Korean Journal of Life Science
Vol. 12. No. 4. 383~391, 2002

생물정보학을 위한 XML의 활용

이완선[1] · 유미애[1,2] · 조환규[1,3*]

부산대학교 [1]생물정보학 센터
[2]자연과학대학 분자생물학과
[3*]공과대학 전자전기정보컴퓨터공학부

XML Application for Bioinformatics

Wan-Seon Lee[1], Mi-Ae Yoo[1,2] and Hwan-Gue Cho[1,3*]

[1]Bioinformatics & Biocomplexity Research Center
[2]Molecular Biology, Pusan National University, [3]School of Computer Sci. and Eng.*
Pusan National University, Busan 609-735, Korea

Abstract

The difficulties in dealing with the Bioinformatics data come more from its idiosyncrasies than from its quantity. Currently researchers need to an easy method for data exchange, manage, update. In order to integrate and manage all kinds of biological data, it is reasonable to adopt XML as standard tool since XML is independent of operating system, programming language and hardware platform. Although XML in Bioinformatics has been used widely as a standard notation abroad, however it is the beginning step in the domestic research. This article reviews a basic concept of XML and how to apply XML modeling in Bioinformatics. In addition we present XML applications for genomic sequences, structures and genetic network modeling.

Key words – Bioinformatics data, XML

서 론

생물정보학 연구 방법들의 발전으로 많은 데이터들이 전 세계 인터넷을 통해 제공되고 있다. 생물정보학 데이터는 주로 서열과 3차원 구조 및 DNA chip 등을 이용한 유전자 발현 프로필(gene expression profile)등으로 구성되어 있으며 앞으로 새로운 종류의 생물학 정보들이 많이 생산될 것이다. 하지만 이들 데이터들은 모두 각각의 성격에 맞도록 특성화되어 있어 데이터간의 교환이 쉽지 않다. 생명현상은 이들 각각의 모든 정보들의 종합임을 되새겨보면, 이들 데이터들을 효과적으로 서로 통합시키고, 연결시키는 과정이 얼마나 중요한가를 알 수 있을 것이다.

이처럼 생물정보학 데이터는 데이터를 통합하고 교환하는데 어려움을 가지고 있는데 이러한 특성에 적합한 언어로 XML (eXtensible Markup Language)을 제안하고자 한다. XML은 태그(tag)를 생성시킬 수 있을 뿐만 아니라 각각의 태그들간의 관계를 정의함으로서 구조적인 문서를 작성할 수 있어 복잡한 생물학 데이터 구조를 설계하는데 용이하다. 그리고 데이터와 표현을 분리함으로서 데이터를 쉽게

*To whom all correspondence should be addressed
Tel : 051-510-2283, Fax : 051-515-2208
E-mail : hgcho@pusan.ac.kr

수정하고 재활용할 수 있으며, 텍스트기반의 마크업 언어이기 때문에 기존의 문서 타입에 따라 다른 프로그램을 이용하여 확인할 필요가 없어 정보를 빠르게 가져올 수 있다.

현재 국외에서는 생물정보학 데이터를 다루는데 XML을 적용을 적용한 사례들이 늘고 있다. 본 총설에서는 외국의 XML 적용 사례들을 서열, 구조, 모델링, 이미지 분석, 존재론으로 나누어 소개하였다.

생물정보학 데이터의 특성

연구자가 자신의 연구와 관련된 데이터를 수집하여 생물학적 의미가 부여된 2차 데이터베이스를 생성하고자 할 경우 1차 데이터베이스로부터 가공되지 않은 데이터(raw data)나 질의어의 결과로부터 자신이 원하는 정보를 추출하여 재가공하는 작업을 거쳐야만 한다. 이는 데이터의 비호환성이 가장 큰 원인이라 할 수 있는데, 이 작업은 연구작업과는 별도로 이뤄지는 것이므로 또 다른 비용과 노력을 들여야 하며 이는 곧 생물정보학 분야의 생산성을 저해하는 요인이 될 수 있다(Fig. 1).

이러한 현상은 생물정보학 데이터가 다음과 같은 특성을 가지고 있기 때문이다[1,10,11,21].

a. 데이터 설계가 복잡하다. : 복잡한 관계속에서 많은 다른 데이터 타입이 존재한다.

b. 새로운 데이터 타입이 증가하고 있다. : 올해 발표된 침팬지의 유전체 지도 작성 후의 전체서열 분석 결과, 마이크로어레이(microarray), 단백질 상호작용 지도(protein interaction map) 등의 데이터들이 생성될 것이고 그 만큼 새로운 데이터 타입도 늘어날 것이다.

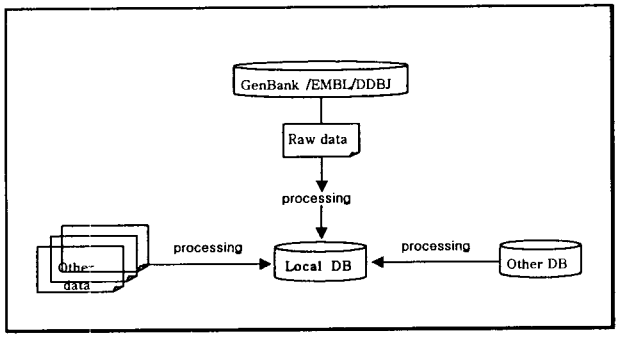


Fig. 1. 데이터 교환 흐름도 : 연구자가 기존의 데이터베이스로부터 원하는 정보를 추출하여 새로운 정보를 만들고자 할 경우 현재는 데이터 재가공이 필수적이다.

c. 데이터 수정이 빈번하다. : 새로운 데이터 타입을 설계하는 것도 필요하지만, 이전의 데이터 타입을 수정하는 것 역시 중요하다. 부분적으로 봤을 때, 새로운 관계가 생성되기도하고 원래 긴밀한 관련을 맺고 있던 부분이 약해지기도 한다. 이것은 예전에는 독립적으로 통합되었던 정보 소스가 이제는 전체의 의미로 업데이트되어야 하기 때문이다.

d. 가공되지 않은 데이터(raw data)를 얻어야 한다. : 과학자들은 종종 가공되지 않은 데이터를 받아 확인을 거쳐 재구성하는 작업을 한다.

e. 데이터가 빈번히 업데이트되고 인터넷을 통해 연구자간의 데이터 교환이 종종 일어난다.

f. 생물학자, 프로그래머, 데이터베이스 관리자 등의 사용자들이 복잡한 질의어(query)를 요구한다.

이러한 특성을 가진 생물정보학 데이터는 현재 서로 다른 형식으로 전세계의 여러 데이터베이스로 산재되어 각각 관리되고 있으며 데이터 교환에 대한 표준화된 형식이 정의되어 있지 않아 데이터 교환은 더욱 어려운 실정이다(Fig. 2)[19].

따라서 이러한 문제점들을 해결하기 위해 표준화된 데이터 형식을 보다 쉽게 만들 수 있고 복잡한 데이터의 구조를 유연하게 표현할 수 있으며 확장성이 뛰어난 방법이 필요한데 그 방안으로 XML을 제안하고자 한다.

XML의 활용

지금까지 사용해 온 HTML은 미리 정해져 있는 태그를 이용하여 문서의 표현정보만을 표현하고 있어 데이터를 전송할 경우 생기게 되는 정보내용의 손실을 피할 수 없었다. 이 역시 생물정보학의 생산성을 저해하는 요인으로 문서의 표현과 구조를 분리하고 있는 XML을 이용하면 이러한 문제점을 해결할 수 있다. XML은 이밖에도 생물정보학 데이터를 기술하는데 여러 장점을 가지고 있다.

1) XML의 기본구조

XML은 eXtensible Markup Language의 약어로 웹상에서 구조화된 문서를 전송 가능하도록 설계된 표준화된 마크업 언어이다. HTML에서 문서의 구조적 정보를 전달할 수 없었던 한계점과 SGML에서 사용의 복잡성을 해결한 문서 표준이다. 즉 HTML과 같은 고정된 형식이 아닌 '확장 표시 언어(eXtensible Markup Language)로 자신만의 마크업 언

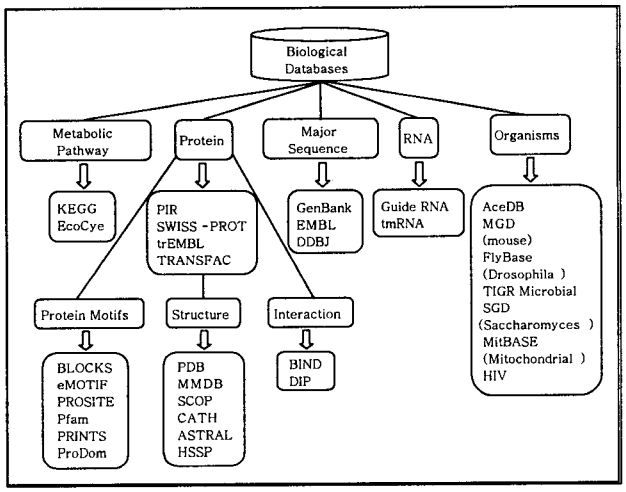


Fig. 2. 생물학 데이터베이스 산재도(Biological Database) : 전세계 데이터베이스들이 서로 다른 데이터 타입을 가지고 산재되어 관리되고 있다.

어(Markup Language)의 작성을 가능하게 하는 메타언어(Meta Language)이다.

HTML (HyperText Markup Language)은 태그의 종류가 한정되어 있는 반면 XML은 문서의 내용에 관련된 태그를 사용자가 직접 정의할 수 있으며 그 태그를 다른 사람들이 사용하도록 할 수 있다. Table 1은 HTML과 XML의 기능을 비교한 것이다.

Table 1. HTML과 XML의 기능별 비교

항목	HTML	XML
사용자 정의 태그 사용	불가능, 제한적	가능, SGML보다는 제한적
문서의 재사용	불가능	가능
응용분야	단순한 구조의 문서 및 소량의 홈페이지 작업	방대한 내용과 구조를 요하는 기술적인 문서, 웹상의 문서 교환 등
문서작성	간단하고 용이함, 논리구조 작성의 어려움	SGML을 단순화시켜 편리하게 작성 가능
문서검색	효과적 검색 어려움	정확한 검색이 가능하고 문서구조에 대한 검색이 가능
링크	HTML	XLL
출력형식 언어	CSS	XSL

하지만, XML은 HTML을 대체하기 위한 것이 아니다. XML과 HTML은 목적이 서로 다르다. HTML은 데이터를 표현(display)것으로 데이터가 보이는 것에 중점을 가지고 디자인 된 것에 반해 XML은 데이터를 설명(describe)하는 것으로 데이터의 의미에 중점을 두고 있다.

하나의 XML 문서가 제대로 표현되기 위해 필요한 필수적인 구성 요소에는 DTD, XSL, XML Parser가 있으며 그 절차를 Fig. 3에 나타내었다[32].

· DTD (Document Type Definition) : 문서의 논리적인 구조를 정의하는 것으로 문서의 내용에 포함되는 요소들을 각각의 엘리먼트(Element)로 정의한다.

· XSL (eXtensible Style Sheet) : 문서의 형식과 스타일을 정의하는 것으로 여러 개의 스타일을 하나의 XML 문서에 적용할 수 있다. 크게 XSLT와 XPATH로 구성되어 있다.

XSLT는 스타일 시트 문서의 뼈대와 같은 부분으로 변환될 문서의 형태를 구성하는 중요한 부분이고, XPATH는 스타일 시트에서 XML 문서의 요소와 대응관계를 표현하기 위한 부분으로 XML 문서 구조에 접근하기 위해 만들어진 언어이다.

· XML Parser : XML문서가 DTD에 합당한지 검사한다.

이 외에도 이름공간(Namespace)나 XML데이터(XML-Data)와 문서 내용 정의(Document Content Definition)와 같은

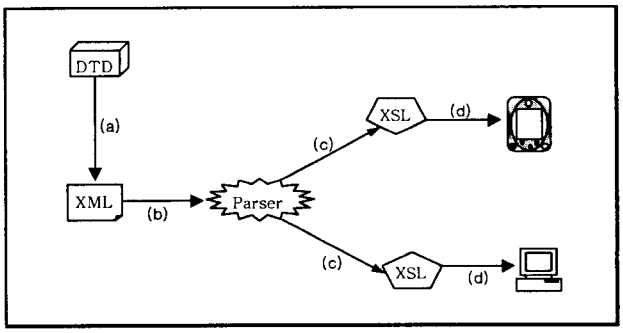


Fig. 3. XML의 구성 : 문서의 논리적인 구조를 정의하고 (a) DTD의 구조에 따라 XML 문서를 생성한다. (b) XML Parser로 DTD가 XML 문서에 합당한지를 검사한 후 (c) XML 문서의 외양 및 프리젠테이션 모양을 제어하는 XSL로 문서를 표현한다. XSL에 정의해주는 것에 따라 다양한 문서를 생성할 수 있다. (d) XSL로 표현된 문서는 PDA나 브라우저 등으로 확인할 수 있다.

더 진보적인 기술들이 존재한다.

하나의 XML 문서가 표현되는 과정을 Code. 1, Code. 2, Fig. 4에 나타내었다. 문서의 구조를 나타낸 DTD를 바탕으로 XML 문서를 생성 후 XSL로 문서 스타일을 만들어 브라우져 상으로 확인한다.

2) XML 활용의 장점

데이터 표준화와 통합의 가장 큰 목적은 데이터를 교환하고 관리하기 위함이다. 이 점에 있어서 XML데이터는 운용체계, 프로그래밍 언어, 어플리케이션, 하드웨어 등에 관계없이 사용할 수 있는 장점을 지니고 있어 여러 형태의 생물정보를 자유롭게 다루는데 적합하다 할 수 있다.

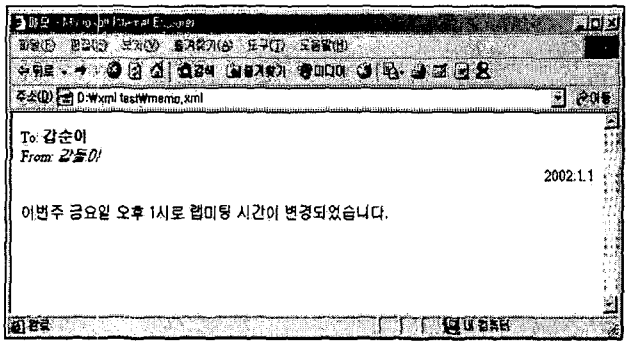
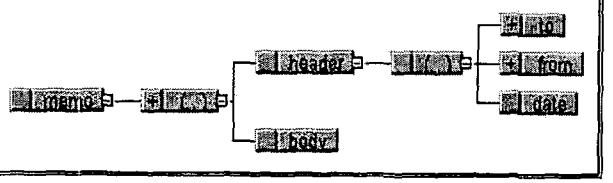


Fig. 4. MEMO.XML을 브라우져로 확인 : 확장된 XML 문서에 XSL을 지정한 결과 화면을 브라우져로 확인한 것이다.

```
<!DOCTYPE MEMO [
<!ELEMENT memo (header , body)+>
<!ELEMENT header (to+ , from+ , date)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT body (#PCDATA)>]>
```



Code. 1. MEMO.DTD : 위쪽은 MEMO.XML을 나타내기 위한 논리적인 구조를 나타낸 MEMO.DTD이고 아래쪽은 다산기술의 DTD Editor1.2를 통해 MEMO.DTD 문서를 시각적으로 보여준 것이다.

```
<?xml version="1.0" encoding="  euc-kr"?>

<!DOCTYPE MEMO SYSTEM "memo.  dtd">
<!-- 외부의 'memo.dtd'문서를 불러온다. -->

<?xml:stylesheet  type="text/ xsl" href="memo. xsl"?>
<!-- ***.xsl 파일로 여러 형태의 문서스타일을  생성 할 수 있다. -->

<memo>
  <header>
    <to>
      <head> To:</head>
      <name> 감순이 </name>
    </to>
    <from>
      <head> From:</head>
      <name> 갑돌이 </name>
    </from>
    <date> 2002.1.1</date>
  </header>
  <body>
     이번주 금요일 오후 1시로 랩미팅 시간이 변경되었습니다 .
  </body>
</memo>
```

Code. 2. MEMO.XML : DTD문서와 XSL 스타일 시트 지정문이 추가된 XML 문서이다.

(1) 복잡한 생물학 데이터 구조 설계의 용이성

사실 자체적으로 사용하는 포맷이 데이터를 더 효과적인 방법으로 표현한다. 하지만 XML은 태그의 사용으로 내용이 커지는 대신 유연성 면에서 이보다 훨씬 많은 이점을 준다. XML에서 사용자는 태그를 정의함에 있어 단순히 새로운 태그를 생성시키는 것 뿐만 아니라 각각의 태그들간의 관계를 정의함으로써 구조적인 문서를 작성할 수 있도록 지원하기 때문이다. 즉, XML은 관련있는 노드(node)로 이루어진 계층적인 트리구조로 이루어져 있으며 이는 관계형 데이터베이스의 '테이블(table), 로우(row), 컬럼(column)'과 유사하다. 이렇게 추상적인 수준에서 작업을 하면 데이터를 저장 할 때 '비트(bit), 바이트(byte)' 수준에서 생각할 필요가 없고 데이터가 나타내는 정보만 생각하면 되므로 생산성이 높아지는 것이다.

(2) 데이터 표현, 수정 및 재활용의 용이성

XML은 웹으로 연결된 애플리케이션과 서비스가 할 일을 줄여준다. 표현에 독립적인 강력한 데이터 구조를 나타내는 XML과 XML을 다른 XML, HTML, 텍스트 기반의 출력 형태로 변환하는 XSLT (eXtensible Stylesheet Language Transformation)을 결합하면 다음과 같은 일을 할 수 있다[32].

데이터와 표현을 분리하므로 애플리케이션 소스를 변경하지 않고도 정보의 형태를 바꿀 수 있다.

요청하는 기기(브라우져, 다른 컴퓨터)에 따라 동일한 데이터에 다른 출력 형태를 적용한다.

생물학 데이터는 여러 소스에서 얻은 정보를 합치고 그것을 요청하는 형태로 변환하는 능력이 아주 중요하다. 이런 면에서 XML 문서는 내용을 가지고 있으면서, 표현에 대한 정의는 포함하고 있지 않기 때문에 같은 문서에 대하여 다양한 표현 방식이 가능하여 재사용적인 측면에서도 뛰어나다 할 수 있다(Fig. 5).

그리고 SQL (Structured Query Language)은 사용자가 요청하는 데이터를 찾아 필터링하여 변형하는데 사용하는 것인데, 이 SQL 기반의 질의 결과를 XML로 표현하게 되면 질의 결과에 있는 정보를 쉽게 변환하고, 전송 할 수 있다(Fig. 6)[32].

현재 생물정보학 데이터들이 각기 다른 데이터 타입을 가지고 전 세계 데이터베이스로 산재되어 관리되고 있는데 이처럼 XML 데이터 페이지를 만들게 되면 사용자는 이를 테이블과 칼럼 형태로 데이터베이스에 저장할 수 있게 되어 관리자, 사용자 모두 최고의 효과를 얻을 수 있다. 즉, 확장성, 신뢰성, 관리성에서 이미 검증된 관계형 데이터베이스와 이것을 이용하는 도구와 애플리케이션에 언제든지 웹으로 정보를 교환할 수 있는 새로운 기술을 추가한 것이다. SQL, XML, XSLT를 결합하면 이와 같이 강력한 기능을 할 수 있는 것이다.

(3) 데이터 교환의 용이성

XML 관련 표준은 웹 기반의 데이터 교환에서 어느 한 부분에 종속되지 않고 플랫폼과 언어에 독립적이다. XML은 텍스트 기반의 마크업 언어로 기존에 해온 문서 타입에 따라 다른 프로그램을 이용하여 확인할 필요가 없다는 것이다. 따라서 정보를 빠르게 가져와서 통합하고 재편성하여 그 정보를 내부뿐만 아니라 외부의 애플리케이션과도 쉽게 교환할 수 있다(Fig. 7).

이처럼 XML은 복잡한 생물학 데이터 구조를 보다 쉽게 설계할 수 있고 기존의 HTML의 단점을 극복하여 데이터 표현과 수정 및 재활용이 가능하며 데이터 교환에도 용이하다.

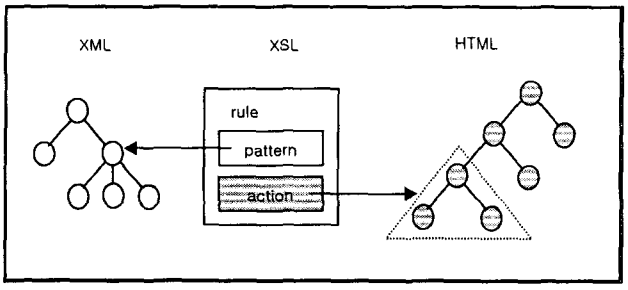


Fig. 5. XSL과 HTML 포맷팅 객체 : XSL 스타일 시트의 구정 요소 중 XSLT의 요소인 규칙(rule)을 이루는 패턴(pattern)과 액션(action)을 타나낸다. 패턴은 XPATH에 해당하는 부분이고, 액션은 변환될 문서의 노드를 구성하는 다른 XML 문서요소나 포맷팅을 위한 요소로 구성된다.

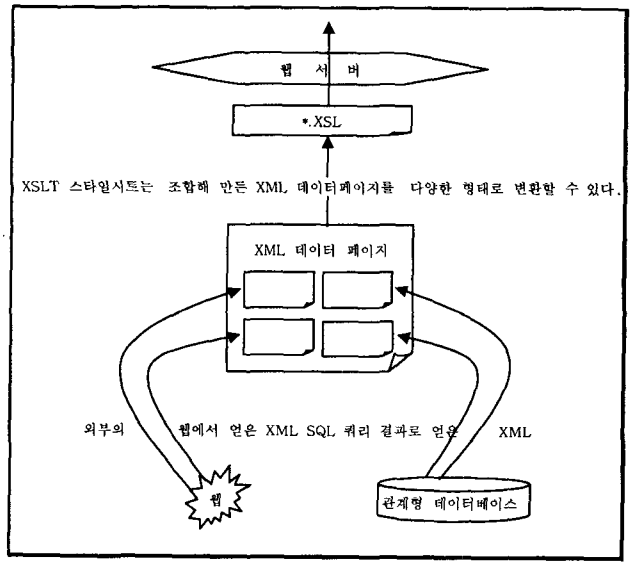


Fig. 6. XML 페이지 생성 및 변환 : SQL 질의어의 결과를 XML로 보여줌으로써 여러질의어와 외부 XML 정보를 합쳐 '데이터 페이지'를 만들 수 있다. 그런 후, XSLT를 이용해 이 XML 데이터 페이지를 브라우져에서 보여주기 위한 출력 형태나 XML 기반으로 한 새로운 형태로 변환 할 수 있다.

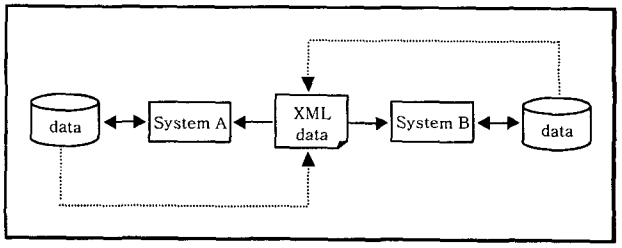


Fig. 7. XML을 이용한 이 기종간의 데이터 전송 : 인터넷상에서 서로 다른 두 시스템 간에 데이터를 전송하는데 있어서 XML을 이용할 경우 시스템의 실제 구현상의 불일치에 따른 문제점들을 극복할 수 있다.

생물정보학에서의 XML 적용 사례

국외에서는 생물정보학에 XML을 적용하는 사례가 많이 있다. 본 보고서에서는 대표적으로 서열, 구조, 모델링, 이미지, 존재론으로 나눠서 설명하고자 한다.

1) 서열(Sequences)

a. AGAVE (An Architecture for Genomic Annotation, Visualization and Exchange, http://www.agavexml.org) : 유전체 서열에 대한 주석을 시각화하여 관리하고 있다. 유전체 주석을 이해하고, 확장성있고 공개되어 있으며 읽을 수 있는 마크업 언어를 제공할 목적으로 시작되었다. 특정 염색체의 중복부위를 나타내는 DNA의 복사본 집합인 콘티그(contig)와 콘티그의 서열 단편 요소(component sequence fragments)의 관계 및 연결되어 있는 각 서열 단편들의 특징을 설명하고 있다. DoubleTwist Genomic Viewer를 통해 정보를 이용할 수 있다[2].

b. BIOML (BIOpolymer Markup Language, http://www.bioml.com/BIOML/) : XML을 사용하여 생물 고분자 물질의 서열에 대한 정보를 표현하고 있다. 현재 이러한 생물 고분자 물질의 서열에 대한 정보를 공유할 수 있는 적절한 방법이 없는 상황에서 BIOML은 웹을 사용하여 과학자들 사이에 이러한 정보를 교환하는 공통적인 도구와 확장 가능한 틀을 제공하는 것이 목표이다. 단백질이나 유전자와 같은 생물 고분자 물질로 구성되는 분자 개체에 대해 알려진 모든 실험 정보를 표현할 수 있는 기반을 제공하고 있으며 무료로 사용할 수 있는 표준이다. BIOML 브라우저를 통해 작성 문서를 볼 수 있다[4].

c. BlastXML (http://www.workingobjects.com) : NCBI Blast 결과를 모델링하는 것으로 Working Object의 PharmTools SDK에서 개발하였다. Working Object에서는 Blast의 검색 결과를 BlastXML 문서로 생성하고 XSL을 적용한 문서를 제공하고 있다[5].

d. BSML (Bioinformatic Sequence Markup Language, http://www.labbook.com) : 유전체 서열 데이터를 표현하기 위한 마크업 언어로 BSML은 서열의 정보를 표현하는 definition과 시각적 표현을 나타내는 display로 구성되어 있다. BSML로 작성된 XML문서를 화면에 나타내기 위해서는 LabBook사에서 개발한 Genomic XML Viewer라는 소프트웨어가 필요하다[6].

e. DAS (A Distributed Annotation System, http://das.wustl.edu/) : 여러 서버에서 얻은 정보를 single client에 통합하는 client-server system이다. 서버는 reference sequence server와 하나이상의 annotation server로 구성된다. Reference server는 서열 지도(sequence map)와 DNA 정보를 제공하고 있으며 annotation server는 유전체 지도에 대한 정보를 제공한다. 가장 큰 특징은 client가 서버에 정보를 요청하여 XML 파일로 정보를 얻는다는 것이다. 원하는 질의에 대한 결과를 XML 파일로 얻으면 원하는 형태로 변환을 쉽게 할 수 있다[9].

f. GAME (Genome Annotation Markup Elements, http://www.bioxml.org/Projects/game) : BDGP (Berkeley Drosophila Genome Project)와 셀레라(Celera)사이에 데이터를 교환하기 위해 개발된 BioXML의 한 부분이다. 특정서열 범위에 관한 정보를 표현하고 서열 분석 프로그램과 전문가에 의해 일반화된 결과들의 차이를 허용하고 있다. 문서에 사용하는 용어는 바이오테크놀로지의 다른 양상과 풍부한 표현 능력을 만들어 낼 수 있도록 정의하고 있다[13].

g. MSAML (Multiple Sequence Alignment Markup Language, http://maggie.cbr. nrc.ca/~gordonp/xml/MSAML) : 다중 서열 정렬(multiple sequence alignment)를 다루고 추출하는데 용이하다[25].

h. PSDML (Protein Sequence Database Markup Language, http://pir.georgetown. edu/) : PIR (Protein Information Resource) 데이터베이스에 저장된 단백질 정보를 이용하여 나타낸 open-standard markup language이다. PIR은 195,891개의 annotation과 classified entry로 구성되어 있으며, 서열의 상동성과 텍스트 검색 기능을 제공한다[30].

2) 구조(Structures)

a. InterPro (ftp://ftp.ebi.ac.uk/pub/databases/interpro/) : protein family, domain, functional site에 대한 내용을 담고 있다[18].

b. ProML (Protein Markup Language, Http://cartan.gmd.de/promlweb) : SCAI (Institute for Algorithms and Scientific Computing)에서 개발한 것으로 단백질의 서열과 구조를 표현하기 위한 표준안이다. PDB (Protein Data Bank)의 데이터를 웹 기반의 변형 툴(converter tool)을 이용하여 ProML형식으로 바꿀 수 있다[29].

3) 모델링(Modeling)

a. MoDL (Molecular Dynamics Language, http://www.oasis-open.org/cover/modl.html) : 단백질 상호작용(protein interaction)과 대사경로(metabolic pathway)에 대한 화학적 시뮬레이션(chemical simulation)을 표현하고 있다[24].

b. PML (*Physiome Markup Language*, http://www.physiome.org.nz/) : AnatML, CellML, FieldM 로 구성되어 있다[28]. AnatML (Anatomical Markup Language, http://www.physiome.org.nz/natml/pages/index.html)은 Physiome Project의 한 부분인 Musculoskeletal modeling project를 통해 얻은 정보를 정리한 것으로, 전체 기관의 기능(organ function)을 통합하기 위한 생리학의 수학적 모델링에 접근하고 있다[3]. CellML (http://www.cellml.org/public/about/ what_is_cellml.html)은 세포구조(physical structure)와 수학적 모델링을 표현하고 있다[7]. 그리고 FieldML (http://www.physiome.org.nz/fieldml/pages/index.html)은 CellML내의 파라메터를 공간적으로 분류하고, AnatML 내의 기하학적 정보를 저장한다[12].

c. SBML (System Biology Markup Language, http://www.cds.caltech.edu/erato/sbml/docs/index.html) : System Biology는 독립된 부분들을 전체 시스템의 의미인 생물학적 프로세싱(biological process)으로 이해하여 실험, 이론, 모델링의 상승효과를 나타내도록 하는 것이다. SBML은 모델(model), 부분(compartment), 기하학(geometry), 종(species), 상호작용(reaction)의 다섯 가지 정보로 구성되어 있으며, 다양한 분석 툴 사이의 정보를 교환하는 것이 주목적이다. 이 포멧을 통해 생화학적 네트웍 모델(biochemical network model)을 표현하였다[31].

4) 이미지(Image)

a. GEML (Gene Expression Markup Language, http://www. geml.org/) : Pattern과 Profile로 구성되어 있는데, Rosetta Inpharmatics에서 이 포멧을 채택하여 시스템을 만들었다. GEML 포멧은 유전자 발현 시스템의 다양성에 있어서 데이터 교환을 쉽게하기 위해 디자인되었고 웹 기반의 유전체 데이터베이스를 가지고 있다. 그리고 어떠한데이터베이스 스키마(database schema)나 이미지 데이터 포멧과 패턴 레퍼런스(pattern reference)에 독립적으로 프로파일 데이터(profile data)를 다룬다[14].

b. MAML (MicroArray Markup Language, http://beamish.lbl.gov) : DNA array 실험에 대한 결과를 XML기반으로 표현한 것으로, GEML처럼 표현되는 DNA의 모든 타입을 묘사하는 방법을 제공하고 있지만 특정 플랫폼에 의존적이다. MGED (Microarray Gene Expression Database) 그룹에서 제안한 MIAME (Minimum Information About a Microarray Experiment)의 정의로 이뤄지고 있으며, 유전자 발현 데이터를 표현하기 위한 최소한의 정보를 다루고 있다[22].

c. GeneX (http://www.ncgr.org/research/genex) : MAML을 수용하고 있으며, 유전자 발현 데이터보다는 어레이(array)에 관한 데이터를 표현하는데 중점을 두고 있다.

d. GEO-XML (Gene Expression Omnibus-XML, http://www.ncbi.nlm.nih.gov/geo/) : NCBI에서 유전자 발현 데이터의 사용이 증가함에 따라 제공하고 있는 유전자 발현과 어래이(array) 데이터이다[16].

5) 존재론(Ontology)

유전체에서 '존재론(ontology)'란 유전자와 단백질의 성질, 기능을 담은 사전이라 할 수 있다. 유전체 정보과학은 컴퓨터를 이용해 유전체가 보존하고 있는 염기배열에서 생명현상의 메카니즘을 해명하는 연구지만, 여기에는 인간과 컴퓨터의 양쪽이 해석할 수 있는 어휘와 기술언어가 필수적이다. 이 같은 통일화된 용어를 정리하는데 XML을 사용하고 있다[20].

a. DAML (Drapa Agent Markup Language, http://www.daml.org) : Ontology 정의 언어를 키워드로 나열하고 있다[8]. 최근 RDF (Resource Description Framwork)기반인 OIL (The Ontology Inference Layer, http://www.ontoknowledge.org/oil/)과 함께 정보를 제공하고 있다[27]. RDF는 웹 기반의 메타 데이터 기술과 교환을 위한 구조로 상이한 메타 데이터간의 어의, 구문 구조를 지원하고 상호 운영성을 지원하여 인터넷상에 존재하는 다양한 형태의 메타 데이터들간의 상호운용이 가능하도록 지원한다.

b. GO (Gene Ontology, http://www.geneontology.org/) : Gene Ontology Consortium에서 진행하고 있는 프로젝트로 분자적 기능(molecular function), 생물학적 프로세싱(biological processing), 세포의 구성요소(cellular component)를 표현하기 위한 일관성 있는 용어들을 정의하는데 XML을 사용하고 있다[17].

c. Troeps (object-based knowledge representation system, http://co4.inrialpes.fr/xml/troeps/, http://exmo.inrialpes.fr/software/#troeps) : 데이터 타입 추출을 통해 사전과 연결하여 객체 셀(object set)에 대한 몇몇 용어들을 뽑아내어 준다. Troeps 시스템은 웹 페이지처럼 HTTP를 통해 객체를 엑세스하여 서버에 알려주는 역할을 한다[33].

d. XOL (XML based Ontology exchange Languag, http://www.ai.sri.com/~pkarp/xol/) : BioOntology Core Group에서 생물정보학 ontology의 교환을 위해 디자인하였다[34].

결 론

생물정보학 연구는 일반적으로 여러 개의 저장소와 네트워크에 분산되어 저장되어 있는 서열, 주석, 분석 결과, 데이터베이스 연결, 그래픽 이미지 등의 다양한 데이터를 필요로 한다. 이러한 데이터를 생성, 관리, 분석, 유통시키기 위해서는 다양한 소프트웨어 응용과 데이터베이스가 요구된다.

생물정보학은 빠르게 변화하고 지수적으로 증가하는 데이터 타입을 가지고 있다. 하지만 생물정보학 데이터를 다루기가 어려운 것은 양적인 문제라기보다는 그 독특한 형식의 문제로 데이터 특성상 연구자간의 데이터 교환 및 관리의 어려움에 있다. 따라서 웹 상에서의 정보 처리 및 관리가 중요한 과제로 떠오르면서 기본적으로 유동적이고 확장성이 뛰어난 XML이 차세대 문서의 표준으로 채택이 되고 있다[23].

XML은 시스템 및 벤더(vender)에 종속되지 않고, 인터넷상의 데이터를 중립적으로 표현하고 교환할 수 있는 기능을 제공한다. 그리고 텍스트 기반의 마크업 언어로 시스템의 하드웨어나 운영체제에 종속적이지 않다. 또한 XML 데이터는 데이터베이스화하여 여러 응용에서 사용할 수 있고 구조화된 정보를 표현하기 용이하며 문서와 표현을 분리함으로서 데이터 재가공이 쉽다는 장점이 있다.

현재 국외에서는 XML을 생물정보학에 적용하려는 시도들이 활발히 이루어지고 있다. 하지만 국내에서는 거의 전무한 상태로, 외국의 사례를 바탕으로 국내 생물정보학에서 XML을 활용 방안을 몇 가지 제안하고자 한다. 먼저, 실제로 각 기업과 연구소가 표준화된 데이터를 교환하고 관리할 수 있도록 여러 단체의 협의 하에 표준화에 대한 공통된 구조가 제정되어야 하며 정부 발주(공공기관) 생물학 프로젝트에서는 모든 결과와 데이터를 XML로 제출하도록 강제할 필요가 있다. 그리고 생물정보학 수업의 일환으로 XML 강좌를 개설하고 생물학 관련 학회에서 XML 튜토리얼, 세션을 만든다면 XML의 응용이 더욱 활성화 될 수 있을 것이다. 또한 BIO XML Tool을 제작, 공급하는 연구소나 단체 및 생물정보학 데이터를 관리하는 BIO XML Bank를 설립하여 생물정보학 연구의 효율성 증대를 이끌어내야 할 것이다.

참 고 문 헌

1. Achard, F., G. Vaysseix and E. Barillot. 2001. XML, bioinformatics and data integration. *Bioinformatics* **17**, 115-125.
2. AGAVE http://www.agavexml.org
3. AnatML http://www.physiome.org.nz/natml/pages/index.html
4. BIOML http://www.bioml.com/BIOML/
5. BlastXMLhttp://www.workingobjects.com
6. BSML http://www.labbook.com
7. CellML http://www.cellml.org/public/about/what_is_cellml.html
8. DAML http://www.daml.org
9. DAS http://das.wustl.edu/
10. David, S. 2001. Bioinformatics-Trying to swim in a sea of data. *Science magazine* **291**(5507), 1260.
11. Davidson, S. B., C. Overton and P. Buneman. 1995. Challenges in Integrating Biological Data Sources.
12. FieldML http://www.physiome.org.nz/fieldml/pages/index.html
13. GAME http://www.bioxml.org/Projects/game
14. GEML http://www.geml.org/
15. GeneX http://www.ncgr.org/research/genex
16. GEO-XML http://www.ncbi.nlm.nih.gov/geo/
17. GO http://www.geneontology.org/
18. InterPro ftp://ftp.ebi.ac.uk/pub/databases/interpro/
19. Kim. TH. 2001. An Ontological Classification and Mapping of Bioinformatics field. *GIW.*
20. Lifschit, S. A Framwork for Molecular Biology Data Integration.
21. Luscombe, N.M., D. Greenbaum and M. Gerstein. 2001. What is Bioinformatics? A Proposed Definition and Overview of the Fild. *Method Inform. Med.* **4**, 346-358.
22. MAML http://beamish.lbl.gov
23. Martin, C.R. 2001 Can we intetrate bioinformatics data on the Internet. *Biotechnology.* **19**, 327-328.

24. MoDL http://www.oasis-open.org/cover/modl.html1
25. MSAML http://maggie.cbr.nrc.ca/~gordonp/xml/MSA-ML
26. NCBI http://www.ncbi.nlm.nih.gov
27. OIL http://www.ontoknowledge.org/oil/
28. PML http://www.physiome.org.nz/
29. ProML Http://cartan.gmd.de/promlweb
30. PSDML http://pir.georgetown.edu/
31. SBML http://www.cds.caltech.edu/erato/
32. Stieve Muench. 2001. *Building Oracle XML Applications.* O'Reilly.
33. Troeps http://co4.inrialpes.fr/xml/troeps/, http://exmo.inrialpes.fr/software/#troeps
34. XOL http://www.ai.sri.com/~pkarp/xol/

(Received April 17, 2002; Accepted May 31, 2002)