

# SWING: Globus를 위한 웹 인터페이스의 설계 및 구현

오영은<sup>o</sup> 김진석

서울시립대학교 컴퓨터통계학과

(iceize, jskim)@venus.uos.ac.kr

## Design and Implementation of SWING: Simple Web Interface for Network of Globus

Young-Eun Oh<sup>o</sup> and Jin Suk Kim

Dept. of Computer Science and Statistics, University of Seoul

### 요약

여러 과학 기술들이 발전함에 따라 여러 지역에 산재해 있는 고성능의 컴퓨팅 자원을 한 데 묶어 그 자원을 이용할 수 있게 해 주는 그리드 기술에 대한 많은 연구가 수행되고 있다. 그리드 환경을 사용자가 쉽게 이용하기 위해서는 작업의 제출 및 진행 상황, 자원 상태에 대한 모니터링 기능을 제공해야 한다. 본 논문에서는 사용자가 그리드 자원을 편리하게 이용할 수 있도록 Globus 툴킷 [1]을 기반의 웹 인터페이스를 제안하고 설계하였으며, 사용자 인증 방법과 작업 제출 및 자원 상태의 모니터링 구조에 대하여 기술한다.

### 1. 서론

과학 기술이 발전함에 따라 대량의 정보를 분석하고 처리하기 위한 여러 가지 방법이 개발되어 왔다. 이러한 방법 중 그리드 (GRID)는 여러 지역에 산재되어 있는 고성능의 컴퓨팅 자원들을 격자 형태로 묶어 어느 곳에서도 그 자원들을 이용할 수 있도록 해주는 서비스이다 [2]. 그러나 그리드 환경에 속한 자원을 이용하기 위해서는 사용자가 사용법을 익히기 위한 많은 노력을 기울여야 하며, 자원들의 상태를 직접 모니터링 하여 작업을 수행하여야 한다.

본 논문에서는 사용자가 편리하게 그리드 자원을 이용할 수 있는 인터페이스를 제안하고 구현하였다. 그리하여 작업의 제출 및 진행 상황에 대한 모니터링 및 자원들의 상태를 쉽게 파악할 수 있도록 하였다.

### 2. 관련 연구

#### 2.1. Grid Portal

그리드 포털은 사용자 작업을 수행하기 위하여 그리드 서비스 및 자원을 활용하는 도구를 제공하는 것뿐만 아니라, 그리드 서비스에 대한 정보를 안전하게 수집할 수 있도록 하는 웹 응용 서버의 한 종류이다 [3]. 이러한 웹 응용 서버의 대표적인 예로는 GSDK (Grid Portal Development Kit) [4]와 GridPort 툴킷 [5]이 있다. Grid 포털을 구성할 때에는 다음과 같은 사항을 고려해야 한다[5].

- 접근 (Access): 새로운 소프트웨어나 플러그인의 다운로드 없이 기존의 웹 브라우저를 이용하여 접근할 수 있어야 한다. 또한 XML과 같은 최신 기술을 지원하지 못하는 웹 브라우저를 위하여 기본적인 HTML로 구현해야 한다.

- 일반적인 그리드 기술과 표준의 사용: 기존에 구성되어 있는 고성능의 자원 구조를 변경시키지 않고 그대로 이용할 수 있어야 한다.
- 보안: SSL을 통한 데이터의 암호화와 HTTPS 프로토콜을 지원하여야 한다.
- Single Sign-on: 그리드 자원들 간의 쉬운 접근을 위하여 GSI (Grid Security Infrastructure) [6]를 통한 Single Sign-on 기능을 제공하여야 한다.

#### 2.2. GSDK

GSDK는 그리드 포털을 구성할 수 있도록 하는 개발 툴킷으로서 다음과 같은 구조로 이루어져 있다.

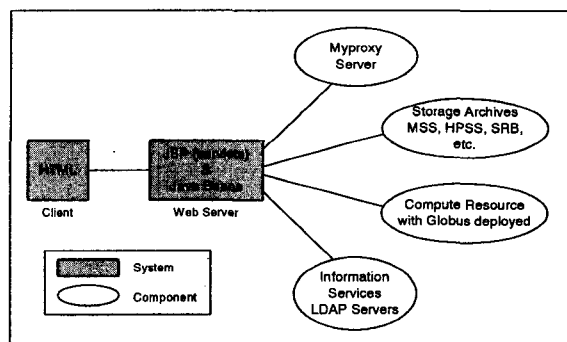


그림 1. GSDK의 구조

<그림 1>에서 사용자는 웹 브라우저를 통해 웹 서버에 안전하게 접근하게 된다. 웹 서버에서는 JSP와 Java Beans를 이용하여 MyProxy [7]를 통한 사용자 인증과 GSIFTP [8]를 통한

파일 전송, LDAP (Lightweight Directory Access Protocol)를 통한 자원에 대한 정보 발견을 하게 되며, 이와 같은 과정을 통하여 작업을 수행할 수 있게 된다. 또한 모든 컴포넌트를 CoG API [9]를 통하여 구현함으로써 포탈 상의 새로운 기능을 구현하려는 경우 확장성이 뛰어나다.

2.3. GridPort 툴킷

GridPort 툴킷은 계산 그리드 (Computational GRID)에서 제공하는 사용자 포탈, 응용 프로그램 인터페이스, 교육 포탈 등의 개발을 도와주는 기술들의 모음이다. GridPort 툴킷은 서버 상의 Perl CGI, 클라이언트 상의 HTML/Javascript를 이용하여 어떤 웹 브라우저에서도 쉽게 볼 수 있도록 구현하였다. <그림 2> GridPort 툴킷의 구조를 보여주고 있다 [5].

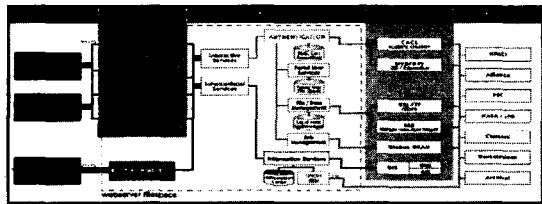


그림 2. Gridport 툴킷의 구조

3. SWING의 설계 및 구현

본 장에서는 새롭게 설계 구현한 그리드 포탈인 SWING (Simple Web Interface for Network of Interface)에 대하여 알아본다. SWING은 Globus 툴킷을 사용자가 쉽게 이용할 수 있도록 편리한 웹 인터페이스를 제공한다. 이러한 기능을 수행하기 위하여 다음과 같은 사항을 고려하였다.

- Globus 툴킷과의 호환성: 사용자 인증, 자원의 상태 검색, 작업의 제출 및 상황 모니터링의 기능을 제공하기 위하여 Globus 툴킷의 명령어를 사용하였다.
- 그리드 서비스의 제공: GSIFTP, GASS (Global Access to Secondary Storage) [10]를 통한 파일 전송 및 PBS와 같은 지역 스케줄러를 지원하며, MPICH-G2 응용 프로그램을 실행할 수 있도록 하였다.
- 자원 상태에 대한 실시간 검색: 기존의 모니터링 응용 프로그램에서 제공하였던 그래프 형태로 자원의 상태를 쉽게 확인할 수 있게 하였다.

3.1. 구조

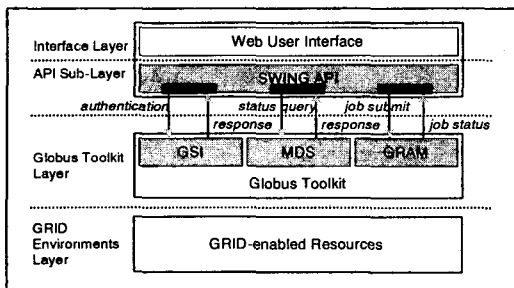


그림 3. SWING과 Globus의 연동 구조

<그림 3>은 SWING과 Globus의 연동 구조를 보여주고 있다. SWING API는 크게 GSI, MDS, GRAM 모듈로 구성되어 있다. GSI를 통하여 사용자 인증을 처리하며, MDS를 통하여 자원의 상태를 검색하고, GRAM을 통하여 작업을 제출하거나 작업의 상황을 모니터링할 수 있다. 이러한 작업의 결과는 HTTPS/SSL을 통하여 클라이언트에게 보내어진다.

3.2. 사용자 인증

SWING에서의 사용자 인증은 MyProxy를 통하여 이루어진다. MyProxy를 통하여 사용자 정보를 서버에 저장하게 되며, 이 정보는 사용자 Proxy 파일을 생성하는 데 사용된다.

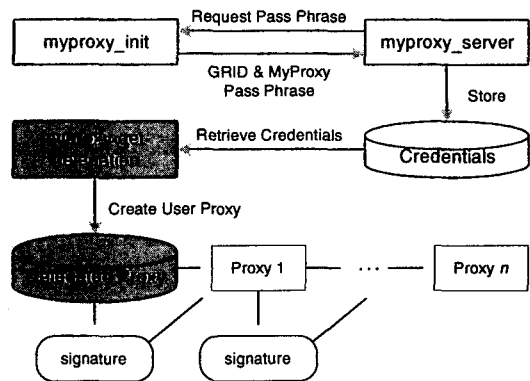


그림 4. SWING에서의 사용자 인증 과정

<그림 4>는 이러한 과정을 보여주고 있으며, 회색 바탕이 SWING에서 제안하는 사용자 인증 처리 부분이다. SWING에서는 MyProxy의 myproxy-get-delegation 명령을 이용하여 Credential로부터 사용자 Proxy를 생성하며, 이렇게 생성된 Proxy는 Single Sign-on을 통하여 그리드 자원을 사용할 수 있게 된다.

3.3. 작업의 제출 및 모니터링

본 논문에서는 그리드 환경에서 수행할 수 있는 작업을 다음과 같은 기준에 따라 분류하였다.

- 사용자 모드: 기본 사용자 모드, RSL 모드
- 작업의 실행 방법: Interactive, Background
- 작업 관리자: fork, PBS, LSF 등
- 실행 파일의 형태: 자원에 이미 있는 경우, 사용자가 업로드한 파일
- 작업의 형태: 단일 작업, MPI 작업

이 중 기본 사용자 모드는 사용자가 쉽게 그리드 자원을 이용할 수 있도록 해주는 모드이며, RSL 모드는 Globus 툴킷에서 자원의 요구 사항을 명시해주는 RSL (Resource Specification Language) [11]을 직접 이용할 수 있도록 해주는 모드이다. 사용자가 작업을 제출하면 여러 기준에 따라 RSL을 생성한 후 이것을 이용하여 작업을 실행시키게 된다. 이렇게 실행된 작업의 정보는 파일에 기록되며, 작업의 상태를 Globus 툴킷에서 제공하는 명령어를 이용하여 업데이트된다. 작업의 상태로는 PENDING, ACTIVE, DONE 등이 있는데, 만약 작업의 상태가 DONE이라면 작업의 결과를 파일에 기록하게 된다. <그림 5>는 작업을 제출할 때의 데이터 흐름을 나타낸다.

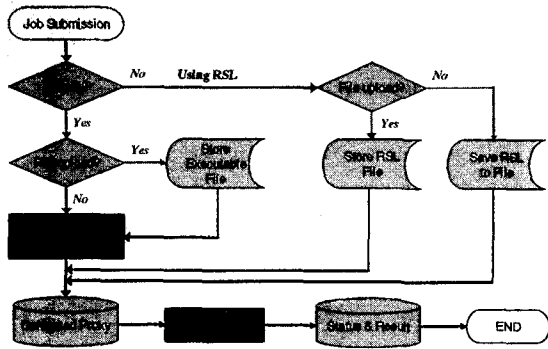


그림 5. 작업 제출에 관련된 데이터 흐름도

### 3.4. 자원 상태의 검색

그리드 환경에서는 여러 자원을 이용할 수 있으므로, 그 자원들에 대한 상태를 모니터링하는 것이 매우 중요하다. 본 논문에서는 자원 상태를 검색하기 위해서 LDAP 라이브러리를 이용하였으며, 데이터를 검색하기 위한 Base DN (Distinguish Name) 값으로 "Mds-Vo-name=local, o=Grid"를 지정하였다. 그리고 검색 필터로는 "objectClass=\*"를 이용하여 정보를 수집하였다. SWING에서는 다음과 같은 자원의 정보를 가져올 수 있으며, 이 중 CPU, 메모리, 파일 시스템의 사용량을 그래프로 보여주었다.

- CPU: 제조사, 모델, 속도, 1, 5, 15분간의 부하량
- 메모리: 전체 용량, 할당된 용량, 사용 가능한 용량
- 스왑: 전체 용량, 할당된 용량, 사용 가능한 용량
- 파일시스템: 전체 용량, 할당된 용량, 사용 가능한 용량
- 운영체제: 이름, 릴리즈 (버전)
- 스케줄러: GRAM에 할당되는 URL 및 스케줄러의 종류

### 3.5. GPDK와 SWING의 비교

	GPDK	SWING
Pre-required Software	Apache, Tomcat CoG Kit	Apache, PHP
API	CoG Java API	Globus Toolkit Command
GSI Module	Use CoG Kit	MyProxy Store Proxy to Session
MDS Module	"	Real-time resource status MDS Browser
GRAM Module	"	GASS support
etc.	Queue Information	File upload support MPICH-G2 support

표 1. GPDK와 SWING과의 비교

### 3.6. 구현 및 테스트 환경

SWING을 구현하기 위하여 운영체제와 웹서버는 각각 레드햇 리눅스 7.2와 아파치 (Apache 1.3.26)를 사용하였다. 그리고 HTTPS 프로토콜을 이용하기 위해 mod\_ssl을 사용하였으며, 언어는 PHP를 사용하였다. 그리고 그리드 자원으로는 2개의 클러스터와 3대의 PC를 이용하였다.

### 4. 결론

본 논문에서는 사용자가 그리드 자원을 쉽게 이용할 수 있게 하는 그리드 포털에 대하여 알아보았으며, 웹 인터페이스를 설계 및 구현하였다. 그러나 본 논문에서 제안한 SWING은 Globus 툴킷에 의존성을 가지고 있으므로, 이는 차후에 개선되어야 할 것이라고 판단된다. 향후 연구 방향으로서는 단순한 웹 인터페이스로서의 역할이 아닌, 글로벌 큐잉 시스템의 포털로서의 역할을 수행하는 것을 목표로 두고 있다.

### 참고문헌

- [1] I. Foster, C. Kesselman and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," *Journal of the International Supercomputer Applications*, 2001.
- [2] Globus Toolkit, <http://www.globus.org>.
- [3] Grid Portal, [http://www.cactuscode.org/Presentations/GridPortals\\_2000.ppt](http://www.cactuscode.org/Presentations/GridPortals_2000.ppt).
- [4] Grid Portal Development Kit, <http://www-itg.lbl.gov/grid/projects/GPDK/index.html>.
- [5] M. Thomas, S. Mock, J. Boisseau, M. Dahan, K. Mueller, and D. Sutton, "The GridPort Toolkit Architecture for Building Grid Portals," *Proc. of the 10th IEEE Symp. on High Performance Distributed Computing*, 2001.
- [6] I. Foster, C. Kesselman, G. Tsudik, and S. Tuecke, "A Security Architecture for Computational Grids," *Proc. of the 5th ACM Conference on Computer and Communications Security Conference*, 1998.
- [7] J. Novotny, S. Tuecke, and V. Welch, "An Online Credential Repository for the Grid: MyProxy," *Proc. of the 10th IEEE Symp. on High Performance Distributed Computing*, 2001.
- [8] B. Allcock, J. Bester, J. Bresnahan, A. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, and S. Tuecke, "Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing," *Proc. of the IEEE Mass Storage Conference*, 2001.
- [9] G. von Laszewski, I. Foster, J. Gawor, W. Smith, and S. Tuecke, "CoG Kits: A Bridge between Commodity Distributed Computing and High-Performance Grids," *Proc. of the ACM Java Grande Conference*, 2000.
- [10] J. Bester, I. Foster, C. Kesselman, J. Tedesco, and S. Tuecke, "GASS: A Data Movement and Access Service for Wide Area Computing Systems," *Proc. of the 6th Workshop on I/O in Parallel and Distributed Systems*, 1999.
- [11] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke, "A Resource Management Architecture for Metacomputing Systems," *Proc. of the IPPS/SPDP Workshop on Job scheduling Strategies for Parallel Processing*, 1998.
- [12] 오영은, 조정우, 김용순, 조호진, 이정훈, 김진석, "WIG: Web Interface for Globus," Technical Report, 서울시립대학교 컴퓨터통계학과, 2002.