

# 그리드 환경에서의 자원 사용 기록 생성 시스템 설계

함재균\*, 박형우\*\*  
한국과학기술정보연구원\*\*\*  
e-mail : jaehahm@hpcnet.ne.kr

## A Design of Resource Usage Log Creation System in the Computational Grid

Jaegyoon Hahm\*, Hyungwoo Park\*\*  
Korea Institute of Science and Technology Information\*\*\*

### 요 약

그리드 환경에서는 사용자가 컴퓨팅 자원을 사용하기 위해 시스템마다 로그인 할 필요가 없다. 어디서나 멀리 있는 자원으로 작업을 던지기만 하면 되는데, 이때 하나의 자원만을 사용하는 것이 아니라 여러 종류, 여러 개의 자원을 동시에 사용할 수 있다. 이러한 환경에서 각 사용자가 자원을 사용한 기록으로 남는 로그(log)처리는 기존의 계산 환경에서의 그것과는 많은 차이가 생기게 된다. 우선 사용자는 자신의 작업에 대한 로그를 보기 위해서 시스템마다 로그인 할 필요가 없어야 한다. 또 작업에 대한 효율적인 모니터링을 위해 정확한 로그를 필요로 하게 된다. 자원 관리자는 정확한 로그를 이용해 자원 관리 및 어카운팅(accounting) 정책을 수립할 수 있을 것이다. 본 논문에서는 이러한 그리드 환경에 적합한 로그 로그 생성 시스템을 설계하고자 한다.

### 1. 서론

계산 그리드(Computational Grid)[1][2]는 지리적으로 분산된 환경에서 각각 독립된 계산 자원들을 묶어, 신뢰성 있고, 사용이 용이하며, 저렴한 계산 환경을 제공하는 기반 구조라 할 수 있다. 그리드의 사용자는 그리드에 묶이는 자원 각각의 물리적인 위치를 상관하지 않을 뿐 아니라 자원의 종류나 운영체제, 그리고 자원에 대한 정책의 다름에 제한 받지 않는다. 사용자는 전세계에 흩어져 있는 컴퓨터들을 마치 자신의 데스크탑 컴퓨터처럼 사용하는 것이 가능케 되는 것이다.

이러한 그리드 환경에서의 계산 방식의 특징이라면, 사용자가 계산작업을 수행함에 있어서 단일 자원에 갇혀 있던 기존의 계산 방식의 한계를 벗어나 네트워크 상에 흩어져 있는 여러 자원들을 동시에 계산에 이용할 수 있도록 한다는 것이다. 기존에는 사용자가 계산을 하기 위해 먼저 원격의 기계에 로그인을 하고 작업을 수행한 다음 그 결과를 모으는 방식을 취했다. 그러나 그리드 환경에서 사용자는 계산을 위해 계정

을 얻어 로그인할 필요가 없이, 위치에 상관 없이 그리드에 연결된 여러 자원들을 한꺼번에 계산에 사용하고, 결과를 확인할 수 있다.

마찬가지로 작업의 수행 과정에서 생성되는 로그에 있어서도, 그리드 이전의 방식에서는 사용자가 작업한 자원 내에 남는 로그를 직접 들어가서 확인해야 했으나, 그리드 환경에서는 앞에서 언급한 계산방식의 특징으로 인해 기존과는 다른 새로운 로그 시스템을 필요로 하게 된다.

따라서 본 연구에서는 그리드 환경에 맞는 로그 생성 시스템을 제안하고자 한다. 2 장에서는 기존의 로그 생성 메커니즘의 한계와 새로운 로그 시스템의 필요성을 설명하고, 3 장에서는 관련된 연구를 소개한다. 4 장에서는 그리드 로그 생성 시스템의 정의와 요구 사항들을 살펴보고, 5 장에서는 새로운 로그 생성 시스템을 설계하여 제시할 것이다. 그리고 6 장에서 향후 관련된 연구 진행 사항과 결론을 맺으며 마칠 것이다.

## 2. 그리드 환경에 따른 기존 로그 시스템의 문제점

그리드를 통해 사용자가 작업을 수행할 때 계산이 이루어지는 데에는 여러 종류의 자원을 거치게 된다. 먼저 작업은 물리적인 네트워크를 통해 전달이 되고, 원격의 계산 자원에서 인증 과정을 거치게 되며, 그 후에 그 자원에 할당이 되어 CPU와 메모리 등을 사용하며 계산이 이루어진다. 계산이 끝난 작업의 결과는 다시 네트워크를 통해 처음 작업을 지시한 사용자에게로 돌아오게 된다. 단지 하나의 계산 자원만을 사용하는 단순한 작업에 대해서도 이렇게 여러 과정을 거치며 각각의 자원에 사용기록을 남기게 된다. 그러나 그리드에서는 하나의 자원만을 사용하는 경우는 거의 없으며 계산을 위해 수없이 많은 동종의 또는 이기종의 자원들을 거치며 계산이 이루어진다. 이렇게 많은 자원들에 사용자의 작업에 대한 로그가 남게 될 것인데, 사용자가 각각의 자원에 일일이 접속하여 로그를 확인한다는 것은 너무 비효율적일 뿐만 아니라, 자신의 작업에 대한 추적을 정확하게 해내기란 거의 불가능하다고 할 수 있다.

그리드 미들웨어의 하나인 글로버스(globus)[3]를 이용한 그리드 작업 수행의 예를 들어보자. 원격의 사용자가 어느 컴퓨터의 글로버스를 통해 작업을 수행시키고자 할 경우, 사용자의 작업 요청은 실행파일과 함께 사용하려는 자원에 할당이 된다. 작업의 수행 과정에서 먼저 글로버스는 사용자에게 대한 인증을 위해 게이트키퍼(gatekeeper)를 통과하게 되고, 게이트키퍼가 잡 매니저(job manager)를 실행시켜 작업이 수행되도록 한다. 잡 매니저는 지정된 작업수행 방식에 따라 컴퓨터에서 실행파일이 수행되도록 한다. 여기서 각각의 과정은 자신만의 로그 파일을 생성하게 된다. 게이트키퍼는 \$GLOBUS\_LOCATION/var/globus-gatekeeper.log 라는 로그 파일에 해당 세션의 요구 시간, 요청자의 글로버스 아이디(ID), 매핑되는 내부 사용자 계정, 요구된 서비스, 실행한 잡 매니저의 프로세스 넘버 등을 기록하게 된다. 잡 매니저는 상당히 소상한 로그를 남기는데, 작업을 수행하는 내부 사용자 계정의 홈 디렉토리에 gram\_job\_mgr\_xxxx.log 라는 이름의 로그 파일을 생성한다<그림 1>. 이 로그 파일에는 잡 매니저가 시작해서 작업이 마칠 때까지의 모든 기록-작업 수행 환경, 시간, 로컬 스케줄러를 사용할 경우 스크립트 생성 과정, 작업 제출(submission)결과 등-을 담고 있다. 그리고, 잡 매니저가 작업을 수행하기 위한 수단으로써 LSF 같은 스케줄러를 이용했을 경우, 해당 스케줄러를 이용한 기록을 스케줄러로부터 얻을 수 있다. 이렇게 각각의 과정에서 로그가 남게 되지만 자원이 분산되어 있듯이 자원 사용에 대한 로그 역시 분산되어 있다. 사용자의 입장에서 자신의 작업의 수행 경로를 한눈에 파악할 수 있도록 하는 서비스는 현재 되어 있지 않다. 결국 그리드 환경에서 분산된 자원을 이용하는 작업의 기록을 사용자에게 한눈에 보여 줄 수 있는 그리드에 적합한 새로운 로그 시스템이 필요하다.

```

9/12 16:07:00 -----
9/12 16:07:00 JM: Entering gram_job_manager main().
9/12 16:07:00 JM: HOME = /home/jaegyoon
9/12 16:07:00 JM: LOGNAME = jaegyoon
9/12 16:07:00 JM: GLOBUS_ID =
/O=Grid/O=Globus/OU=hpcnet.ne.kr/CN=Jae Hahm
.....
9/12 16:07:05 JMI: testing job manager scripts for type lsf
exist and permissions are ok.
9/12 16:07:05 JMI: job manager type is lsf.
.....
JM_SCRIPT: starting to build LSF job script
JM_SCRIPT: checking environment
JM_SCRIPT: LSF job script successfully built
JM_SCRIPT: submitting LSF job script
JM_SCRIPT: job submitted successfully!
.....
    
```

그림 1 잡 매니저 로그 파일

## 3. 관련 연구

그리드 환경에서의 로그 시스템에 대한 연구는 그 필요성은 많은 공감대를 형성하고 있으나 실제 연구는 그리 많지 않고 진행되고 있지 않다. 관련 연구동향을 NASA의 Grid Monitoring Infrastructure와 NWS(Network Weather Service)를 중심으로 정리하면 다음과 같다.

### 3.1 Grid Monitoring Infrastructure

미국의 NASA에서 수행하고 있는 IPG(Information Power Grid)에서는 자원 및 작업을 모니터링하고 관리하기 위해서 새로운 작업관리 구조를 설계하였다[4]. 이 구조에서는 센서(Sensors), 액추에이터(Actuators), 이벤트 서비스(Event service)라는 세개의 기본 컴포넌트를 이용해서 사용자가 모니터 등을 만들 수 있도록 하였다. 센서는 각 자원에 대한 정보를 파악하는 일을 하고, 액추에이터는 사용자가 자원을 관리하기 위해 특정한 명령을 하기 위한 것이다. 그리고 이벤트 서비스는 센서에서 생성된 정보들을 외부에 제공하는 역할을 한다.

### 3.2 NWS(Network Weather Service)

미국의 NMI 프로젝트에서 제안하는 그리드 미들웨어 패키지에 포함된 자원 모니터링 서비스이다[5]. NWS는 그리드 상의 자원들의 상태를 파악하고 예측할 수 있도록 하는 서비스이다. NWS는 크게 네가지 컴포넌트로 구성되어 있는데, 네임 서버(Name server)는 매우 낮은 레벨에서 프로세스와 데이터 이름을 매핑시켜 주는 디렉토리 서버이고, 메모리 서버(Memory server)는 각 컴포넌트로부터 얻어진 정보를 저장하는 서버이다. 또 센서(Sensor)는 자원으로 부터 성능 측정치를 모으는 역할을 하고, 포어캐스터(Forecaster)는 특정한 시간동안의 자원의 성능에 대한 예측값을 제공한다. NWS에서는 자원의 상태를 주기적으로 측정하여 서버에 저장하고 그 정보에 근거해서 질의에 응답

하거나, 예측 서비스를 제공한다.

#### 4. 그리드 로그(Grid Log) 생성시스템 요구사항

그리드 환경에서 자원 및 작업에 대한 모니터링(monitoring)과 관리(management)의 문제는 새로운 연구 주제이다. 따라서 자원에 대한 어카운팅(accounting) W/G(Working Group) 중심으로 그리드 로그에 관한 연구가 GGF(Global Grid Forum) 등에서 이루어지고 있다. 본 장에서는 그리드 로그의 정의와 요구사항 등을 설명한다.

##### 4.1 정의

그리드의 가장 큰 특징이라고 한다면 '분산'이라는 키워드일 것이다. 그리드 환경에서는 분산된 서로 다른 자원들을 이용해 작업이 수행된다. 작업에 대한 로그 역시 분산된 각 자원들에 기록이 된다. 이 분산되어 있는 기록을 어떻게 통합하여, 얼마나 정확하게 사용자에게 제공하느냐가 그리드에 적합한 로그의 문제영역이 될 것이다. 이는 사용자의 관점에서 작업에 대한 로그의 정의라고 할 수 있다.

또 다른 측면에서, 각 자원의 관리자는 자원을 사용한 이에 대해 비용을 요구할 수 있는 근거가 있어야 한다. 이는 각 자원에서 생성하는 로그에 기반하는데, 이러한 측면에서의 로그는 기존 환경에서와 별반 다를 바가 없다. 단지 그리드 환경에서 요구되는 것이 있다면, 사용자에게 대한 명확한 신원 확인과 사용자가 가진 자원 사용 기록과의 일관성이다.

결국 그리드 로그는 그리드 자원을 사용함에 있어서 계산에 참여하는 모든 자원에 대한 사용기록의 전체적이고 일관적인 사용 기록이라 할 수 있다.

##### 4.2 그리드 로그에 필요한 어트리뷰트(attributes)

GGF의 Resource Usage WG에서는 OGSA(Open Grid Service Architecture)의 관점에서 자원 사용 기록 서비스를 정의하고 이에 필요한 자원 정의(Resource Definition)을 만들어 가고 있다. 여기서 제안된 자원의 정의와 현재 글로벌스 기반의 그리드 환경을 고려하여 볼 때 다음과 같은 어트리뷰트들이 그리드 로그에 포함되어야 할 것이다.

- GRID\_ID : 그리드에서 유일하게 사용자를 인식하게 하는 이름
- LOCAL\_ID : 자원 내부에서 사용되는 계정의 이름
- CLIENT\_CONTACT\_STR : 작업을 요구한 기계에 대한 접근 포인트
- JOB\_ID : 사용자가 요구한 작업의 이름
- SUBJOB\_ID : 여러 자원에서 실행되어질 때 세부 작업의 이름
- RESOURCE\_ID : 각 자원의 이름(IP 주소 또는 도메인 네임)
- USED\_LOG\_CONTACT\_STR : 로그에 대한 쿼리를 할수 있는 접근 포인트

- NODES : 클러스터인 경우 자원이 실행된 계산노드 이름
- TIME : 작업이 거처간 시간(시작, 끝 등)
- EXE\_FILE : 사용자 작업의 실행파일 이름
- USED\_QUEUE : 자원에서 사용된 큐의 이름
- NUM\_CPU : 사용된 cpu 개수

위에서 언급한 어트리뷰트 들은 매우 기본적인 것들이며, 실제로는 더욱 자세한 항목들이 기재된다.

#### 5. 그리드 로그 시스템 설계

본 절에서는 사용자가 두가지 방식으로 로그를 얻을 수 있는 시스템을 설계하였다. 한가지는 사용자가 작업을 원격으로 수행하게 되면, 작업이 거치는 각각의 자원마다 그 수행 과정을 사용자에게로 보내주는 것이다. 이렇게 함으로써 사용자는 자신의 작업에 대한 종합된 로그를 자신의 기계에서 확인할 수가 있다. 또 하나의 방법은 사용자가 작업의 ID 또는 사용자 자신의 ID 를 통해 어떤 자원으로 질의(query)를 보내어 그 자원에서만 로그를 알아낼 수가 있다.

<그림 2>에서 이 시스템의 개요를 설명하고 있다. 사용자가 던진 작업에 대해서, 각 자원들은 사용자의 작업이 자신을 거쳐간 기록을 사용자에게 보낸다. 이 기록들은 사용자 기계의 로그 파일에 차곡차곡 쌓이게 되고, 사용자는 로그 파일을 통해 네트워크 이동에

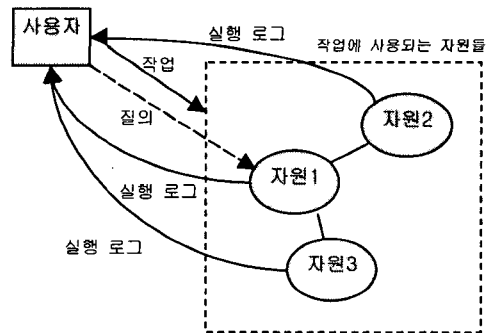


그림 2 그리드 로그 개념

소요된 시간, 각 자원에 작업이 제출된 시간, 실제로 실행된 시간, 또 에러가 발생했을 경우 위치와 원인 등을 파악할 수가 있다. 이러한 시간의 흐름에 따르는 로그 파일을 통해서 비정상적으로 시간이 오래 걸리는 부분을 알아낼 수도 있다.

그리고 사용자는 필요에 따라 자신의 작업에 참여한 자원에 질의를 보내어 해당 자원에 기록된 좀더 자세한 내용의 로그를 받을 수 있다. 각 자원에 있는 로그 매니저(log manager)는 질의가 올 때 해당 작업에 대한 로그를 검색해서 답변해주는 역할을 하며, 또 자신이 관리하는 자원에서 수행되는 작업들에 대한 로그 파일을 수집하는 역할을 한다.

<그림 3>은 로그 매니저가 자원 내에서 차지하는 위치와 역할을 설명하고 있다. 이 구조는 현재 글로버스를 기반으로 설계되어 있다. 글로버스에서는 사용자가 던진 작업이 게이트키퍼를 통해 인증을 받고 게이트키퍼는 인증을 통과한 작업에 대해서 잡 매니저(job manager)를 생성한다. 잡 매니저는 작업을 수행하기 위해 바로 프로세스를 생성시키거나(fork), 또는 시스템에 설치되어 있는 LSF 나 PBS 같은 지역 스케줄러(local scheduler)를 통해 작업을 실행하게 된다. 로그 매니저는 작업이 수행될 때 잡 매니저와 지역 스케줄러로부터 해당 작업에 대한 로그를 수집하여 저장한다. 이 로그 매니저는 해당 작업의 로그를 사용자에게 다시 보내주는 일을 하며, 추후에 사용자의 질의에 대해 응답하는 역할을 한다.

[3] Foster I. Kesselman C., "Globus: A Metacomputing Infrastructure Toolkit", Intl J. Supercomputer Applications, 11(2):115-128, 1997  
 [4] Waheed A. Smith W. George J. Yan J. "An Infrastructure for Monitoring and Management in Computational Grids", Proceedings of the 5<sup>th</sup> Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers. March, 2000  
 [5] Wolski R. Spring N. Hayes J. "The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing", Journal of Future Generation Computing Systems, Vol 15, No 5-6, 757-768, October 1999

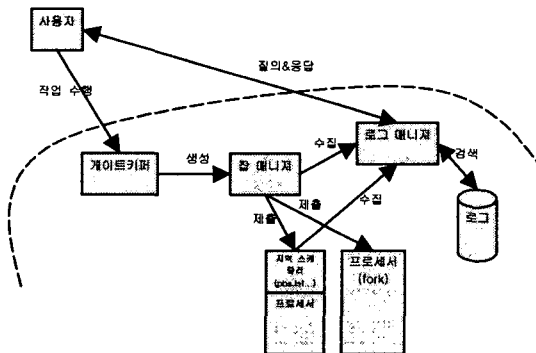


그림 3 자원 내부의 세부 구조

## 6. 결론 및 향후 계획

설계된 그리드 로그 시스템은 사용자가 원격에 접속하지 않아도 자신의 작업에 대한 로그를 쉽게 확인할 수 있도록 하여, 작업의 성능과 병목(bottleneck) 부분도 확인할 수 있게 된다. 또한 이 로그를 이용하여 자원에 대한 어카운팅 시스템을 수월히 설계할 수 있을 것이라 기대하며, 작업에 대한 모니터링이 편해지리라 생각한다.

본 로그 시스템은 글로버스를 미들웨어로 사용하는 시스템에 우선적으로 적용될 것이다. 이를 위해서 글로버스의 GRAM(Grid Resource Allocation Manager)에 대해서 수정을 가해야 할 것이다. 추후에 N\*Grid 미들웨어에 포함되어 모니터링 및 작업관리를 위해 쓰이게 될 것이다. 사용자 기계와 자원의 로그 매니저 간의 로그 형식에 대한 통일과 필요로 하는 어트리뷰트에 대한 정제 과정등이 필요로 할 것이다.

## 참고문헌

[1] Foster I. Kesselman C. "The Grid: Blueprint for a New Computing Infrastructure", Morgan Kaufmann Publishers, Inc., 1999  
 [2] Foster I. Kesselmann C. Tuecke S. "The Anatomy of the Grid", Intl J. Supercomputer Application, 15(3), 2001