

그리드 환경을 위한 정형화된 웹 기반 데이터 검색 시스템

이 상 근[†] · 황 석 찬[†] · 최 재 영^{††} · 노 경 태^{†††}

요 약

데이터베이스에 저장되어 있는 데이터를 그리드 시스템에서 사용하기 위해서는 먼저 시스템에 필요한 데이터를 수집하여 데이터베이스나 파일 시스템에 저장한 후, 데이터베이스의 데이터 및 그 인덱스를 관리하는 전용의 모듈을 별도로 구현하고, 그리드 시스템에 연동시켜야 한다. 웹 기반의 데이터 검색 시스템을 구현하여 데이터베이스에 있는 데이터들을 그리드 시스템에서 바로 사용할 수 있다면, 전용의 데이터 모듈 없이 그리드 시스템에서 바로 데이터를 처리할 수 있으므로, 보다 효율적이고 사용하기 쉬우며, 데이터베이스의 변경에 유연하게 대처할 수 있을 것이다. 본 논문에서는 웹 기반 데이터베이스의 데이터를 바로 그리드 시스템과 연계시켜 사용할 수 있는 검색 시스템을 제안한다. 이 검색 시스템을 이용하여 버추얼 스크리닝 작업을 수행하는 바이오 그리드 시스템에 적용하여 UB Grid (Universal Bio Grid)를 구축하였다. 개발자는 웹에서 제공되는 데이터를 그리드 시스템에 통합시키기 위한 시간과 노력을 줄일 수 있었으며, 사용자는 보다 쉽게 효율적으로 UB Grid 시스템을 사용할 수 있었다.

Formalized Web-based Data Searching System for GRID Environment

Sangkeon Lee[†] · Seogchan Hwang[†] · Jaeyoung Choi^{††} · Kyoung Tai No^{†††}

ABSTRACT

To interact database data with GRID system, implementation and installation of data manipulation module which manipulates database data and its index is required. Developing a search system searching data on web-based database, and integrating it with grid system, it is possible that searching data on web and use it directly on GRID system without independent data module. So, we can build easy and effective grid system, and the system could have more flexible architecture adapting data change. In this paper, we propose a searching system which interacting web-based database with GRID systems. We integrated the searching system with a bio grid system which runs virtual screening jobs. As a result, UB Grid (Universal Bio Grid) is constructed. Developer could reduce time and effort required to integrate web data to GRID system, and user could use UB Grid system easily and effectively.

키워드 : 그리드(GRID), 검색 엔진(Search Engine), 웹(Web), 데이터베이스(Database)

1. 서 론

1990년대 중반에 등장한 그리드 컴퓨팅은 기존의 병렬 및 분산 컴퓨팅이 가지고 있는 성능과 확장성에 한계를 극복하여, 높은 성능을 제공하고, 더 넓은 의미에서 자원을 공유할 수 있으며, 보다 발전된 기능을 가지는 응용 프로그램을 개발할 수 있는 기반을 제공한다[1]. 최근 그리드 컴퓨팅은 빠른 연산 및 대규모 데이터 처리가 필요한 과학 및 공학 분야의 연구에 활발히 적용되고 있다.

물리학, 생물학 등의 기초 과학 연구에서는 웹 기반의 검색이 가능한 정형화된 형태로 데이터를 제공하는 경우가 많다. 이 데이터들을 그리드 시스템에서 사용하려면, 먼저 시스템에 필요한 데이터를 수집하여 데이터베이스나 파일

시스템에 저장한 후, 이 데이터에 접근하는 별도의 모듈을 구현하여 그리드 시스템에 연동시켜야 한다. 그러나 이러한 데이터 접근 모듈은 데이터베이스를 가지고 있는 서버의 보안성 및 안정성에 심각한 문제를 야기시킬 수 있으며, 또한 범용적으로 사용하기가 어렵다.

본 논문에서는 웹을 통해 제공되는 데이터를 검색하고, 검색 결과를 바로 그리드 시스템에 연결시켜서 사용할 수 있는 웹 기반 검색 시스템을 제안한다. 이 검색 시스템은 XML로 작성된 서버 정보를 해석하여 웹 사이트에서 데이터를 수집한다. 서버에 대한 정보 파일만을 추가함으로써 다른 데이터 서버를 추가할 수 있으므로 확장이 용이하며, 서버에서 제공하는 데이터의 위치 등이 변경되더라도 해당 서버에 대한 정보 파일을 수정함으로써 검색 시스템의 기능을 손쉽게 변경할 수 있다. 그리고 검색 결과 중에서 사용자가 작업을 위해 선택한 데이터만 작업 노드에 전송할 수 있고, 여러 웹 사이트에서 제공하는 검색 등을 통합하여

[†] 준 회원 : 숭실대학교 대학원 컴퓨터학과

^{††} 종신회원 : 숭실대학교 컴퓨터학부 교수

^{†††} 비 회원 : 연세대학교 생명공학과 교수

논문접수 : 2003년 5월 12일, 심사완료 : 2003년 11월 27일

그리드 시스템에서 사용할 수 있으므로 사용자가 보다 쉽게 사용할 수 있는 그리드 시스템의 구축이 가능하다. 또한 본 검색 엔진은 웹으로 데이터를 제공하는 서버에 HTTP를 통해 데이터의 읽기만을 요청하므로 데이터가 위치한 서버의 보안성 및 안정성을 유지할 수 있다.

UB Grid(Universal Bio Grid)는 버추얼 스크리닝(virtual screening) 작업을 수행하는 바이오 그리드 시스템으로, 연구의 작업 환경을 위해 구현되었다. 본 논문에서 제안한 검색 시스템을 버추얼 스크리닝(virtual screening) 작업을 수행하는 UB Grid 시스템에 적용시킨 결과, 웹에서 제공되는 데이터를 그리드 시스템에 통합하기 위해 필요한 시간과 노력을 줄일 수 있었다. 또 사용자가 사용하기 쉬운 그리드 시스템의 구축이 가능하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구 및 기반 기술에 관해 기술한다. 3장에서는 검색 엔진에 대해 설명한다. 4장에서는 검색 엔진을 UB Grid 시스템에 적용한 결과에 대해 자세히 설명한다. 5장에서는 시스템의 시험 결과에 대해 평가한다. 마지막 5장에서는 본 논문의 결론을 맺고 향후 연구 방향을 제시한다.

2. 관련 연구 및 기반 기술

미국 아르곤 국립 연구소, 시카고 대학 등에서 개발한 Globus Toolkit[2]은 그리드 시스템의 구축에 필요한 미들웨어와 도구들을 제공한다. Globus Toolkit이 제공하는 기능들을 살펴보면 자원 할당, 작업 제출 및 관리, 메타 정보 시스템, 공개키 기반의 보안 기능 등이 있다.

현재 그리드 시스템을 생명 공학과 연계시킨 많은 연구들이 진행되고 있다. 대표적인 연구 중 하나로서 Virtual Laboratory Project[3]는 신약 개발을 위한 분자 모델링을 수행한다. Virtual Laboratory Project는 데이터에 접근하는 방법으로 여러 개의 화합물들이 저장된 화합물 데이터베이스에 대한 인덱스 파일을 생성하고, 데이터를 관리하는 CDB (Chemical Data Bank) Management and Intelligent Access Tools라는 브로커를 구현하여 데이터들에 접근하는 방법을 사용하였다.

GDMP(Grid Data Management Pilot)[4]는 Globus Toolkit을 이용하여 그리드 시스템을 구축하기 위한 데이터 관리 시스템이다. GDMP는 Globus Toolkit의 도구들을 이용하여 파일을 복제하고 복제된 카탈로그를 관리하는 기능을 제공한다. GDMP는 European DataGrid[5], US ATLAS[6], GryPhyN(Grid Physics Network)[7] 등 많은 그리드 프로젝트에 사용되고 있다.

CDB Management and Intelligent Access Tools나 GDMP 등은 정적으로 데이터 카탈로그를 생성하는 방법을 사용한다. 최근에는 동적으로 카탈로그를 생성하여 데이터를 관리할 수 있는 시스템에 관한 연구가 활발히 진행되고 있다. 대표적인 연구로 Chimera[8]는 가상 데이터 카탈로그

(Virtual Data Catalog)를 생성한다. 응용 프로그램이 데이터를 요청하면 VDL(Virtual Data Language)을 이용하여 데이터 요청을 데이터베이스에 대한 데이터 정의와 질의로 변환하여 데이터를 전송하는 방법으로 동작한다.

CDB Management and Intelligent Access Tools나 GDMP는 데이터베이스 혹은 파일 형태로 필요한 데이터를 저장한 다음, 이 데이터를 위한 정적인 인덱스를 생성하고, 그 인덱스를 통해 데이터에 접근한다. 따라서 높은 성능을 제공하지만, 그리드 시스템의 구축 이전에 데이터의 수집과 인덱스의 생성이 선행되어야 한다. Chimera는 가상적인 데이터 카탈로그와 데이터 요청을 위한 언어를 제공하여 보다 동적인 데이터 접근 기능을 제공하지만, 데이터베이스 혹은 파일 형태로 데이터를 저장하는 과정이 필요하다.

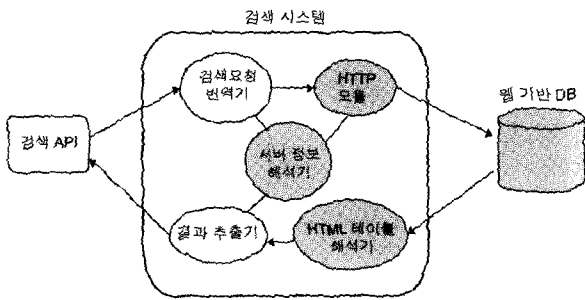
3. 검색 시스템

과학 기술 분야의 연구에 이용되는 많은 데이터베이스들은 사용자들이 웹을 통해 검색할 수 있는 형태로 제공되고 있다. 사용자가 HTML 폼에 입력한 검색 인자들은 HTTP 요청으로 변환되어 웹 서버로 전달되고, 웹 서버에서 검색 결과로 생성한 데이터 목록은 HTML 테이블에 담긴 형태로 제공되는데, 이 데이터들은 일정한 패턴을 가진다. 이러한 데이터들을 추출하는 패턴과 결과를 해석하기 위한 정보들을 XML 파일로 저장한 다음에, 이를 분석하여 사용자의 검색 요청을 처리하면, XML 파일 작성만으로 검색 기능을 손쉽게 확장할 수 있다.

본 논문에서 제안하는 웹 기반 검색 시스템으로 검색 기능을 제공하고, 이를 응용 그리드 시스템에 통합하여 기존에 웹으로 구축된 데이터를 그리드 시스템에서 바로 사용할 수 있도록 한다. 따라서 이 검색 시스템과 통합된 그리드 시스템은 데이터베이스의 추가 구축에 필요한 시간과 비용을 줄일 수 있으며, 보다 손쉽게 데이터를 추가할 수 있다.

검색 시스템의 전체 구조는 (그림 1)과 같다. 클라이언트 프로그램이 검색 API를 호출하면 검색 시스템의 번역기가 검색 요청을 해석하여 HTTP 요청으로 변환한다. HTTP 모듈은 HTTP 요청을 웹 기반 데이터베이스에 보낸 다음, 검색 결과가 포함된 HTML 문서를 전송받아 테이블 해석기에 전달한다. 전달된 HTML 문서는 HTML 테이블 해석기에 의해 검색 결과의 추출이 쉬운 자료 구조로 변환된다. 마지막으로 검색 결과 추출기가 데이터와 메타 데이터를 추출한다. 서버 정보 해석기는 XML로 작성된 서버 정보를 해석한다. 해석된 서버 정보는 검색 요청 번역기, HTTP 모듈, 결과 추출기의 동작을 정의한다.

검색 결과를 전부 클라이언트로 전달하는 것은 비효율적이므로, 검색 결과를 검색 시스템의 레지스트리에 저장하고, 클라이언트에는 메타 데이터와 검색 엔진의 레지스트리에서 데이터를 지정하는 키 값의 목록만을 전달한다.



(그림 1) 검색 시스템의 구조

3.1 서버 정보 파일

(그림 2)는 서버 정보를 담고 있는 파일의 예를 보인 것이다. address 태그는 검색 엔진의 주소를, search 태그의 타입 속성은 검색 결과가 어떤 형식의 데이터인지를 나타낸다. action 태그는 웹 기반 데이터베이스 검색 페이지에서 사용자의 입력을 받는 폼을 기술하며, URL 태그의 method 속성은 HTML 문서의 폼 상의 액션을 나타내고, param 태그는 폼 상에 존재하는 텍스트 상자, 리스트 등을 사용자가 검색에 사용하는 인자의 이름과 매핑시키는데 필요한 여러 정보들을 나타낸다.

result 태그는 검색 결과로 전송된 HTML 문서를 재가공하여 데이터를 추출하는 방법을 명시한 것으로, meta 태그는 메타 정보의 이름과 문서상에서의 위치를 표현하고, data 태그는 데이터의 문서상에서의 위치를 나타낸다.

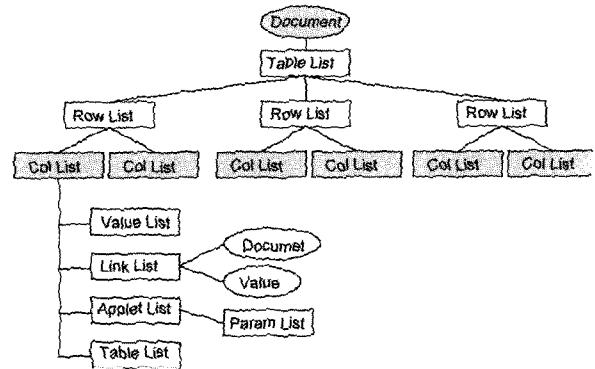
```

- <server_info >
  < address > chemdb.kordic.re.kr </ address >
- < search type = "MDLMOL" >
- < action >
  < url method = "get" > /chi-bin /DB /textsearch.cgi </ url >
  < param name = "CAS" type = "alone" scalar = "false"
    set = "R1 = V3" > cNumber </ param >
  < param name = "IUPAC" type = "alone" scalar = "false"
    set = "R1 = V2" > iName </ param >
  < param name = "CommonName" type = "alone"
    scalar = "false" set = "R1 = V1" > cName </ param >
  < param name = "Formuls" type = "alone" scalar = "false"
    set = "R1 = V4" > mFormular </ param >
- < result >
  < meta type = "CAS" > t[1].r[1].c[0].I[0]-t[1].r[0].
  c[0].t[0].r[0].c[1].v[0] </ meta >
  < meta type = "CommonName" > t[1].r[1].c[0].I[0]-
  t[1].r[0].c[0].t[0].r[1].c[1].v[0] </ meta >
  < meta type = "IUPAC" > t[1].r[1].c[0].I[0]-t[1].
  r[0].c[0].t[0].r[2].c[1].v[0] </ meta >
  < meta type = "Formula" > t[1].r[1].c[0].I[0]-t[1].
  r[0].c[0].t[0].r[3].c[1].v[0] </ meta >
  < meta type = "MW" > d[1].r[1].c[0].I[0]-t[1].r[0].
  c[0].t[0].r[4].c[1].v[0] </ meta >
  < data > t[1].r[1].c[3].I[0] </ data >
</ result >
+ </ action >
</ search >
</ server_info >
    
```

(그림 2) XML 형식의 서버 정보 파일

3.2 검색 결과의 해석

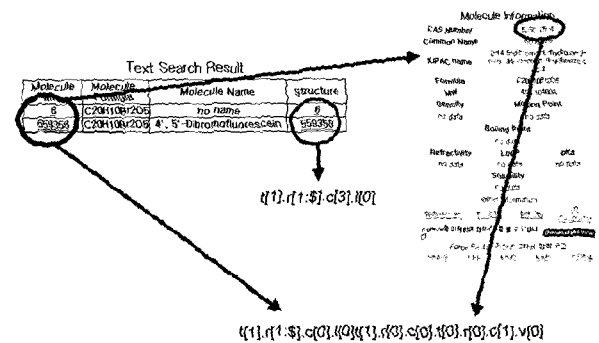
Web Grid의 검색 시스템은 검색 결과로 생성된 HTML 문서에서 검색 결과를 추출하는데 필요한 계층적 자료 구조와 함께 자료 구조에 접근할 수 있는 문법을 제공한다. HTML 문서를 해석하기 위한 자료 구조는 (그림 3)과 같다. 문서는 테이블들을 포함하고, 각 테이블은 행과 열을 가지고 있으며, 행과 열로 표현되는 테이블의 셀에는 내부 테이블이나 값의 리스트, 링크의 리스트, 애플릿 등이 존재한다. 링크는 데이터 값 자체에 대한 링크와 링크를 포함하는 다른 문서 구조에 대한 링크로 구분된다. 애플릿의 경우에 데이터 값이 존재하는 위치는 PARAM 태그의 VALUE 속성의 값이 된다.



(그림 3) 테이블 해석에 사용되는 자료 구조

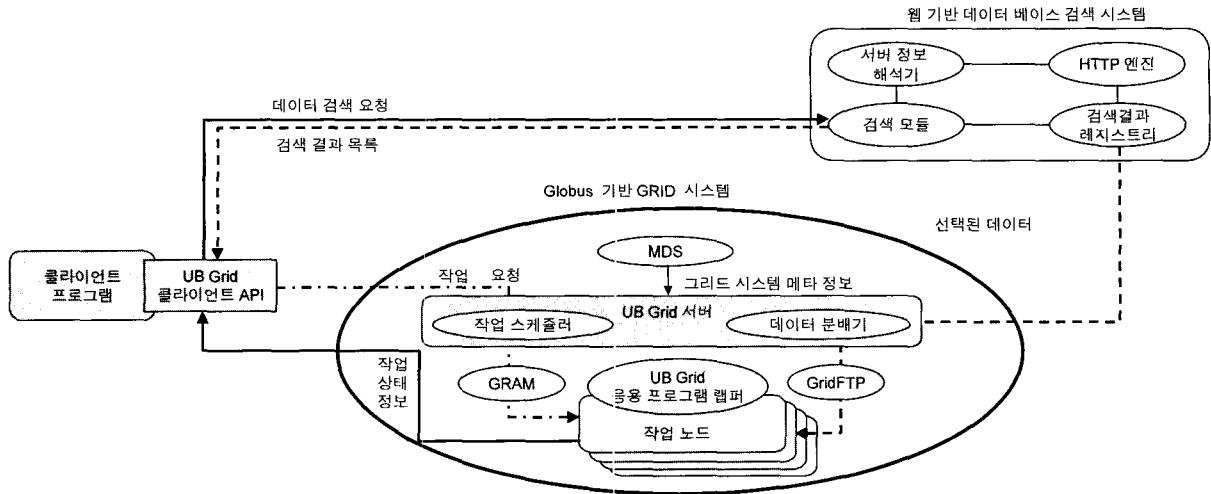
3.3 결과 추출 문법

검색 결과로 생성된 HTML 문서에서 데이터를 추출하려면, 검색 엔진이 해당 HTML 문서를 해석하여 (그림 3)에서 보인 형태의 자료 구조를 만들고, 프로그래밍 언어의 배열과 유사한 형태의 문법으로 이를 참조한다.



(그림 4) 데이터 참조 문법의 예

여기에 사용되는 문법에서 t는 테이블 및 내부 테이블, r은 줄, c는 열, l은 링크, v는 값, a는 애플릿, p는 애플릿의 인자 등을 나타내고 괄호([]) 안의 숫자는 데이터의 위치나 범위를 나타낸다. '-' 기호는 마지막 링크가 가리키는 값이 최종 결과 값이 아니라, 자료 구조를 따라 값을 더 탐색할 필요가 있는 다른 문서에 대한 링크라는 것을 나타낸다. 검



(그림 5) 검색 엔진이 적용된 UB Grid 시스템의 구조

색 엔진의 검색 결과로 전송된 문서에서 데이터가 담긴 테이블 부분과 그 부분을 참조하기 위한 문법의 예는 (그림 4)와 같다.

3.4 캐시 기능

정형화된 웹 기반 검색 시스템이 적용된 UB Grid 시스템에서 데이터의 요구 패턴을 분석한 결과, 중복된 요청들은 일반적으로 높은 지역성을 가지고 발생하였다. 그러나 개별적인 검색 요청 사이에서 중복 전송의 발생 여부는 사용자가 검색하고자 하는 데이터에 의해 결정되므로 예측하기가 불가능하다. 따라서 캐시에 저장된 데이터는 단일 검색 요청의 문맥 안에서 유지하는 것이 효율적이다.

4. UB Grid

UB Grid 시스템은 버추얼 스크리닝 작업을 수행하는 바이오 그리드 시스템이다. 버추얼 스크리닝은 컴퓨터 연산을 통해 병원균의 단백질과 신약 후보가 되는 화합 물질의 결합 에너지를 계산한 다음, 계산된 에너지를 이용해 신약 후보 물질을 도출하는 작업으로, 신약 개발에 필요한 화학 실험의 회수를 줄이는데 사용된다.

기존에는 연구자가 유닉스 시스템 상에서 여러 명령어를 입력하여 버추얼 스크리닝 작업을 수행하였다. UB Grid 시스템은 작업 대상 데이터를 검색하고 지정하여, 그리드 시스템 상에서 작업을 분배하고 수행할 수 있도록 한다.

UB Grid 시스템에서 화합물 데이터를 그리드 시스템에 연결하기 위해 본 논문에서 제안한 정형화된 웹 기반의 검색 시스템을 이용한다. 그 결과 숭실대학교 분자설계연구센터 Organic Chemical Database 웹 사이트[9]에서 제공하는 화합물 데이터를 이용해 버추얼 스크리닝 작업을 손쉽게 수행할 수 있다.

UB Grid 시스템의 구조는 (그림 5)와 같다. 그리드 미들

웨어로 Globus Toolkit을 사용하고, 버추얼 스크리닝을 위한 응용 프로그램으로 Autodock[10], Babel[11]을 이용한다. 검색 엔진으로 검색된 데이터는 GridFTP[12]를 이용해 작업 노드로 전송한다.

화합물 데이터베이스에 접근하는 전용 모듈을 구현하여 데이터에 접근하는 경우, 전용 모듈에 버그가 있으면 화합물 데이터베이스를 운영하는 시스템의 안정성을 저하시킨다. 그러나 검색 시스템을 적용한 경우에는 데이터베이스를 운영하는 시스템에 직접 설치가 되지 않고 데이터를 읽기만 하므로, 이런 문제점이 발생하지 않는다. 또한 웹 사이트에서 사용자가 직접 화합물 데이터를 검색하는 것과 유사한 인터페이스를 제공하므로 사용자가 쉽게 사용할 수 있다. 그리고 검색 시스템에서 제공하는 검색 API를 이용하여 손쉽게 검색 기능을 사용자 클라이언트에 통합할 수 있다.

5. 실험

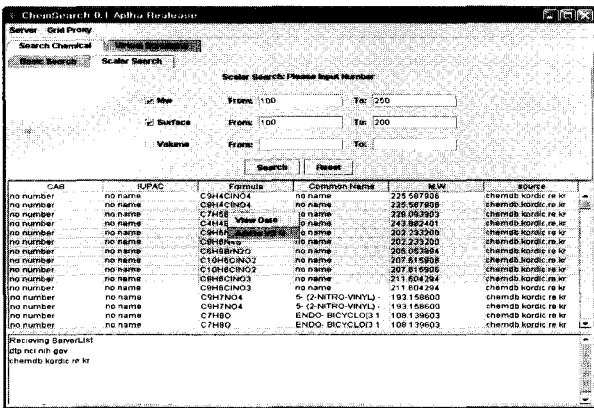
이 장에서는 검색 엔진을 이용하여 구축된 UB Grid 시스템을 이용한 실험에 대해 설명한다. 실험을 위해 KISTI의 클러스터 테스트베드를 이용하였다. 테스트 베드는 4개의 클러스터로 이루어져 있으며, 각각의 클러스터는 24 노드로 LSF[13]를 이용하여 구성되어 있다. 실험은 KISTI 테스트 베드 중 하나인 mercury 클러스터에서 수행되었다. 테스트 베드에 관한 자세한 정보는 <http://gridtest.gridcenter.or.kr>에서 얻을 수 있다.

실험을 위해 시스템에 다음과 같은 조건으로 작업을 제출하였다. 검색 엔진을 이용하여 분자량은 100부터 250까지이고 표면적은 100Å²부터 200Å²까지인 화합물 데이터를 검색하였다. 747개의 검색 결과 중 200개의 화합물을 선택하여 작업 큐에 추가하였다. 작업에 사용한 단백질은 에이즈 바이러스의 단백질이다.

KISTI 슈퍼컴퓨팅센터의 그리드 테스트 베드에 설치된

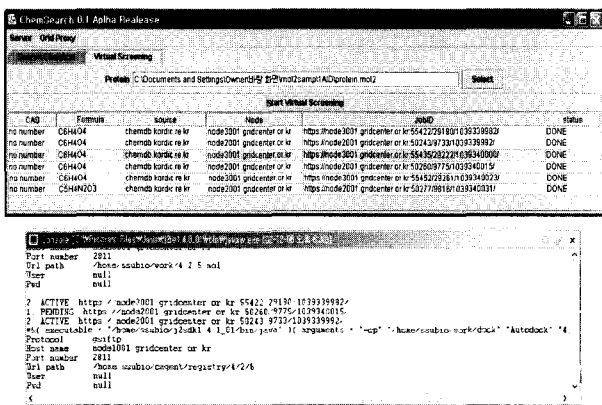
UB Grid 시스템을 이용해 화합물 데이터를 검색하고 버추얼 스크리닝을 수행하는 과정은 (그림 6)~(그림 8)의 과정과 같다.

(그림 6)은 화합물을 검색한 후 검색 결과 목록에서 필요한 화합물을 선택하여 작업 큐에 추가하는 화면이다. 클라이언트 프로그램은 분자설계연구센터 Organic Chemical Database의 검색 옵션과 유사한 인터페이스를 가지도록 구현하였다. 검색 결과로 수집된 데이터의 수와 웹 사이트를 통해 검색한 데이터의 수가 일치하였고, 메타 데이터들도 정확히 수집되었다.



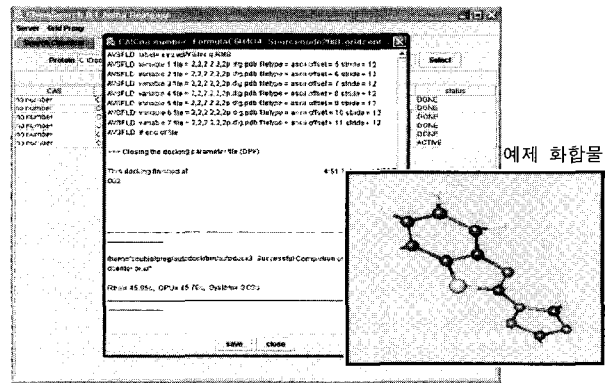
(그림 6) 화합물을 선택하여 작업 큐에 추가

(그림 7)은 UB Grid 시스템에서 에이즈 바이러스의 단백질을 선택하고 가상 스크리닝 작업을 수행하는 모습이다. 위 창은 작업의 상태를 보여주고, 아래 창은 작업 수행 중 작업 상태의 변화 등을 보여준다. JobID 부분은 Globus Toolkit에서 할당받은 작업 ID를 보여준다.



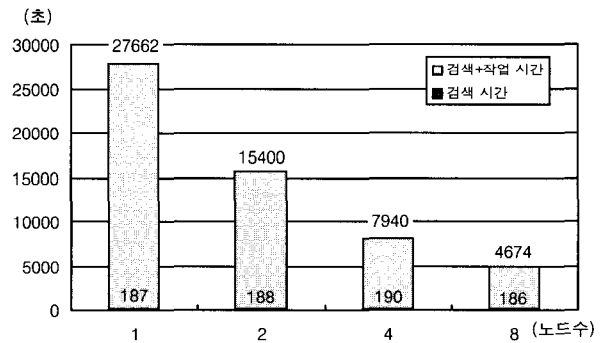
(그림 7) 작업 ID 할당 및 작업 상태 변화

(그림 8)은 작업이 끝난 예제 화합물에 대한 버추얼 스크리닝의 결과를 보여준다. 버추얼 스크리닝의 결과는 로그 파일로 예제 화합물을 여러 각도에서 단백질에 결합시킬 때 발생하는 에너지 값들과 작업이 수행된 노드, 시간 등이 기록된다.



(그림 8) 버추얼 스크리닝의 결과

(그림 9)는 클러스터의 1대, 2대, 4대, 8대의 노드에서 동시에 화합물 검색 및 버추얼 스크리닝을 수행한 작업 시간을 보여준다. 그림 9에서 화합물의 검색 시간은 전체 작업 시간의 아래 부분에 진하게 포함되어 있다. 검색 시점의 네트워크 속도에 따라 약간의 차이가 있지만, 화합물 검색에 소요된 시간은 동일한 데이터를 검색하므로 거의 일정한 값을 유지하였다. 화합물 검색 시간을 포함한 작업 시간의 경우, 클러스터에서 작업을 수행하는 노드를 증가시킴에 따라 200개의 화합물을 처리하는 시간이 확장성 있게 감소하였다.



(그림 9) 화합물 검색 및 버추얼 스크리닝 작업 수행 시간

UB Grid 시스템을 이용하여 전체 그리드 시스템의 성능과 확장성을 크게 감소시키지 않으면서, 다량의 화합물을 쉽게 검색하고, 그리드 시스템을 이용하여 다량의 버추얼 스크리닝 작업을 훨씬 편리하고 효율적으로 수행할 수 있음을 확인할 수 있다. 사용자가 본 시스템을 사용하지 않고, 다량의 화합물을 웹에서 검색한 후, 그 결과를 수집하여 작업 노드로 전송하고 수동으로 작업을 요청한다면, 작업의 수행에 보다 많은 시간과 노력이 필요하다.

6. 결 론

본 논문에서 구현한 검색 시스템을 UB Grid 시스템에 적용하여 그리드 시스템에 웹에서 검색이 가능한 여러 데

이터베이스들을 손쉽게 통합할 수 있었다. 또 데이터를 별도의 저장 장치에 저장하고 인덱스를 생성하여 검색 기능을 제공하는 기존의 연구와 비교하여, 현재 서비스되고 있는 웹 사이트에 관한 정보 파일을 작성하는 것만으로 손쉽게 데이터베이스를 그리드 시스템에 연결할 수 있었다. 그리고 데이터가 위치한 서버의 보안성 및 안정성을 해치지 않고 데이터에 접근할 수 있었다. 데이터베이스를 제공하는 웹 사이트의 내용이 변경되었다고, 그에 따라 XML 서버 정보를 수정함으로써 적응력있게 대체할 수 있다.

본 논문에서의 연구 내용은 기존에 존재하는 정형화된 웹 데이터베이스들을 그리드 시스템에 통합하는 기능을 제공하여 그리드 시스템의 구축과 사용에 편의성을 제공하였다. 앞으로는 데이터의 안정성 및 전송 성능을 향상시키기 위해서, 데이터 복제 기능을 추가하고, 검색 엔진의 서버 정보 파일을 보다 쉽게 작성할 수 있는 도구에 관한 연구를 수행할 계획이다.

참 고 문 헌

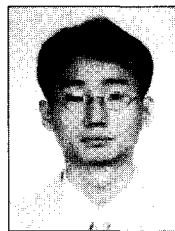
[1] I. Foster, C. Kesselman, S. Tuecke, "The Anatomy of the Grid : Enabling Scalable Virtual Organizations," International J. Supercomputer Applications, 15(3), 2001.
 [2] I. Foster, C. Kesselman, "Globus : A Metacomputing Infrastructure Toolkit," International J. Supercomputer Applications, 11(2), pp.115-128, 1997.
 [3] Rajkumar Buyya, Kim Branson, Jon Giddy and David Abramson, "The Virtual Laboratory : A Toolset for Utilising the World-Wide Grid to Design Drugs," 2nd IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2002), Berlin, Germany, May, 2002.
 [4] Asad Samar, Heinz Stockinger, "Grid Data Management Pilot (GDMP) : A Tool for Wide Area Replication," IAS-TED International Conference on Applied Informatics (AI2001), Innsbruck, Austria, February, 2001.
 [5] <http://www.eu-datagrid.org>.
 [6] <http://www.usatlas.bnl.gov>.
 [7] <http://www.griphyn.org>.
 [8] I. Foster, J. Voeckler, M. Wilde, Y. Zhao, "Chimera : A Virtual Data System for Representing, Querying and Automating Data Derivation," Proceedings of the 14th Conference on Scientific and Statistical Database Management, Edinburgh, Scotland, July, 2002.
 [9] <http://chemdb.kordic.re.kr>.
 [10] <http://www.scripps.edu/pub/olson-web/doc/autodock>.
 [11] <http://smog.com/chem/babel>.
 [12] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder, S. Tuecke, "GridFTP Protocol Specification," GGF GridFTP Working Group Document, September, 2002.

[13] S. Zhou, "LSF : Load sharing in large-scale heterogeneous distributed systems," In Proc. Workshop on Cluster Computing, 1992.



이 상 군

e-mail : seventy9@ss.ssu.ac.kr
 2001년 숭실대학교 컴퓨터학부(학사)
 2003년 숭실대학교 컴퓨터학과(석사)
 2003년 현재 숭실대학교 컴퓨터학과 (박사과정)
 관심분야 : 그리드, 초고속 컴퓨팅(HPC), 전자상거래



황 석 찬

e-mail : seogchan@ss.ssu.ac.kr
 1998년 숭실대학교 컴퓨터학과(석사)
 2003년 숭실대학교 컴퓨터학과(박사)
 2003년 현재 숭실대학교 컴퓨터학과 박사후 과정
 관심분야 : 분산/병렬 컴퓨팅, 그리드컴퓨팅, 시스템 소프트웨어, 미들웨어



최 재 영

e-mail : choi@comp.ssu.ac.kr
 1984년 서울대학교 제어계측공학과(학사)
 1986년 미국 남가주대학교 전기공학과 (석사)
 1991년 미국 코넬대학교 전기공학부 (박사)
 1992년~1994년 미국 국립 오크리지연구소 연구원
 1994년~1995년 미국 테네시 주립대학교 연구교수
 2001년~2002년 미국 국립 슈퍼컴퓨팅 응용센터(NCSA) 초빙 연구원
 1995년~현재 숭실대학교 정보과학대학 컴퓨터학부 부교수
 관심분야 : 고성능컴퓨팅(HPC), 병렬/분산처리, 시스템 소프트웨어



노 경 태

e-mail : ktno@camd.ssu.ac.kr
 1978년 연세대학교 화학과 졸업
 1980년 한국과학기술원 화학과(석사)
 1983년 한국과학기술원 화학과(박사)
 1983년~2003년 숭실 대학교 화학과/생명정보학과 교수
 1997년~2003년 숭실 대학교 분자설계연구소 소장
 2002년~2003년 사단법인 분자설계연구소 소장
 2003년~현재 연세대학교 생명공학과 교수
 관심분야 : 컴퓨터를 이용한 신약개발, 생물정보 관련 S/W 개발