

# 슈퍼컴퓨팅 환경하의 정책기반 그리드 어카운팅 시스템 설계

성주영<sup>0</sup> 김경수 김법균 황호진 박동선 정성중 \*장행진  
전북대 \*KISTI  
[zezejong@orgio.net](mailto:zezejong@orgio.net)<sup>0</sup>

## Design of Policy-based Grid Accounting System on Supercomputing Environments

Joo-Young Seong<sup>0</sup> Kyoung-Su Kim Beob-Kyun Kim Ho-Jeon Hwang Dong-Sun Park  
Sung-Jong Chung Haeng-Jin Jang  
Dept. of Information & Communication, Chonbuk National University

### 요 약

그리드는 지리적으로 분산되어 있는 이기종의 유휴 자원들을 서로 연결하여 하나의 가상 컴퓨터처럼 동작하는 환경이다. 특히, 그리드 어카운팅은 각 자원 제공의 기준을 적용할 수 있기 때문에 그리드 시스템 구축에 필수적인 요소이다. 본 논문에서는 기존의 그리드 어카운팅 시스템들에서 발생할 수 있는 부하 집중, ID 중복 등의 문제점들을 해결하기 위해 슈퍼컴퓨터 환경에서의 새로운 정책기반 그리드 어카운팅 시스템을 설계하였다. 제안한 시스템은 Globus Toolkit을 미들웨어로 사용하며, 각 자원 제공자가 기존 시스템 관리 기능을 넘지 않는 한 도 내에서 정책기반 그리드 어카운팅을 사용할 수 있도록 하였다.

### 1. 서 론

인터넷이 보편화되고 컴퓨터 및 네트워크의 성능이 향상됨에 따라 분산된 슈퍼컴퓨팅 기반의 고성능 어플리케이션들은 더 큰 컴퓨팅 파워를 요구하고 있다. 그리드[1]는 지리적으로 분산된 고성능, 대용량의 자원들과 첨단 장비들을 원격에서 동시에 사용하여 단일 시스템처럼 사용하는 환경이다. 이런 분산된 공유 자원들을 더 체계적으로 관리하고, 서로 다른 어카운팅 정보를 생성하는 이기종 사이트들의 어카운팅 정보를 수집, 분석하기 위한 어카운팅 시스템은 그리드에서 반드시 필요하다.

본 논문은 분산된 슈퍼컴퓨팅 자원을 통합한 그리드 환경에서 로컬 사용자들을 위하여 정책 기반의 어카운팅 모델을 제안한다. 현재 구현되어 있는 VUS와 템플릿 어카운팅의 방법은 정보관리 서버의 부하 집중, ID 중복 등의 문제점이 발생한다. 이런 문제점을 해결하기 위해 제안한 모델에서는 Client가 Server에 접속했을 때 로컬 사이트의 정책에 기반한 User ID 발급 방법과 필요한 정보들과 원하는 프로그램을 찾아 로그인하는 방법 및 흐름을 제시하였다. 또한 MDS(Meta computing Directory Service) 기능을 돕는 정책기반 MDS Server(Policy-based MDS Server)를 추가하여 Client가 리소스를 더 정확히 찾을 수 있도록 하였다.

### 2. 그리드 환경에서의 어카운팅 시스템

#### 2.1 그리드와 Globus Toolkit

그리드는 분산된 고성능, 대용량의 자원들과 첨단장비들을 원격에서 동시에 사용하는 것을 목적으로 설계되었다. 그러므로 그리드에서는 보안, 스케줄링, 자원 할당, 자원 결합, 자원 정보 수집 등의 서비스가 제공되어야 한다. 그리드는 그리드 기반요소, 그리드 미들웨어, 그리드 개발환경, 그리드 어플리케이션으로 구성된다. 그 중 위의 서비스를 제공하기 위해 필수적인 요소가

그리드 미들웨어이다[9][10][11].

그리드 미들웨어는 그리드 응용 서비스와 네트워킹 서비스 기능들을 제공하기 위한 도구로서 Globus, Legion, Condor, Cactus[1] 등이 있다. Globus Toolkit은 그리드에서 필요로 하는 다양한 서비스들을 독립적인 요소로써 제안하고, 기존의 각 시스템 및 네트워크의 관리 정책에 협력하여 그리드를 이루어 나가므로 전세계적으로 가장 많이 사용되고 있다. Globus Toolkit[2][10]은 크게 그리드 보안, 정보 서비스, 자원 관리, 데이터 관리 등으로 나뉜다. 보안을 담당하는 부분은 GSI[3]로서 single-sign-on과 분산된 각 자원에 대한 사용자의 인증을 proxy가 대행하여 그리드 환경에 노출된 자원들을 안전하게 하고 사용자들이 그리드 환경에 더 쉽게 접근할 수 있게 한다. 분산된 각 로컬 자원의 정보 서비스는 MDS[4]가 담당한다. MDS는 그리드 환경 안에 존재하는 자원들의 상태정보 및 종류를 사용자에게 제공하며 이 서비스를 위해서 LDAP[5]를 이용한다. 자원관리는 GRAM[6]이 담당하며 어카운팅 시스템과 더불어 그리드에서 핵심이 되는 부분이다. 마지막으로 데이터 관리를 위해 GASS, GridFTP 등을 제공한다.

#### 2.2 기존의 어카운팅 시스템

어카운팅 시스템은 분산된 자원을 공유하고 사용하기 위해 각 자원에 대한 접근 및 사용 권한, 해당 사이트에 대한 인증, 사용된 자원에 대한 과금 서비스(billing service)의 역할을 한다. 이기종의 각 사이트들은 작업을 수행하고 서로 다른 형태의 어카운팅 정보를 생성한다. 어카운팅 시스템은 각 사이트에서 생성된 어카운팅 정보를 수집하고 분석함으로써 요금을 산출할 수 있다.

##### 2.2.1 VUS (Virtual Users' Accounts Systems)

폴란드의 슈퍼컴퓨팅 센터[12]에서 구축하여 현재 상용

서비스를 하고 있다. VUS[7]는 Globus를 기반으로 구현되지 않고 독자적으로 구현되었다.

VUS에서 모든 작업은 서로 다른 버추얼 어카운트 상에서 실행되고 하나의 작업은 단일 어카운트에서 실행된다. 각 로컬의 VUM(VU' Accounts Manager)들은 실제로 작업을 수행하기 위해 VUID(VU' ID)를 발급하고 VUSD (VU' Server Daemon)에게 그것에 관련하여 정보를 전송한다. 이때 VUS는 각 로컬의 LM(Log Manager)과 통신하여 자원 사용 정보를 수집하고 관리한다. VUS가 모든 어카운팅 정보를 유지 관리하며 실제 사용자 머신과 작업을 수행하는 머신 사이에 중계 역할을 담당한다.

이 모델의 경우, 모든 어카운팅 정보를 VUS가 유지 관리하므로 VUS에 부하가 집중된다는 단점이 있어, 이로 인해 발생하는 여러 문제점이 있을 수 있다.

### 2.2.2 템플릿 어카운팅

T.Hacker가 설계한 모델[8]로 사용자 계정간의 바인딩에 대한 부분을 모델링한 시스템이다.

자원을 제공하는 사이트에서 Template Account을 발급하여 실제 사용자의 계정과 일시적으로 바인딩시키는 형식을 취한다. 특히, 대부분의 그리드 사용자들이 사용하는 어플리케이션이나 Site에 대해 지역성(locality)을 가진다는 점에 착안하여 특정 사용자 계정에 이미 바인딩되어 있는 Template Account의 경우 일정 시간동안은 이미 바인딩되어 있는 계정에 우선권을 주고, 시간이 지나면 바인딩을 해제하여 다른 사용자 계정과 바인딩 되도록 하는 방법이다.

그러나, 이 모델의 경우 Dynamic Account가 현존하는 운영체제에서는 지원하지 않는다는 점에서 현실성이 없다. 이 모델이 성공하기 위해서는 운영체제 수준에서 Template Account를 필요할 때 발급하고 사용되지 않을 때는 삭제하는 기능을 지원해야 한다.

## 3. 정책기반 어카운팅 시스템

### 3.1 전체 구조

앞에서 언급한 바와 같이 어카운팅은 로컬 유지나 자원의 관리 등에서 중요한 역할을 한다. 현재의 그리드 환경의 어카운팅에서는 여러 가지 방법이 있으나 문제점이 발생한다. 또한 MDS가 리소스의 연결포인트의 정보만을 가지므로 Client가 원하는 프로그램 정보나 노드의 사용가능 등의 정보를 알 수 없다. 이를 해결하기 위해 그림 1과 같이 정책기반 어카운팅 시스템의 구조를 설계하였다.

그림1에서처럼 Client, Server, Super MDS Server, Policy-Based MDS Server로 구성되면 각 부분의 설명은 다음 절에서 설명한다.

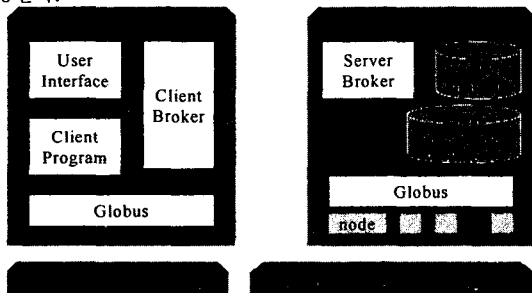


그림 1 정책기반 어카운팅 시스템 구조

### 3.2 각 부분의 역할

#### 3.2.1 Client

위의 그림1에서처럼 User Interface, Client Program, Client Broker 등으로 구성된다.

##### 3.2.1.1 User Interface

Client가 job을 실행시키기 위해 로그인 했을 때 Client Program을 통해 proxy를 띄워주는 역할을 한다.

즉, Client가 id를 넣었을 때 인증된 id이면 Client Program으로 들어가 password를 넣게 된다. 이 password와 id에 대응하는 CN을 통해 proxy를 띄워주게 된다. 이 proxy를 통해 Client는 job을 submit 할 수 있다.

##### 3.2.1.2 Client Program

Client가 job을 쉽게 submit하게 하기 위해 GUI 형태로 Client와 User Interface, Client Broker를 연결해 주며 로그인, Job Specification, MDS Browsing으로 구성된다. Job Specification은 Client가 실행하고자 하는 job의 설정에 관한 부분으로 실행 파일, 데이터 파일, specification (node 수, speed/MB, disc 할당 등) 등을 넣어주게 된다. MDS Browsing은 Super MDS Server와 Policy-Based MDS Server를 Browsing한다.

##### 3.2.1.3 Client Broker

Client와 Server를 Client Program을 통해 연결시켜주는 부분으로 proxy가 띄워진 후 Client가 Job Specification을 하면 Super MDS Server와 Policy-Based MDS Server를 통해 MDS Browsing을 보여주는 역할과 Client가 원하는 Server 선택 시 그 Server Broker와 협상하는 역할을 하게 된다. 또한 협상으로 원하는 Server를 모두 찾게 되면 Globus를 통해 job을 submit하게 한다.

#### 3.2.2 Server

그림1에서처럼 Server Broker, Local Policy-Based MDS Server, grid-mapfile 등으로 구성된다.

##### 3.2.2.1 Server Broker

Client가 Client Program의 MDS Browsing에서 선택한 각 Server의 Broker로써 Client Broker와 각각의 정책에 따라 협상을 하는 역할을 한다.

만약 협상 결과 job을 받아들이면 grid-mapfile에 이 Client의 CN을 추가한다. 본 논문에서는 각 Server가 여러 개의 ID를 가지고 각 Client의 CN과 일대 일로 mapping 되게 함으로써 일대 다로 했을 때의 복잡성 등의 문제를 줄였다. 따라서 grid-mapfile은 CN과 ID를 일대 일로 mapping하는 역할을 하게된다.

##### 3.2.2.2 Local Policy-Based MDS Server

본 논문에서 제안한 CN과 ID의 일대 일 mapping 방식에 의해 각 ID가 사용가능한 software과 node 수를 다음 표1과 같이 DB로 저장해 두는 역할을 한다. 이 DB는 각 Server마다 존재하며 Policy-Based MDS Server에 각 Server가 제공하는 Software와 그 Software의 사용가능한 최대 node 수의 정보를 주어 Client Broker와 Server Broker의 협상 시 도움을 준다.

표 1 Local Policy-Based MDS Server

G1	STAR_CD	3~5
G2	STAR_CD	4~10
...	...	...

3.2.3 Super MDS Server

앞에서 언급한 Globus Toolkit의 MDS의 역할을 한다. 즉, 각 지역에 퍼져 있는 리소스의 contact point, 상태, 사용가능한 node 등의 정보를 제공해 준다. 현재는 다음 표2와 같이 각 지역 리소스의 contact point를 제공하는 역할만을 하고 있다.

표 2 Super MDS Server

전북대	001.chonbuk.ac.kr
KISTI	001.kisti.ne.kr
포항공대	001.postech.ac.kr
...	...

3.2.4 Policy-Based MDS Server

본 논문에서 추가한 것으로 Client Broker와 Server Broker의 협상 시 더 정확한 정보를 제공해 주게 된다. Super MDS Server의 부족한 정보를 보완하는 역할을 하며 각 Server의 Local Policy-Based MDS Server의 자료를 제공받아 각 Site에서 Software의 가능한 최대 node 수를 알려준다. Policy-Based MDS Server는 다음 표3과 같이 구성된다. Client에게 MDS Browsing 할 때 Super MDS Server의 내용과 함께 보여준다.

표 3 Policy-Based MDS Server

전북대	STAR_CD	7
KISTI	STAR_CD	10
포항공대	STAR_CD	5
...	...	...

3.3 전체의 흐름

제안한 정책기반 어카운팅 시스템의 전체 흐름도는 그림과 같다.

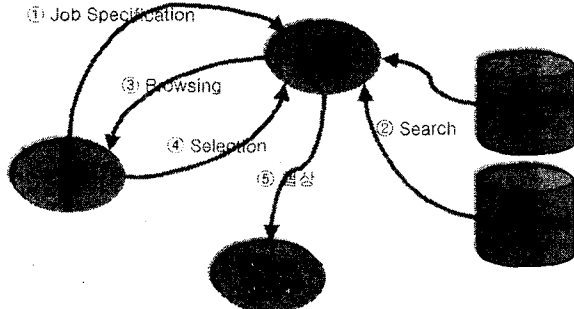


그림 2 전체 흐름도

Client가 User Interface를 통해 proxy를 띄운 후

1. Client Program을 통해 Client Broker에게 Job Specification을 알려주게 된다.
2. 이 정보를 가지고 Client Broker는 super MDS Server와 Policy-Based MDS Server의 정보를 가져오게 된다.
3. 두 Server에서 가져온 정보로 Client Program을 통해 Client에게 각 리소스의 Site, Software, node 수 등의 정보를 Browsing한다.
4. 이 MDS Browsing 정보를 보고 Client는 적절한 리소스를 선택하게 된다.
5. Client Broker는 선택된 Server의 Broker와 협상을 하게 된다. 이 때 Server Broker는 그 Server의 정책에 따라 수락하거나 거절하게 된다. 거절할 경우 Client Broker는

Client에게 다른 Server를 선택하게 하여 다른 Server와 협상하게 되고, 수락하게 되는 Server Broker는 grid-mapfile에 이 CN을 추가한다.

이 과정을 거쳐 적절한 리소스를 Client가 선택하게 된 후 Client가 Client Program을 통해 job을 실행시키면 Client Broker는 Globus에 job 실행 명령을 내리고 Globus는 선택된 Server의 Globus를 부른다. Server의 Globus는 GRAM의 Job Manager를 통해 job을 ID에 맞는 node에 submit한다.

4. 결론 및 향후 연구과제

차세대 인터넷을 위한 그리드 환경에서 사용자나 자원의 공유, 사용에 대한 접근 및 권한, 과금 등의 관리를 위해 어카운팅 시스템은 중요한 역할을 한다. 기존의 어카운팅 방법은 관리를 한 곳에서 함으로써 부하가 집중되고, ID의 중복에 의한 복잡성 등의 문제점이 발생한다. 이를 해결하기 위해 정책기반 어카운팅 시스템을 설계하여 각 Server에서 여러 개의 ID를 두어 각 CN과 일대일로 mapping함으로써 해결하였다. 또한 Policy-based MDS Server를 추가함으로써 Client Broker가 MDS Server의 각 슈퍼 컴퓨터의 contact point만을 가지고 자원들과 연결하는 것에 추가로 가능한 Software와 node 수를 추가로 알려주어 Client의 선택을 돕고 Server Broker와의 협상에 도움을 주었다.

그러나 본 논문에서 설계한 모델에서는 Client가 job을 submit할 때 그것의 specification을 넣어야 하고, MDS Browsing을 통해 Client가 직접 Server를 선택해야 한다. 이것은 향후에는 실행파일과 데이터파일을 통해 Client Broker가 job의 specification을 알 수 있어야 하고 그것을 통해 Super MDS Server와 Policy-Based MDS Server에서 Client의 선택없이 적절한 Site를 찾아 Server Broker와 협상하고 실패했을 때도 Client Broker가 다시 Site를 찾아야 할 것이다.

5. 참고 문헌

- [1] I. Foster, C. Kesselman(eds), Q.677, "The Grid : Blueprint for a New Computing Infrastructure" Morgan Kaufmann Publishers, 1998.
- [2] I. Foster, C. Kesselman(eds), S. Tuecke Q.25, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", Intl. J. Supercomputer Applications, 2001.
- [3] I. Foster, C. Kesselman, G. Tsudik, S. Tuecke, " A Security Architecture for Computational Grids", Proc. 5th ACM Conference on Computer and Communications Security Conference, pp. 83-92, 1998.
- [4] K. Czajkowski, I. Foster, S. Fitzgerald, "Grid Information Service", 2001.
- [5] I. Foster, G. von Laszewski, " Usage of LDAP in Globus" .
- [6] K. Czajkowski, I. Foster, et al. "A Resource Management Architecture for Metacomputing Systems", Proc. of the 4th Workshop on Job Scheduling Strategies for Parallel Processing, 1998
- [7] M. Lawenda, N. Meyer, Q.31, " VUS specification", Poznan SuperComputing and Networking Center March. 2001.
- [8] Thomas J. Haker, Brian D. Athey, Q.16, "Account Allocations on the Grid", Center for Parallel Computing University of Michigan. 2000.
- [9] <http://www.gridforum.org>
- [10] <http://www.globus.org>
- [11] <http://www.gridforumkorea.org>
- [12] <http://www.man.poznan.pl>