

사이트 가격 정책 기반의 그리드 어카운팅 모델 설계

황 호 전[†] · 안 동 언^{**} · 정 성 종^{***}

요 약

그리드 자원의 안정적인 수요와 공급이 뒷받침되기 위해서는 유틸리티 모델을 지향하는 그리드 어카운팅 모델이 필수적이다. 현재 개발된 대부분의 그리드 어카운팅 시스템들은 로컬 어카운팅 시스템을 변경함으로써 사이트의 자율성을 침해하거나, 사이트의 가격 정책에 따른 가변 비용을 고려하지 않은 채로 그리드 서비스에 대한 과금을 산출하고 있다. 본 논문에서는 그리드 컴퓨팅 환경에 참여하는 사이트의 자율성을 보장하면서, 다양한 과금 정책에 능동적으로 대처할 수 있는 사이트 가격 정책 기반의 그리드 어카운팅 모델을 제안한다. 또한 그리드에 참여하는 엔티티들간에 어카운팅 정보를 상호 교환할 수 있도록 GGF-UR 포맷으로 변환하여 Resource Usage Service가 가능하도록 제안한다. 그리고 본 논문에서 제안하는 그리드 어카운팅 모델은 기존의 어카운팅 시스템들에 비해 사이트의 자율성을 보장하면서 정교한 과금 정책을 펼 수 있는 비즈니스 모델로 평가된다.

키워드 : 그리드 어카운팅, 과금 정책, 자원 사용량

Design of Grid Accounting Model Based on Site Price Policy

Ho-Joen Hwang[†] · Dong-Un An^{**} · Seung-Jong Chung^{***}

ABSTRACT

Grid accounting model is necessary to support the demand and supply of grid resource. Most grid accounting systems currently in use intrude upon site autonomy by modifying local accounting systems or calculate the cost of grid service without regard to site price policy. In this paper we propose and design Grid accounting model based on site price policy. This model assures autonomy of sites participating in grid computing and be able to actively cope with diverse billing services. Also this proposed model enables to provide RUS to Grid entities by transforming basic accounting information into GGF-UR format and allows this entities to exchange resource usage information. In this paper, proposed Grid accounting model enables sites to have autonomy of them and is evaluated for business model to enforce elaborate charging policy, compared with other systems.

Key Words : Grid Accounting Model, Charging Policy, Resource Usage

1. 서 론

그리드[1-3]는 지리학적으로 분산되어 있는 다양한 컴퓨팅 자원들을 통합하는 메타 컴퓨팅 환경을 제공하는 분산 및 병렬 컴퓨팅 분야의 새로운 개념이다. WWW은 단순히 정보 공유만을 제공하는데 비해, 그리드는 프로세서, 메모리, 어플리케이션 그리고 데이터 및 대용량 저장장치등 다양한 자원들을 공유하여 고성능 메타 컴퓨팅 환경을 제공한다. 따라서 그리드는 단일 머신상에서 처리할 수 없는 복잡하고 거대한 형태의 문제를 해결할 수 있어 바이오, 천문학, 기상, 인공지능 등 고속 연산 및 대용량 데이터 처리가 요구되는 분야에서 널리 활용되고 있다.

그리드 컴퓨팅 환경은 단일 시스템 이미지, 대용량 저장 장치, 데이터베이스, 어플리케이션 그리고 특수 목적의 컴퓨팅 장치 등 다양한 형태의 자원들을 제공하는 자원 제공자와 이를 활용하는 자원 소비자로 구성된다. 자원 제공자와 자원 소비자는 각자의 목적을 지니고 그리드 컴퓨팅 환경에 참여한다. 자원 제공자 측면에서는 고성능 컴퓨팅 자원들을 사용자에게 공유하여 유휴 자원의 활용률을 향상시킬 수 있고, 자원 소비자 측면에서는 단일 시스템상에서 처리할 수 없는 문제를 그리드 컴퓨팅 자원들을 사용함으로써 해결할 수 있게 된다.

이와 같이, 그리드 서비스를 상업화하기 위해서는 유틸리티 서비스가 필수적인 요소이다. 자원 소비자는 그리드에서 제공하는 강력한 컴퓨팅 파워를 사용할 수 있고, 자원 제공자는 그리드 서비스에 대한 처리 비용을 지불받게 된다. 그러함으로써 자원 제공자는 유휴 자원의 활용률을 높일 수 있고, 그리드에 안정적인 자원을 공급할 수 있게 된다. 따라

※ 본 연구는 전북대학교 정보검색시스템연구센터 지원으로 수행되었음.

† 준 회 원 : 전북대학교 컴퓨터공학과 박사과정

** 중 심 회 원 : 전북대학교 전자정보공학부 부교수

*** 정 회 원 : 전북대학교 전자정보공학부 교수

논문접수 : 2005년 9월 26일, 심사완료 : 2005년 12월 12일

서 그리드 서비스에 대한 자원 사용량을 측정하고, 다양한 가격 정책에 따라 사용자의 처리 비용을 산출할 수 있는 그리드 어카운팅 모델이 필요하다.

그리드 어카운팅 모델을 통해 유틸리티 컴퓨팅 모델[4, 5]을 실현하기 위해서는 그리드 컴퓨팅 자원을 사용할 때 발생하는 비용 처리를 위한 사용자 계정 서비스가 필요하다. 사용자 계정 서비스가 이루어지기 위해서는 어카운팅, 요금 부과 알고리즘 및 बैं킹 서비스 등도 함께 제공되어야 한다. 어카운팅은 사용자의 작업을 처리하는데 어떠한 종류의 자원들을 얼마만큼 사용했는지를 측정한다. 요금 부과 알고리즘은 어카운팅 관련 자료들을 분석해 사이트 가격 정책에 따라 과금을 책정한다. 그리고 बैं킹 서비스는 자원 제공자와 소비자간의 온라인 과금 결제 서비스를 제공한다. 따라서 그리드 어카운팅 모델은 유틸리티 컴퓨팅 모델을 지향하게 하는 출발점이 된다.

본 논문에서는 그리드에 참여하는 모든 노드에 로컬 어카운팅 시스템을 설치하여 그리드 서비스에 대한 자원 사용량을 측정한다. 그리고 각각의 그리드 노드에 설치된 로컬 어카운팅 시스템으로부터 측정된 어카운팅 정보를 사이트 가격 정책에 따라 처리 비용을 산출함으로써 유틸리티 컴퓨팅 모델을 지향하는 그리드 어카운팅 모델을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 그리드 어카운팅 시스템 및 그리드 어카운팅 표준화에 관련된 워킹그룹들을 살펴보고, 3장에서는 유틸리티 모델을 지향하는 사이트 가격 정책 기반의 그리드 어카운팅 모델을 설명한다. 4장에서는 실험 환경 및 실험 결과를 기술하고, 마지막으로 5장에서 결론 및 향후 연구에 대해 기술한다.

2. 관련 연구

현재 그리드에 관련된 기술 표준화는 GGF(Global Grid Forum)[6]와 GFK(Grid Forum Korea)[7]등에서 주도적으로 이끌어가고 있다. GGF는 그리드 미들웨어, 그리드 자원 관리, 그리드 보안, 그리드 서비스 구조 등 다양한 분야의 워킹그룹들이 존재하며, 특히 그리드 어카운팅 표준화에 관련된 어 활동중인 워킹그룹들도 존재한다.

GGF UR-WG(Usage Record Working Group)[8]은 그리드에 참여하는 각 사이트들간의 자원 사용량 정보를 교환하기 위해 공통된 포맷의 Usage Record를 정의하였다. GGF RUS-WG(OGSA Resource Usage Service Working Group)[9]에서는 자원 사용량 정보를 필요로 하는 자원 제공자, 사용자, 그리드 리소스 브로커 또는 बैं킹 서비스에게 각 사이트에서 그리드 작업을 처리하는데 드는 자원 사용량 데이터를 표준 XML 문서로 제공하는 Resource Usage 서비스를 정의하였다. 그리고 GSAX(Grid Service Accounting Extensions)[10]에서 그리드 어카운팅 시스템 프레임워크를 제안하였다. GESAWG(Grid Economic Services Architecture Working Group)는 자원 소비자로부터 OGS(Open Grid Services Infrastructure) 기반 그리드 서비스 사용에 대한 요금을 청

구하는 경제 기반의 서비스 모델로 확장하였으며, 이를 위해서 Resource Usage 서비스와 합의 계약에 기반을 둔 아키텍처를 제안하였다.

DataGrid 프로젝트에서 개발된 DGAS(DataGrid Accounting System)[11]는 Economic Grid Market 개념을 도입하였다. 그리드 리소스 브로커가 사용자의 작업을 처리하는데 드는 비용을 예측하여 적절한 가격의 컴퓨팅 엘리먼트에 작업을 제출한다. 작업을 넘겨받은 해당 사이트에서는 사용자의 작업을 실행시키고, 주기적으로 자원 사용량을 모니터링하여, 이 정보를 사용자에게 전달한다. 사용자측은 Job Cost Computation 루틴을 통해 처리 비용을 계산한다. 최종적으로 작업이 완료되면, 사용자의 전체 비용을 산출하여 자원 제공자와 사용자간의 GridCredit를 결제하는 어카운팅 시스템 아키텍처를 제안하였다.

스웨덴 국가 그리드, SweGrid에서 개발된 SGAS(SweGrid Accounting System)[12]는 각 로컬 사이트에서 사용자의 작업을 처리하기 전에 사용자의 계좌 잔액에 근거한 Time-limited Hold를 획득한 다음, 작업이 완전히 처리되면 해당 작업의 자원 사용량을 수집하여 VO(Virtual Organization) Bank에게 사용자의 과금을 처리하도록 하였다.

그러나 DGAS나 SGAS는 사용자 작업의 처리 비용을 산출하는데 있어 그리드에 참여하는 각 사이트의 시스템 자원 운영 현황을 고려하지 않았다. 사실상 그리드 서비스에 대한 수요가 비용을 결정하기 때문에 그리드 컴퓨팅 환경에 자원을 제공하는 각 사이트의 자원 운영 현황에 따라 사용자 작업 처리 비용을 결정해야 한다. 왜냐하면 사용자 작업을 처리하기 위해 그리드 자원을 소비하는 시간대가 Peak Time인지, 아니면 Off-Peak Time인지에 따라 그리드 서비스의 수요가 다르다. 또한 평일인지, 주말인지에 따라 그리드 서비스 수요가 다를 것이다.

그리고 그리드 컴퓨팅 환경에 참여하는 사이트의 컴퓨팅 노드들마다 컴퓨팅 파워나 작업 부하량이 다를 수 있기 때문에, 사용자 작업의 처리 비용을 산출하는데 있어 컴퓨팅 노드들의 자원 운영 현황이 고려되어야 한다. 따라서 본 논문은 Globus Toolkit[6] 기반의 그리드 컴퓨팅 환경에서, MPI(Message Passing Interface) 병렬 작업에 대해 각각의 컴퓨팅 노드들로부터 자원 사용량을 측정하고, 사이트의 가격 정책에 따라 처리 비용을 결정함으로써 유틸리티 컴퓨팅 모델을 지향하는 그리드 어카운팅 모델을 제안한다.

3. 유틸리티 모델을 지향하는 사이트 가격 정책 기반의 그리드 어카운팅 설계

그리드는 정보 공유뿐만 아니라 슈퍼 컴퓨터, 클러스터링, 다양한 고성능 연구 장비 등과 같은 자원들까지도 공유가 가능하기 때문에, 서로 다른 관리 도메인상에 있는 정보 및 자원들을 공유하여 하나의 유기체적인 VO(Virtual Organization)[13]를 구축하여 거대하고 복잡한 문제를 해결할 수 있는 컴퓨팅 환경을 제공한다.

그리드 컴퓨팅 환경에서 적절한 인증 절차 및 자원 접근에 대한 권한을 가진 사용자는 대규모의 문제를 해결하기 위해 그리드상에서 제공하는 수많은 자원들을 동시에 이용하고, 자원 사용량에 따른 처리 비용을 지불하게 된다. 따라서 그리드 컴퓨팅 환경에 참여하는 자원 제공자측의 다양한 가격 정책에 따라 사용자 작업의 처리 비용을 산출할 수 있는 그리드 어카운팅 모델을 설계한다.

사용자가 그리드 컴퓨팅 자원을 사용할 때 발생하는 처리 비용을 지불하기 위해선 그리드 사용자의 Account Reference [10] 정보가 필요하다. 사용자는 그리드 스케줄러에게 사용자의 작업 명세서와 Account Reference 정보를 전달하면, 그리드 스케줄러는 사용자의 요구사항에 따라 스케줄링이 이루어진다. 그리고 그리드 스케줄러는 작업이 할당된 각 사이트에 사용자의 작업 명세서와 Account Reference 정보도 함께 전달한다. 사용자의 작업이 완료된 후, 해당 사이트에서는 사용자의 Account Reference 정보를 통해 과금을 청구하게 된다. 본 논문에서는 그리드에 참여하는 사이트의 그리드 어카운팅 시스템은 서비스를 제공받는 사용자의 Account Reference 정보를 인지한다는 가정하에서 설계 구현한다.

3.1 사이트 가격 정책 기반의 그리드 어카운팅 설계

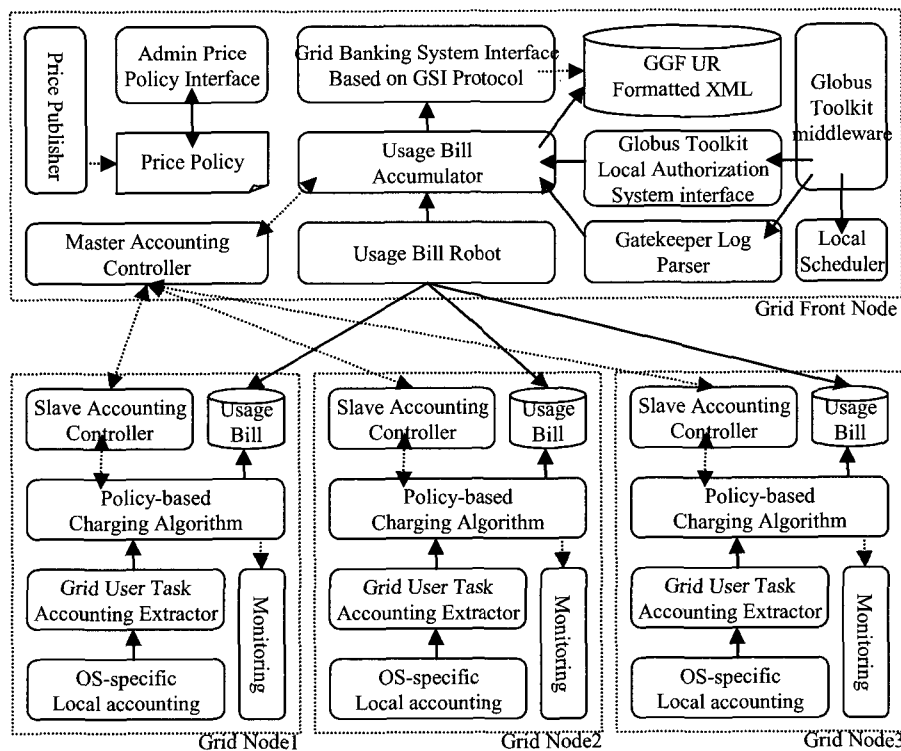
그리드 어카운팅 시스템이 사이트의 가격 정책을 반영하기 위해, (그림 1)과 같이 각 그리드 노드마다 사용자의 자원 사용량을 측정하는 로컬 어카운팅 시스템과 그에 따른 비용을 산출하는 구성요소를 배치한다. 따라서 그리드 노드

의 자원 운영 현황에 따라 가변 비용을 산출할 수 있도록 처리한다. 즉 사용자 작업을 처리하는 컴퓨팅 자원의 가치와 그 자원을 사용하는 시간이 Peak Time인지에 따라, 또는 시스템 작업 부하량 변화에 따라 처리 비용을 차등 산출할 수 있도록 한다.

(그림 1)은 그리드에 참여하는 한 사이트에서 사용자 작업에 대한 가격 정책 기반의 어카운팅 정보 수집 및 GGF UR(Usage Record)[8] 서비스를 제공하는 노드별 주요 구성요소들간의 상호 관계를 나타낸다.

그리드 프론트 노드는 Globus Toolkit Middleware를 통해서 GRAM(Grid Resource Allocation and Management) 클라이언트가 요청한 작업을 로컬 스케줄러에게 제출한다[6, 14]. 그리드 사용자의 어카운팅 정보 및 처리 비용을 생산하기 위해 Master Accounting Controller는 Slave Accounting Controller에게 해당 노드로부터 그리드 사용자 작업의 어카운팅 정보 수집하고, 처리 비용을 산출하도록 지시한다. Master Accounting Controller는 모든 Slave Accounting Controller로부터 작업 완료 신호를 받으면 Usage Bill Accumulator를 호출한다. Usage Bill Accumulator는 Usage Bill Robot을 호출하여 모든 노드에 분산되어 있는 Usage Bill을 한 곳으로 수집한 후, 사용자 작업 단위의 어카운팅 및 과금을 취합, 정리하여 DB에 저장한다.

그리드 노드의 Slave Accounting Controller는 Master Accounting Controller의 지시에 따라 Grid User Task Accounting Extractor를 호출하여 로컬 어카운팅 시스템에서 측정된 모든 사용자 프로세스들 중에서 그리드 사용자만



(그림 1) 사이트 가격 정책 기반의 그리드 어카운팅 주요 구성요소

의 어카운팅 정보만을 추출한다. 그리고 Policy-based Charging Algorithm을 통해 자원의 가치 및 가격 정책에 따라 처리 비용을 계산하여 사용자별 Usage Bill을 작성한다.

따라서 본 논문에서 제안하는 사이트 가격 정책 기반의 그리드 어카운팅 모델은 그리드 프론트 노드와 그리드 노드로 나뉘어 살펴 볼 수 있다. 그리드 프론트 노드에서는 사용자 작업 전체의 자원 사용량과 그에 따른 처리 비용을 취합하여 작업 단위의 어카운팅 및 지불 비용을 취합하도록 한다. 그리고 그리드 노드에서는 사용자 작업이 이루어지는 Session Time 동안 소비한 자원 사용량을 측정하고, 가격 정책에 따라 사용자별 처리 비용을 계산한다.

3.2 경제 기반 그리드 컴퓨팅 환경을 위한 사이트 가격 정책

그리드 자원의 안정적인 수요와 공급이 뒷받침되기 위해서는 유틸리티 모델을 지원하는 그리드 컴퓨팅 환경을 구축해야 한다. 자원 제공자는 유희자원들을 활용함으로써 경제적 이윤을 얻을 수 있고, 자원 소비자는 투명한 가격으로 언제 어디서든지 원하는 만큼의 자원을 사용할 수 있어야 한다.

따라서 사용자들이 경제 원리에 준하는 그리드 자원을 사용할 수 있도록 하기 위해선 자원의 가치를 표현, 해석할 수 있어야 한다. 자원 제공자측은 그들 소유의 자원 가치를 표현할 수 있어야 하고, 또한 사용자들에게 투명한 가격을 제공하기 위해 가격 정책을 공표해야 한다. 그리고 그리드 스케줄러는 각 사이트에서 표방하고 있는 가격 정책들을 해석할 수 있어야 한다. 왜냐하면 그리드 스케줄러는 각 사이트에서 공시하고 있는 가격 정책들을 인지하여, 사용자의 요구사항에 부합하는 자원을 선택하여 작업 할당이 이루어져야 하기 때문이다.

자원의 가치[15, 16]는 그 자원이 가지고 있는 컴퓨팅 파워, 자원의 구입 가격, 자원의 수요, 소비자의 예상하고 있는 자원의 가치, 희소성 등 다양한 항목들이 고려되어 결정된다(그림 2). 예를 들어 Resource Value = Function(computing power, cost of physical resources, profit margin ...) 등으로 결정되고, Computing Power = Function(CPU frequency, storage, software, network bandwidth ...) 등으로 결정된다. 그리고 그리드에 참여하는 사이트의 자원 사용에 대한 처리 비용은 그 자원의 가치와 사이트의 가격 정책을 반영해서 산출해야 한다. 사이트의 가격 정책은 자원 사용시간, 시스템 작업 부하량, 우선순위, 이용가능한 노드 수, 디스크 및 메모리 용량, 자원 사용 시간대 및 요일등이 고려된다.

본 논문에서는 여러 가지 유틸리티 모델 중 유틸리티 컴퓨팅 개념을 도입한다. 자원의 가치는 사용자의 작업을 처

<Resource Value> ::=	<literal>
	<function>
	<arithmetic expression>

(그림 2) Determining Resource Value

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <DOCUMENT>
  <SiteAddress>grid.chonbuk.ac.kr</SiteAddress>
  <ResourceValue per="sec">1G</ResourceValue>
  <!-- G : Grid Current Unit -->
- <DiscountTerm>
  <OffPeakTime from="20" to="8">10%</OffPeakTime>
  <Workload loadavg="0.6">15%</Workload>
- <UsageDuration>
  <Day>20%</Day>
  <Week>30%</Week>
  <Month>40%</Month>
</UsageDuration>
- <Holiday rate="30%">
  <Weekend>SAT, SUN</Weekend>
  <Date>Mar 1, May 5, Jun 6, Jul 17, Oct 3</Date>
</Holiday>
</DiscountTerm>
  <Payment mechanism="pay later" />
</DOCUMENT>
```

(그림 3) Sample Pricing Policy Written in XML

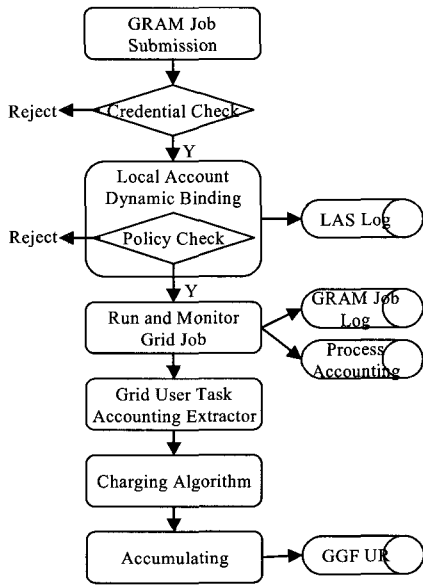
리하기 위해 사용한 CPU의 속도에 따라 결정하고, 가격 정책은 CPU 사용시간, 시스템 작업 부하량, 우선 순위, 작업 수행 시간대 및 요일 등을 고려한다.

(그림 3)은 그리드에 참여하는 한 사이트의 자원의 가치와 가격 정책을 XML로 나타낸것이다. 자원의 가치는 사용자의 작업을 처리하는 데 소비한 CPU Time을 초당 1G(Grid Current Unit)로 계산한다. 그리고 가격 정책에 따라 Off-Peak Time에 사용을 사용할 경우 10%, 일정 이하의 시스템 작업 부하량 상태에서 자원을 이용할 경우 15%의 명시된 가격 할인을 받는다. 또한 자원 사용 시간이 하루, 일주일, 한달을 사용할 경우 각각 20%, 30%, 40%의 할인을 받을 수 있도록 한다. 만약 할인 항목이 중복될 경우, 가장 큰 할인율을 적용한다. 따라서 최종적으로 사용자의 작업을 처리하는데 드는 비용은 자원 가치, 자원 사용 시간 그리고 가격 정책에 따라 산출한다.

3.3 그리드 사용자의 작업 어카운팅 정보 수집

그리드 사용자는 자신의 작업을 처리하기 위해 작업 명세서와 요구사항을 작성하여 그리드 스케줄러에게 제출한다. 그리드 스케줄러는 MDS(Metacomputing Directory Service)로부터 사용자의 요구사항을 만족하는 자원을 검색한 후 사용자의 계좌 정보와 작업 명세서를 해당 사이트에 전달한다. 그리드 작업을 전달받은 해당 사이트에서는 사용자의 작업을 완료한 후에, 자원 사용량 정보를 수집하고, 가격 정책에 따라 과금을 산출한다.

(그림 4)는 그리드에 참여하는 사이트의 사용자 어카운팅 정보를 추출 및 병합하는 과정이다. LAS를 통해 그리드 사용자의 작업을 처리할 수 있도록 미리 정의된 로컬 계정을 일시적으로 생성한 후에 해당 로컬 스케줄러에게 작업을 요청한다. 작업이 끝나면 각 노드에 설치된 로컬 어카운팅 시스템에서 만들어진 프로세스 어카운팅 정보로부터 그리드 사용자의 어카운팅 정보만을 추출한다. 마지막으로 그리드 작업이 실행되는 로컬 계정과 처리되는 Session Time 정보

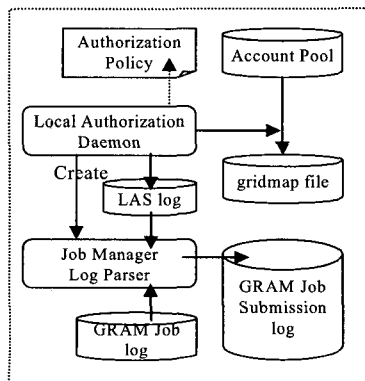


(그림 4) 그리드 사용자 어카운팅 정보 수집 과정

를 이용하여 각각의 노드에 분산되어 있는 어카운팅 정보를 병합한다.

GRAM 작업이 처리되는 과정을 살펴보면, 그리드 스케줄러로부터 사용자 작업을 할당받은 사이트의 Gatekeeper가 Job Manager를 생성한다. 생성된 Job Manager는 로컬 스케줄러에게 사용자 작업을 처리하도록 요청한다. 이 때, Job Manager는 그리드 사용자의 DN(Distinguish Name)과 매핑된 로컬 계정으로 작업이 실행될 수 있도록 로컬 스케줄러에게 요청한다[17]. Globus Toolkit 미들웨어는 그리드 사용자의 로컬 사이트 접근 제어를 위해 Gridmap 파일 방식을 이용한다.

본 논문에서도 (그림 5)와 같이, Gridmap 파일 방식의 Identity-based LAS(Local Authorization System)는 그리드 사용자의 로컬 사이트의 접근 권한을 설정하고, 미리 정의된 Account Pool내의 로컬 계정과 사용자의 DN과 다이내믹하게 일대일 매핑되도록 처리한다. 그러므로써, 로컬 사이트에서 수행하는 그리드 사용자의 작업들에 대해 구별 가능한 어카운팅 정보를 수집, 취합할 수 있게 된다.



(그림 5) Identity-based LAS for Grid Accounting

(그림 5)의 Identity-based LAS의 동작은 그리드 사용자의 DN 매핑 요청시, 사이트 권한 정책에 따라 미리 정의된 Account Pool내의 임의의 로컬 계정을 만들고(그림 6), 그리드 작업을 처리할 수 있는 환경으로 초기화한다. 그리고 생성된 로컬 계정과 사용자의 DN 정보를 Gridmap 파일에 등록한다(그림 7). GRAM을 통해 그리드 사용자의 작업이 완료되면, 그리드 사용자는 자기 자신의 DN 매핑 해제를 요청한다. Identity-based LAS는 매핑 해제 요청을 받으면, LAS 로그파일에 그리드 사용자의 DN, 그 사용자의 DN과 매핑된 로컬 계정, 그리고 사용자와 사이트간의 작업 처리를 위해 접속한 시간을 나타내는 Session Time을 기록한다(그림 8), Job Manager Log Parser 프로세스를 생성하여 GRAM Job 로그파일을 분석한다. GRAM Job 로그파일로부터 어카운팅 정보 취합 및 Resource Usage Service를 위한 로컬 계정, 작업 시간 시간, Job Handle, RSL(Resource Specification Language) 정보를 추출한다<표 1>. 마지막으로 Gridmap 파일에서 사용자의 DN를 제거한 후, 로컬 계정을 삭제한다.

```
// Project or Group Name( Local Account for Grid user )
Power { gridusr01, gridusr02, gridusr03 }
Limited { gridusr04, gridusr05, gridusr06, gridusr07, gridusr08 }
Project { gridusr09, gridusr10 }
```

(그림 6) Account Pool Configuration File

```
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang gridusr09
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=jeongjin2 gridusr01
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=kyongsu gridusr04
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=test gridusr05
```

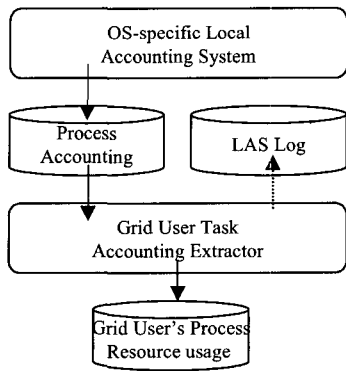
(그림 7) Sample Gridmap File

// Grid User's DN	Local Account	Session Time (Unix Time)
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=test	gridusr07	1110846160:1110848026
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang	gridusr09	1110863533:1110865232
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=jeongjin2	gridusr01	1110860277:1110866872
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=kyongsu	gridusr04	1110934223:1110936034
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=test	gridusr05	1110932587:1110937688
/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang	gridusr09	1110969172:1110970227

(그림 8) Sample LAS Log File

<표 1> Data Format of GRAM Job Submission log

char	LocalUserID[32];	// Local User ID
time_t	JobBeginTime	// Unix Time
char	JobHandle[256];	// Job Handle ID
char	RSL[4096];	// User's Job Specification



(그림 9) 그리드 노드에서의 자원 사용량 추출

본 논문에서 구현한 사이트 가격 정책 기반의 그리드 어카운팅 모델은 그리드에 참여하는 각 사이트의 운영체제에 특화된 로컬 어카운팅 시스템을 토대로 그리드 사용자의 자원 사용량을 측정한다(그림 9). 이를 위해 각각의 그리드 노드에 설치된 로컬 어카운팅 시스템에서 측정된 프로세스 어카운팅 정보 파일로부터, Grid User Task Accounting Extractor는 LAS 로그파일의 로컬 계정과 Session Time 정보를 기초로 그리드 노드별 사용자 작업의 자원 사용량 정보를 수집한다. 즉 그리드 노드에서 사용자의 작업을 처리하기 위해 접속한 Session Time 동안 매핑된 로컬 계정의 자원 사용량을 추출한다.

그리고 (그림 1)에서 그리드 프론트 노드의 Usage Bill Accumulator는 사용자의 작업 단위 어카운팅 정보를 생산하기 위해 Identity-based LAS의 로그파일, Job Manager Log Parser가 GRAM Job 로그파일을 분석하여 사용자 작업의 Job Name, GRAM Job Handle, RSL 및 로컬 계정 정보를 저장하고 있는 로그파일(그림 5), 마지막으로 각 그리드 노드에서 수집된 Usage Bill들을 병합한다. 즉 사용자의 DN, 그 DN과 매핑된 로컬 계정, Session Time 정보로 구분하여 각 그리드 노드에서 수집된 프로세스 어카운팅 정보를 취합하여 그리드 사용자 작업 단위의 어카운팅 정보를 생산한다.

3.4 사이트 가격 정책에 따른 자원 사용량의 비용 산출

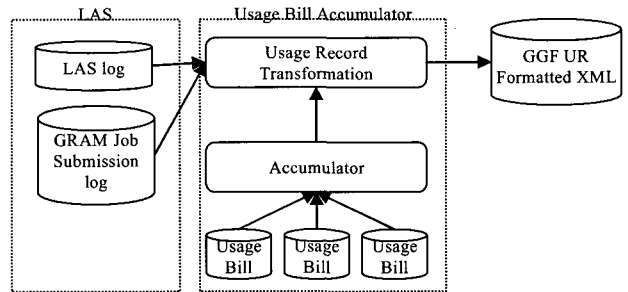
그리드에 참여하는 사이트의 가격 정책에 따른 가변 비용을 산출하는 구성요소들을 모든 노드에 배치한다. 이는 사용자의 작업을 처리하는 노드들의 컴퓨팅 파워나 시스템 작업 부하량에 따라 처리 비용이 다르기 때문이다. 그래서 모든 노드에 시스템 제반사항들을 기록하는 Monitoring 구성요소를 두어 운영체제의 /PROC 파일 시스템을 통해 CPU 속도와 일정 시간 간격마다 시스템 작업 부하량을 실시간으로 측정할 수 있도록 한다.

그리고 모든 노드의 Policy-based Charging Algorithm 구성요소에서는 자원의 가치와 CPU 사용시간, 시스템 작업 부하량, 작업 수행 시간대 및 요일등을 고려한 가격 정책에 따라 처리 비용을 계산한다. 즉 그리드 사용자 프로세스의 자원 사용량 정보를 기초로 처리 비용을 계산하여 Usage

Bill을 작성한다. 이 Usage Bill은 그리드 사용자의 자원 사용량 정보와 그에 따른 과금 정보를 포함하고 있다.

3.5 RUS를 위한 UR 변환

GGF UR-WG에서는 그리드에 참여하는 사이트들간의 어카운팅 정보를 상호 교환하기 위해 공통된 Usage Record 포맷을 권장하고 있다. 따라서 Resource Usage Service를 위해 그리드 사용자 작업의 기본 어카운팅 및 자원 사용량 데이터를 Usage Record 포맷으로 변환해야 한다. 이를 위해서 각 그리드 노드에서 수집된 Usage Bill들을 사용자별로 어카운팅 정보를 취합한 다음, (그림 5)의 LAS 로그 파일과 GRAM Job Submission 로그파일들을 분석해 GGF Usage Record 포맷으로 변환한다(그림 10).



(그림 10) GGF UR Transformation

4. 실험 환경 및 결과

4.1 실험 환경

본 연구에서는 PBS(Portable Batch System) 로컬 스케줄러를 사용하는 리눅스 클러스터링 환경에서 MPI 그리드 작업에 대한 자원 사용량을 측정하였고, 또한 사이트 가격 정책에 따라 자원 사용량의 처리 비용을 산출하였다. 최종적으로 Resource Usage Service가 가능한 그리드 어카운팅 모델을 제안하였다. 개발 환경은 다음과 같다<표 2>.

<표 2> 실험 환경

항목	내용
클러스터링 환경 (CPU, Memory)	리눅스 클러스터링 시스템 - 그리드 프론트 노드(1GHz, 512MB) - 그리드 노드 노드 1(933Mhz, 512MB) 노드 2(866Mhz, 512MB) 노드 3(866Mhz, 256MB) - NFS 서버(800Mhz, 256MB)
운영체제 및 로컬 어카운팅 시스템	Red Hat Linux 7.1(Kernel Version 2.4.2-2) psacct 6.3.2-5
로컬 스케줄러 및 그리드 미들웨어	Globus Toolkit 2.4 OpenPBS 2.3.16
MPI 라이브러리 및 컴파일러	MPICH 1.2.6, GCC 2.96

4.2 실험 결과 및 평가

본 논문에서 제안하는 사이트 가격 정책 기반의 그리드 어카운팅 모델을 구현하기 위해 다음과 같은 일련의 과정을 수행하였다. 첫 번째로 그리드 사용자는 작업 명세서를 작성한 후, 해당 사이트에 제출한다. 두 번째로 해당 사이트의 Gatekeeper는 작업 처리를 요청한 사용자에게 인증 과정을 거친다. 인증 과정을 마친 후, Identity-based LAS는 접근이 허용된 사용자의 DN에 대해 자원 사용에 대한 권한이 부여된 로컬 계정으로 매핑시켰다. 그리고 Job Manager는 로컬 스케줄러에게 작업 처리를 요청함으로써 작업이 수행된다. 마지막으로 그리드 어카운팅 시스템은 작업 부하량이 적은 시간대에 사용자의 어카운팅 정보를 수집하여, 사이트의 가격 정책을 토대로 사용자별 자원 사용량에 과금을 처리하도록 하였다.

그리고 제안된 그리드 어카운팅 모델을 평가하기 위해 그리드 관련 프로젝트에서 개발된 어카운팅 시스템들과 비교하였다. 평가 대상이 되는 그리드 어카운팅 시스템들에서 나타나는 주요 특징들만을 비교 설명하였다. 왜냐하면 평가 대상 시스템의 어카운팅 추적 및 수집 과정들이 자세히 기술되어 있기 않기 때문이다. 그리고 일부 시스템의 경우 운영체제에 특화된 로컬 어카운팅 시스템을 변경한 사례도 있기 때문이다. 따라서 비교 평가시, 그리드 어카운팅 시스템 상에서 나타난 특징들을 고려하였다.

Globus Toolkit에서 보안을 담당하는 GSI 구성요소[6]로부터 사용자 인증과 Identity-based LAS로부터 자원 사용에 대한 권한을 획득한 그리드 사용자는 자신의 MPI 병렬 작업을 처리하기 위해 RSL을 사용하여 작업 명세서를 해당 사이트에 제출하였다. 그리드 사용자의 작업은 로컬 스케줄러를 통해 여러 노드에 분산시켜 동시에 처리하게 된다.

(그림 11)는 각 노드에 설치된 로컬 어카운팅 시스템의 로그파일에서 그리드 사용자의 프로세스가 소비한 자원 사용량을 추출하여 가격 정책에 따라 산출된 정보를 포함하고 있는 Usage Bill이다.

이 Usage Bill에는 사용자의 MPI 병렬 작업을 처리한 노드명, 로컬 계정, Session Time, 자원의 가치, 자원 사용 시간(CPU Time), 할인 항목 및 할인을 그리고 자원 사용 시간에 따른 과금정보를 담고 있다. 본 논문에서의 과금은 자원 가치에 자원 사용 시간을 곱 하였다. 그리고 가격 정책에 따라 중복된 할인 항목이 있을 경우, 가장 큰 할인율을 적용시켜 과금을 산출하였다.

(그림 11)의 Usage Bill를 보면 그리드 사용자의 작업이 gc01노드상에서 gridusr09 로컬 계정으로 Session Time 동안 11,662ms(11,508ms+154ms) CPU Time을 소비하였다. 그리고 이 사용자의 작업이 수행한 시간이 Off-Peak Time이기 때문에 가격 정책에 따라 10%의 할인율을 적용하였다.

(그림 12)는 그리드 사용자의 MPI 병렬 작업을 처리하는데 드는 총 자원 사용량과 그에 대한 과금을 취합한 그림이다. 각각의 그리드 노드에 흩어져 있는 Usage Bill들을 사용자별로 통합하기 위해 Identity-based LAS 로그파일<표 1>

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <DOCUMENT>
  <GridNode>gc01</GridNode>
  <BillDate>1110988803</BillDate>
  <LocalUserID>gridusr09</LocalUserID>
  - <SessionTime>
    <!-- Unix Time -->
    <BeginTime>1110969172</BeginTime>
    <EndTime>1110970227</EndTime>
  </SessionTime>
  <!-- Resource Value -->
  <ResourceValue>1.3G</ResourceValue>
  <!-- Resource Usage Summary -->
  - <Usage>
    <!-- CPU Time (ms) = User Time + System Time -->
    <UserTime>11508</UserTime>
    <SystemTime>154</SystemTime>
    <ElapsedTime>23052</ElapsedTime>
  </Usage>
  <!-- Discountable Item and Rate -->
  - <Discount>
    <Item>OffPeakTime</Item>
    <Rate>10%</Rate>
  </Discount>
  <CurrentCharges>12.9G</CurrentCharges>
</DOCUMENT>
```

(그림 11) Usage Bill for MPI Task

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <DOCUMENT>
  <BillDate>1110988834</BillDate>
  <LocalUserID>gridusr09</LocalUserID>
  <GridUserDN>/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang</GridUserDN>
  - <SessionTime>
    <BeginTime>1110969172</BeginTime>
    <EndTime>1110970227</EndTime>
  </SessionTime>
  <TotalAmount>18.2G</TotalAmount>
  <!-- Grid User Job Accounting Summary -->
  + <GridNode name="gc01">
  - <GridNode name="gc02">
    <ResourceValue>1G</ResourceValue>
  - <Usage>
    <UserTime>4832</UserTime>
    <SystemTime>135</SystemTime>
    <ElapsedTime>23054</ElapsedTime>
  </Usage>
  - <Discount>
    <Item>OffPeakTime</Item>
    <Rate>10%</Rate>
  </Discount>
  <CurrentCharges>3.6G</CurrentCharges>
  </GridNode>
  - <GridNode name="gc03">
    <ResourceValue>1G</ResourceValue>
  - <Usage>
    <UserTime>2319</UserTime>
    <SystemTime>55</SystemTime>
    <ElapsedTime>92374</ElapsedTime>
  </Usage>
  - <Discount>
    <Item>Workload</Item>
    <Rate>15%</Rate>
  </Discount>
  <CurrentCharges>1.7G</CurrentCharges>
  </GridNode>
</DOCUMENT>
```

(그림 12) Usage Bill for MPI-based Program

의 DN, 로컬 계정, Session Time을 참조하였다. 이 로그파일에는 언제 어떤 그리드 사용자가 어느 로컬 계정으로 매핑되어 자원을 사용하고 있는지를 기록하고 있다. 따라서 모든 노드에 분산되어 있는 Usage Bill들을 병합하기 위해 로컬 계정과 Session 시작 시간 및 종료 시간을 취합하기 위한 키로 사용하였다.

(그림 12)는 “/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang”라는 DN를 가진 사용자가 gridusr09라는 로컬 계정으로 Session 시작 시간에서 종료 시간까지 3개의 노드를 사용하여 자신의 작업을 수행할 것 알 수 있다. 그리고 각

```
<?xml version="1.0" encoding="UTF-8" ?>
- <JobUsageRecord xmlns="http://www.gridforum.org/2003/ur-wg" xmlns:urwg="http://www.gridforum.org/2003/ur-wg"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.gridforum.org/2003/ur-wg
  file:/Users/bekah/Documents/GGF/URWG/urwg-schema.09.xsd">
  <RecordIdentity urwg:recordId="458.grid.chonbuk.ac.kr" urwg:createTime="2005-03-17 1:0:34" />
- <JobIdentity>
  <LocalJobId>458.grid.chonbuk.ac.kr</LocalJobId>
</JobIdentity>
- <UserIdentity>
  <GlobalUsername>/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang</GlobalUsername>
</UserIdentity>
  <JobName>prime</JobName>
  <Charge>18</Charge>
  <Status>completed</Status>
  <WallDuration>00:01:32</WallDuration>
  <CpuDuration>00:00:19</CpuDuration>
  <EndTime>2005-3-16 19:36:21</EndTime>
  <StartTime>2005-3-16 19:35:49</StartTime>
  <MachineName>IATLab Linux Cluster</MachineName>
  <Host>grid.chonbuk.ac.kr</Host>
  <SubmitHost>iat.chonbuk.ac.kr</SubmitHost>
  <Queue>workq</Queue>
  <ProjectName>Grid Accounting</ProjectName>
</JobUsageRecord>
```

(그림 13) GGF UR Formatted XLM

노드별 CPU 사용 시간에 따른 과금과 전체 자원 사용에 대한 과금이 기술되어 있다. 이와 같이 그리드 사용자의 어카운팅 정보 및 과금 정보를 취합하기 위해, 사이트의 작업 부하량을 분산시키기 위해 Cron Daemon을 설정하였다. 즉 정해진 시간마다 시스템 전체 노드의 작업 부하량이 일정 이하일 때 동작하도록 하였다.

마지막으로 그리드에 참여하는 사이트들간의 어카운팅 정보를 상호 교환하기 위해 공통된 포맷으로 변환하여 그리드 사용자의 Resource Usage Service를 제공하기 위해 (그림 10)과 같이 <표 1>의 GRAM Job Submission Log 파일과 (그림 12)의 Usage Bill를 통합하여 (그림 13)과 같이 GGF Usage Record로 변환하였다. 그리드 사용자의 MPI 병렬 작업에 대한 자원 사용량 및 어카운팅 정보를 XML형식으로 DB에 저장하였다. DB에 저장된 GGF Usage Record는 Grid Banking System Interface를 통해 그리드 사용자의 과금이 결제되도록 설계하였다.

(그림 13)의 Usage Record를 보면, "/O=Grid/O=Globus/OU=chonbuk.ac.kr/CN=hjhwang"라는 DN를 가진 그리드 사용자는 iat.chonbuk.ac.kr 클라이언트에서 grid.chonbuk.ac.kr 사이트로 prime이라는 작업을 제출하였다. 그리고 해당 사

이트에서 사용자의 작업을 처리하는데 걸리는 시간은 1분 32초, 총 CPU 사용시간은 19초이며 작업 상태는 completed이다. 또한 자원 사용량과 사이트의 가격 정책에 따라 산출된 과금은 18G이다.

이와 같이 본 논문에서 제안하는 사이트 가격 정책 기반의 그리드 어카운팅 모델은 DGAS[11], SGAS[12]와 마찬가지로 Dynamic Account Management, User Accounting, Project-User Mapping, Consistent Usage Record, 그리고 투명한 사이트 가격 정책을 집행할 수 있도록 설계하였다. 그러나 SGAS의 경우 그리드에 참여하는 로컬 어카운팅 시스템을 일부 변경함으로써 사이트의 자율성을 침해한다는 단점을 지니고 있다. 따라서 <표 3>에서도 알수 있듯이, 본 논문에서 제안하는 그리드 어카운팅 모델은 사이트의 로컬 스케줄러가 무엇이든간에 상관없이 운영체제에 특화된 로컬 어카운팅 시스템으로부터 그리드 사용자의 작업들에 대해 구별 가능한 어카운팅 정보를 생산할 수 있게 하였다. 그리고 그리드에 참여하는 사이트의 각 노드마다 자원 운영 현황에 따라 정교한 Charging Schemes를 수립할 수 있도록 설계하였다. 그러므로써 그리드에 참여하는 시분할 클러스터링 및 서로 다른 컴퓨팅 파워를 가진 자원들을 제공하는

<표 3>기존 어카운팅 시스템과의 평가

Evaluation Areas	Proposal Model	DGAS	SWAG
Site Autonomy	로컬 어카운팅 시스템을 수용	모니터링 서비스를 통해 주기적으로 자원 사용량 수집	로컬 어카운팅 시스템을 일부 변경
Account Management	다이나믹 어카운트	다이나믹 어카운트	다이나믹 어카운트
Project-User Mapping	지원	지원	지원
GGF UR format	지원	지원	지원
Charging Schemes	노드/사이트별 정교한 과금 정책 가능	가격 정책 개념이 미흡	가격 정책 개념이 미흡
Where to Pay User's Bill	뱅크를 통해 자원 제공자와 사용자간의 과금 결제	자원 제공자와 소비자간의 과금 결제	뱅크를 통해 자원 제공자와 사용자간의 과금 결제

사이트에 적용하는데 적합한 모델로 평가된다.

5. 결론 및 향후 연구

그리드는 지리적으로 분산된 고성능 컴퓨팅 파워, 대용량 저장장치 및 데이터베이스, 첨단 실험 장비등의 자원들을 통합, 상호 공유할 수 있도록 하는 메타 컴퓨팅 환경을 제공한다. 그리드 자원의 안정적인 수요와 공급이 뒷받침되기 위해서는 유틸리티 모델을 지원하는 그리드 컴퓨팅 환경을 구축해야 한다. 그리드 비즈니스를 활성화하고, 사용자들이 경제원리에 준하는 그리드 컴퓨팅 자원을 활용할 수 있도록 하기 위해서는 사이트 가격 정책 기반의 그리드 어카운팅 모델이 필수적이다.

그러므로 본 논문에서 제안하는 사이트 가격 정책 기반의 그리드 어카운팅 모델은 투명한 가격 정책에 따라 자원 사용량에 대한 처리 비용을 산출하였으며, 특히 사이트의 자율성을 보장하면서 다이나믹한 사용자 계정 서비스를 통해 그리드 사용자들에 대해 식별 가능한 어카운팅 정보를 생산해낼 수 있는 모델이다. 또한 사이트의 자원 운영 현황에 따라 가격을 차등 적용할 수 있는 정교한 과금 체계를 수립할 수 있다. 그리고 사용자 어카운팅 정보를 공통된 포맷의 GGF Usage Record로 변환함으로써 그리드에 참여하는 Entity들에 대해 Resource Usage Service를 제공할 수 있다.

본 논문에서 제안하는 사이트 가격 정책 기반의 그리드 어카운팅 모델이 활성화가 되면 그리드에 참여하는 자원들 간의 다양한 형태의 경제 모델도 등장하게 될 것이다. 이를 위해서는 다양한 종류의 자원 소비 데이터를 측정할 수 있고, 그리드 환경을 인지할 수 있는 Grid-aware Local Accounting System 연구가 필요할 것이다. 그리고 그리드 스케줄러 및 बैं킹 시스템과 연동할 수 있고, 더 나아가 사용자의 बैं킹 계좌 잔액에 따른 시스템 자원 사용량을 제한할 수 있는 메커니즘을 제공하는 그리드 어카운팅 시스템 구축이 반드시 필요하다. 왜냐하면 자원 제공자의 과금 정책에 따라 사용자의 बैं킹 계좌 잔액 정보를 실시간으로 확인하여 사용자의 자원 사용량 제한 및 접근 권한을 결정할 수 있기 때문이다.

참 고 문 헌

- [1] I. Foster, C. Kesselman, "Globus : A Metacomputing Infrastructure Toolkit," International J. Supercomputer Applications, Vol.11, No.2, pp.115-128, 1997.
- [2] Ian Foster, Carl Kesselman, "The Grid: Blueprint for a New Computing Infrastructure," Morgan Kaufmann Publishers, 1998.
- [3] I. Foster, C. Kesselman, S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," International J. Supercomputer Applications, Vol.15, No.3, 2001.
- [4] Michael Treaster, Nadir Kiyancilar, Gregory A. Koenig, and William Yurcik, "A Distributed Economics-based Infrastructure for Utility Computing," ACM Computing Research Repository Technical Report cs.DC/0412121, December, 2004.
- [5] G. A. Koenig and W. Yurcik, "Design of an economics-based software infrastructure for secure utility computing on supercomputing clusters," in 12th Intl. Conference on Telecommunication Systems - Modeling and Analysis, 2004.
- [6] The Global Grid Forum, <http://www.ggf.org>.
- [7] The Grid Forum Korea, <http://www.gridforumkorea.org>.
- [8] Usage Record, GGF Working Group, <http://forge.gridforum.org/projects/ur-wg>.
- [9] OGSA Resource Usage Service, GGF Working Group, <http://forge.gridforum.org/projects/rus-wg>.
- [10] Anthony Beardsmore, Keith Hartley, et al, "GSAX(Grid Service Accounting Extensions)," GGF OGSA Resource Usage Service Working Group, 2002.
- [11] Guarise, A., Piro, R., and Werbrouck, A. DataGrid Accounting System - Architecture - v1.0. DataGrid-01-TED-0126-1_0. EU DataGrid 2003. http://server11.infn.it/workload-grid/docs/DataGrid-01-TED-0126-1_0.pdf.
- [12] Erik Elmroth, Peter Gardfjäll, Olle Mulmo, Åke Sandgren, Thomas Sandholm, "A Coordinated Accounting Solution for SweGrid," Draft 0.1.3, October, 2003.
- [13] M. Baker and R. Buyya, "Cluster Computing at a Glance," High Performance Clustering Computing Vol.1, Prentice Hall, 1999.
- [14] Karl Czajkowski, Ian Foster, et al, "A Resource Management Architecture for Metacomputing Systems," Procs. of the 4th Workshop on Job Scheduling Strategies for Parallel Processing, 1998.
- [15] Rajkumar Buyya, Heinz Stockinger, Jonathan Giddy, David Abramson, "Economic Models for Management of Resources in Grid Computing," CERN, 2001.
- [16] R. Buyya, D. Abramson, and J. Giddy, "A Case for Economy Grid Architecture for Service Oriented Grid Computing," <http://www.buyya.com/papers/ecogrid.pdf>.
- [17] Laura F. McGinnis and William Thigpen and Thomas J. Hacker. Accounting and Accountability for Distributed and Grid Systems. 2nd IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2002), 22-24 May, 2002, Berlin, Germany 2002.



황 호 전

e-mail : hjhwang@mail.chonbuk.ac.kr
 1997년 전북대학교 컴퓨터공학과(공학사)
 1999년 전북대학교 컴퓨터공학과
 (공학석사)
 1999년~현재 전북대학교 컴퓨터공학과
 박사과정

관심분야 : 분산 및 병렬처리, 그리드



안 동 언

e-mail : duan@moak.chonbuk.ac.kr
 1981년 한양대학교 전자공학과 (공학사)
 1987년 KAIST 전산학과(공학석사)
 1995년 KAIST 전산학과(공학박사)
 1995년~현재 전북대학교 전자정보공학부
 부교수
 2001년~2002년 전북대학교 정보검색시스템
 연구센터 센터장

관심분야 : 정보검색, 한국어정보처리, 문서분류, 문서요약



정 성 종

e-mail : sjchung@moak.chonbuk.ac.kr
 1975년 한양대학교 전기공학과(공학사)
 1981년 Houston대학교 전자공학과
 (공학석사)
 1988년 충남대학교 전산공학과(공학박사)
 1985년~현재 전북대학교 전자정보공학부
 교수

1996년~1998년 전북대학교 전자계산소 소장
 2001년~현재 전북대학교 BK21 전자정보사업단 단장
 관심분야 : 정보검색, Grid