한 것이다.

GetEntry 함수 -27 개	
1. getEntry	DB 명과 A.num 를 입력, entry 리턴
2. getClone_DDBJEntry	Clone 정보를 입력, DDBJ 엔트리 리턴
3. getDDBJCONEntry	A.num 입력, 플랫파일의 DDBJ Contig 리턴
4. getDDBJEntry	A.num 입력, 플랫파일의 DDBJ 엔트리 리턴
5. getDDBJVerEntry	버전정보를 포함한 A.num 를 입력받아 DDBJ 리턴
6. getFASTA_CDSEntry	A.num 입력, DDBJ FASTA 엔트리의 CDS 정보리턴
7. getFASTA_DDBJCONEntry	A.num 입력, Fasta 포맷의 DDBJ contig 엔트리 리턴
8. getFASTA_DDBJEntry	A.num 입력, Fasta 포맷의 DDBJ 엔트리 리턴
9. getFASTA_DDBJVerEntry	v_A.num 입력, Fasta 포맷의 DDBJ 엔트리 리턴
10. getGene_DDBJEntry	Gene 이름으로 DDBJ 엔트리 리턴
11. getLocus_DDBJEntry	Locus 정보로 DDBJ 엔트리 리턴
12. getPID_DDBJEntry	PID 정보로 DDBJ 엔트리 리턴
13. getProd_DDBJEntry	Product 정보로 DDBJ 엔트리 리턴
14. getQVEntry	A_num 로 DDBJ 엔트리의 Quality value 를 리턴
15. getXML_DDBJEntry	A_num 로 xml 포맷의 DDBJ 엔트리 리턴
16. getEMBLEntry	A_num 로 플랫파일의 EMBL 엔트리 리턴
17. getFASTA_PIREntry	A_num 로 Fasta 포맷의 PIR 정보를 리턴
18. getPIREntry	A_num 로 플랫파일포맷의 PIR 정보를 리턴
19. getFASTA_SWISSEntry	A_num 로 Fata 포맷의 Swissprot 엔트리 리턴
20. getSWISSEntry	A_num 로 플랫파일 포맷의 Swissprot 엔트리 리턴
21. getFASTA_PDBEntry	A_num 로 Fasta 포맷의 PDB 엔트리 리턴
22. getPDBEntry	A_num 로 플랫파일 포맷의 PDB 엔트리 리턴

23. getDADEntry	A_num 로 플랫파일 포맷의 DAD 엔트리 리턴
24. getFASTA_DADEntry	A_num 로 FASTA 포맷의 DAT 엔트리 리턴
25. getPID_DADEntry	PID 정보로 DAD 엔트리 리턴
26. getFASTA_PRFEntry	A_num 로 FASTA 포맷의 PRF 엔트리 리턴
27. getPRFEntry	A_num 로 플랫 파일 포맷의 PRF 엔트리 리턴
SRS 함수 – 2 개	
28. searchSimple	String 형태의 query 를 입력받아, SRS 정보 리턴
29. searchParam	파라미트를 포함한 query 를 입력받아 SRS 정보 리턴
DDBJ 함수 – 6 개	
30. getFFEntry	A_num 를 입력, 플래파일의 DDBJ 엔트리 리턴.
31. getXMLEntry	A_num 를 입력, XML 포맷의 DDBJ 엔트리 리턴
32. getFeatureInfo	A_num 와 feature 를 입력, feature 의 세부정보리턴
33. getAllFeatures	A_num 를 입력, DDBJ 엔트리의 모든 feature 리턴
34. getRelatedFeatures	A_num, Start, stop 입력, DDBJ 엔트리의 feature 리턴
35. getRelatedFeaturesSeq	A_num, Start, stop 입력, DDBJ 엔트리의 feature 와 서열 정보 리턴
XEMBL 함수 – 1 개	
36. getNucSeq	포맷정보와, A_num 입력, BSML, AGAVE 포맷 리턴

[부록 표] 외부 서열 데이터베이스 검색 서비스 내 함수들

(Accession Number 를 A.num, 버전정보를 포함한 Accession Number. v_A.num 라고 표기)