

그리드 서비스를 위한 사용자 데이터 관리 시스템 설계

Design of User Data Management System for Grid Service

오영주, 김법균, 안동언, 정성중

Young-Ju Oh*, Beob-Kyun Kim, Dong-Un An, Seung-Jong Chung

Abstract - Grid computing enables the fundamental computing shift from a localized resource computing model to a fully-distributed virtual organization with shared resources. In the grid computing environment, grid users usually get access rights by mapping their credential to local account. The mapped local account is temporally belongs to grid user. So, data on the secondary storage, which is produced by grid operation, can increase the load of system administration or can issue grid user's privacy. In this paper, we design a data management system for grid user to cover these problems. This system implements template account mechanism and manages local grid data.

Key Words : Grid, Accounting, Data Management, 'grid-mapfile'

1. 서 론

그리드는 지리학적으로 분산된 컴퓨팅 자원들을 상호 연동하여 조직과 지역에 관계없이 사용할 수 있는 인터넷 환경이라 할 수 있다[1].

그리드 환경의 구축에 있어서 가장 많이 사용되고 있는 미들웨어로 Globus Toolkit을 들 수 있다. Globus Toolkit이 가장 많이 사용되는 이유는 분리될 수 없는 단일 시스템이 아니라 그리드에서 필요로 하는 다양한 서비스들을 독립적인 요소로써 제안하고 있기 때문이다[2].

Globus Toolkit에서는 외부 사용자가 로컬 자원을 사용하기 위해서는 각 자원의 실제 로컬 계정을 얻는 대신 'grid-mapfile'[3]을 이용한다. 그리드 사용자는 자신의 DN(Distinguished Name)과 로컬 계정의 쌍이 'grid-mapfile'에 기록함으로써 로컬 계정이 가지는 권한을 획득한다. 즉, 그리드 사용자는 'grid-mapfile'에 자신의 DN을 등록 또는 삭제함으로써 동적으로 로컬 자원에 대한 사용 권한을 획득 및 상실하게 된다.

이러한 특성으로 인해, 작업의 완료 또는 중단된 경우 로컬 자원에 데이터가 남게 되는데 이렇게 저장된 데이터들은 시스템 관리 측면 및 사용자의 개인정보 보호 측면에서 문제점을 낳게 된다.

따라서 본 논문에서는 이와 같은 문제점을 보완하기 위해 GUDMS(Grid User Data Management System)를 설계 및 구현하였다. 이 시스템을 통해 디스크 공간을 효율적으로 관리할 수 있으며 로컬 어카운트 활용도를 높이고, 결과적으로 사이트 전체의 활용도를 높이면서 그리드 사용자들에게 더욱더 많은 기회를 제공할 수 있는 이점을 제공할 수 있다.

2. 장 관련 연구

2.1 GSAX(Grid Service Accounting Extensions)[4]

Global Grid Forum(GGF)[5]의 RUS-WG에서 제안한 프레임 워크로 OGSA(Open Grid Services Architecture) [6]에 기반을 두어 설계되었다.

어카운팅 기능에 서비스 개념을 도입하여 어카운팅 프레임 워크 내의 각 기능별 요소를 서비스화 하고 각 Virtual Organization(VO)에게 빌링 서비스 등에 관한 자율성을 부여 하였으며 작업이 실행되는 동안 동적으로 타 서비스를 이용하는 경우를 고려하였고 그리드 사용자의 잔고에 따라 진행 중인 작업의 지연 및 중단 여부를 결정할 수 있도록 고안되었다.

2.2 DGAS(DataGrid Accounting System) [7]

EGrid의 프로젝트 중의 하나로 진행 중이며, 시스템 내의 각 구성 요소간의 메시지 전달을 위한 수단으로 XML을 사용하고 있으며 각 구성요소를 DGAS API로 모두 구현하고 있다.

사용자의 그리드 환경 이용을 위한 권한 부여는 사용자의 정보와 잔고를 기준으로 결정하고 있으며, 각 자원을 사용하기 위해서는 사용자 잔고의 일부를 동결하고 잔고가 부족할 경우 진행 중인 작업을 중단 또는 지연시킬 수 있다.

VO내의 모든 시스템들은 동일한 가격 결정권자 (Price Authority)가 제어하도록 설계되어 있어 각 시스템의 자율성이 제한받을 수 있는 여지가 있다.

2.3 PGAM (Policy based Grid Account Manager)

특정 시스템에 국한되지 않고 일반적인 그리드 환경에 적용할 수 있고, 각 사이트의 자율성을 보장하며, 어카운팅과 같은 부가 서비스에도 이용할 수 있도록 설계한 시스템이다.

특히, 그리드 환경 구축 시 가장 많이 사용되는 Globus

Toolkit을 기반으로 설계하였으며, 그리드 접근 제어 시스템의 요구 사항을 반영하도록 설계 및 구현하였다.

그리드 사용자가 사용 권한을 얻기 위해서 서버가 요구하는 사용자와 관련된 정보를 제공하면 서버 측면에서는 이 정보들을 바탕으로 DN과 로컬 아이디를 1:1로 Template하게 정책에 맞춰 바인딩 시켜주는 시스템이다.

2.4 AIService(Accounting Information Service) [8]

그리드 환경의 특성 때문에 어카운팅 정보의 종류 또한 다양하게 존재한다.

여러 종류의 OS 상에서 각각 나타내는 어카운팅 정보의 형식과 실제 값들이 많은 차이를 보임에도 불구하고 Unix/Linux의 어카운팅 정보들은 일부 요소를 제외하고 유사성을 보이는데 공통적으로 일치하는 요소들을 통합하여 어카운팅 정보를 수집하고 수집된 정보를 사용자에게 실시간으로 보여주는 모니터링 툴을 설계하였다.

3. GUDMS 시스템 설계

3.1 DN과 로컬 ID 바인딩 특성

Globus Toolkit에서 로컬의 계정을 얻기 위해서는 'grid-mapfile'을 참조하여 DN과 ID를 동시에 기입해야 작업의 실행 권한을 얻을 수 있다. 기본적으로, 다수의 DN이 하나의 로컬 계정에 대해 결합 가능하며, 각 DN은 신뢰 있는 Certification Authority(CA)로부터 발급받은 인증서로 인증된다.

아래 표 1은 'grid-mapfile'의 내용이다

```
"/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=youngju5" gw01
"/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=kyun" gw01
"/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=iat1" gw03
```

(표 1) 'grid-mapfile'의 예시

다수의 DN이 하나의 로컬 계정에 대해 결합 가능하므로, 다수의 그리드 사용자가 하나의 로컬 계정을 공유하게 된다. 이로 인해 여러 가지 문제점이 발생하게 되는데 그 예는 아래와 같다.

① youngju5 와 kyun이 gw01에 바인딩 되면 관리자는 프로세스의 실제 사용자가 누구인지 판별해줘야 하는데, 이런 구분을 하기 위해서는 시스템 커널을 통한 복잡한 메커니즘이 필요하거나 만약 사용자의 이름이 똑같은 경우에는 처리가 불가능하게 된다.

② 동일한 계정을 다수의 사용자가 공유함으로써 각 사용자의 데이터 보안과 사용자 정보의 보호 부분에서 문제를 야기할 수 있다.

그래서 본 시스템에서는 DN과 로컬 계정을 1:1 바인딩을 전제로 설계하였다. 즉 바인딩 되는 ID는 영구적인 ID가 아니라 Temporary 한 ID이므로 job이 모두 종료되었을 때는 해당 바인딩의 권한을 상실하게 된다.

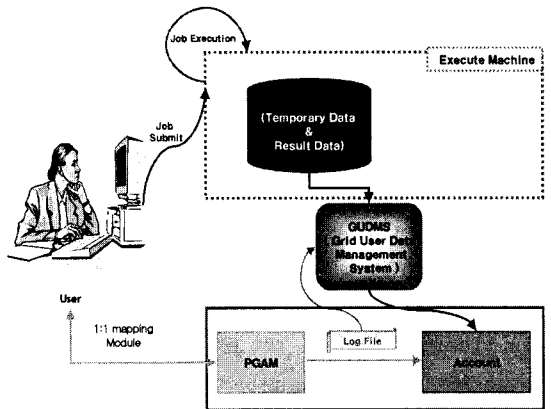
3.2 작업 실행 후 파생된 데이터 관리

바인딩이 되면 사용자는 로컬에서 작업을 실행할 수 있으며, 작업이 실행되는 중간 또는 작업이 종료되는 경우 로컬 디스크에 파일로 남게 되는데 이러한 데이터들을 백업 또는 폐기 처분해 주지 않으면 남아있는 파일들을 다른 사용자가 가져갈 수도 있으며, 여기저기 산재되어 있는 정보들은 디스크 공간 활용도면에서 매우 비효율적이므로 주기적이며 체계적으로 관리 해줘야 할 필요성이 있다

작업이 종료되면 곧바로 가져갈 수 있는 경우는 별로 없다. 이런 경우 바로 폐기시킬 수도 없고 무기한 보존할 수도 없으며 계속 보관하고 있을 시에는 시스템에 디스크 공간 관리상에서 비효율적이게 된다. 그렇다고 부족한 디스크의 용량을 일일이 시스템마다 늘려준다면 시스템 구축 시 비용이 만만치가 않게 되는데 GUDMS 시스템은 적은 비용으로 효율적으로 관리할 수 있다는 장점을 제공한다.

3.3 GUDMS 시스템의 역할

본 시스템은 작업이 실행되고 발생되어진 로그 파일을 참조하여 작업이 종료된 시각을 얻어와 작업이 종료된 시점으로부터 일주일의 시간이 지나면 tar 파일로 묶어서 파일을 압축시켜주는 역할과 압축이 된 시간부터 일정 시간이 지나도록 사용자가 찾아가지 않고 아무런 조치가 없으면 압축된 파일을 완전히 폐기시켜주는 역할을 한다.



(그림 1) 시스템 전체 구성도

위의 그림 1은 시스템 전체 구성도이다. 작업이 실행된 후 발생된 데이터들이 시스템의 로컬 디스크(Temporary Data & Result Data)에 남게 된다.

GUDMS 시스템은 작업이 종료된 종료시점을 로그 파일로부터 얻어와 일정 시간이 지나면 자동으로 Temporary Data & Result Data로부터 파일을 가져와 압축 및 폐기 시키는 시스템이다.

본 시스템은 무엇보다도 리소스 디스크 공간의 활용도를 효율적으로 관리할 수 있으며, 저비용으로 시스템의 성능을 최대화시키고, 로컬 어카운트 활용도를 높이고, 결과적으로 사이트 전체의 활용도를 높이면서 그리드 사용자들에게 더욱

더 많은 기회를 제공할 수 있는 이점을 제공할 수 있다.

4. 시스템 구현

4.1. 어카운팅 시스템 구현

표 2는 어카운팅 시스템 구현 환경을 표로 작성한 것으로 어카운팅 정보의 수집과 서비스를 위해 필요한 정보를 DB로 구축하기 위해 MySQL[9]을 사용하였으며, 다양한 시스템에서도 동작할 수 있도록 Java[10]로 구현하였다

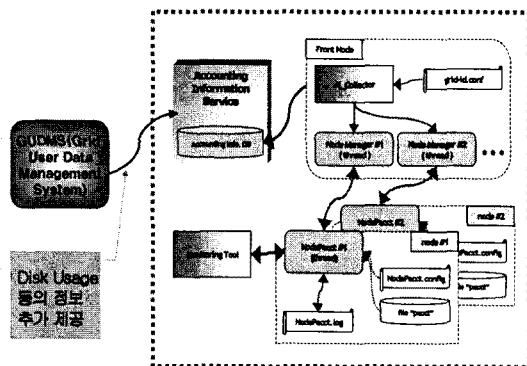
Machine	IBM RS/6000, Linux Cluster
OS	AIX 4.3.2 / Linux 8.0
Environment	Java 1.4.2 / gcc
DataBase	Mysql 4.0.12
Middleware	Globus Toolkit 3.2.1
Job Manager	LoadLeveler 2.1
Job Run /System Test	GridSphere

(표2) 시스템 구현 환경

그리드 환경의 특성 때문에 어카운팅 정보의 종류 또한 다양하게 존재한다. 여러 종류의 OS상에서 각각 나타내는 어카운팅 정보의 형식과 실제 값들이 많은 차이를 보임에도 불구하고 유닉스나 리눅스의 어카운팅 정보들은 일부 요소들을 제외하고 유사성을 보인다.

이미 구현된 그리드 어카운팅 정보 수집 시스템[11]은 GSAX의 Usage Record - Working Group(UR-WG)[12]에서 제안한 필드들을 기반으로 했기 때문에 다른 그리드 환경과 연동이 쉽게 적용 가능하다.

아래 그림 2는 기존에 구현된 어카운팅 정보 수집 시스템의 전체 구조와 본 논문에서 설계한 GUDMS시스템과의 연동관계를 나타낸 그림이다.



(그림 2) 어카운팅 시스템과 GUDMS 시스템

4.2 GUDMS 구현

GUDMS 시스템은 로컬의 디스크에서 일정 시간이 지나면

데이터를 가져와 백업 및 폐기시키는 시스템이므로 가장 중요한 문제는 백업을 하기에 어떤 시스템이 좋은 지부터 먼저 고려해야 한다.

DB는 대용량의 레코드 수일 때 효율적이나 본 시스템에서 발생하는 데이터는 하루에 수십 개 정도로 얼마 되지 않으며, DB에 대용량의 파일을 사용할 경우에는 인덱싱하고, 입력과 삭제하는데 시간이 너무 많이 소요되며, allocation과 de-allocation 하는데 시간이 너무 많이 소요되어 비효율적 이므로 DB보다는 File System을 이용하여 백업을 하는 것이 효율적이라 사료된다.

그러나 File System은 백업 하는데 저장할 수 있는 파일 사이즈가 시스템 별로 조금씩 다르기는 하나 일부 오래된 리눅스 시스템들은 저장되는 용량이 한정되어 있으므로 대량의 데이터는 스크립트를 이용하여 처리해줘야 한다.

5. 결론

본 논문에서 설계 및 구현한 시스템은 GSAX에서 연구된 설계 요소들을 적용한 시스템이며, 실시간으로 발생하는 정보를 수집 즉시 DB에 저장하고 사용자가 원하는 시점에서 바로 어카운팅 정보를 제공할 수 있도록 설계하였다.

본 논문에서 설계한 GUDMS 시스템은 관리자가 일일이 관리하지 않고도 정기적이며 체계적으로 로컬 자원의 리소스 디스크 공간을 효율적으로 사용할 수 있도록 설계하였다.

백업 및 폐기 기간은 리소스 별로 관리자가 권한을 다르게 줄 수 있도록 지정할 수 있도록 설계하였다.

그로인해 각 자원에 남은 사용자들의 데이터를 효율적으로 관리함으로써 디스크 공간 활용도가 높아지고 관리자는 직접 관리하지 않아도 되므로 편리해지는 이점을 얻었으며, 기존의 어카운팅 시스템에 GUDMS 시스템에서 얻은 Disk Usage를 비롯한 부가정보를 전송해줘 보다 풍부한 어카운팅 정보를 제공한다.

참 고 문 헌

- [1] Foster, C. Kesselman(eds), "The Grid :Blueprint for a New Computing Infrastructure" Morgan Kaufmann Publishers, 1998.
- [2] <http://www.globus.org>
- [3] <http://www.globus.org/security/v1.1/grid-mapfile.html>
- [4] A. Beardsmore et al, "GSAX (Grid Service Accounting Extensions)", (draft), GGF6, 2002
- [5] <http://www.gridforum.org/>
- [6] <http://www.globus.org/ogsa/>
- [7] <http://eu-datagrid.web.cern.ch/eu-datagrid/>
- [8] Seung Jong Chung, "Accounting Information Service(AIServ) Administrator and User's Guide", <http://kmi.moredream.org>, KISTI, 2004
- [9] <http://www.mysql.com>
- [10] <http://java.sun.com>
- [11] 장경익, 김법균, 안동언, 정성중, "슈퍼컴 그리드 어카운팅 정보수집 시스템 설계 및 구현", 정보처리학회 .10권 2호, 추계학술 발표 논문집 , pp.1213-1216, November 2003.
- [12] <http://www.psc.edu/~lfm/Grid/UR-WG>