

그리드 환경에서 Replica 최적화를 위한 Data Location Service 연구

박희용, 이무훈, 심의규, 최의인

한남대학교 컴퓨터공학과

e-mail : {hypark, mhlee, ekshim, eichoi}@dblab.hannam.ac.kr

A Study on a Data Location Service for optimal Replica in a Grid environment

Hee-Yong Park, Moo-Hun Lee, Eui-Kyu Shim, Eui-In Choi
Dept of Computer Engineering, Hannam University

요 약

단순한 데이터통신을 위한 넷(Net)이 정보 교환의 혁명을 일으켰던 웹(Web)으로 발전하였고, 현재 웹은 신경망 형태의 인터넷 구조를 갖는 그리드(grid)를 향해 발전하고 있다. 정보의 교환 및 분산된 자원을 공유하기 위한 그리드 컴퓨팅은 자원의 발견 뿐 만 아니라, 접근 속도와 제한된 자원의 공유를 비롯한 여러 문제점을 가지고 있다. 특히, 데이터에 대한 접근 속도와 제한적인 데이터 공유 문제를 해결하기 위해 Replica 서비스가 제안되었으나, 이러한 Replica 서비스를 원활하게 지원하기 위해서는 Replica 경로 및 정보들을 목록으로 구성해야만 한다. 현재 그리드 컴퓨팅 분야에서 이러한 목록을 구현하는 것과 동시에 최적의 조건을 찾아가는 기법에 대한 연구가 활발히 진행 중이다. 따라서, 본 논문에서는 Replica 서비스를 최적화하기 위한 기존의 연구들을 분석하고 Data Location Service를 이용하여 Replica 서비스를 최적화하는 방법을 제안하였다.

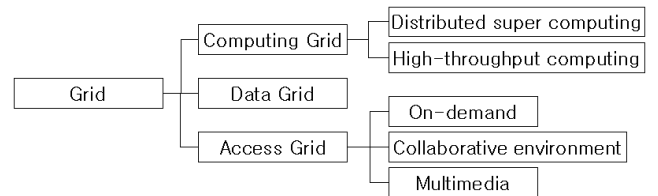
1. 서론

웹을 통하여 정보 교환에 대한 개념이 바뀌게 되었고, 차세대 인터넷 개념으로 정보의 교환 및 모든 자원의 공유를 지원하는 그리드 컴퓨팅이 새롭게 떠오르고 있다. 그리드는 1998년 이안 포스터(Ian Foster)가 처음으로 이론화하여 제시하였으며, Globus 프로젝트와 더불어 다양한 분야로 연구되어져 왔다. 이안 포스터는 그리드를 수많은 조직으로부터 형성된 동적인 가상조직(VOs ; Virtual Organizations)으로서 단순한 정보나 데이터 교환을 하는 것이 아닌 computational 자원, storage 자원, software, 뿐만 아니라 전문 지식과 인적자원까지도 포함하는 모든 자원에 대한 공유라고 정의하였다[1].

현재 수많은 나라와 연구기관들이 그리드를 위한 프로젝트를 진행 중에 있으며, 데이터 그리드의 중심 기관인 CERN에는 이미 전 세계에서 온 수 천명의

연구자들이 참여하고 있다.

그림 1은 그리드를 크게 3가지로 분류해 놓은 것이다[4]. 본 논문에서는 데이터 그리드에 중점을 두고 있다.



(그림 1) Grid의 분류

가상조직으로 이루어진 그리드 내에서 공유된 데이터 자원을 사용하고자 할 때, 몇 가지 문제점이 부딪히게 된다. 그러한 문제점들 중 일부를 살펴보면 사용자로부터 요구된 자원의 위치, 데이터 자원의 제공자 수, 자원 접근을 위한 시간에 대한 문제를 생각해 볼 수 있고, 이러한 문제점의 해결 방안

은 이미 여러 기법으로 연구되어 제시되고 있으며, 그러한 연구 결과 중 하나가 Replica 서비스이다.

Replica 서비스는 위에서 언급한 그리드의 분류 중 데이터 그리드에 속하는 서비스이며, 필요에 따라 생성되거나 삭제되어야 할 Replica들을 관리하고, 목록을 생성해 유지시키는 Replica management를 필요로 한다[2]. 또한 최선의 Replica 선택을 위한 Replica selection을 요구하게 되는데 본 논문에서는 Replica selection을 최적화하기 위한 Data Location Service를 제안한다.

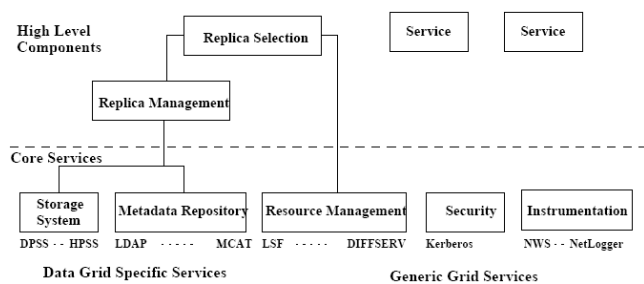
본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대해 기술하고, 3장에서는 본 논문에서 제안하고 있는 Data Location Service에 대해 기술한다. 마지막으로 4장에서 결론을 맺고 문제점을 제시하며 향후 연구 방향에 대해서 언급한다.

2. 관련 연구

2.1 Globus Data Grid

Globus Data Grid architecture[2]는 Grid 환경을 통해 분산된 데이터와 storage 자원 관리를 위한 기반구조를 제공한다. Globus의 데이터 그리드는 core 서비스 계층과 high-level 서비스 계층으로 구성되어있다. 단, 이 계층들에서는 어플리케이션과 도구들이 재사용이 가능하다는 가정을 기반으로 한다.

그림 2에서는 Globus의 데이터 그리드 구조를 설명하고 있다.



(그림 2) Globus Data Grid Architecture

2.1.1 Core Service

Core 서비스들은 high-level 서비스들과 어플리케이션들이 그리드 내에서 storage 자원에 접근할 수 있도록 storage 자원들을 추상화시킨다. storage 시스템과 데이터 액세스 그리고 메타데이터(metadata) 메커니즘으로 storage 자원들에 있는 데이터를 접근하고 관리하기 위한 기본적인 메커니즘을 제공한다.

2.1.2 high-level Service

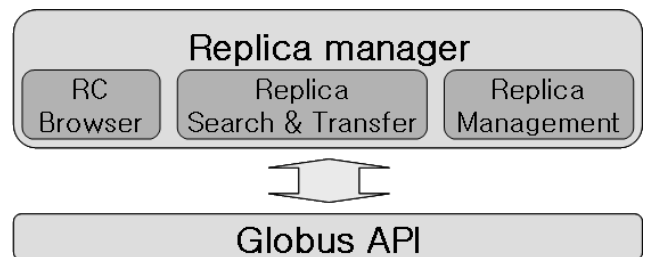
Replica management[2]는 storage 자원에서 Replica의 생성과 삭제에 대한 관리를 수행하고, Replica들은 필요에 따라서 원본 데이터와 똑같이 복제되어 그리드 내의 성능 향상에 이용된다. 또한 Replica manager는 Replica 관리를 위해서 Replica의 사이트들에 대한 주소와 파일 인스턴스 등을 포함한 Replica 목록들을 유지하여야 한다.

Replica selection[2]은 그리드의 전역에 걸쳐 분산되어 있는 Replica들 중에서 필요한 것을 선택하는 과정이다. Replica selection 프로세스는 Replica 목록에서 성능과 데이터 접근에 대한 특징을 이용하여 Replica를 선택하는 어플리케이션의 사용을 허가한다.

2.2 웹 기반 Replica Manager

Globus 툴킷에서 제공하는 Replica management 서비스는 커맨드 라인 기반 인터페이스를 제공하는데 사용자에게는 매우 불편한 방식이다. 사용자가 이용하기 편한 인터페이스를 제공하고, 웹 포털을 통하여 웹 기반 인터페이스를 제공해 주는 GPDK의 새로운 기능으로 추가된다면 그리드 사용자들은 Replica management를 좀 더 쉽게 사용할 수 있을 것이다.

Replica manager[5]는 위의 요구사항을 위해 제안 되었고 이를 통해 Replica의 효율적인 관리 및 사용을 도모하며, 그림 3과 같이 세 가지 모듈로 구성할 수 있다. 그리고 Globus API를 기반으로 동작하게 된다.



(그림 3) Replica Manager 구성 모듈

- RC Browser : Replica 목록 브라우징 기능
- Replica Search & Transfer : 파일 패턴 기반 Replica 검색 및 선택된 Replica 전송
- Replica Management : Replica 등록 및 삭제 기능

사용자의 요구사항은 웹 인터페이스를 통해 전달

되며 입력된 요구 사항은 서블릿(servlet) 엔진을 통하여 Globus API로 전달된 후 처리된다. 처리 결과는 서블릿 엔진을 통해 다시 사용자에게 반환된다.

웹 기반 Replica manager는 Grid 웹 포탈 구축을 가능하게 하는 GPKD와의 통합을 고려하여 자바 서블릿과 JSP(Java Server Page)를 사용하여 구현되었다.

3. Data Location Service

그리드는 분산된 사이트들을 통하여 분배되고, 여러 조직들에 소속될 수 있는 많은 자원들을 가질 수 있다. 그 사이트들은 WAN(Wide Area Network) 링크들로 연결되어 있고, 이러한 링크는 사이트들의 여러 특성에 따라서 각각의 사이트들에 대한 대역폭과 대기시간을 서로 다르게 설정할 수 있다. 이것은 그리드 환경에서 스케줄링 결정 시 중요하게 작용할 수도 있으며, 데이터의 전송시간과 크기, 사용하는 대역폭과 함께 컴퓨팅 비용에 중대한 영향을 끼치기 때문이다. 컴퓨팅 비용을 최소화하기 위해서는 위와 같은 요소들이 최적화를 이루어야 한다.

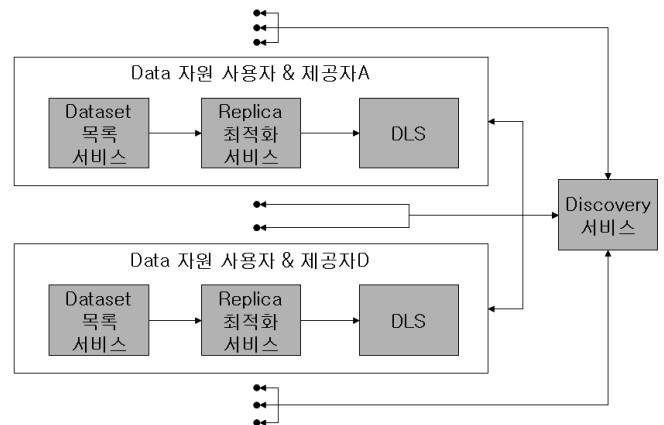
그리드 환경에서는 사용자를 위해서 데이터 자원에 대한 접근 방법과 Replica Location Service가 서로 조합되어야 할 필요성이 있다. 사용자의 에이전트는 접근하려는 자원에 대한 논리적 이름(filename)과 실질적인 물리 경로를 동시에 파악할 수 있어야만 최적화된 비용을 사용하여 자원을 이용할 수 있기 때문이다. 이와 같이 자원의 위치, 네트워크 경로 그리고 접근을 위한 대기시간 등을 고려하여 원하는 data 자원에 최선의 선택을 통해 접근하는 것을 가능하게 해주는 것이 Data Location Service(DLS)이다. 이 서비스를 위해 그리드 내에서는 데이터 자원의 위치, 네트워크 경로, 접근 대기시간 등을 모니터링하는 서비스로부터 정보를 얻어야 한다. 이러한 정보를 수집하는 서비스는 반드시 경량(light-weight)이어야만 하는데, 그렇지 못할 경우 DLS의 의미가 무색해지기 때문이다. 이렇게 얻어진 정보를 이용하여 최적화된 자원에 접근할 수 있다.

DLS는 좀 더 효율적인 Replica 정보를 제공할 수 있다. Data Location 프로세스가 수많은 Replica 목록들 사이에서 최적화된 Replica를 선택하는 어플리케이션의 이용을 허가하기 때문인데 이 어플리케이션은 데이터 접근에 대한 특징과 시간, 접근 자원의 적재(load) 상태와 성능에 따라서

Replica를 선택하게 된다. 사용자 어플리케이션에 의한 Replica의 선택은, 자원의 논리적 이름(file name)에 대한 접근이 요청되면 DLS는 최적화된 선택을 하기 위해서 물리 경로 정보를 포함하고 있는 Replica 목록에서 선택된 정보를 가져오는 과정으로 이루어진다. 이러한 결정을 하는 과정에서 중대한 영향을 끼치는 요소들은 네트워크의 속도, 데이터의 위치, 파일의 크기, 데이터 전송 시간과 기타 사항들과 같은 표준화된 자료들이다.

DLS는 폴트 톨러런트(fault tolerant) 기능을 가지고 있어야 한다. 사용자가 데이터 자원에 접근하여 작업을 수행 중에 있을 때, 접근 중인 데이터 자원 제공자가 오프라인(offline)이 된다면 작업 중인 프로세스는 현재 작업 데이터에 대한 손실을 입을 수밖에 없게 된다. 작업 데이터 손실을 막기 위해서 DLS는 집중적이지 않아야 하고 폴트 톨러런트 기능으로 다른 자원 제공자에게 접근하여 진행 중이던 작업을 계속적으로 할 수 있어야 한다. 그러므로 항상 최적화되어 있는 Replica 목록을 유지하여야 하고 갱신하여야 한다.

본 논문에서 제안하고 있는 DLS의 구조는 그림 4와 같이 표현할 수 있다.



(그림 4) Data Location Service 구조

Discovery 서비스가 사용가능한 데이터 자원들을 찾아내기 위해서 Dataset 목록 서비스에 접근하기를 원한다. 하지만 데이터 자원에 대한 구체적인 정보가 없다면 접근 후에 사용가능 여부를 알게 될 것이고, 사용이 불가능하거나 원하는 데이터 자원이 아닐 경우 새로운 데이터 자원을 찾아가는 비용이 증가하게 된다. 하지만 Replica 최적화 서비스를 이용해 데이터 자원에 대한 구체적인 정보들을 수집하고 명세하여 DLS에게 전달하면 DLS는 사용가능한

데이터 자원의 모든 위치를 파악해 리스트로 만들어 Discovery 서비스에게 전달한다. 그 밖에도 DLS는 데이터 자원의 제공뿐만 아니라 다른 네트워크의 데이터 자원 사용을 위한 네트워크 비용도 측정한다. 다른 데이터 자원의 사용을 위해서 위에서 언급했던 최적화된 데이터 자원 선택을 위한 조건들과 과정들을 제공하는 동시에 받을 수도 있어야 하기 때문이다.

4. 결론

본 논문에서는 데이터 그리드 환경에서 데이터 자원에 대한 접근을 위한 Dataset 목록 서비스와 데이터 자원의 효율적인 관리를 위한 Replica 서비스를 효율적으로 사용/관리할 수 있는 Data Location Service를 제안하였다. 결론적으로 Globus 프로젝트에서 제시되었던 Replica management와 Replica selection에 DLS를 추가하여 데이터 자원에 대한 접근과 데이터 자원 제공에 대한 성능을 개선할 수 있다.

본 논문에서는 DLS를 이용한 Replica의 최적화는 데이터 자원 접근과 제공에 대해 향상된 성능을 제시하고 있지만, Discovery 서비스로의 정보 전달 인터페이스에 대한 문제점을 야기할 수 있다. 작게는 하나의 조직 내에서 서비스되는 Discovery 서비스와 넓게는 다수의 조직으로 이루어진 가상 조직에서의 Discovery 서비스를 구분해 줄 시스템의 필요한 것이다.

향후에는 각각의 Discovery 서비스의 요청이 있을 때, 3장에서 언급한 조건들을 비롯하여 요청한 사용자의 위치 파악과 모니터링, Discovery 서비스에 대한 구체적인 구분을 통하여 그에 맞는 서비스를 제공해 줄 시스템에 대한 연구가 필요하다.

참고문헌

- [1] Ian Foster, Carl Kesselman, Steven Tucke "The Anatomy of the Grid", Internateional J. Supercomputer Application, 2001
- [2] Sudharshan Vazhkudai, Steven Thcke, Ian Foster "Replica Selection in the Globus Data Grid", In International Workshop on Data Models and Databases on Clusters and the Grid (DataGrid 2001). IEEE Computer Society Press, 2001.
- [3] Ian Foster, Carl Kesselman, Jeffrey M. Nick,

Steven Tucke "The Physiology of the Grid", <http://www.globus.org/research/papers/ogsa/pdf>

[4] 이춘희, "그리드 컴퓨팅(Grid Computing)", 한국정보처리학회 학회지, 제10권 제1호, 2003.

[5] 김은성, 박형우, 이상산, 정진욱, "데이터 그리드 위한 웹 기반 Replica Manager의 설계 및 구현", 한국정보처리학회 춘계학술발표 논문집, 제9권 제2호, 2002.

[6] Jang-uk In, Paul Avery, Richard Cavanaugh, Laukik Chitnis, Mandar Kulkarni, Sanjay Ravka, "SPHINX: A fault-tolerant system for scheduling in dynamic grid environments", in the processings of the 19th IEEE International Parallel & Distributed Processing Symposium(IPDPS 2005), Denver, Colorado, April, 2005

[7] <http://public.web.cern.ch/>

[8] <http://www.globus.org/wsrp>

[9] Allcock. B., Bester. J., Bresnahan. J., Chervenak. A.L., Foster. I., Kesselman. C., Meder. S., Nefedova. V., Quesnel. D. and Tuecke. S., Secure, "Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing", In Mass Storage Conference. 2001.

[10] W.Allcock, A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, S. Tuecke, "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets", Journal of Network and computer Applications, 23:187-200, 2001.