

DGNP : 고성능 계산을 위한 동적 그리드 이름 프로토콜

권오경⁰, 박형우, 이상산
한국과학기술정보연구원 슈퍼컴퓨팅센터
{okkwon⁰, hwpark, sslee} @kisti.re.kr

DGNP : Dynamic Grid Naming Protocol for High Performance Computing

Ohkyoung Kwon⁰, Hyoungwoo Park, Sangsan Lee
Supercomputing center, Korea Institute of Science and Technology Information(KISTI)

요 약

인터넷 기반의 컴퓨팅 환경이 발전함에 따라 자원과 정보의 공유를 위한 그리드 컴퓨팅이 나타나게 되었다. 그리드 컴퓨팅에서는 그리드 응용 수행시 프로그램과 데이터간의 위치가 다르고 분산되어 있는 경우가 많다. 현재 그리드 미들웨어 시장 표준인 글로버스 툴킷(Globus Toolkit™)에서 사용하는 GASS(Global Access to Secondary Storage)는 원격에 있는 관련 데이터들을 로컬 시스템의 데이터에 접근하는 것처럼 처리하는 것을 지원한다. GASS에서는 원격지의 파일 시스템의 접근시 URL(Uniform Resource Locator)을 이용하는데, 그리드 환경에서의 I/O를 위한 파일 시스템은 복수의 파일 형태로 분산되어있기 때문에 하나의 그리드 응용을 수행할 때 URL을 동적으로 바꿔야 하는 문제점을 갖고 있다. 위치에 상관 없고 고성능을 지향하고 데이터의 성격을 쉽게 표현할 수 있는 이름 체제가 필요하다. 그래서 본 연구는 동적이고 다양한 표현이 가능한 URI(Uniform Resource Identifiers)를 사용하여 그리드 상에서 데이터를 사용하는 방법을 제안하고자 한다.

1. 서 론

인터넷 기반의 컴퓨팅 환경이 발전함에 따라 네트워크 연결 이상의 새로운 서비스 개념이 나오기 시작했다[1]. 그 중 대표적인 것 중의 하나가 자원과 정보의 공유를 위한 그리드 컴퓨팅이다. 사용자들이 이때까지 접하지 못했던 여러 슈퍼컴퓨터의 큰 자원을 그리드를 통해 얻을 수 있게 된다.

그리드 응용 프로그램은 원격 데이터를 사용하거나 출력하는 일이 빈번하다. 뿐만 아니라 사용하는 데이터의 파일 시스템도 다양하다. 글로버스 툴킷은 이를 위해 데이터를 표현하는 방법을 URL을 사용한다[2]. 표준화된 이름을 가지고 여러 파일 시스템을 쉽게 이용할 수 있다. 하지만 URL 시스템은 데이터의 정적인 성격만 표현이 가능하다. 그리드의 데이터는 쉽게 변하고 이동성 또한 많다. 위치에 상관 없고 고성능을 지향하고 데이터의 성격을 쉽게 표현할 수 있는 이름 체제가 필요하다. 그래서 본 연구는 동적이고 다양한 표현이 가능한 URI를 사용하여 그리드 상에서 데이터를 사용하는 방법을 제안하고자 한다.

제한한 이름 체제를 지원하기 위해 글로버스 툴킷의 응용프로그램인 globus-gass-server를 수정하였다. 그리고 위와 같은 URI기반 응용 프로그램이 작동할 수 있는 환경을 위해 글로버스 툴킷의 작업수행기인 globusrun을 고쳤다.

2. 관련 연구

글로버스 툴킷에서는 GASS (Global Access to Secondary Storage) 컴포넌트를 통해서 URL을 사용해서 파일 접근을 이룬다. WebFS[3]와 UFO[4] 또한 GASS처럼 URL를 통해 파일 이름을 정함으로써 다른 파일 시스템에서의 접근을 가능하게 한다. 하지만 이것은 정적인 파일 정보만 가질 수 있다.

글로버스 툴킷의 리플리카 카탈로그 시스템은 논리적 이름을 등록시킴으로써 동적인 시스템을 연결할 수 있는 이름 구조를 가지고 있다[5]. 이것은 데이터의 여러 특징을 반영 하지 못하고 확장성이 부족하다.

Avaki에서 제안한 SGNP(Secure Grid Naming Protocol)에서는 동적으로 물리적 주소로 바인딩할 수 있는 확장성 있는 이름 체제와 프로토콜 바인딩, 암호화 정보까지 가지고 있다[6]. 하지만 이것 또한 고성능 컴퓨팅 환경을 위한 특징을 반영하고 있지는 못하다.

3. URI 기반 그리드 데이터 이름 프로토콜

3.1 그리드 데이터 주소의 특징

본 연구에서 제안하는 이름 체제의 특징은 다음과 같다.
장소 의존적이지 않음 (Location Independent) : 그리드 미들웨어의 디팩토 표준인 글로버스 툴킷에서는 URL을 사

용해서 데이터간의 접근을 한다. 하지만 URL은 데이터의 주소에 대한 물리적인 주소이다. 위치가 바뀌면 작업을 실행시킬 때마다 컴파일을 새로 해야 한다. 하지만 주소를 고정시키지 않고 논리적인 주소 체제를 사용한다면 그리드에 맞는 능동적인 이름을 사용할 수 있게 된다. 글로버스 툴킷의 리플리카 서비스처럼 논리적인 주소(URN, Uniform Resource Name)를 URL로 만들게 된다. 그리고 URL주소 자체 또한 환경변수를 통해서 동적으로 변할 수 있다. 즉 그림 1에서 보듯이 W3C에서 제안하고 있는 URI 주소 체제를 사용하게 됨으로써 URL과 URN을 동시에 사용하게 된다[7].

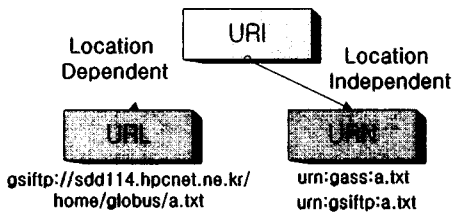


그림 1 W3C의 데이터 이름 주소 체계

데이터를 자동으로 변환(Transformation) : 파일 시스템이 다르면 파일의 데이터 형식도 다르게 된다. 이름에서 프로토콜을 보고 변환할 수 있고 텍스트인지 바이너리 인지 에 따라서 데이터를 바꿀 수도 있다.

그리드 I/O 패턴 지원 (파일의 부분 데이터 읽기) : 그리드 I/O 패턴을 글로버스 툴킷의 요소인 GASS에서 부분적으로 지원을 한다. GASS에서 전체 데이터의 부분만 가지고 오고 싶을 때는 읽고 나서 다시 후 처리를 해줘야 한다. 파일에 대해서 부분 요소를 이름을 이용해서 쉽게 나타낼 수 있다.

데이터의 동적, 정적인 성격 : 파일의 모드가 읽기 전용인 경우에는 정적인 성격을 가지고, 패턴이 쓰기거나 첨가인 경우에는 동적인 성격을 가진다.

고성능을 지향 : 파일의 데이터를 서버 쪽에서 처리해서 부분 데이터만 가져오게 함으로써 네트워크의 대용량을 줄일 수 있다. 파일의 일부 라인과 문자열을 정규표현식 등을 이용해서 가져오게 한다. 파일의 정적인 성격인 경우에는 로컬 캐싱을 지을 필요가 없게 되므로 네트워크를 통해서 다시 가져올 필요가 없게 된다.

3.2 URN 주소 상세 설계

URN형태로 위의 특징을 고려해서 다음과 같이 표현된다 [8].

```

<URN> ::= "urn:" <NID> ":" <NSS>
<NID> ::= <protocol> | <logical-ns>
<protocol> ::= gass | gsiftp | http | file | ftp
  
```

<NID>은 데이터가 저장되어 있는 곳에 대한 네임스페이스 이름이다. <protocol>에서 프로토콜 바인딩에 대한 이름을 정할 수 있다. 예를 들면 gsiftp를 사용하면 GridFTP 프로토콜을 사용하는 파일이고, gass는 globus-gass-server에 의해서 사용되는 것이며, file은 로컬 파일 시스템이다. <logical-ns>는 글로버스 툴킷의 리플리카 위치 서비스에서 지원되는 네임스페이스 이름이다. 위치 서비스는 5장에서 언급하겠다.

```

<NSS> ::= <filename> [<properties>]
<filename> ::= 1*<URN chars>
<properties> ::= 1 * (:<property>)
<property> ::= rmode=<rmode> |
  line=1*<number> - 1*<number> |
  byte=1*<number> - 1*<number> |
  text=<textmode>
<rmode> ::= ronly | wronly | rdwr |
  append
<textmode> ::= text | binary
  
```

<NSS>은 데이터에 대한 이름과 속성을 정하는 것으로 파일에 대한 이름과 속성을 추가할 수 있다. <filename>을 통해서 파일 이름을 정하고 <property>를 사용해서 3.1에서 제안한 이름의 특징을 나타낼 수 있다. 데이터를 자동으로 변환하는 것은 <textmode>를 통해서 클라이언트에서 바뀌게 된다. 설정이 안되면 기본 모드는 binary이다. 파일의 부분 데이터를 읽는 특징은 line과 byte를 통해서 가능하다. line은 <textmode>가 text일 때만 유효하다. line과 byte가 함께 사용할 경우에는 먼저 line을 통해서 가져온 후에 그 중에서 선택된 byte를 다시 추출한다. line과 byte는 해당되는 프로토콜이 서버에서 지원을 해야지 사용이 가능하다. 4장에서 구현에 대해서 더 자세히 다루도록 하겠다. rmode를 통해서 파일의 읽기 전용(ronly), 쓰기 전용(wronly), 읽기 쓰기(rdwr), 쓰기 추가(append)형태의 모드를 설정할 수 있다. 기본 모드는 rdwr이다. 앞에서 언급했지만 파일 모드가 ronly인지와 아닌 것에 따라서 파일의 동적인 성격과 정적인 성격을 구분할 수 있다.

4. 구현

4.1 DGNP 을 지원하는 globus-gass-server

3.2에서 설계한 URN주소에서 파일 데이터의 line과 byte의 부분 요소만 가져오기 위해서는 서버가 지원을 해야 한다. GridFTP를 수정하는 것 보단 쉽게 수정할 수 있는 GASS 서버 응용 프로그램인 globus-gass-server를 수정하기로 하였다. http프로토콜 위에 GS를 지원하게 되는데 여기에 데이터를 전송하기 전에 전 처리를 거쳐서 보내게 된다. line과 byte에 대한 정보는 프로토콜의 헤드 정보에 덧붙여서 보내지게 된다. textmode인 경우에는 바뀐

다 해서 네트워크에 보내는 데이터의 차이가 크지 않으므로 클라이언트에서 처리하도록 한다.

4.2 DGNP 기반 글로벌스 툴킷

3장에서 제안한 이름 구조 체제를 지원하기 위해서 글로벌스 툴킷의 GASS 모듈을 수정하였다. 즉 URL구조로 되어 있는 구조를 URI기반 GASS 모듈로 바꾸었다. 현재 GASS 캐쉬 구조에서는 모든 데이터가 동적인 데이터라고 가정되어 있다. 프로그램이 수행된 후에 무조건 데이터의 내용이 지워진다. 만약 이름이 `rwmode=ronly`이면 GASS 캐쉬의 내용을 지울 필요가 없다. 캐쉬 항목의 Tag에 그런 항목을 들어갈 수 있다. `globus_gass_transfer` 모듈에서 파일의 데이터가 옮겨진 후에 변환 부분을 삽입하였다. `text` 항목에 맞춰서 데이터를 바꾸면 된다.

`globus_gass_file` 모듈을 URI기반 API로 수정을 함으로써 자동으로 URN 형태의 주소는 URL로 바뀌게 되고, 환경 변수가 있는 URL 주소 또한 실제적인 URL주소로 바뀌게 되었다. 동적인 주소가 실제적인 물리적 URL주소로 바뀔 때 `shell` 환경 변수를 사용해서 하게 되는데 글로벌스 툴킷 상에서 작업을 수행하기 위한 언어인 RSL을 통해서 전달이 된다. 필요한 사항인 IP 주소와 파일 시스템의 절대 경로가 포함된다. 위의 사항을 반영하기 위해 글로벌스 툴킷에서 작업을 수행하는 프로그램인 `globusrun`을 고쳤다(그림 2).

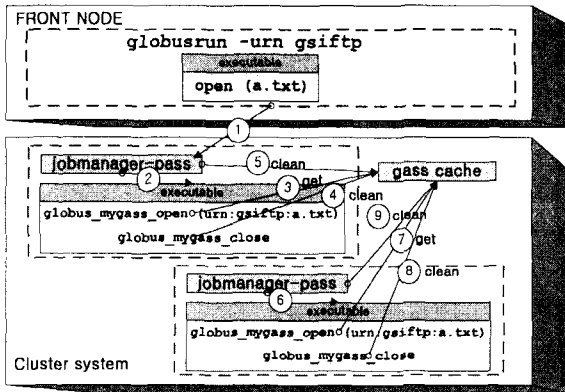


그림 2 urn:gsiftp:a.txt가 globusrun을 통해서 실행되는 모습

5. 결론 및 향후 연구

그리드 응용 프로그램은 원격 데이터를 사용하거나 이동한 일이 빈번하다. 글로벌스 툴킷은 이를 위해 원격의 여러 시스템의 데이터를 표현하는 방법을 URL을 사용해서 통합한다. 하지만 URL 이름 체제는 데이터의 정적인 성격만 표

현이 가능하다. 그리드의 데이터는 쉽게 변하고 이동성 또한 많다. 그래서 W3C에서 제안하고 있는 이름 체제인 URI를 사용해서 위치에 상관 없고 고성능 컴퓨팅을 지향하고 데이터의 성격을 쉽게 표현할 수 있는 이름 체제를 제시하였다. 장소 의존적이지 않고, 데이터를 자동으로 변환 가능하며, 파일의 부분 데이터 읽기, 데이터의 동적, 정적인 성격, 고성능을 지향한다.

현재 그리드는 OGSA기반의 서비스 지향적인 구조로 발전해나가고 있다. 여기에 발맞추어 제안한 DGNP이 OGSA기반에 맞게 차후에 연구가 되어야 할 것이다. 서비스 단위로 등록시킬 때 사용하는 파일의 위치 정보를 바인딩 할 수 있다면 좀더 자동적으로 파일을 찾을 수 있다. DGNP를 글로벌스 툴킷의 리플리카 위치 서비스와 혼합해서 하이 레벨 서비스를 만들 수 있을 것이다.

6. 참고문헌

- [1] I. Foster and C. Kesselman, editors. *The Grid: Blueprint for a Future Computing Infrastructure*. Morgan Kaufmann Publishers, 1999.
- [2] J. Bester, I. Foster, C. Kesselman, J. Tedescoy, S. Tuecke, "GASS: A Data Movement and Access Service for Wide Area Computing Systems", Sixth Workshop on I/O in Parallel and Distributed Systems, May 1999.
- [3] A. Vahdat, P. Eastham, and T. Anderson. *WebFS: A global cache coherent file system*. Technical report, Department of Computer Science, UC Berkeley, 1996.
- [4] A. D. Alexandrov, M. Ibel, K. E. Schauer, and C. J. Scheiman. *Extending the operating system at the user level: The UFO global file system*. In 1997 Annual Technical Conference on UNIX and Advanced Computing Systems (USENIX'97), January 1997
- [5] Replica Selection in the Globus Data Grid. S. Vazhkudai, S. Tuecke, I. Foster. *Proceedings of the First IEEE/ACM International Conference on Cluster Computing and the Grid (CCGRID 2001)*, pp. 106-113, IEEE Computer Society Press, May 2001.
- [6] *Secure Grid Naming Protocol(SGPN) : Draft Specification for Review and Comment*, GWD-R, 5-February, 2002
- [7] *Naming and Addressing: URIs, URLs.*, <http://www.w3.org/Addressing/>
- [8] *URN Syntax*, <http://www.ietf.org/rfc/rfc2141.txt>