

G-03-SD-01-23R-2

## 연구보고서

# 클러스터 자원 제공을 통한 컴퓨팅 그리드 구축 (Construction of Cluster Computing Grid)

2003. 12. 15.

연구기관 : 서울그리드센터

### 주 의

1. 이 보고서는 한국과학기술정보연구원에서 위탁연구과제로 시행한 연구보고서입니다.
2. 이 보고서 내용을 발표할 때에는 반드시 한국과학기술정보연구원에서 시행한 위탁연구과제의 연구결과임을 밝혀야 합니다.

클러스터 자원제공을 통한 컴퓨팅 그리드 구축

2003

한국과학기술정보연구원

연구보고서

클러스터 자원 제공을 통한 컴퓨팅 그리드 구축  
(Construction of Cluster Computing Grid)

2003. 12. 15.

연구기관 : 서울그리드센터

한국과학기술정보연구원

# 제 출 문

한국과학기술정보연구원장 귀하

본 보고서를 “클러스터 자원 제공을 통한 컴퓨팅 그리드 구축”에 대한 연구의 최종보고서로 제출합니다.

2003년 12월 15일

수탁기관 : 서울시립대정보기술연구소

수탁기관장 : 김성환 (인)

연구책임자 : 이용우 (서울그리드센터)

공동연구책임자 :

참여연구원 : 민현수 (서울그리드센터)

유병덕 (서울그리드센터)

김규성 (서울그리드센터)

박순필 (서울그리드센터)

김영호 (서울그리드센터)

윤홍수 (서울그리드센터)

# 최종연구보고

1. 위탁과제명 : 클러스터 자원 제공을 통한 컴퓨팅 그리드 구축
2. 수탁기관 : 서울시립대정보기술연구소
3. 연구책임자 : 이용우
4. 일정계획 대 진도
5. 주요 연구개발 추진 내용 및 최종결과
6. 추진상의 문제점 및 앞으로의 추진 대책
7. 주요 연구 기자재 및 시설 사용 계획

# 요 약 문

## I. 제 목

클러스터 자원 제공을 통한 컴퓨팅 그리드 구축

## II. 연구개발의 목적 및 중요성

### 1. 연구의 목적

서울그리드센터의 그리드 컴퓨팅 테스트베드의 구축을 위한 노력의 일환으로 클러스터 그리드 컴퓨팅 테스트베드를 구축하였다. 구축된 클러스터 그리드 컴퓨팅 테스트베드를 KISTI의 국가 그리드 테스트베드에 연결하여 클러스터 자원을 제공함으로써 국가 컴퓨팅 그리드 테스트베드 구축에 참여하는데 목적이 있다.

### 2. 연구의 중요성

하드웨어의 기하급수적인 발전에 힘입어 값싼 고성능 PC의 대중화가 이루어졌다. 이런 고성능 PC를 기반으로, 상용 유닉스의 성능에 버금가는 리눅스를 기본 운영체제로 채택하는, 가격대비 성능비가 뛰어난 리눅스 클러스터링 시스템이 개발되고 있다. 이 클러스터링 시스템은 어느 나라를 막론하고 경쟁적으로 구축, 튜닝(최적화), 적용되고 있는 실정이다. 또한 이러한 시스템은 특정기관에서 보유한 슈퍼컴퓨터와도 경쟁할 수 있거나 이를 대치가능하고, 또한 많은 Toolkit의 개발로 이러한 시스템과도 협업이 가능하게 되었다. 그러나 이런 단일 클러스터 시스템은 변화하는 메타컴퓨팅(그리드) 환경의 개념과는 거리가 먼 것이 사실이다. 이에 이런 클러스터링 시스템 간에 상호 연결·이용은 그 어느 때보다 중요한 이슈가 되었다.

### Ⅲ. 연구개발의 내용 및 범위

다음과 같은 연구를 하여서 서울그리드센터의 서울시립대학교 캠퍼스 클러스터 그리드 테스트베드를 구축한 후 이를 국가 클러스터 그리드 테스트베드에 연결한다.

- 가. 사용자 계정 통합 및 그리드 전환 기술 연구
- 나. 자원 할당 및 분할 및 그리드 전환 기술 연구
- 다. 자원 운영 통합 및 그리드 전환 기술 연구

### Ⅳ. 연구개발결과

서울그리드센터에서 구축한 클러스터 그리드 테스트베드 중 16대의 컴퓨팅 파워를 국가 클러스터 그리드 테스트베드에 제공하여 성공적으로 운영하였다.

### Ⅴ. 연구개발의 활용에 대한 건의

국내 클러스터 그리드 컴퓨팅 구축 및 전환 기술이 개발되고 제대로 된 완성품이 나오게 되면 그리드 사업은 본 궤도를 타게 되어 국가 그리드 사업이 매우 활발히 진행되고 산업체에서 이를 이용한 사업이 매우 활발하여 질 것이다. 현재의 클러스터 그리드 컴퓨팅을 계획하는 곳은 대부분 이 연구의 혜택을 얻게 되므로 우리의 개발 산출물을 사용하게 될 수 있는 경우 중복 투자 없이 소기의 목적을 달성할 수 있게 된다.

### Ⅵ. 기 대 효 과

본 연구의 개발 결과물들과 성과들은 서울그리드센터의 테스트베드로서 국가그리드사업에 참여하는 기관들이 사용할 수 있게 개방될 것이다. 연구의 결과로서 구축될 테스트베드는 그리드 응용연구를 테스트하고 서비스할 수 있게 하는 테스트베드로서 유용하게 활용될 수 있을 것이다. 개발된 그리드 전환 기술은 국가 그리드에 참여하는 기타 기관들에게 유용하게 전수될 수 있으며 그 자체로 상업적인 패키지 기술로 선보일 수 있을 것이다. KISTI에서 연구하는 미들웨어 개발에 유용하게 사용될 수도 있다.



# Summary

## I . Title

Construction of Cluster Computing Grid

## II. Objective of the study and its importance

### 1. Objective

As a part of efforts to build Seoul Grid Center's GRID testbed, we have built a cluster GRID computing testbed. In this research, we contribute the computing power of 16 PCs from the Seoul Grid cluster computing Grid testbed to the national cluster computing Grid testbed.

### 2. Significance

Thanks to unbelievably rapid H/W developments. Now cheap PCs with high performance prevail. With the help of this trend, competing commercial Unix's performance, cheap, yet, powerful Linux clustering systems are being developed, tuned, optimized and applied competitively all around the world. These days, GRID enables us to connect the separately operated cluster systems into a metacomputing Cluster system. Indeed, it's time to do it.

## III. Content and scope of the study

In this research, we contribute the computing power of 16 PCs from the Seoul Grid cluster computing Grid testbed to the national cluster

computing Grid testbed. We have proceeded the research on next themes in building the Seoul Grid testbed.

A. to develop the technology to provide the participating organizations of the CC-GRID with a unified user accounting system.

B. to develop the technology to provide the participating organizations with a unified resource allocation and scheduling system.

C. to develop the technology to provide the participating organizations with a unified resource management system.

#### **IV. Result of the study**

In this research, we have contributed the computing power of 16 PCs from the Seoul Grid cluster computing Grid testbed to the national cluster computing Grid testbed.

#### **V. Suggestion for utilization**

When our CCGrid technologies are completed, Korean domestic grid related business would be on its right track and be prosperous. The utilization of these infrastructure could spin off variety of business. We are sure that the future plans to build CCGrid system in Korea would almost have benefit from our grid technologies without duplicated investment into CCGrid Technologies.

#### **VI. Expected Effects**

All outputs of our project along with Seoul Grid Center testbed will be open to all working and researching organizations that participate in N\* Grid Projects. The testbed will be used to test the GRID applications. The technologies obtained from this project could be transferred to other

organizations. They can be commercial packages itself, and can, if needed, be efficiently used to develop the middleware about which KISTI is researching.

# Contents

I . Objective and Significance	
I – I . Objective .....	1
I – II . Significance .....	1
II . Contents and Scope	
II – I . Final Goal .....	2
II – II . Yearly Goal .....	2
III . Research Activities	
III – I . R&D State of the Art in Korea .....	4
III – II . R&D state of the Art in the World .....	6
III – III . Standardization .....	4
IV . Requirement Analysis	
IV – I . Project Requirement & Achievement .....	9 1
IV – II . Required Function .....	13
IV – III . Check Points (Problems) .....	1 3
V . Seoul GRID System	
V – I . System Design and Construction .....	2 3
V – II . Seoul Grid Super Cluster System .....	6 3
VII . Conclusions	
VII – I . Conclusions .....	8
VII – II . Further Works .....	8

# 목 차

I. 연구의 목적 및 중요성	
I - I. 연구의 목적 .....	1
I - II. 연구의 중요성 .....	1
II. 연구의 내용 및 범위	
II - I. 2차년도 연구 목표 .....	2
II - II. 단계별 목표 .....	2
III. 연구 현황	
III - I. 국내 연구 현황 .....	4
III - II. 세계의 연구 현황 .....	6
III - III. 표준화 .....	4
IV. 요구 사항 분석	
IV - I. 과제별 요구 내용 및 수행 사항 .....	9
IV - II. 필요한 기능 .....	3
IV - III. 문제점 .....	3
V. 서울 그리드 시스템	
V - I. 시스템 설계 및 구축 .....	2
V - II. 서울 그리드 슈퍼 클러스터 시스템 .....	8
VI. 결론	
VI - I. 결론 .....	8
VI - II. 향후 연구 방향 .....	38

# 표차례

<표 III-1> 그리드 테스트베드 .....	9
<표 III-2> 그리드 프로젝트 현황 .....	4
<표 V-1> 프로토타입 I의 하드웨어 구성 .....	3
<표 V-2> 프로토타입 I 시스템의 소프트웨어 구성 .....	3
<표 V-3> 슈퍼 클러스터 시스템의 구성 .....	3

# 그림차례

<그림 III-1> 정보통신부 국가 그리드 추진 체제 .....	5
<그림 III-2> 국가 그리드 계산 그리드 테스트베드 .....	5
<그림 III-3> 테라그리드 레이아웃 .....	7
<그림 III-4> 테라그리드 계획도 .....	7
<그림 III-5> Euro Grid 중 HPC Reseach Grid의 참여 시스템 .....	8
<그림 III-6> Concurrent Standardization .....	81
<그림 III-7> 한국 IT 표준화 추진체계 .....	8
<그림 V-1> 서울시립대학교 네트워크 구성 .....	2
<그림 V-2> 프로토타입 I 의 구성도 .....	3
<그림 V-3> 시스템 A 구성도 .....	3
<그림 V-4> 시스템 B 구성도 .....	3
<그림 V-5> 서울그리드 256노드 슈퍼 클러스터 시스템 .....	63
<그림 V-6> 슈퍼 클러스터 시스템 구성도 .....	6

## 제 1 장 연구의 목적 및 중요성

### 제 1 절 연구의 목적

서울그리드센터의 그리드 컴퓨팅 테스트베드의 구축을 위한 노력의 일환으로 클러스터 그리드 컴퓨팅 테스트베드를 구축하였다. 구축된 클러스터 그리드 컴퓨팅 테스트베드를 KISTI의 국가 그리드 테스트베드에 연결하여 클러스터 자원을 제공함으로써 국가 컴퓨팅 그리드 테스트베드 구축에 참여하는데 목적이 있다.

### 제 2 절 연구의 중요성

하드웨어의 기하급수적인 발전에 힘입어 값싼 고성능 PC의 대중화가 이루어졌다. 이런 고성능 PC를 기반으로, 상용 유닉스의 성능에 버금가는 리눅스를 기본 운영체제로 채택하는, 가격대비 성능비가 뛰어난 리눅스 클러스터링 시스템이 개발되고 있다. 이 클러스터링 시스템은 어느 나라를 막론하고 경쟁적으로 구축, 튜닝(최적화), 적용되고 있는 실정이다.

또한 이러한 시스템은 특정기관에서 보유한 슈퍼컴퓨터와도 경쟁할 수 있거나 이를 대치가능하고, 또한 많은 Toolkit의 개발로 이러한 시스템과도 협업이 가능하게 되었다. 그러나 이런 단일 클러스터 시스템은 변화하는 메타컴퓨팅(그리드) 환경의 개념과는 거리가 먼 것이 사실이다. 이에 이런 클러스터링 시스템 간에 상호 연결·이용은 그 어느 때보다 중요한 이슈가 되었다.



## 제 2 장 연구의 내용 및 범위

### 제 1 절 2차년도 연구 목표

다음과 같은 연구를 하여서 서울그리드센터의 서울시립대학교 캠퍼스 클러스터 그리드 테스트베드를 구축한 후 이를 국가 클러스터 그리드 테스트베드에 연결한다.

- \* 사용자 계정 통합 및 그리드 전환 기술 연구
- \* 자원 할당 및 분할 및 그리드 전환 기술 연구
- \* 자원 운영 통합 및 그리드 전환 기술 연구

### 제 2 절 단계별 목표

#### 가. 1단계 목표

서울 그리드 센터의 1차년도 목표는 캠퍼스 그리드 시스템을 구축하는데 있다. 그 동안 독자적으로 운영하고 있는 6개의 클러스터 시스템을 기가비트고속통신 학내망을 이용하여 캠퍼스 클러스터 그리드 컴퓨팅 환경을 구축하는데 1차년도 목표를 둔다. 독자적으로 운영되고 있던 6개의 클러스터 시스템은 다음과 같다. 시립대학교 슈퍼클러스터시스템(256 CPU), 물리학과 클러스터시스템, 전자전기컴퓨터공학부 클러스터 시스템, 컴퓨터-통계학과 클러스터시스템, 기계정보학과 클러스터시스템, 전자계산소 클러스터 시스템.

#### 나. 2단계 목표

1차년도에 완성한 서울시립대학교 캠퍼스 그리드 시스템과 인근 연구 및 학술기관들의 보유 클러스터 시스템들을 연결하여 서울 동북부 지역 글로벌 그리드 환경을 구축하고 관련 전환 기술을 개발하는데 2차년도 목표를 둔다. 대상기관은 다음과 같다. 연구기관으로 KIST, KIAS(고등과학원) 등이다. 대학교로는 국민대학교, 서울여자대학교, 한성대학교, 고려대학교 등이다.

#### 다. 3단계 목표

서울시의 산하기관들의 클러스터 시스템을 1, 2차년도 완성한 서울그리드센터의 클러스터 그리드 환경에 연결하여 구축하고 관련 전환 기술을 개발하는데 3차년도 목표를 둔다.

#### 라. 4단계 목표

서울시 상암동에 추진하고 있는 서울 DMC에 구축된 DMC GRID 시스템을 1,2,3년도까지 완성된 서울그리드센터의 클러스터 그리드 환경에 연결하여 구축하고 관련 전환 기술을 개발하는데 4차년도 목표를 둔다.

#### 마. 5단계 목표

4차년도까지 달성된 서울그리드센터의 클러스터 그리드 환경이 서울시 동북부 지역이외의 전역을 커버할 수 있도록 클러스터 그리드 시스템을 확대 구축하고 이에 필요한 전환 기술을 개발한다.

## 제 3 장 연구 현황

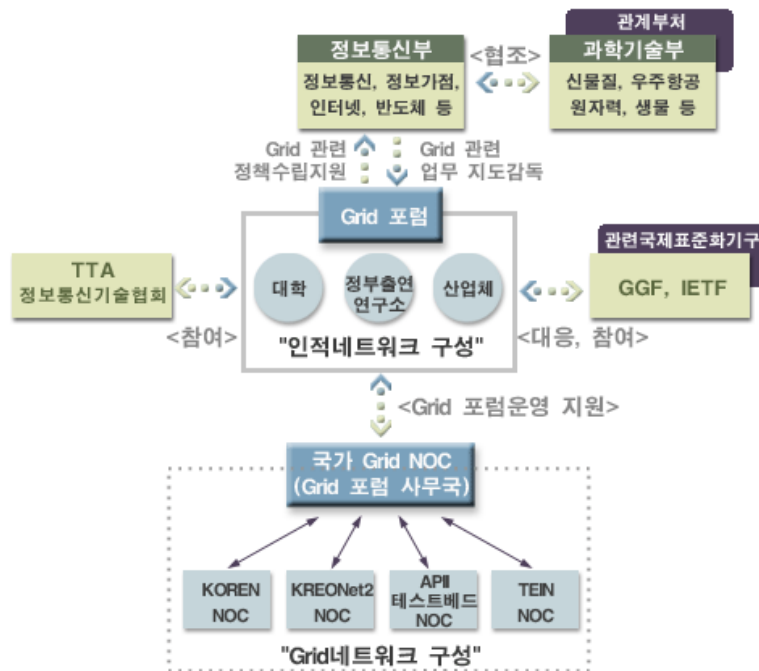
### 제 1 절 국내 연구 현황

CC-GRID는 지역적으로 분산되어 있는 클러스터들을 연동해서 그리드 컴퓨팅 환경을 구축하는 것이 목표이다. 따라서 과거부터 클러스터를 직접 운영을 해 오고 있는 기관 및 대학들이 큰 관심을 기울이고 있으며, 실제로 현재 CC-GRID를 연구하고 있는 곳 또한 바로 이들 기관 및 대학이다. 많은 곳에서 소규모로 CC-GRID를 연구하고 있는 것으로 보이나, 아직까지 눈에 띄는 성과를 올린 곳은 없다. 국내에서 CC-GRID를 전문적으로 연구하고 있는 곳은 대표적으로 KISTI의 국가 그리드 테스트베드 구축 연구를 꼽을 수 있다.

우리나라는 초고속 인터넷망이 이미 세계적인 수준으로 구축되어 운용되고 있기 때문에 GRID 응용 프로젝트 수행을 위한 네트워크 구성은 용이한 편이다. 다만, 정부출연연구소 및 산업체 연구소들이 기관별로 사업을 추진함으로써 연구 효율성 및 산업간 시너지 효과가 미흡한 편이다. 이러한 문제점들을 극복하기 위해 2001년 5월 정보통신부가 “차세대 인터넷 기반구축을 위한 국가 그리드(GRID) 기본 계획”을 수립하였다. 이 계획에 의하면 정보통신부가 “국가 GRID” 관련 사업을 총괄하고 정부출연연구소들은 GRID 관련 응용 프로젝트들을 발굴, 추진한다. 또한 대학 및 산업체 연구소는 응용 프로젝트 참여, 추진 및 기술개발 능력을 확보하는데 중점을 둔다.

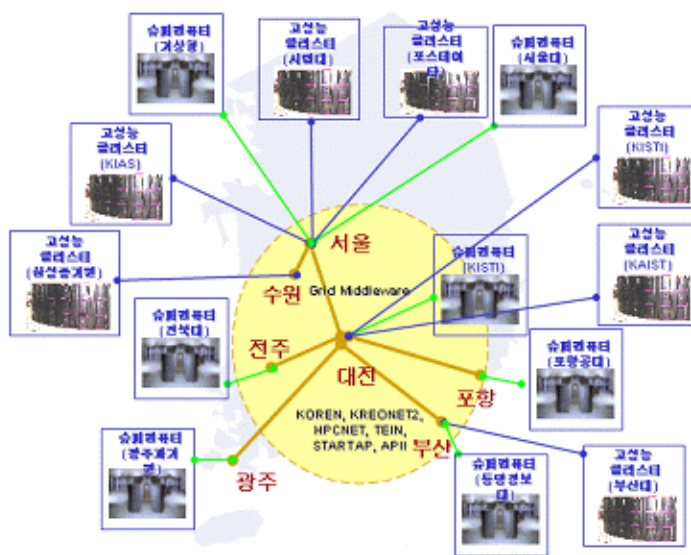
이에 따라 정보통신부는 2002년부터 2006년까지 “국가 그리드 구축사업”을 추진하고 있다. 이 사업의 구체적인 목표는 첫째, 슈퍼컴퓨터와 고성능 클러스터 중심의 계산 그리드 구축 및 관련 기술개발, 둘째, 데이터 액세스 그리드, 협업 액세스 그리드, 몰입형 액세스 그리드 등과 같은 첨단 액세스 그리드 구축 및 관련 기술개발 셋째, 응용 그리드 포털, 응용 그리드 테스트베드, 바이오 및 그리드 응용연구 등이 포함된 응용 그리드 구축 및 관련 기술개발 등이 있다. 또한 그리드의 핵심 기술인 그리드 미들웨어 연구개발 및 멀티미디어 3차원 브라우징 기술 개발을 오는 2005년까지 상용화함으로써 세계적으로 초기 연구단계인 그리드 표준 발굴에 나설 계획이다. 여기에는 2001년부터 5년간 총 435억원의 예산이 투입될 계획이다.

국내에서는 KISTI 슈퍼컴퓨터센터가 본 연구사업으로 <그림 III-2>와 같이 서울그리드센터(서울시립대), KAIST, 서울대, 포항공대, 전북대, 부산대 등을 연결하여 국가 그리드 테스트베드 서비스를 제공하였다.



<그림 III-1. 정보통신부 국가 그리드 추진 체제>

현재, 국가 그리드 기본 계획에는 항공우주 분야뿐 만 아니라 다양한 응용 분야의 연구들이 포함되어 있다. 나노물질의 구조를 파악하고 우수한 물질의 개발을 위한 나노 그리드, 신약 개발과 새로운 단백질의 구조 규명 등 생물학 분야의 연구를 위한 바이오 그리드, 환경 오염 문제의 해석 등을 위한 환경 그리드 등이 국가 그리드의 일부로서 연구되고 있다. 아울러, 세계적으로 경쟁력을 가지는 그리드 기술의 확보를 위하여 그리드 미들웨어의 세부 기술들과 연계하여 자동화된 그리드 컴퓨팅 구현을 위한 기술 개발을 수행 중에 있다. (<http://www.kisti.or.kr>)



<그림 III-2. 국가 그리드 계산 그리드 테스트베드>

## 제 2 절 세계의 연구 현황

국외에서 많은 기관 및 대학에서, 클러스터뿐만 아니라 슈퍼컴퓨터 및 일반 피씨들까지 포함하는 그리드 테스트베드를 구축 중에 있다. 본 연구보고서에서는 클러스터가 들어가는 그리드 테스트베드를 중심으로 현황을 조사하여 주요 테스트베드를 분석하였다.

### 가. 테라그리드[1]

2001년 8월 9일 미국의 과학재단(NSF)은 5,300만 달러의 기금을 투자하여 테라바이트 급 슈퍼 그리드 컴퓨팅 연결을 시도한다고 발표하였다. 이 슈퍼컴퓨팅 시스템을 DTF(Distributed Terascale Facility)라고 하며, 초당 11.6조 계산을 하고, 총 450조 바이트의 데이터를 저장할 수 있다. 이 종합적인 인프라를 테라그리드(TeraGrid)라고 하며, 여기에 송수신 속도는 모든 광통신(All Optical Network)을 통하여 초당 40억 비트(4기가) 속도로 전송하거나 받을 수 있다.

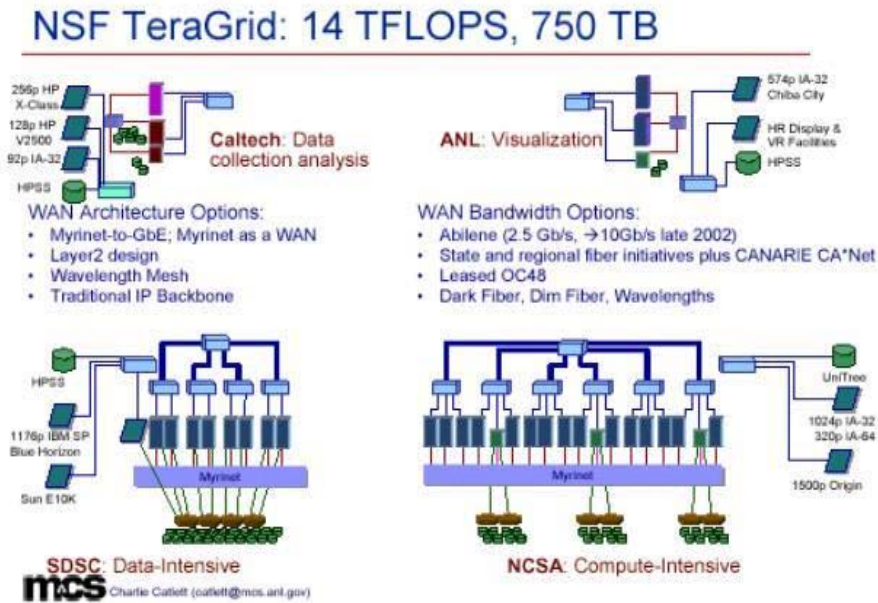
이를 위해 NSB(National Science Board)는 3년간의 NSF 기금 수상자를 선정했는데, 일리노이즈 소재의 NCSA(National Center for Supercomputing Applications)와 샌디에고의 SDSC(San Diego Supercomputer Center)의 컨소시엄, 시카고의 ANL(Argonne National Laboratory), 파사다나의 Caltech(California Institute of Technology)이다.

Terascale이란 초당 1조 이상의 부동소수점연산(Floating-point operations)을 의미한다. 이 DTF 프로젝트는 2002년 중반부터 운영되며, 2003년 4월까지의 최대 11.6 teraflops의 성능에 도달할 예정이다. 이들 사이트들은 이를 활용한 폭풍(storm), 기후(climate), 지진(earthquake) 등의 사전예측 등을 한다. 또한 엔진의 효율성 향상, 화학과 분자, 물리학, 재료 등을 연구한다.

여기에 참가하는 파트너는 IBM, Intel 및 Qwest Communications사를 비롯해서, Myricom, Oracle, Sun도 참여한다.

NCSA는 컴퓨터를 연구하는데, Intel의 64비트 이타니움(Itanium) 프로세서 베이스 IBM의 리눅스 클러스터(Linux cluster)를 지원한다. 240 테라바이트 스토리지에 8 teraflops까지의 성능을 가능하게 한다. SDSC는 데이터와 지식경영을 연구하는데, Intel의 64비트 이타니움(Itanium) 프로세서 베이스 IBM의 리눅스 클러스터와 225 테라바이트 스토리지로 지원하고 여기에 Sun사의 차세대 하이엔드 서버가 지원된다. ANL은 소프트웨어를 연구하는데, 1 teraflops의 IBM 리눅스 클러스터가 지원된다. Caltech은 과학 데이터를 연구하는데, 4 teraflops의 McKinly Cluster와 32-노드 IA-32 cluster가 지원하며, 온라인 연결시 86 테라바이트의 스토리지를 관리하도록 한다.

<그림 Ⅲ-3>와 <그림 Ⅲ-4>에서는 테라그리드의 레이아웃과 계획도를 소개하였다.



<그림 Ⅲ-3. 테라그리드 레이아웃>



<그림 Ⅲ-4. 테라그리드 계획도>

## 나. Euro Grid [2]

유로그리드는 Bio Grid, Meteo Grid, CAE Grid, HPC Research Grid 4개의 작업 그룹으로 구성된다. 이 중 HPC Research Grid는 유로그리드의 참여자들이 컴퓨팅 자원들을 사용할 수 있도록 하는 그리드 환경을 만드는데 그 목적이 있으며, 본 과제의 연구 목적과

유사하다. 이 연구는 각 기관에서 보유하고 있는 전용의 계산 자원들을 사용할 수 있게 함으로써, 유럽 계산 그리드 환경을 구축할 예정에 있다.

HPC Research Grid는 분산 어플리케이션의 개발을 위한 테스트베드로써 사용되어질 것이며, 유로그리드 파트너들 사이의 공동 작업 연구를 위한 활동 무대로도 사용되어질 것이다. 최종적인 목적은 유럽에 분산되어 있는 계산 자원들을 이용한 그리드에서 풍부한 생산 시스템을 완성하는 것이며, 중요한 과학적인 문제들을 풀이하는데 있어서 각기 다른 사이트들 사이에 밀접한 협력을 갖게 하는 것이다.

이 연구에서는 유로그리드의 다른 워크패키지들(Bio Grid, Meteo Grid, CAE Grid)같이 특정 분야에 한정짓지 않고, 일반적인 분산 어플리케이션의 개발에 대해서 전념하고 있다. 과학 연구를 위한 커다란 가치를 가지는 분산 어플리케이션들은 다른 소프트웨어 모듈들의 결합을 바탕으로 여러 가지 물리 시뮬레이션 또는 복잡한 계산 사슬에 의존하는 것과 관련되고 있다. 이러한 분산 어플리케이션들은 다른 계산 자원들의 통일성 있고 상호 보완적인 사용을 요구하고 있다.

순차적인 소프트웨어 모듈들의 연결은 UNICORE의 현재 기능을 통해서 달성될 수 있다. 소프트웨어 모듈들의 실시간 상호 작용을 포함하는 더욱 대규모의 어플리케이션들은 이식 가능하고 잘 개발된 MPI2, CORBA등과 같은 표준들을 사용함으로써 개발되어질 것이다. 이 연구에서는 인터넷 대역폭과 같은 HPC Research Grid의 Qos, 관리와 작동 이슈들 그리고 보안 이슈들과 관련된 주제들을 가지고 연구를 진행할 것이다.



<그림 III-5. Euro Grid 중 HPC Research Grid의 참여 시스템>

#### 다. GSC(UK Grid Support Centre) [3]

여러 그리드 테스트베드가 영국 내에서 개발되고 있는 중이다. 연합 연구 장비 모임 또는 그와 비슷한 기관에서 자금을 지원한 자원들은 이런 테스트 베드를 늘리는데 좋은 기반이 될 것이다. 주목받고 있는 "mini-grids"는 아마 급격하게 설립될 것이고, 국가 그리드가 어떻게 나아가야 하는지를 보여 줄 것이다.

## 라. The Distributed ASCI Supercomputer(DAS) [4]

미리넷을 기반으로 한 펜티엄 프로 클러스터 4개로 만들어진 200 노드의 광범위 분산 시스템인 DAS(Distributed ASCI Supercomputer)는 ASCI(Advanced School for Computing and Imaging)에서 설계된 광범위 분산 클러스터이다. DAS는 아래의 다섯 네덜란드 대학교에서 병렬 및 분산 컴퓨팅을 연구할 목적으로 만들어졌다.

- (1) Vrije Universiteit, Amsterdam
- (2) University of Amsterdam
- (3) Delft University of Technology
- (4) Leiden University
- (5) University of Utrecht

DAS는 처음 4개의 대학교에 위치한 클러스터로 구성이 되어있다. 첫 번째 클러스터(VU에 있는)는 128 노드를 포함하고 있으며, 다른 세대의 클러스터는 24개의 노드를 가지고 있다(전부 200 노드임). 이 시스템은 광범위의 병렬 시스템을 만드는데 다년간의 경험을 가진 Parsytec (Aachen, Germany)에서 만들어졌다. Parsytec는 자사의 CC 시리즈에 사용했던 것과 같은 패키징 기술을 DAS에도 사용하였다. DAS의 운영체제는 레드햇 리눅스이다. DAS 시스템은 NWO (과학연구를 위한 네덜란드 기관), SION (네덜란드에 있는 과학재단) 그리고 참여대학이 자금을 제공한다.

## 마. 그 외의 그리드 테스트베드 소개

본 절에서는 앞에서 자세히 소개한 그리드 테스트베드를 비롯한 세계의 수많은 테스트베드를 간단히 소개한다. 각 테스트베드별로 주요 연구분야 혹은 특징, 응용 분야, 국가/주관기관에 대해서 조사하였다.

<표 III-1. 그리드 테스트베드>

프로젝트 명	주요 연구 분야 혹은 특징	응용(적용 분야)	국가/주관 기관
Wide Grid (WWG)/ECONOMY GRID[5]	연구 주요 초점 : 전세계적인 그리드 컴퓨팅 차원에서 경제적이고 시장경제적인 자원 관리와 스케줄링 시스템. 연구와 개발은 두개의 축으로 수행됨 : GRACE(GRid Architecture for Computational Economy), Grid Resource Broker (GRB, aka Super Schedulers/MetaSchedulers)	시뮬레이션, 제약, 네트워크와 퍼지 이론 시뮬레이션	호주/모나쉬대학



Polder Metacomputer [6]	분산 병렬 컴퓨터 시스템에서 계산 문제를 최적으로 매핑하기 위한 시스템을 구성하는데 그 목적이 있음.	분리 이벤트 시뮬레이션(DES)	네덜란드/ 암스테르담 대학
Asia Pacific Bioinformatics Network [7]	Asia Pacific Bioinformatics Network (APBioNet)는 비영리, 비국가주관의 기관이다. 이 기관은 아시아 태평양 지역의 바이오인포메틱스의 발전을 도모하는데 초점을 두고 있다. 1998년 이후, 이 기관의 과제는 회원국 간의 바이오인포메틱스의 계몽, 훈련, 교육, 자원과 연구 등을 개척하는 것이다.  또한 EMBnet과 같은 다른 국제적인 기관들과도 기술적인 협력하고 연락하는 것을 포함한다. 바이오인포메틱스 기반 자원을 더 진보시키기 위하여, 이 기관은 또한, APAN(Asia-Pacific Advanced Network project)과도 동반자 관계를 형성하고 있다.	APBioNet은 특히, 바이오인포메틱스 분야에서 기반자원, 데이터와 정보 교환, 교육 프로그램, 국제협력 그리고 국제회의 등을 형성하고 발전시키는데 많은 노력을 기울이고 있다.	아시아지역 12개국에서 300명 이상의 회원과 20개 이상의 기관들이 참여
NASA Information Power Grid (IPG)[8]	IPG는 NASA를 중심으로 분산컴퓨팅 및 분산데이터 관리 환경을 구축하는 과제로서 어플리케이션 사용자들에게 분산된 컴퓨팅 자원들과 데이터 자원들을 통합하여 사용할 수 있게 하고, 고가의 실험 장비들을 온라인으로 사용 가능하게 하며 협력 연구가 필요한 연구원들 사이에 온라인으로 협업을 가능하게 하는 것이 목표이다.	NASA의 고성능 계산 그리드	미국/IPG 는 다음과 같은 기관들의 협업 환경을 구축하기 위한 노력이다. NASA Ames, NASA Glenn, NASA Langley Research Centers,, the NSF PACI programs at SDSC, NCSA,  재 정 지 원 : Computing , Information and Communications Technology

			(CICT) program at NASA Ames Research Center
NPACI: Metasystems [9]	NPACI의 Metasystems은 분산된 고성능 컴퓨팅과 데이터 자원들에 접근하기 위해서 통합되고, 보안이 유지된, 사용하기 쉬운 환경을 제공한다. thrust area 는 다르고 지역적으로 떨어져 있는 자원들을 하나의 단일한 자원으로 사용하는 것을 허락하기 위한 안정되고 효과적인 기반을 제공하는 컴퓨팅을 지원할 수 있다. 이 연구는 다음과 같이 3가지 프로젝트로 구성되어 있다. 1. Globus, 2. Application-Level Scheduler 3. Network Weather Service	제 삼의 목표 에이즈 치료제, 전 세계적인 최적화환경에 대한 접근, Leary Tetrahedron, 지구 내부 구조(활동) 모델링, 은하계 해석 등.	미국/UC San Diego (UCSD)의 Fran Berman와 San Diego Supercomputer Center (SDSC) 등이 참여
ThaiGrid [10]	다음과 같은 목적으로 글로벌 기술, 다른 지역 기술 그리고 소프트웨어 툴을 기반으로 그리드 인프라를 구성한다.  - 태국에 있는 슈퍼컴퓨팅 사이트 간의 자원 공유를 용이하게 한다. - 새롭게 뜨고 있는 그리드 컴퓨팅 기술 연구하고 사용한다. - 그리드에 기초한 응용프로그램, 소프트웨어 툴, 그리고 알고리즘의 개발에 대한 테스트베드 시스템을 구축한다. - 태국에 있는 고성능 컴퓨팅 연구 그룹간에 협업작업을 모의 실험한다.	태국 국가 그리드 테스트베드  흥미를 끄는 실험 결과를 다음 사이트에서 볼 수 있다. <a href="http://prg.cpe.ku.ac.th/thaigrid/result.htm">http://prg.cpe.ku.ac.th/thaigrid/result.htm</a>	태국/Kasetsart University 그리고 King Mongkut's Institute of Technology North Bangkok 을 중심으로
The Alliance Virtual Machine Room[11]	NCSA와 이의 국가 계산 과학 협업 동반자들은 현재 협업 VMR을 구성하기 위해 기본시설을 구축하고 있다. 목표는 6군데로 따로 위치하는 자원들을 마치 단일한 자원인 양 접근하여 사용 관리하기 위하여 이 자원들을 묶는 대단위 작업.	사용자의 관점에서, VMR(virtual machine room) 실제의 것 동일하다. 그 공간은 하나의 사이트에 있는 자원과 서비스의 다양성을 제공해준다. 물론 VMR안에서 이	미국/과학계산을 위한 NCSA 분과(Division)

		<p>런 자원들과 서비스들은 지역적으로 떨어진 곳에서 다른 OS 버전으로 돌아가고 있는 시스템들로부터 제공되어 진다. 자원들의 산개 되어 있는 특성은 보이지 않으며, 시스템을 사용할 때 장애가 될 것이라고 예상되어 지지도 않는다. 이상적으로 VMR은 투명한 개념이다.</p>	
Micro Grid[12]	<p>MicroGrid 프로젝트의 목표는 편리한 수단과 그리드 자원 관리 문제의 과학적인 연구를 제공 해주는 가상 그리드 기반 시설을 개발하고 구현하는데 있다. 가상 그리드 기반 시설은 반복 가능하고, 제어 가능한 실험을 동적인 자원 관리 기술로 가능케 할 수 있어야 한다. MicroGrid 실험은 완전한 규모에서 실험을 보완하고 그리드 설정의 광범위한 다양성을 넘는 제어 알고리즘의 과학적 평가를 지원한다. 추가된 이득은 줄어든 모의 실험에 대한 노력과 증가된 실험의 관찰능력이 포함될 것이다.</p>	<p>- 확장 가능한 (scalable) 클러스터 자원을 사용한 그리드 응용 모의실험 지원</p> <p>- 응용프로그램 하여금 Identical API을 실행하는 하고 마치 실제적인 그리드 환경에서 인 양 반응하게끔 그리드 소프트웨어 환경을 지원하기 위하여</p> <p>- 그리드 자원들의 수행 특성들이 환경 설정되어 질 수 있도록 모의 실행 지원, , MicroGrid 기반자원의 능력들을 확장, 개선할 수 있도록 하는 개방된 소프트웨어 환경</p>	<p>미국/NFS의 그리드 애플리케이션 개발 소프트웨어 프로젝트로서 Rice University</p>
Alliance Grid Technologies [13]	<p>Alliance의 목표를 실제로 바꾸는 것은 그리드의 기술 전반에 걸친 청사진에 대한 이해력이 필요하다. 이러한 미래에 대한, 세부적인 계획으로부터, 기술을 최고조로 배치운용하는 노력들을 개발하여 오고 있다. : 계산용 클</p>	<p>NCSA는 미국전역에 걸쳐서 계산적인 과학(예, 바이오인포메틱스)을 개선하고 미국 국가 그리드의 부분이 될 시스템과 응용프로</p>	<p>미국/학교, 회사, 정부기관 등으로 구성된 50개 이상의 기관</p>

	러스터, 그리드 소프트웨어, 액세스 그리드 등	그램을 개발하는데 도움이 될 고성능, 고생산성 시스템 제공한다. 어떤 팀은 VMR을 위해 일을 한다.(VMR을 대하여는 전문 내용을 보라)	
DSG GridPortal [14]	이 프로젝트는 CLRC e-Science와의 협력 연구과제로서 영국 고성능 컴퓨터 회를 위한 그리드 서비스 포털을 개발하는 것이다. 현재의 목표는 존재하는 웹 포털 소프트웨어 예를 들면, GridPortal, WebFlow, 그리고 JiPANG등을 구축하여 평가하는 것이다. 이 연구의 초점은 특별한 응용 프로그램을 위한 웹포털을 빨리 구축하기 위하여 사용되어질 수 있는 툴과 라이브러리들을 개발하는 것이다.	응용프로그램 데이터베이스/ 특성화된 장치로서 고성능 컴퓨팅 환경 제공 / 저장장치	영국/ 포츠마우스 대학
G-WAAT[15]	이 시스템은 PSC, SNL, 그리고 HLRS 등의 컴퓨터 자원을 통합하고자 만들어 짐	여러 분야에 적용 가능한 CCGrid testbed	미국/ Pittsburgh Supercomputing Center (PSC) Sandia National Laboratories (SNL), High Performance Computing Center Stuttgart (HLRS)
인터넷 영화 프로젝트 (Internet Movie Project)[16]	전세계의 여러사람 자원으로부터 공동, 협업 환경을 힘입어 레이트레이싱 기법(POV-Ray raytracer)을 이용하여 영화를 만드는 것.	현재로서는 주로 단편 영화 개발	img.org
Irish Computational Grid (ICG)[17]	아일랜드에 있는 유수의 교육기관, 대학, 연구기관 등 고등교육기관 (HEA:Higher Education Authority)로 구성되어, 이 기관들이 보유한 자원들을 HEAnet에 의해 서로 연결하는 프로젝트이다. Grid-Ireland는 그	모든 계산 영역 및 액세스 그리드 :	아일랜드/ 국가 계산용 그리드

	리드서비스를 제공하는 HEAnet를 관리하는 가상의 상위 레벨기관(manager virtual organization)이다.		
--	---	--	--

<표 III-2. 그리드 프로젝트 현황 (단위 : 백만) : ITA 자료>

프로젝트명	국가	기관	예산	기간
GriPhyN	미국	NSF	\$ 13.5	2000 - 2005
e-Science	영국	영국정부	£ 98	2000 - 2002
EuroGrid	유럽연합	EC	Eur 2	2001 - 2004
EU DataGrid	유럽연합	EC	Eur 10	2001 - 2004
LCG(Ph1)	스위스	CERN	CHF 30	2002 - 2004
NEESGrid	미국	NSF	USD 10	2000 - 2004
TeraGrid	미국	NSF	USD 53	2001 - 2004
IPG	미국	NSF	\$ 9.5	2001 - 2004
ITBL	일본	RIKEN	-	-

## 제 3 절 표준화 활동

본 절에서는 우리의 연구에 해당하는 분야에서의 표준화 활동을 GGF, W/G, R/G를 중심으로 살펴보았다.

### 가. GGF WG/RG에 기준한 연구 활동 및 분석

#### (1) 사용자 계정 통합 및 그리드 전환기술 기초연구

관련 RG는 Accounting Models Research GROUP([http://www.gridforum.org/5\\_ARCH/ACCT.htm](http://www.gridforum.org/5_ARCH/ACCT.htm))이 있다. 이 R/G은 Distributed Accounting Working Group(DAWG) 이라고도 한다. 여기서는 Grid environment상에서 계정관리와 관련된 문제를 다루고 있다. 이 연구 그룹은 아래와 같이 3 가지의 목적을 가지고 있다.

- 프로토타입 그리드 어카운팅 아키텍처의 개발
- 다른 그리드 컴포넌트와 존재하고 있는 시스템들간의 상호작용의 정의
- 프로토타입 시스템을 구축하고 평가하는데 사용될 수 있는 명세서의 개발

이 연구 그룹에서는 위의 목적을 가지고 다음과 같은 초안과 개요를 만들고 있다.

- ① Account Allocations on the Grid
- ② An Economy Grid Architecture for Service-Oriented Grid Computing
- ③ Distributed Accounting on the Grid
- ④ Distributed Accounting Working Group (DAWG) Charter
- ⑤ Resource Accounting - Current Practices
- ⑥ Simplifying Administration and Management Processes in the Polish National Cluster

## **(2) 자원 할당 및 분할 및 그리드 전환기술 기초연구**

자원 할당, 분할 및 그리드 전환 기술 기초 연구는 Scheduling & Resource Management Area([http://www.gridforum.org/3\\_SRM/srm.htm](http://www.gridforum.org/3_SRM/srm.htm))에서 다루고 있다. 현재 GGF 내에서 이 분야에 대해서 전문적으로 연구하는 R/G는 없는 것으로 보인다. 단, 이와 관련된 W/G이 3곳이 있는데, 그것은 다음과 같다.

- ① Scheduling Attributes  
(<http://www-unix.mcs.anl.gov/~schopf/ggf-sched/WG/sa-wg.html>)  
- 그리드 환경에서 다른 scheduling instance들 사이의 통신을 위한 속성들을 정의하는 문서들을 만드는 것을 목표로 삼고 있다.
- ② Scheduling Dictionary([http://www.gridforum.org/3\\_SRM/SD.htm](http://www.gridforum.org/3_SRM/SD.htm))  
- 로컬과 그리드 레벨에서의 스케줄러들 사이에 공통적인 용어를 정의하는 사전  
을 개발 중이다.(현재 워킹그룹 웹사이트는 접근을 할 수 없는 상태이다.)
- ③ Distributed Resource Management Application API Working Group  
(<http://www-unix.mcs.anl.gov/~schopf/ggf-sched/WG/drmaa-wg.html>)  
- 하나 또는 그 이상의 분산된 자원 운영 시스템에서 job을 운영하기 위한 API를  
개발하는 워킹 그룹이다.

현재 Resource Management만을 따로 다루고 있는 Workig Group는 존재하지 않는 것으로 보인다.

## **(3) 자원 운영 통합 및 그리드 전환기술 기초연구**

자원 운영 통합에 관한 연구는 그리드 환경을 구성하는 자원들에 대한 정보를 알아야 구현될 수 있는 것으로 특히 Information service에 큰 관련을 두고 있다. GGF에서 자원 운영 통합 및 그리드 전환 기술 기초 연구를 다루고 있는 부분은 Grid performance and Information services( [http://www.gridforum.org/4\\_GP/Perf.htm](http://www.gridforum.org/4_GP/Perf.htm) )이다.

- ① Discovery and Monitoring Event Description ( DAMED )
- ② Grid Monitoring Architecture Working Group (GMA-WG)
- ③ Network Measurements Working Group (NM-WG)
- ④ Grid Notification Framework (GNF-WG )
- ⑤ Metacomputing Directory Services (MDS-WG)

위의 W/G중 자원 운영 통합 기술과 관련된 W/G는 Grid Notification Framework(GNF-WG,[http://www.gridforum.org/1\\_GIS/GNF.htm](http://www.gridforum.org/1_GIS/GNF.htm))과 Metaompting Directory Servces(MDS-WG)이다. 이 둘의 W/G의 web site는 <http://www-unix.mcs.anl.gov/gridforum/gis/>이다.

다음의 사이트에서도 유사한 정보를 얻을 수 있다.

- Discovery and Monitoring Event Description (<http://www-didc.lbl.gov/damed/>)
- Grid Monitoring Architecture Working Group  
(<http://www-didc.lbl.gov/GGF-PERF/GMA-WG/>)
- Metacomputing Directory Services([http://www.gridforum.org/1\\_GIS/MDS.htm](http://www.gridforum.org/1_GIS/MDS.htm))

#### (4) 프로토타입 연구를 위한 실험시스템 구축

현재 본 과제에서는 다양한 형태의 소규모 실험시스템들이 구축이 된 상태이고, 이후에는 더 많은 시스템들을 묶어서 사용할 수 있는 대규모 실험시스템을 구축할 예정이다. 이와 관련된 W/G는 밑에 소개하였다. 밑에 소개된 워킹 그룹과 연구 그룹에서는 실제로 그리드 환경을 구성하기 위한 테스트베드를 구축하거나, 테스트베드를 구축하기 위해서 고려해 봐야 할 사항들에 대한 연구를 진행 중에 있다.

- ① Open Grid Service Infrastructure Working Group  
(<http://www.gridforum.org/ogsi-wg/>)
  - 이 워킹 그룹의 목적은 the Grid Service Specification 과 OGSA(Open Grid Service Architecture) 환경과 관계되는 기술적인 문서들을 포함하는 여러 가지 문서들을 검토하고 다듬는 것이다.
- ② Application and Test Beds([http://www.gridforum.org/7\\_APM/APPS.htm](http://www.gridforum.org/7_APM/APPS.htm))
  - 광범위한 어플리케이션 커뮤니티와 개발자들 그리고 디렉터들 사이의 다리를 제공하려는데 목적이 있다. 이 연구 그룹의 홈페이지는 제한된 사용자만 접근할 수 있기 때문에 현재 상황에서는 들어갈 수가 없다.
- ③ Grid User Services([http://www.gridforum.org/7\\_APM/GUS.htm](http://www.gridforum.org/7_APM/GUS.htm))
  - 그리드 환경에서 일어나는 사용자 서비스들에 대해서 연구하는 그룹이다.
- ④ Grid Computing Environment(<http://www.computingportals.org/>)
  - framework, portals, Problem Solving Environments, 그리고 Grid-based

computing environments와 Grid service들의 결합과 공유에 대해서 연구 중에 있다.

⑤ Grid Globus를 이용한 testbed : <http://www.globus.org/research/testbeds.html>

#### (6) 연구 활동 및 분석

현재 Accounting은 모든 Grid 사용자가 하나의 account를 사용하는 것에서, 하나의 account 계정으로 전화 되는 것을 거쳐, 분산 account으로 가면서 관리의 불편함을 줄이는 추세이다. 이 Distributed Accounting는 distributed Accounting Working Group에서 연구를 주도하고 있다. (<http://people.nas.nasa.gov/~thigpen/accounts-wg/>)

자원 할당 및 분할은 Scheduling & Resource Management Area에 속하는 부분으로 GGF에서는 Scheduling과 Resource Management부분을 함께 다루고 있는 것으로 보인다. 현재 Resource Management 부분을 연구 하고 있는 Working Group는 없는 것으로 보인다. Distributed Resource Management Application API Working Group(DRMAA)는 Resource Management를 연구하는 것이 아니라 Resource Management Application API를 연구하는 W/G 이다.)

그러므로 Scheduling Attributes W/G에서 그리드 환경에서 다른 스테줄링 인스턴스들 간에 통신을 위한 attribute를 정의한 문서를 만드는 것을 목표로 하고 있는데 이곳에서 Resource Management에 대한 표준화 작업이 이루어진다고 할 수 있다.

자원 운영 통합은 그리드 performance and Information services area에서 다루어지는데, GIS(Grid Information Service)(<http://www-unix.mcs.anl.gov/gridforum/gis/>)에서 표준화 작업을 하고 있는 것 같다.

### 나. 향후 계획

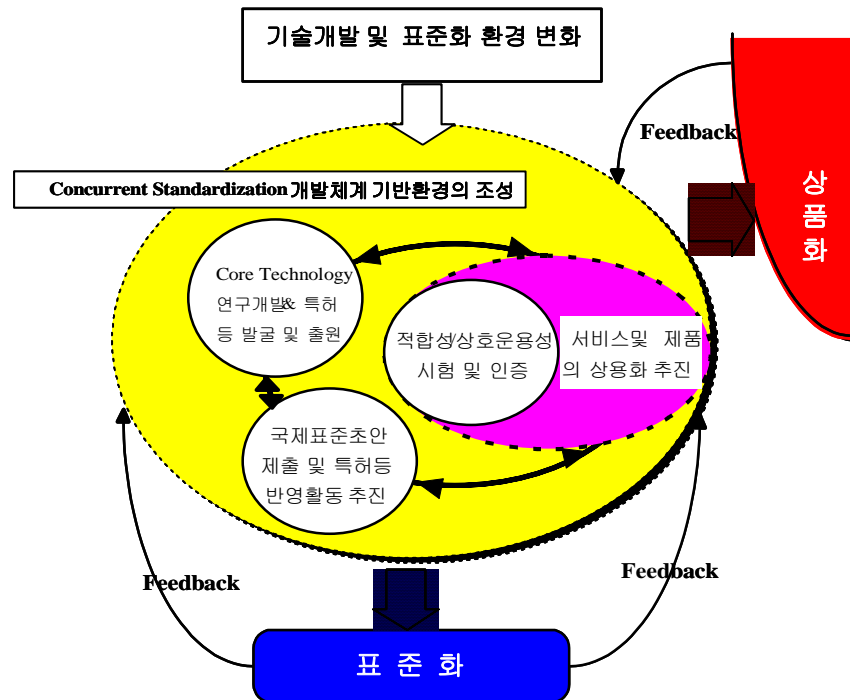
각 워킹 그룹 별로 나와 있는 문서 및 논문들은 대부분 수집 및 분석이 되어 있는 상태이다. 그러나 워킹 그룹 중에는 GGF의 홈페이지에서 링크가 깨져 있거나, 업데이트가 잘 안 되어 있는 곳이 있어서 문서 및 그들의 노하우를 쉽게 얻거나 접하는 것이 어려운 실정이다. 워킹 그룹들이 모두 국외에 있는 관계로, E-mail이나 팩스를 사용해서 그들과 의견 및 정보들을 주고받는 형태로 본 과제의 연구를 진행 중에 있으며 이는 향후에도 그대로 적용될 것이다.

또한 이미 구축되어 있는 워킹 그룹들의 대규모 테스트베드에 대한 연구를 진행 중에 있으며, 이 연구들은 본 과제에서 수행 중인 실험시스템의 구축에 사용되어질 것이다.

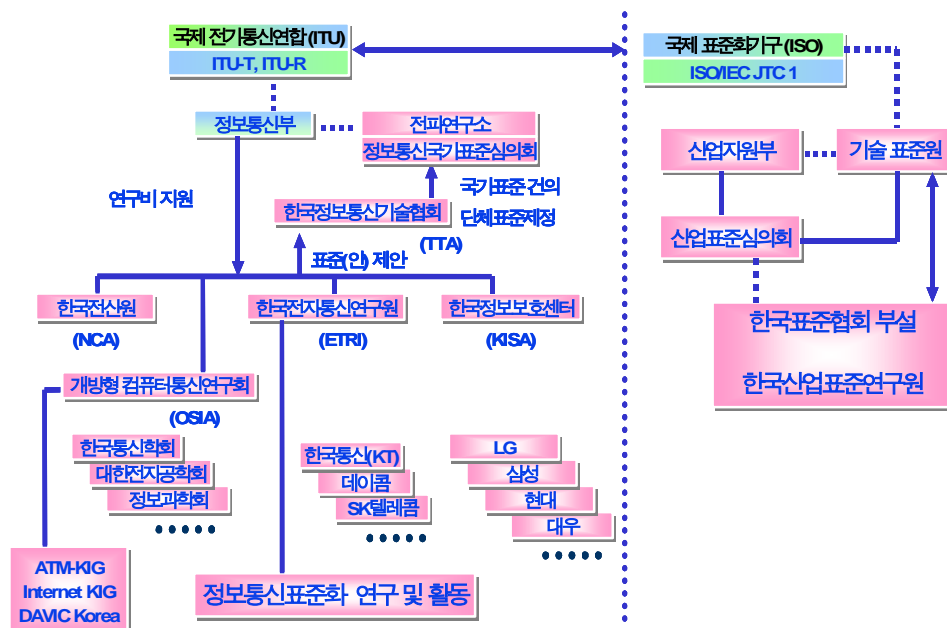


## 다. 국내 활동 분석

국내에서는 동국대학교를 중심으로 그리드 표준화 관련 연구를 하고 있다. <그림Ⅲ-6>와 <그림 Ⅲ-7>은 동국대학교의 관련 홈페이지에서 발췌한 도표이다.



<그림 III-6. Concurrent Standardization>



〈그림 III-7. 한국 IT 표준화 추진체계〉

## 제 4 장    요구 사항 분석

### 제 1 절 과제별 요구 내용 및 수행 사항

#### 가. Globus 설치

(1) Globus toolkit v2.2.4 or higher.

##### (가) 설치

① 다운로드

<http://www.globus.org/gt/2/install/download.html>에서 Globus Toolkit 2.2.5를 다운 받는다.

② Globus를 위한 계정 만들기

③ 설치전에 점검해야 할 사항

㉠ perl 버전 5.005이상이 설치되어 있어야 한다.

㉡ hostname 이 FQDN(Fully-Qualified Domain Name)으로 설정되어 있는지 확인한다.

④ globus 툴킷의 설치

㉢ globus 툴킷의 설치는 globus 계정으로 한다.

㉣ 압축을 해제할 디렉토리(globus의 홈디렉토리)하여 다음과 같은 환경변수를 설정한다.

```
$ export GLOBUS_LOCATION=/usr/local/globus
```

㉤ Globus 패키지 툴을 설치한다.

```
$ tar zxvf gpt-1.0.tar.gz
```

```
$ cd gpt-1.0
```

```
$ ./build_gpt
```

```
$ /usr/local/globus/sbin/globus-build -install-only bundle options  
flavors
```

Data Management    : Client, SDK, Server 번들

Information Services : Client, SDK, Server 번들

Resource Management : Client, SDK, Server 번들 설치

㉥ globus-build뒤의 설정을 한다.

```
$ /usr/local/globus/sbin/gpt-postinstall
```

㉦ GSI 소프트웨어의 setup을 위해 슈퍼 유저의 권한을 얻어 다음과 같이 GSI 설치 명령을 실행시킨다.

- ```
# /usr/local/globus/setup/globus/setup-gsi
```
- ㉞ globus package 설치가 제대로 되었는지 확인한다.
- ```
$ /usr/local/globus/sbin/gpt_verify
```
- ```
You have a coherent development tree
```
- ```
No packages are missing from your install tree
```
- 이 명령의 결과로 위와 같은 메시지가 나오면 설치과정은 끝난다.
- ⑤ 인정서 얻기 및 성공적인 설치 여부 확인
- ㉟ 사용자 계정에 환경 설정
- ```
$ export GLOBUS_LOCATION=/usr/local/globus
```
- ```
$ GLOBUS_LOCATION/etc/globus-user-env.sh
```
- ㊱ 유저 인증서 요청하기
- ㊲ host에 대한 gatekeeper 인증서 요청하기
- ㊳ CA의 인증서 설치하기
- 다음 명령을 실행하여 gatekeeper를 띄운다.
- ```
$ /usr/local/globus/bin/grid-proxy-init
```
- ```
$ /usr/local/globus/bin/globus-personal-gatekeeper -start
```
- ```
GRAM contact:
```
- ```
ccsgrid3.uos.ac.kr:52637:/O=Grid/O=Globus/OU=ccsgrid3/CN=ccsgrid3
```
- 다음 명령을 수행하여 정상적인 결과가 나오는지 확인한다.
- ```
$ /usr/local/globus/bin/globusrun -o -r W
```
- ```
"ccsgrid3.uos.ac.kr:52637:/O=Grid/O=Globus/OU=ccsgrid3/CN=ccsgrid3"
```
- ```
'&(executable=/bin/date)'
```
- ⑥ gatekeeper를 서비스로 등록하기
- ㉟ /etc/services 파일에 다음과 같이 추가한다.
- ```
globus-gatekeeper      2119/tcp      # Globus Gatekeeper
```
- ㊱ /etc/xinetd.d/ 디렉토리 안에 "globus-gatekeeper" 파일 만들기
- ```
service globus-gatekeeper{
socket_type  =    stream
protocol    =    tcp
wait        =    no
user        =    root
server       =    /usr/local/globus/sbin/globus-gatekeeper
server_args  =    conf /usr/local/globus/etc/globus-gatekeeper.conf
disable     =    no
}
```
- ㊲ xinetd을 재구동한다.
- ```
# /etc/rc.d/init.d/xinetd restart
```
- ㊳ /etc/grid-security/grid-mapfile에 인증서의 subject와 username을 추가해

야 한다.

```
"/O=Grid/O=Globus/OU=ccsgrid3/CN=ccsgrid3" globus
```

위의 작업의 결과로 정상적으로 동작하는지 확인한다.

```
$ grid-proxy-init
```

```
$ globusrun -o -r localhost '&(executable=/bin/date)'
```

## 나. Grid 소프트웨어 설치

### (1) MPICH-G2 v1.2.5

#### (가) 설치

<http://www.niu.edu/mpi/mpich>로부터 MPICH의 source file을 가져온다.

Globus가 정상적으로 실행되는지의 여부를 확인하고, GLOBUS\_LOCATION이라는 환경변수를 Globus가 설치된 directory를 가리키도록 설정한다.

MPICH file의 압축을 풀어내고, 필요한 경우 적절한 patch를 적용시킨다.

```
$ gunzip -c mpich.tar.gz | tar xvf -
```

```
$ patch -p0 < patch.all
```

Configure script를 실행한다.

이 때 MPICH device를 globus2로 지정하고 globus flavor를 추가로 기술한다.

```
$ ./configure --with-device=globus2:-flavor=gcc32dbg
```

make 와 make install을 순서대로 실행한다.

```
$ make
```

```
$ make install
```

### (2) gsissh bundle : Grid Testbed에 접속하기 위한 소프트웨어

#### (가) 설치

GPT(Globus Packaging Toolkit) (<ftp://ftp.globus.org/pub/gt2/2.2/2.2.4/gpt/>)

소스 파일 다운로드

GSI-SSH (<ftp://ftp.ncsa.uiuc.edu/aces/gssapi-openssh/bundle/2.5/>)

바이너리 파일 다운로드

### ① GPT 설치

다음과 같이 GT2가 설치될 호스트명이 FQDN(Fully-Qualified Domain Name)으로 등록되어 있는지 확인한다. 만약, 호스트명을 변경한 경우에는 네트워크를 다시 시작해준다.

```
#vi /etc/hosts
111.222.333.444 ccsgid3.uos.ac.kr
```

```
#hostname
ccsgid3.uos.ac.kr
```

```
#cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=ccsgid3.uos.ac.kr
GATEWAY=111.222.333.1
```

```
#/etc/rc.d/init.d/network restart
```

GT2가 설치될 디렉토리를 생성한다. GT2는 globus의 유저로 /usr/local/gt2에 설치되는 것으로 가정한다.

```
#mkdir /usr/local/gt2
#chown globus.globus /usr/local/gt2
```

GT2를 설치하기 위해 GPT\_LOCATION과 GLOBUS\_LOCATION의 환경변수를 설정해야 하며, 다음의 과정을 통해 설치한다.

```
#export GPT_LOCATION=/usr/local/gt2
#export GLOBUS_LOCATION=/usr/local/gt2
```

```
#tar xvzf gpt-2.2.5-src.tar.gz
#cd gpt-2.2.5
#./build_gpt
```

### ② GSI-SSH 설치

GSI-SSH를 설치하기 위해서 globus의 권한으로 다음의 과정을 통해 설치한다.

```
#!/usr/local/gt2/sbin/gpt-install W
gsi_openssh_bundle-2.5-i686-pc-linux-gnu-gcc32.tar.gz
#!/usr/local/gt2/sbin/gpt-postinstall (y) 선택
```

GSI를 설정하기 위해서 루트의 권한으로 다음과 같이 실행한다.

```
#/usr/local/gt2/setup/globus/setup-gsi 루트의 권한으로 실행 (y) (q) 선택
```

GSI-SSH패키지가 제대로 설치되었는지 확인하기 위해서 globus의 권한으로 다음과 같이 실행한다.

```
#/usr/local/gt2/sbin/gpt-verify
```

GSI-SSH를 설치하기 위한 모든 과정이 끝났으며, /usr/local/gt2/bin에 gsissh 및 gsiscp가 생성되어 있을 것이다.

루트CA를 설치하기 위해서 gridca.gridcenter.or.kr로부터 CA인증서 (x.0, x.signing\_policy)를 다운로드 받은 후, /etc/grid-security/certificates에 복사한다. x.0과 x.signing\_policy의 퍼미션을 다음과 같이 변경한다.

```
#chmod 644 x.0
```

```
#chmod 644 x.signing_policy
```

### ③ 테스트베드에 유저DN(Distinguished Name) 등록

테스트베드에 접속하기 위해서는 유저 인증서가 필요하다. gridca.gridcenter.or.kr에서 유저인증서 요청을 생성하고, 유저 인증서를 발급 받는 과정은 다음과 같다.

gridca.gridcenter.or.kr에 접속한다. GridCA의 로그인ID를 생성한다. 패스워드는 메일로 확인이 가능하다. 로그인ID와 패스워드로 GridCA에 로그인한다. 메뉴에서 “Generate a new CSR for an user”를 선택한다. CSR생성 폼에서 Organization Unit Name과 Common Name을 입력한다. (Organization Name1과 Organization Name2는 변경하지 않는다) Pass Phase를 위한 패스워드를 입력한다. (패스워드는 프록시를 생성할 때, 암호화된 유저 비밀번호를 복호화하는데 사용된다) Generate 버튼을 누른다.

CSR메뉴를 클릭하면 방금 생성한 유저 인증서 요청의 CSR과 비밀번호를 확인할 수 있다. GridCA 관리자에 의해서 유저 인증서 요청이 승인되면, CERT메뉴로부터 유저 인증서와 비밀번호를 확인할 수 있다.

유저 인증서(Certificate)와 비밀키(Private Key)를 각각 usercert.pem과 userkey.pem으로 복사한 다음 ~/.globus 디렉토리에 복사해 준다. ~/.globus 디렉토리가 없을 경우, 만들어 준다. usercert.pem과 userkey.pem의 퍼미션을 다음과 같이 변경한다.

```
#chmod 644 usercert.pem
```

```
#chmod 400 userkey.pem
```

유저 인증서의 DN과 IP를 테스트베드 사용 신청서에 적어서 테스트베드 관리자에게 메

일로 보내면 테스트베드의 grid-mapfile에 등록되어 테스트베드에 접속할 수 있게 된다.  
유저 인증서의 DN은 다음과 같이 확인 가능하다.

```
#grid-cert-info -subject  
O=Grid, O=Globus, OU=ccsgrid3, CN=ccsgrid3
```

#### ④ 테스트베드에 접속

GSI-SSH의 경로를 지정하기 위해서 다음과 같이 환경 변수를 추가한다.

```
#vi .bashrc  
export GLOBUS_LOCATION=/usr/local/gt2  
export GPT_LOCATION=/usr/local/gt2  
. $GLOBUS_LOCATION/etc/globus-user-env.sh
```

```
#vi .bash_profile  
PATH=$GLOBUS_LOCATION/bin:$GLOBUS_LOCATION/sbin:$PATH  
export PATH
```

```
#. .bashrc  
#. .bash_profile
```

유저 프록시를 생성한 다음 테스트베드에 접속하기 위해서 다음과 같이 실행한다. 테스트베드에 접속후에는 로그인 공지사항을 반드시 확인하도록 한다.

```
#grid-proxy-init  
#gsissh -p 2222 ccsgrid4.uos.ac.kr  
.  
. 사용중  
.
```

테스트베드로부터 로그아웃한 후에는 보안을 위해서 다음과 같이 유저 프록시를 반드시 삭제한다.

```
#grid-proxy-destroy
```

\* gsincftp bundle.

## 다. 방화벽 설치

방화벽은 학교자체의 방화벽을 사용했고, port는 아래와 같이 open 했다.

2119: globus gatekeeper  
2135: MDS query  
2222: gsissh  
2811: gridFTP  
20000 ~ 25000: MPICH-G2 & GLOBUS\_TCP\_PORT\_RANGE

## 라. 사용자 계정 발급

globus, jhkwak, voxel, giljael, kis001 ~ kis005, ngrid001~ngrid011 총 20개의 계정을 발급했다.

## 마. SCMSWEB 설치

# : root shell

\$ : scms shell

(1) 설치파일 및 메뉴얼 다운로드

<http://hpcnc.cpe.ku.ac.th/moin/InstallingSCMSWeb> or

[http://testbed.gridcenter.or.kr/software/SCMSWeb/kisti\\_scmsweb2.3.tar.gz](http://testbed.gridcenter.or.kr/software/SCMSWeb/kisti_scmsweb2.3.tar.gz)

(2) Python, PyXML, Tcl, Tk, libjpeg and zlib

#rm /usr/bin/python

#ln -s /usr/bin/python2 /usr/bin/python

(3).Installation Configuration

(가)압축해제

\$vi tar.sh

#!/bin/sh

gunzip=gunzip

tar=tar

\$gunzip -c -d Imaging-1.1.1.tar.gz | \$tar xf -

\$gunzip -c -d Pmw.0.8.5.tar.gz | \$tar xf -

\$gunzip -c -d scebase-1.0.tar.gz | \$tar xf -

\$gunzip -c -d libhal-2.4.tar.gz | \$tar xf -

\$gunzip -c -d scms-2.3.1.tar.gz | \$tar xf -

\$gunzip -c -d PyRRDtool-0.1.tar.gz | \$tar xf -

\$gunzip -c -d rrdtool-1.0.40.tar.gz | \$tar xf -

\$gunzip -c -d scmsweb-2.3.tar.gz | \$tar xf -



(나)환경변수 설정(클러스터 환경에 맞게 수정한다.)

```
$vi environ
prefix=/home/scms/scmsweb-2.3
python=/usr/bin/python2
cluster_url=http://seoulgrid.uos.ac.kr
email=squad@uos.ac.kr
grid_name=ccsgrid
org_name=UOS
cluster_name=ccsgrid pysite_dir=$prefix/lib/pythonx.X/site-packages (replace
X.X with python version which greater than 2.1)
html_dir=$HOME/public_html
html_url=/~scms/
cgi_dir=$HOME/public_html/cgi-bin
cgi_url=/~scms/cgi-bin
mds_host=ngrid-mds.gridcenter.or.kr
mds_basedn=mds-vo-name=ngrid-mds,o=grid
probe_dir=$prefix/share/scmsweb/grid
timeout=30
```

(4) Preparation

```
$cd kisti-scmsweb-2.3
$./tar.sh
$source environ
```

```
$mkdir -p $prefix ; mkdir -p $pysite_dir
$export PYTHONPATH=$pysite_dir:$pysite_dir/PIL
$export PATH=$PATH:$prefix/bin:$prefix/sbin
```

(5) Python Imaging Library

```
$cd Imaging-1.1.1
$patch -p0 < ../Imaging-1.1.1.patch ; ./BUILD
$cp -Rp PIL $pysite_dir
$cp -p _imaging.so _imagingtk.so PIL.pth $pysite_dir ; cd ..
```

(6) Python Mega Widget

```
$cp -Rp Pmw $pysite_dir
```

(7) libsce

```
$cd scebase-1.0
$./configure --prefix=$prefix
$make ; make install ; cd ..
```

(8) libhal

```
$cd libhal-2.4
```

```
libhal-2.4/src/x86-linux/.libs/libhal.lai에서
libdir='/home/users/staff/ssy/software/lib'
==>libdir='/home/scms/scmsweb-2.3/lib'으로 변경
```

```
libhal-2.4/src/x86-linux/.libs/libhal.la에서
libdir='/home/users/staff/ssy/software/lib'
==>libdir='/home/scms/scmsweb-2.3/lib'으로 변경
```

```
libhal-2.4/src/x86-linux/libhal.la에서
libdir='/home/users/staff/ssy/software/lib'
==>libdir='/home/scms/scmsweb-2.3/lib'으로 변경
```

```
./configure --prefix=$prefix
$make ; make install ; cd ..
```

(9) scms

```
$cd scms-2.3.1
$./configure --prefix=$prefix --with-libsce=$prefix --with-libhal=$prefix
--with-initrddir=$prefix/etc/init.d --sysconfdir=$prefix/etc
--with-piddir=$prefix/var/run --with-python=$python
$make ; make install ; cd ..
```

(10) RRDtool

```
$cd rrdtool-1.0.40
$./configure --prefix=$prefix
$make ; make install ; cd ..
```

(11) PyRRDtool

```
$cd PyRRDtool-0.1
$patch -p0 < ../PyRRDtool-0.1.patch
```

```
$sed -e "s%@RRDTOOL_PREFIX@%$prefix%g" setup.py > psetup.py
$python psetup.py build
$cp build/^(cd build;ls|grep lib)`/RRDtoolmodule.so $pysite_dir ; cd ..
```

#### (12) SCMSWEB

```
$cd scmsweb-2.3
$./configure --prefix=$prefix --with-htmldir=$html_dir
      --with-htmlurl=$html_url --with-cgidir=$cgi_dir --with-cgiurl=$cgi_url
      --with-python=$python
$make ; make install
```

#### (13) 설정 변경

\$prefix/bin/sce\_rcp, sce\_rsh 을 시스템에 맞게 scp, ssh등으로 변경  
 /home/scms/public\_html/cgi-bin/graph\_gen.cgi 의 tool부분을  
 tool = '/home/scms/scmsweb-2.3/bin/rrdtool'로 변경

#### (14) sce.conf 변경

kisti-scmsweb-2.3/Sample.sce.conf 참조

#### (15) Test

##### (가) PATH추가

PATH에 \$HOME/scmsweb-2.3/bin, \$HOME/scmsweb-2.3/sbin을 추가한다.

##### (나) rms 테스트

```
$scms_start
ju007: Starting rms on 150.183.249.207 : [ OK ]
[생략]
```

```
$ps -ax | grep rms
7145 ?          S      0:00 /home/scms/scmsweb-2.3/sbin/rms
[생략]
```

```
$sce_host
```

```
ccsgrid3 node301 node302 node303 node304 node305 node306 node307 node308
node309 ccsgrid4 node401 node402 node403 node404 node405 node406 node407
```

```
$sce_rms
```

host	idle	nice	pid	system	users
jupiter	84.0	0	0	4.5	11.4

ju001                    0.0            0        0        69.3        30.6

[생략]

(16) xml파일 만들기

\$scmsweb-cs

\$

(17) 웹페이지 설정

@웹서버 실행시키기(Listen 24580으로 설정)

\$chmod 711 /home/scms/public\_html

\$chmod 755 /home/scms/public\_html/cgi-bin

/home/scms/public\_html/cgi-bin/graph\_gen.cgi 의 tool부분을  
rrdtool이 있는 경로로 변경.

예)tool = '/home/scms/scmsweb-2.3/bin/rrdtool'

(18)

<http://monitor.gridcenter.or.kr/>

## 바. grid-mapfile 링크

#

# KISTI : /home/kisti/kis\*\*\*

#

# kis001 :

# kis002 : ChoongNam university, 168.188.48.{185,219}

# kis003 : DHCP 15.183.234.160 ~ 175

# kis004 : 150.183.234.105

#

"/O=Grid/O=Globus/OU=hpcnet.ne.kr/CN=Hongduk Yi at KISTI" kis001

"/O=Grid/O=Globus/OU=cnu.ac.kr/CN=globuslinux" kis002

"/O=Grid/O=Globus/OU=hpcnet.ne.kr/CN=KumWon Cho" kis003

"/O=Grid/O=Globus/OU=gridcenter.or.kr/CN=Sangwan Kim" kis004

"/O=Grid/O=Globus/OU=supercomputing.re.kr/CN=smlee" kis005

```

#
# Researchers from partners of KISTI : /home/ngrid/ngrid***
#
#  ngrid001 : POSTECH, 141.223.165.{131,231,145,148}
#  ngrid002 : Seoul national university, 147.46.118.96
#  ngrid003 : KAIST
#  ngrid004 : Pusan national university, 164.125.130.167
#  ngrid005 : Sookmyung women's university, 203.252.201.20
#  ngrid006 : Pusan national university
#  ngrid020 : Super computing, KAIST 143.248.8.31
#  ngrid021 : Seoul national university
#

"/O=Grid/O=Globus/OU=postech.ac.kr/CN=KSC" ngrid001
"/O=Grid/O=Globus/OU=snu.ac.kr/CN=floydfan" ngrid002
"/O=Grid/O=Globus/OU=kaist.ac.kr/CN=kaistCFD" ngrid003
"/O=Grid/O=Globus/OU=gridcenter.or.kr/CN=cfdkwon" ngrid003
"/O=Grid/O=Globus/OU=me.pusan.ac.kr/CN=sdhong" ngrid004
"/O=Grid/O=Globus/OU=sookmyung.ac.kr/CN=grid1" ngrid005
"/O=Grid/O=Globus/OU=grid.pusan.ac.kr/CN=gridnode1" ngrid006
"/O=Grid/O=Globus/OU=SNU/CN=dcslab" ngrid007
"/O=Grid/O=Globus/OU=kaist.ac.kr/CN=globus_at_wwf" ngrid008
"/O=Grid/OU=GlobusTest/OU=simpleCA-dss3.grid.or.kr/OU=grid.or.kr/CN=nano
tech" ngrid009

```

## 제 2 절 필요한 기능

본 과제는 지역적으로 분산되어 있는 클러스터들을 그리드 환경으로 엮는 것을 목적으로 하고 있다. 그러기 위해서는 앞에서 언급한 것과 같이 기존의 사용 방식과 사용 어플리케이션 프로그램들을 그대로 사용할 수 있게 하면서 편리하게 그리드로 전환하여 주는 기술과 소프트웨어가 필요하다. 즉 다음과 같이 제 1항에서 요구하는 사항을 충족시켜 주 있는 기능이 필요하다.

가. 그리드를 적용함으로써 발생하는 사용자 계정 관리 부담이 거의 없어야 한다.

나. 자원을 보다 원할 하게 통합되어 운영을 하려면 자원 데이터 스키마(schema)를 XML로 만들어 보다 융통성 있게 사용할 수 있도록 만들어야 한다.

## 제 3 절 문제점

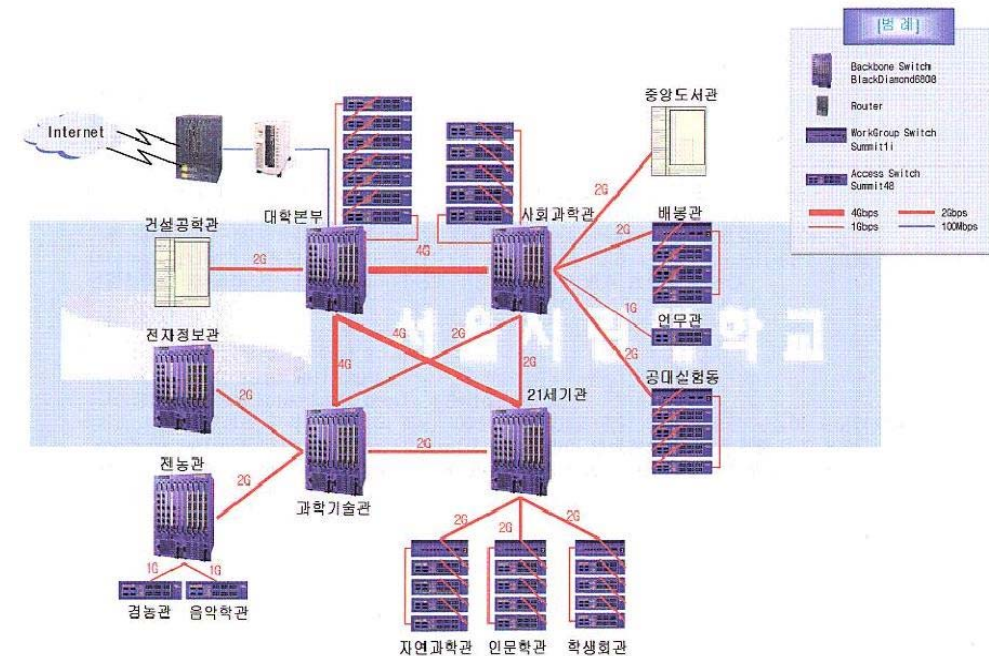
기존 기술 및 소프트웨어의 연구 및 개선이 필요하며 새로운 기술과 소프트웨어의 개발이 필요하다. 그리드 기술은 매우 복잡한 기술이다. 현재 나와 있는 기술과 소프트웨어를 가지고 기존의 클러스터들을 묶어서 그리드 환경을 구축하는 것은 용이하지가 않다. 클러스터에서 클러스터 그리드로 전환하는 데에도 역시 많은 노력이 요구된다. 우리도 본 과제를 수행하는데 있어서 사용된 그리드 프로그램인 글로버스는 현재 구현된 기술은 그리드가 추구하는 기술 중 일부분 구현 되었지만 그래도 설치 및 설정을 하는 데 몇 배의 노력이 들어갔다. 보다 많은 사람이 그리드를 이용하려고 하면 보다 쉽고 간단한 그리드 구축 프로그램이 필요하다고 생각된다.

본 과제에서 사용된 프로그램은 글로버스와 보다 상위 레벨 프로그램이 MPI-G2이다. MPI-G2는 클러스터에서 사용된 MPI 프로그램을 완벽히 지원하지 못하는 것 같다. 좀 더 개선을 해야, 고도의 정밀을 요구하는 계산에 그리드가 적용될 수 있을 것 같다.

## 제 5 장 서울 그리드 시스템

### 제 1 절 시스템 설계 및 구축

클러스터 그리드의 구축을 위해서는 우선 로컬 인트라넷 상태에서의 성능 평가가 선행되어야 한다. 일반적으로 그리드[18]는 지역적으로 분산되어 있는 자원들이 서로 통신을 하기 때문에 각 자원들 간의 통신 속도가 전체 컴퓨팅 성능에 큰 영향을 끼친다. 서울시립대학교의 경우 교내 네트워크가 기가비트 이더넷으로 구성되어 있기 때문에, 빠른 전송속도를 기반으로 한 클러스터 그리드 환경의 구축에 적합하다.[19] <그림 V-1>에 시립대학교의 교내 네트워크 구성도를 보였다.



<그림 V-1. 서울시립대학교 네트워크 구성>

클러스터 그리드 환경은 크게 2가지 방식으로 구성할 수 있다. 하나는 LSF[20]나 PBS[21]와 같은 자원 관리 시스템으로 작업 및 자원을 관리하는 클러스터 시스템의 마스터 노드에 글로버스[22]를 설치하여 그리드 환경을 구성하는 것이고, 다른 하나는 PC와 같은 단일 시스템에 글로버스를 설치하여 그리드 환경을 구성하는 것이다. 전자의 경우 이미 클러스터 시스템으로 사용 중인 시스템을 글로버스를 이용하여 쉽게 그리드 환경으로 구성할 수 있는 장점이 있으며, 후자의 경우 클러스터 시스템에 속하지 않는 보통의 PC들을 클러스터로 구축할 수 있게 하고, 동시에 그리드 환경으로도 구성할 수 있게 하는 장점을 가지고 있다. 또한 후자의 경우 Peer to Peer 방식으로 그리드 환경에 참여하게 되기 때문에, 각 PC가 독립적으로 자신의 자원을 다른 사용자에게 제공할 수도 있으며 자신이 다른 시스템의 자원을 사용할 수도 있다.

본 연구에서는 전자와 후자의 시스템 구성 방식을 구별하기 위해 전자의 경우 클러스터 그리드 컴퓨팅 시스템, 후자의 경우에는 P2P(Peer to Peer) 클러스터 그리드 컴퓨팅 시스템 또는 P2P 그리드 컴퓨팅 시스템이라는 호칭을 사용하였다.

<표 V-1>은 사용한 시스템들에 관한 정보 중 프로토타입 I에 대한 정보를 보여주고 있다. 여기 보여주는 49대의 PC들은, 시스템A의 39노드와 시스템B의 10개 노드로 구성되어 있다. 두 장소의 PC들 모두 Pentium 4-1.7Ghz급의 성능을 가지며, 시스템A가 시스템B보다 메모리와 네트워크에서 더 나은 성능을 보이고 있다. 다양한 구성 방식별로 테스트를 진행하기 위하여 시스템A는 클러스터 그리드 컴퓨팅 시스템 방식으로 시스템B는 P2P 그리드 컴퓨팅 시스템 방식으로 구축하였다. 본 연구에서 구축한 프로토타입 I시스템은 현재 서울 그리드 연구소(Seoul GRID Institute)의 테스트베드의 일부로 활용되고 있다.

<표 V-1. 프로토타입 I의 하드웨어 구성>

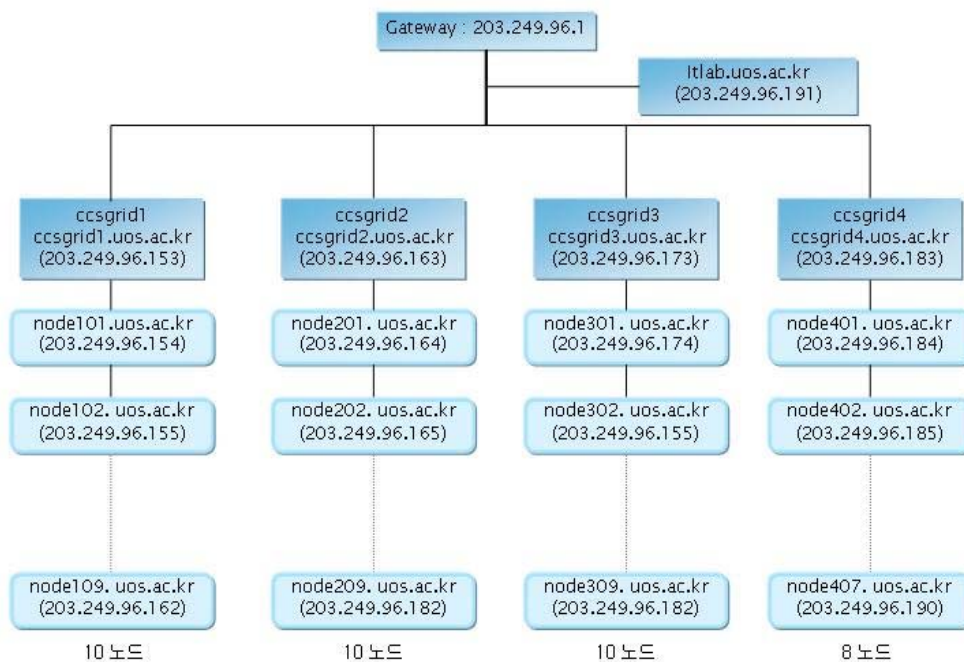
구분	시스템 A	시스템 B
위치	서울시립대 본관	서울시립대 전자정보관
노드 수	39	10
프로세서	P4-1.7Ghz	P4-1.7Ghz
메모리	512MB	256MB
디스크	10GB	10GB
OS	Redhat Linux 7.3	Redhat Linux 7.3
네트워크	Gigabit Ethernet	Fast Ethernet

<그림 V-2>는 프로토타입 I의 시스템 구성도를 보여 주고 있다. 시스템A의 경우 10노드씩 3그룹의 클러스터와 8노드의 1그룹 클러스터로 구축하여 보았다. 또한 하나의 PC는 외부 접속을 허용하는 노드로써 프로그램 컴파일 및 실제 프로그램을 수행하기 위해 사용되어지며, 각 클러스터는 계산 자원만을 제공한다. 시스템B의 경우 각 노드가 독립된 시스템으로 그리드 환경을 구성하며, 각 노드에서의 프로그램 컴파일 및 그리드 환경에서의 프로그램 실행이 가능하다. 각 장소에 따른 자세한 시스템 구성도를 <그림 V-3>과 <그림 V-4>에 나타내었다.



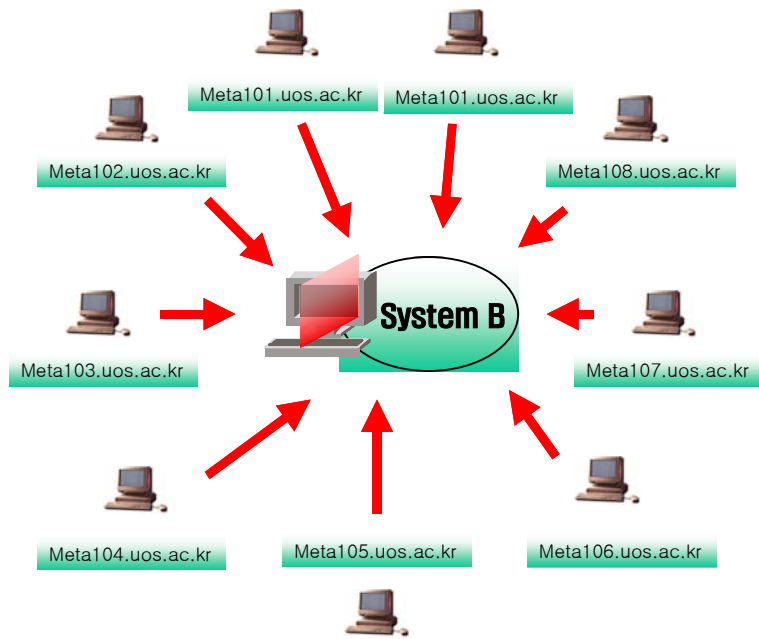


<그림 V-2. 프로토타입 I의 구성도>



<그림 V-3. 시스템A 구성도>

<표 V-2>에는 그리드 환경을 구축하는데 사용한 Software를 나타내었다. 시스템A의 클러스터의 경우 각 클러스터에는 오픈 소스인 OpenPBS[23]를 설치하여 작업을 관리하도록 하였으며, 마스터 노드에 글로버스를 설치하여 클러스터 그리드 환경을 구축하도록 하였다. 글로버스를 통하여 들어오는 그리드의 작업은 클러스터 내부에서는 PBS의 관리를 받게 된다. 시스템B의 경우 각 노드가 단일한 시스템으로 독립되도록 각 노드별로 글로버스를 설치하여 P2P 그리드 컴퓨팅 환경을 구성하였다. 각 노드에는 별도의 Job Manager를 사용하지 않았다.



<그림 V-4. 시스템B 구성도>

<표 V-2. 프로토타입 I 시스템의 소프트웨어 구성>

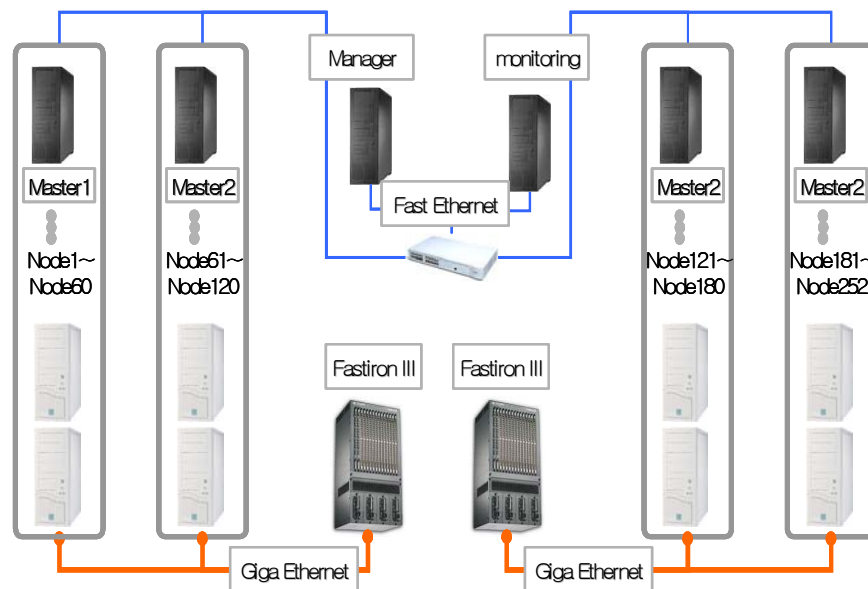
시스템명칭	시스템 A					시스템B
	ccsgrid1	ccsgrid2	ccsgrid3	ccsgrid4	itlab	globus
Local Job Scheduler	OpenPBS				fork	
통신 라이브러리	MPICH-G2(v1.2.5) (마스터 노드에만 설치)				MPICH-G2	
그리드 미들웨어	globus 2.2.4 (마스터 노드에만 설치)				globus 2.2.4	
기타 설치 S/W	gsissh, gsincftp				gsissh, gsincftp	

메시지패싱 방식의 병렬 프로그램 라이브러리인 MPI를 사용하기 위하여 글로버스가 설치된 시스템마다 MPICH-G2[24]를 설치하였다. MPICH-G2는 글로버스로 구축된 그리드 환경에서 MPICH의 사용을 가능하게 한 라이브러리로서, 기존의 MPI로 짜여진 코드의 경우 MPICH-G2로 다시 컴파일하는 과정을 거쳐서 그리드 환경에서 프로그램을 수행할 수 있게 한다.

## 제 2 절 서울 그리드 슈퍼 클러스터 시스템



<그림 V-5. 서울그리드 256노드 슈퍼 클러스터 시스템>



<그림 V-6. 슈퍼 클러스터 시스템 구성도>

위 그림은 2003년 1월 완성된 서울 시립대 256 node 슈퍼 클러스터 시스템의 구성도이다. 이 슈퍼 클러스터는 4개의 클러스터를 하나의 클러스터로 돌아갈 수 있도록 구성되어 있다. 2003년 6월에 발표한 슈퍼컴퓨터 톱500 순위에서 392위를 차지하였다.

<표 V-3. 슈퍼 클러스터 시스템의 구성>

구분	슈퍼 클러스터 시스템
위치	서울시립대 21세기관
노드 수	256
프로세서	P4-2.0Ghz
메모리	Master:1GB node:512MB
디스크	Master: 80GB
OS	Redhat Linux 7.3
네트워크	Gigabit Ethernet
Local Job Scheduler	LSF
통신 라이브러리	MPICH-G2
그리드 미들웨어	globus 2.2.4
기타 설치 S/W	gsissh, gsincftp

## 제 6 장 결론

### 제 1 절 결론

KISTI의 전체적인 진행 상황에 맞추어서 국가그리드 테스트베드가 국내 최초로 본 연구에서 구현되고 성공적으로 가동되었다. 정통부에서 KISTI를 거쳐서 발주한 여러 그리드응용 연구가 이 국가그리드 테스트베드를 사용하여 좋은 연구결과를 얻었다. 이 연구 결과들은 2003년 12월 1일부터 2일까지 서울의 조선포털에서 열린 2003 Winter Grid Forum Korea Workshop에서 일반에게 공개되었다.

서울시립대학교에 위치한 서울그리드센터는 서울시립대학교에 구축된 Peer to Peer 방식의 서울그리드 테스트베드 중에서 16대를 할애하여 참여하였다. 가장 안정적인 서비스를 제공하였다는 자부심은 있으나 국가그리드 테스트베드에 접속하는 연결 네트워크의 대역폭이 낮아서(90Mbps) 고성능 컴퓨팅을 하기에는 미흡한 점이 많이 있었다. 서울그리드센터는 이점을 충분히 예견하고 KISTI를 통하여 지난 1년 이상을 반복하여 국가초고속 선도망과 연구전산망을 연결하여 줄 것을 서면과 구두로 신청하였다. 아직까지 알 수 없는 이유로 우리의 신청이 실현되지 않고 있어서 이와 같은 문제점이 생겼다. KISTI는 우리의 신청사항을 조속히 반영할 수 있도록 협조하여 주기를 본 연구보고서에서 다시 한번 요청하고 본 기관도 계속적으로 관계요로에 요청을 할 것이다.

### 제 2 절 향후 연구 방향

국가그리드 테스트베드 구축과 별개로 서울그리드센터는 서울그리드 테스트베드를 구축하고 있다. 이번 연구까지 포함하여 2년의 기간동안 연구가 진행되어 1단계 캠퍼스 테스트베드 구축 사업은 많은 진전을 보았다. 2단계의 외부 기관들과의 연결 연구도 본 연구로 여러 현안 문제들을 해결하는 진전을 보았다. 예를 들면 방화벽 문제 등이 그것이다. 따라서 내년도부터는 2단계 서울그리드 테스트베드 구축계획을 본격적으로 수행하고자 한다.

그 후는 본 연구보고서에서 설명한 바와 같이 5단계에 걸쳐서 후속 연구를 차근차근 수행할 예정이다. 이를 통하여 세계제일의 서울그리드 테스트베드를 구축 할 예정이며, 이 서울그리드 테스트베드는 KISTI에서 추구하는 국가그리드 테스트베드의 주요한 부분으로 국가그리드 발전에 기여할 것이다.

## 참고문헌

- [1] <http://www.teragrid.org>
- [2] <http://www.eurogrid.org>
- [3] <http://www.grid-support.ac.uk>
- [4] <http://www.cs.vu.nl/das>
- [5] <http://www.csse.monash.edu.au/~raj कुमार/ecogrid/wwg>
- [6] <http://www.science.uva.nl/research/scs/SCS4.html>
- [7] <http://www.apbionet.org>
- [8] <http://www.ipg.nasa.gov>
- [9] <http://www.npaci.edu/Thrusts/Metasystems/index.html>
- [10] <http://prg.cpe.ku.ac.th/thaigrid>
- [11] <http://www.ncsa.uiuc.edu/SCD/Alliance/VMR>
- [12] <http://www-csag.ucsd.edu/projects/grid/microgrid.html>
- [13] <http://archive.ncsa.uiuc.edu/alliance/alliance/GridTech.html>
- [14] <http://l59.csm.port.ac.uk/dsgPortal/about/about.html>
- [15] <http://www.hlr.de/people/resch/PROJECTS/GWAAT.html>
- [16] <http://www.imp.org>
- [17] <http://www.cs.tcd.ie/grid-ireland>
- [18] I. Foster, C. Kesselman, S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International J. Supercomputer Applications, 15(3), 2001.
- [19] 이영석, 박동국, 윤창일, 이용우, "Intranet 구성 요소가 데이터 전송

성능에 미치는 영향에 대한 연구”, 한국정보처리학회 춘계학술발표논문집, 제9권, 제1호, pp.1269-1271, 2002.

[20] LSF Homepage: <http://www.platform.com>

[21] PBS Homepage: <http://www.pbspro.com>

[22] Globus Project: <http://www.globus.org>

[23] Open PBS Homepage: <http://www.openpbs.org>

[24] MPICH-G2 Homepage: <http://www3.niu.edu/mpi/>