

웹서비스를 이용한 바이오인포매틱스

자원통합 서비스 설계 및 구현

박성호^o 이기정 김용국
세종대학교 소프트웨어공학과

skyreon@hotmail.com fumaro@nownuri.net ykim@sejong.ac.kr

A design and implementation of Bioinformatics

Resource Integration Service using Web services

Seongho Park^o Kijung Le Yongguk Kim
Dept. of Software Engineering, Sejong University

요 약

바이오인포매틱스에 대한 이슈가 대두됨에 따라 바이오정보를 제공하려는 데이터베이스가 세계 곳곳에서 개발 및 서비스되고 있다. 이러한 바이오 정보를 통합하고 분석하는 환경을 제공하는 방법으로 웹서비스 기반 방식이 대두되고 있는 가운데, 본 논문에서는 웹서비스를 이용한 바이오인포매틱스 자원통합서비스를 설계 및 구현한다.

1. 서론

바이오정보들이 매일 발견되어 속아져 나오고 있지만 바이오 정보를 제공하는 데이터베이스의 환경과 분석 프로그램들이 이질적인 특성 때문에 바이오 정보를 통합하고 분석하는데 용이한 환경을 제공하기란 쉬운 일이 아니다. 이에 대한 접근 방법으로 데이터베이스의 통합, 웹브라우저를 통한 통합접근, OMG 표준 인터페이스 기반 통합과 같은 방식이 사용되고 있으며, 최근에는 웹서비스 기반 통합 방식이 중요 이슈가 되고 있다[1]. 웹서비스의 가장 큰 장점은 서로 다른 플랫폼간의 연동이 가능해지며 네트워크가 구축되어있는 곳이 있다면 어느 곳이나 웹서비스를 이용할 수 있다는 점이다. 이에 본 논문에서는 웹서비스 기반 통합 방식을 이용하여 바이오인포매틱스 자원통합 서비스를 설계 및 구현한 시스템에 대해 기술 하고자 한다. 다음의 2장 관련연구에서는 웹서비스를 적용한 사례와 웹서비스 설계 및 동작에 대해 기술하고, 3장에서는 결론 및 향후 과제에 대하여 논한다.

2. 관련연구

최근에 웹서비스를 이용하여 바이오 정보를 통합하려는 사례를 살펴보면 다음과 같다. 2000년에 등장한 일본의 DDB[2]는 바이오 데이터베이스 서비스를 시초로, 최근에는 바이오 데이터를 웹서비스로 제공하기 위한 연구 및 시험 서비스 중에 있으며, 2001년에 Canada에서 열린 MOBY-DIC(Model Organism Bring Your own Database Interface Conference)회의에서 MOBY라는 생물학 데이터베이스 서비스가 등장하여 2002년부터는 Bio-MOBY라는 웹서비스 개념을 적용한 생물학 데이터베이스 서비스를 연구하고 있다. 그 외에 영국의

MyGrid[4], 일본의 KEGG[5]들이 각자의 목적에 맞게 웹서비스의 개념을 적용하여 바이오인포매틱스 관련 바이오정보 통합에 대해 연구하고 있다.

2.2 웹서비스 구조

웹서비스는 XML 포맷을 사용하여 웹상에서 SOAP(Simple Object Access Protocol) 통신 방식으로, 인터넷을 통하여 단일 또는 다수의 비즈니스 업체간에 기존에 사용했던 애플리케이션 또는 서로 다른 플랫폼과의 상관 관계없이 상호운영이 가능하도록 해주는 표준화된 기술로서 그림 1에 도시된 바와 같이 레지스트리 서비스(Registry Service), 클라이언트 서비스(Client Service), 서비스 제공자(Service Provider)로 구성된다.

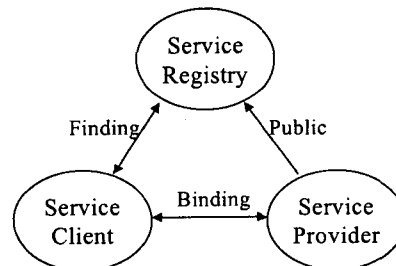


그림 1. 웹서비스의 기본 구조

2.3 웹서비스의 기술

웹서비스 기술에는 크게 WSDL(Web Services Description Language), SOAP(Simple Object Access Protocol), UDDI(Universal Discovery Description & Integration)로 구성

된다. 각각에 대하여 살펴보면 다음과 같다.

2.3.1 WSDL(Web Services Description Language)

WSDL은 CORBA의 IDL(Interface Definition Language)에 해당하는 언어로 웹서비스를 하기위한 인터페이스 사양들을 XML형식으로 서비스를 기술하기 위한 방법을 정의하며, types, message, operation, portType, binding, port, service 요소로 하나의 서비스를 여러 부분으로 나누어서 기술한다.

2.3.2 SOAP(Simple Object Access Protocol)

SOAP은 CORBA의 IIOP, RMI의 JRMP와 유사한 프로토콜로 XML을 사용하여 RPC 및 메시징 통신을 하는 방식으로 다른 통신 프로토콜에 비하여 구조화되어있고 방화벽을 쉽게 통과할 수 있다는 점과, SMTP, JMS등의 다른 전송 프로토콜들과도 함께 사용할 수 있다는 장점이 있다. 하지만, 반대로 보안에 취약하다는 문제점도 있다.

SOAP은 XML 데이터를 교환하는 방식에 따라서 메시징(Messaging) 방식과 RPC(Remote Procedure Call) 방식으로 구분된다.

- 메시징(Messaging): 메시징(Messaging) 방식은 동기식(Synchronous)과 비동기식(Asynchronous) 전송을 모두 지원하는 Document-driven 방식으로 메시지의 전달을 하며 Java API로는 JAXM(Java API for XML Messaging)이 있다.
- RPC(Remote Procedure Call) : RPC(Remote Procedure Call) 방식은 원격 웹서비스의 Procedure를 호출하고 결과를 얻는 동기식(Synchronous) 전송 방식으로 Java API로는 JAX-RPC(Java API for XML-based RPC)가 있다.

2.3.3 AXIS(Apache eXtensible Interaction System)

AXIS는 아파치 SOAP의 차기 버전으로 기존 SOAP의 DOM(Document Object Model) 방식이 아닌 SAX(Simple API for XML) 방식을 사용하였기 때문에 메모리를 적게 사용하면서 보다 향상된 성능을 보인다.

- DOM(Document Object Model) : DOM은 W3C에 등록된 표준으로 XML문서에 있는 객체(Object)를 모두 메모리에 올려서 XML 문서를 파싱(Parsing)하는 방식으로 객체(Object) 구조가 트리 형태로 구성되어 있어서 쉽게 XML문서를 생성할 수 있고 객체의 조작을 쉽게 할 수 있다는 장점을 갖는다. 반면에 메모리의 사용량이 많고 속도가 느리다는 단점도 있다.
- SAX(Simple API for XML) : SAX은 XML문서를 순차적으로 이벤트를 발생시켜 XML문서를 파싱(Parsing)하는 방식으로 XML문서의 일부분만을 처리하는 경우 매우 유용하며 메모리를 적게 차지하면서 속도가 DOM에 비해 빠른 장점을 갖는다. 반면에 XML문서를 생성시키거나 조작하기 어렵다는 단점도 있다.

2.3.4 UDDI(Universal Discovery Description & Integration)

UDDI는 비즈니스 서비스를 제공하는 웹서비스들의 서비스 정

보들을 담고 있는 저장소로 비즈니스 이름, 주소, 전화번호들이 정의되는 Yellow Page 같이 구성되어 있다. 이처럼 UDDI는 비즈니스 정보들이 구조화되어 저장되어 있다. 최근 공용 UDDI 레지스트리를 제공하는 회사로는 IBM[6], Microsoft[10], SAP[15], HP[16]등의 회사가 있으며 자신만의 사설 UDDI 레지스트리를 만들어 서비스하는 연구도 많이 이뤄지고 있다.

2.5 웹서비스 시스템 구조

그림 2은 현재 구현된 웹서비스 시스템의 전체 구조로 클라이언트에게 웹서비스를 제공해주기 위한 서버인 SeqService, MedService, BioDB Site 서버와 웹서비스를 등록 및 검색하기 위한 Registry Server, 클라이언트용 웹브라우저 서버인 Bio-Services Site서버로 구성된다.

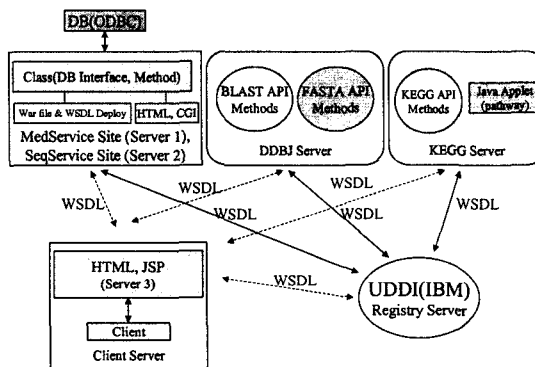


그림 2. 시스템 전체 구조

2.5.1 SeqService Site & MedService Site

SeqService Site는 Protein Name을 입력받아서 해당 단백질(Protein)에 관련된 시퀀스(Sequence)와 LOCUS Name, Definition, 관련된 Medline Number들을 제공해주는 서비스를 하며 MedService Site는 Protein Name을 입력받아서 단백질(Protein)에 관련되는 Medline의 Abstract 정보를 제공해주는 서비스를 하는 사이트를 구축한다.

2.5.2 BioDB Site

• DDBJ Site

최근 DDBJ에서 제공되는 데이터베이스 서비스는 Blast, ClustalW, DDBJ, ExClustalW, Fasta, GetEntry, Gtop, SRS, TxSearch가 있으며 각각의 서비스에 대하여 WSDL문서가 기술되어 SOAP 서비스와 함께 웹서비스가 제공된다.

• KEGG Site

KEGG Site는 KEGG 데이터베이스와 GENES, SSDB, PATHWAY의 데이터베이스들에 대한 검색 서비스를 제공한다. KEGG의 WSDL문서에서는 각각의 데이터베이스에 해당되는 오퍼레이션(operation)이 등록되어있고 오퍼레이션에 해당되는 메소드가 SOAP 서비스로 제공되고 있다.

2.6 Bio-Services Site(Client Server)

Bio-Services Site는 클라이언트에게 웹서비스를 제공해주는

서버로 Registry Server에서 해당 서비스를 검색하는 부분과 서비스 제공자와 클라이언트간의 인터페이스를 처리하는 부분으로 구성된다.

• 클라이언트 인터페이스 화면

그림 3은 구현된 웹서비스의 클라이언트 인터페이스 화면으로 왼쪽 화면에 있는 Service입력 창은 레지스트리에서 원하는 서비스를 검색하기위한 입력 창을 보여준다.

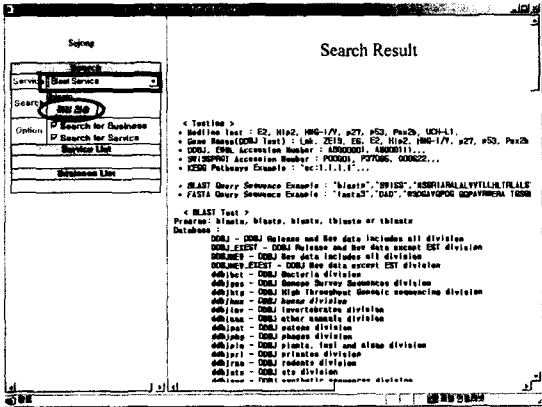


그림 3. 클라이언트 인터페이스

그림 4는 레지스트리 서버에서 해당 서비스를 검색한 결과화면으로 레지스트리 정보 중에서 서비스의 Description, End-Point URL, Service Name, Business Name들의 정보들과 Yellow Page에 해당하는 Service List 부분, White Page에 해당하는 Business List 부분들의 정보들을 읽어서 출력한 화면을 보여준다.

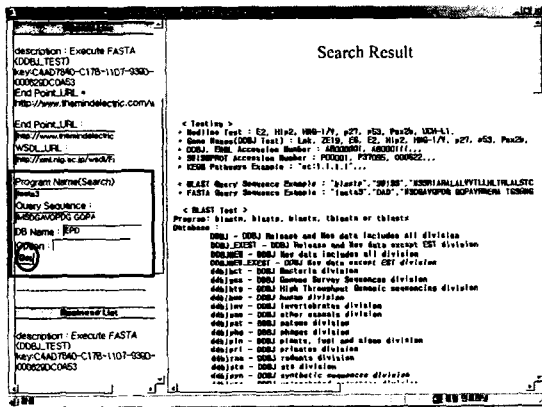


그림 4. 레지스트리 검색 결과

2.8 웹서비스 동작

그림 5은 구현된 웹서비스가 어떻게 동작하는지를 보여주는 그림이다. 먼저 SeqService, BioDB Site, BioDB Site(DDBJ, KEGG)들이 Registry Server에 서비스를 등록한다. 레지스트리는 각 서비스가 등록된 정보인 WSDL문서, End-Point URL, Service Name, Business Name들을 저장하고 있다가 Client가

Registry Server에 Find Service를 요청하면 해당 서비스의 WSDL문서와 서비스에 관련된 정보들을 Client에게 보낸다. Client는 해당 서비스의 WSDL문서와 서비스 정보, SOAP Method Name들을 참조하여 제공되는 서비스의 SOAP Method를 호출하면 해당 서비스는 Method를 실행하고 출력된 Data를 Client에게 보내준다.

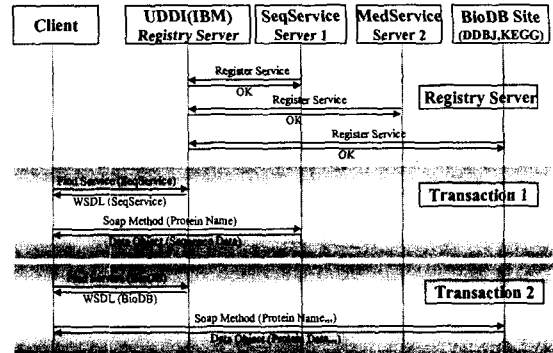


그림 5. 웹서비스 동작

3. 결론 및 향후 과제

최근에 바이오 정보를 통합하고 분석하는 환경을 제공하기위한 많은 연구가 있는 가운데 웹서비스 기반 방식을 사용한 연구가 많이 이뤄지고 있다.

본 논문에서는 이런 바이오 정보를 통합하고 분석할 수 있는 환경을 개발하는 것을 목적으로 웹서비스 기반 방식을 사용하여 바이오인포매틱스의 전체적인 서비스를 설계 및 구현하였다. 하지만, 아직은 웹서비스가 보안의 취약점 등의 약간의 문제점이 있으므로 향후 과제로 QOS(Quality of Service)에 대한 연구, 보안(Security)에 관한 연구 등의 추가 연구가 필요하다 하겠다.

참고문헌

- [1] S.Jeelani Basha, Ben Galbraith, Mack Hendricks, Romin Irani, James Milbery, Tarak Modi, Andre Tost & Alex Toussaint, Professional Java Web Services.
- [2] DDBJ project: <http://www.ddbj.nig.ac.jp>
- [3] BioMOBY project: <http://www.biomoby.org>
- [4] MyGrid project: <http://phoebus.cs.man.ac.uk/twiki/bin/view/Mygrid/WebHome>
- [5] KEGG project: <http://www.genome.ad.jp/kegg>
- [6] IBM registry: <https://uddi.ibm.com>
- [7] UDDI : <http://www.uddi.org>
- [8] Wheeler, D. L. *et al.* Database resources of the National Center for Biotechnology Information: 2002 update. *Nucleic Acids Res.* 30, 13-16 (2002).
- [9] EBI project: <http://www.ebi.ac.uk>
- [10] Microsoft registry : <http://uddi.rte.microsoft.com>
- [15] SAP registry : <http://udditest.sap.com>
- [16] HP registry : <http://uddi.hp.com>