

# XML 기반의 Usage Reporting 시스템의 설계 및 구현

성진우, 이영주, 이상동, 김중권  
슈퍼컴퓨팅센터 한국과학기술정보연구원  
e-mail:jwsung@kisti.re.kr

## A Design and Implementation of Reporting System Based on XML

JinWoo Sung, YoungJoo Lee, SangDong Lee,  
JoongKwon Kim  
Supercomputing center,  
Korea Institute of Science and Technology Information(KISTI)

### 요 약

컴퓨터에서 사용자의 usage정보는 Accounting의 기본 자료가 될 뿐만 아니라 그리드 컴퓨팅과 같은 환경에서도 교환되어지는 중요한 자료이다. 본 논문에서는 다수의 컴퓨팅 시스템으로부터 usage정보를 수집하여 제공하는 시스템을 구축하였다. 이 서버에서의 특징은 정보관리를 위하여 일반적으로 사용하는 DB를 사용하지 않고 XML파일로 처리한 것이다. 이렇게 함으로써 서버는 보다 단순해 질 수 있으며, 개인이나 중소 규모의 전산센터에서도 쉽게 usage정보 제공이 가능해 질 수 있다.

### 1. 서론

산업사회가 고도화됨에 따라 고성능 컴퓨터의 이용성과 초고속 네트워크의 성장은 오늘날 우리가 계산하는 일과 컴퓨터를 사용하는 방법을 변화시키고 있다[1]. 그리고 최근에는 인터넷과 컴퓨터의 활용이 많아지고 있으며, 고성능 컴퓨팅 자원의 수요 또한 급격히 증가하고 있다. 그러나 한정된 자원으로 인하여 다수의 사용자가 요구하는 만큼의 자원을 충분히 사용하는 것이 매우 제한적으로 이루어지고 있다.[2] 이런 부족한 자원 문제를 해결하기 위하여 그리드와 같이 분산된 유휴 자원을 활용하는 기술들이 계속 연구되고 있다.

그리드 환경에서 자원제공자는 제공한 자원의 양에 비례하는 정당한 대가를 받기를 원하며, 필요한 경우는 그 대가를 사용자에게 청구할 수 있어야 한다.

따라서 누가 어떤 자원을 얼마만큼 사용했는지 정확한 사용량을 수집하고 통계를 내는 어카운팅(accounting)기능은 계산그리드를 활성화시키기 위한 필수적인 기능이다.[3] 컴퓨터에서 사용자의 사용량(usgae)정보는 어카운팅의 기본 자료가 될 뿐만

아니라 그리드 환경에서도 교환되어지는 중요한 자료이다.[4] 이러한 자료들의 관리를 위하여 데이터베이스를 활용하는 것이 일반적이다. 그러나 XML 데이터베이스가 아닌 일반 관계형 데이터베이스인 경우에는 추가되어야 하는 기능(예: 포맷변환기)들로 인하여 복잡해 질수가 있다.

본 논문에서는 이기종 시스템으로부터 사용자 작업의 사용량 정보를 일반 데이터베이스를 사용하지 않고 XML 파일로 관리하고 정보를 보고(reporting)를 해주는 '사용량 보고 시스템(Usage Reporting System, URS)'을 구현하고자 한다. 논문의 구성은 2장에서는 관련연구, 3장은 시스템 구성, 4장은 구현 결과, 5장은 결론으로 되어 있다.

### 2. 관련 연구

#### 2.1 선행 연구 조사

사용자의 usage정보를 XML 문서로 처리하여 정보를 제공하는 대표적인 사례는 다음과 같다. 첫째는 Rod Mach가 Java 코드로 구현한 PbsAccounting(ver. 1.4.2)[5]이고, 둘째는 Devin

Kowatch가 Perl 코드로 작성한 UsageRecord(ver.10.0)[6]가 있다. 이들의 코드는 PBS(Portable Batch System) log가 생성되는 시스템에 위치하여 PBS log를 XML 문서로 변환하여 정보를 필요로 하는 서버로 정보의 제공이 가능한 기능을 가지는 것이 특징이다[4]. 이와는 달리 서버/클라이언트 형태로 된 시스템도 있다. 샌디에고 슈퍼컴퓨터센터(SDSC)에서 개발한 SNUPI 시스템[7]이 있다. 서버/클라이언트 형태의 구성은 다수의 usage 정보를 처리할 수 있는 확장성을 제공한다.

## 2.2 기술 동향

### 2.2.1 UR(Usage Record)-WG과 RUS(Resource Usage Service)-WG

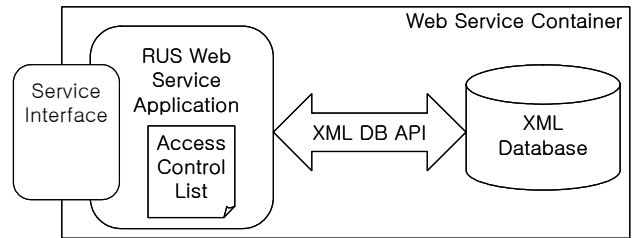
GGF(Global Grid Forum)[8]에는 다수의 Working Group(WG)이 있으며, 그 중에서 Scheduling and Resource Management(SRM) 영역에 속한 UR-WG에서는 GRID 환경에서의 Accounting과 usage data의 교환에 대하여 연구를 활발히 진행하고 있으며, CUR(Common Usage Record) XML schema를 제안하는 것을 최종 목표로 하고 있다. 그리고 RUS-WG에서는 OGSA환경의 그리드 서비스에서 자원의 사용내역이 나타나도록 하는 것을 목표로 하고 있다.

### 2.2.2 UR-WG 표준안 적용 사이트 조사

- Teragrid project[9]  
9곳의 다른 사이트에 있는 자원 시스템으로부터 usage 데이터를 전송하는데 UR schema를 적용한다. 각 사이트는 집중관리되는 usage 자료를 reporting 등의 용도로 이용한다.
- EGEE[10] project의 CSAR[11]  
CSAR 서비스에서는 각 작업에 UR schema를 적용하여 RUS(Resource Usage Service) 시스템으로 upload한다.
- EGEE project의 gLite[12]  
12기관에서 80명 이상이 공동으로 개발하는 그리드 미들웨어인 gLite내의 Grid Accounting 시스템인 DGAS에 UR schema가 적용되었다.
- National Grid Service, UK[13]  
UR schema를 채택하기로 결정하였으며, 현재 작업은 진행중이다.
- Michigan대학의 MGRID 프로젝트[14]  
MGRID에서는 PBS와 Condor의 usage record에

UR schema를 적용하였으며, 데이터를 저장 후 accounting과 통계에 사용한다.

- RUS-WG[15], GGF  
WS-I Resource Usage Service(RUS) Working Group에서는 usage record를 저장할 때 UR schema를 적용한다. 그림 1은 RUS-WG에서 제시하는 RUS 시스템의 구조이다.



(그림 1) RUS 시스템 구조[16]

## 3. 시스템 구성

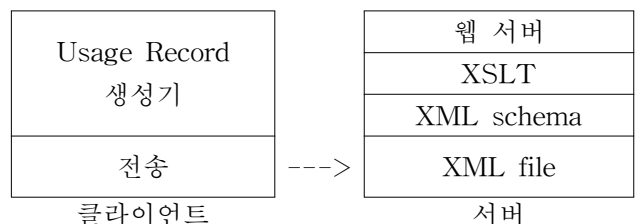
본 논문에서 제안하는 시스템은 크게 서버와 클라이언트 두 부분으로 구분되며, 여기서 클라이언트는 정보가 생성되고 제공되는 컴퓨팅 노드를 말한다. 서버는 클라이언트로부터 수집된 정보를 저장, 검색할 수 있는 기능을 제공하도록 설계하였다.

### 3.1 시스템 처리 구성도

<표1>은 Reporting 시스템이 usage 정보를 처리하는 구성도를 나타낸다. 클라이언트에서는 원시 accounting 자료로부터 reporting에 필요한 정보를 추출하는 Usage Record 생성기와 이를 서버로 전송하는 전송 프로그램으로 구성되어 있다. 그리고 서버의 구성은 data가 저장되어 있는 XML file, XML schema, 그리고 출력에 해당하는 XSLT 및 데이터 검증을 위한 XML schema로 구성되어 있다.

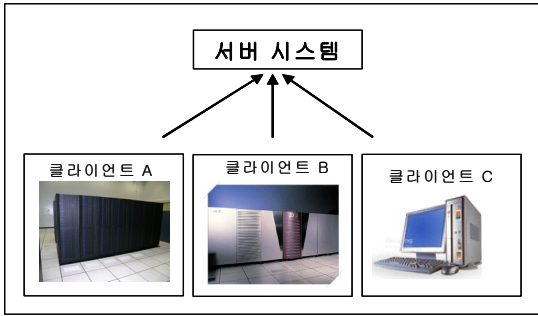
일반적인 구성에서는 <표1>에서의 XML file 대신에 데이터베이스(XML 데이터베이스이든 아니든)가 위치되어 데이터의 저장과 외부 검색에 대하여 결과를 되돌려 주는 구성의 형태이다.

<표 1> 시스템 처리 구성도



### 3.2 시스템 환경

시험 환경은 그림 2와 같이 1대의 서버와 3대의 클라이언트로 구성되어 있다.



(그림 2) 시험 환경

시험에 사용된 서버의 사양은 P3 500MHz이며, O.S는 Linux 2.4.20이다. 클라이언트 시스템의 정보는 표 2에 나타나 있다.

<표 2> 클라이언트 시스템 구조

클라이언트	A	B	C
O.S	Linux	SUPER-UX	Linux
구조	cluster	Vector	
Processor	Xeon 2.8 GHz	312.5 MHz	P4 1.7 MHz
CPU수	512	8	1
Scheduler	PBSpro	NQS	openPBS

### 3.3 시스템의 구성 요소

#### 3.3.1 클라이언트측 요소

##### □ Usage Record 생성기

사용자의 작업이 종료되면 log 파일을 모니터하고 있던 프로그램에 의해 Usage Record 생성기가 실행된다. Usage Record 생성기는 각 클라이언트 시스템의 로컬 스케줄러에서 제공하는 log를 읽어서 Usage Record를 생성시킨다. 여기서 Usage는 사용자의 작업이 노드에서 수행될 때 사용한 자원(CPU time, Memory, Disk 등)의 양을 나타내는 정보이다.

##### □ 전송 프로그램

XML file의 전송은 ftp 프로그램을 사용한다.

#### 3.3.2 서버측 요소

##### □ XML file

Reporting에 필요한 usage 정보들은 XML format으로 저장되며, file 내의 element들은 표 3의 항목명으로 구성되어 있다.

<표 3> Usage Record 항목

1 :Username	2 :ProjectName	3 :Tvne
4 :AuthName	5 :JobId	6 :Queue
7 :Qwait	8 :GridId	9 :FromHost
10:ExecHost	11:QueueTime	12:StartTime
13:EndTime	14:ReaProcessors	
15:MaxProcessors	16:Nodes	17:NumNodes
18:NodeTvne	19:PerProcessors	
20:proconsumptionRate	21:Cputime	22:UserCpu
23:SvstemCpu	24:ReaTime	25:ConnectTime
26:Walltime	27:ReaMemory	28:MaxMemory
29:Memory	30:IOread	31:IOwrite
32:Disk	33:File	34:Network
35:JobName	36:Status	37:Class
38:Jobtvne	39:Shift	40:QOS
41:Charge	42:Comment	43:UsageType

##### □ XML schema

XML 파일내의 데이터는 유효성이 검증되어야 한다. 데이터를 검증하는 방법은 DTD(Document Type Definition)와 XML schema가 있다. 본 시스템에서는 XML schema로 하였다.

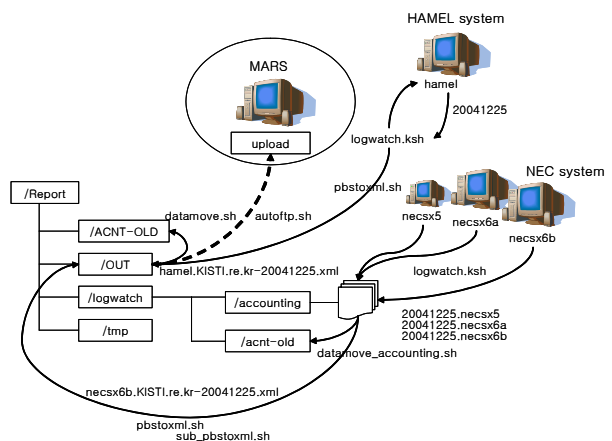
XML 문서는 schema파일에 의하여 데이터의 검증을 거치므로 노드간 혹은 사이트간의 부적합한 정보가 교환되는 것을 방지할 수가 있다.

##### □ XSLT

XSLT는 XML 문서를 다양한 종류의 문서로 생성하는 역할을 한다. 본 시스템에서는 HTML 문서 형태의 출력으로 구축하였다.

### 4. 구현 결과

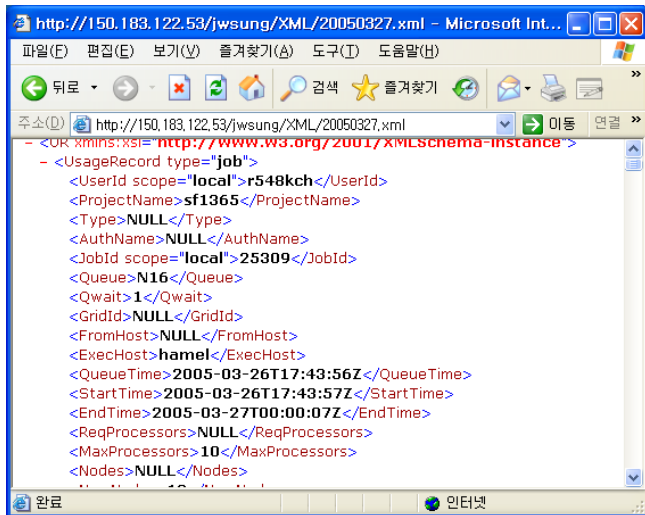
그림 3은 클라이언트 시스템(HAMEL과 NEC시스템)과 서버 시스템(MARS)와의 data flow를 나타낸 그림이다.



(그림 3) 서버/클라이언트 data flow

그림 4는 XML 파일 포맷의 usage 정보 파일이다. 각 계산 노드에서 사용자의 작업이 종료할 때 마다 이 파일이 노드에서 서버 시스템(MARS)로 전송이 된다. 서버에서는 전송되어 오는 XML파일들을 정

보 관리와 정보 제공을 위하여 병합작업을 한다.



(그림 4) XML 파일

그림 5는 웹을 통하여 외부의 사용자에게 정보를 제공하고자 할 경우에 나타나는 화면이다. 그림 5의 HTML 파일은 XSLT파일을 이용하여 생성이 된다. 그리고 여러 계산노드(hamel, nec)로부터 usage정보가 취합되어 나타난 것이 그림 5에 나타나 있다.

(그림 5) 여러 노드로부터 통합된 usage 정보

## 5. 결론

컴퓨팅 시스템에서는 사용자 작업이 수행된 후 생성되는 사용량(usage) 정보는 시스템 관리 및 과금(accounting)을 위하여 중요한 정보이다. 이러한 정보는 데이터베이스를 이용하여 저장하고 웹을 통하여 정보를 제공하는 것이 일반적인 시스템 구성이다.

본 논문에서는 KISTI에서 시스템 usage 정보를 데이터베이스를 사용하지 않고 XML 파일로 처리하는 사용량 보고 시스템(Usage Reporting System, URS)을 구현했던 내용을 소개하였다. URS를 통하

여 다수의 시스템 usage 정보를 수집하여 저장하고 처리해 보았다. DB를 사용하지 않기 때문에 reporting 시스템의 구조가 간단해 질수 있는 장점이 있다. usage 정보를 제공하여야 하는 개인이나 중소규모의 전산센터에서 이 시스템을 이용하여 쉽게 구현이 가능하다.

향후 과제로는 RUS-WG에서 제시하는 XML DB를 사용하는 시스템을 구현한 후 처리성능시험을 하고자 한다.

## 참고문헌

- [1] Rajkumar Buyya, Steve Chapin, David DiNucci, "Architectural Models for Resource Management in the Grid", The 1st IEEE/ACM International Workshop on Grid Computing
- [2] 이상산, "국가 그리드 기반 구축을 통한 고성능 자원 통합 활용", 정보통신연구진흥, vol.13.
- [3] 김상완,곽재혁, 황영철, 이필우, "어카운팅을 위한 그리드 사용자 환경 설계", 2005, 한국컴퓨터종합학술대회
- [4] 성진우, 이상동, 이영주, 김중권, "XML을 이용한 openPBS Usage 정보 처리 시스템의 설계 및 구현", 2003 KOSTI
- [5] [pbsaccounting.sourceforge.net](http://pbsaccounting.sourceforge.net)
- [6] [cvs.sdsc.edu/cgi-bin/cvsweb.cgi/tg-acctmgmt/usageRecord/perl/](http://cvs.sdsc.edu/cgi-bin/cvsweb.cgi/tg-acctmgmt/usageRecord/perl/)
- [7] Viter Hazlewood, Ray Bean, Kenneth Yoshimoto, "SNUPI: A grid accounting and performance system employing portal services and RDBMS back-end", The 2th LCI International Conference on Linux Clusters: The HPC Revolution 2001
- [8] [www.gridforum.org](http://www.gridforum.org)
- [9] [www.teragrid.org](http://www.teragrid.org)
- [10] [public.eu-egee.org](http://public.eu-egee.org)
- [11] [www.csar.cfs.ac.uk](http://www.csar.cfs.ac.uk)
- [12] [glite.web.cern.ch/glite](http://glite.web.cern.ch/glite)
- [13] [www.ngs.ac.uk/index.html](http://www.ngs.ac.uk/index.html)
- [14] [www.mgrid.umich.edu/projects/accounting.html](http://www.mgrid.umich.edu/projects/accounting.html)
- [15] [www.doc.ic.ac.uk/%7Eesjn5/GGF/rus-wg.html](http://www.doc.ic.ac.uk/%7Eesjn5/GGF/rus-wg.html)
- [16] John D. Ainsworth, Jon MacLaren, John M. Brooke, "Implementing a Secure, Service Oriented Accounting System for Computational Economies", CCGrid 2005