



ACCOUNT MANAGEMENT PRODUCTIZED

DESCRIPTION OF DJRA1.4 DELIVERABLE

Document Filename:	BG-DJRA1.4-v.1.1
Activity:	JRA1
Partner(s):	PSNC, VU, EENET, KTH, IMCS UL
Lead Partner:	PSNC
Document classification:	PUBLIC

Abstract: The report describes the current state of deployment of the account management software developed in research activity (JRA1). The deliverable DJRA1.4 is a demonstrator and the report is a prove of what has been done until December 2007. It aims to be a help for administrators that are going to install it on their sites.





Document review and moderation

	Name	Partner	Date	Signature
Released for moderation to				
Approved for delivery by				

Document Log

Version	Date	Summary of changes	Author
0.1	25/10/2007	Initial version	Michal Jankowski Norbert Meyer
0.2	07/12/2007	Description of the testbed started Added VUS test procedures	Michal Jankowski
0.3	28/12/2007	DGAS tests added updated testbed resources	Hardi Teder
0.9	31/12/2007	Limitations described DGAS tests updated Minor enhancements	Michal Jankowski
1.0	7/01/2008	Minor updates, modified executive summary, abstract and summary	Norbert Meyer
1.1	8/01/2008	Minor enhancements, references updated	Michal Jankowski



Contents

GLOSSARY.....	4
1 EXECUTIVE SUMMARY.....	6
2 OBJECTIVES.....	8
3 INSTALLATION AND CONFIGURATION GUIDE.....	10
3.1 VUS.....	10
3.2 DGAS.....	12
3.2.1 CE Client.....	13
3.2.2 HLR SERVER.....	15
4 TEST PROCEDURES.....	19
4.1 VUS.....	19
4.1.1 Testing different authorization plugins.....	19
4.1.2 Testing the dynamic mapping/unmapping.....	20
4.2 DGAS.....	20
4.2.1 HLR Server Side General Tests.....	20
4.2.2 HRL Regression test suite.....	21
4.2.3 CE Client Side TESTS.....	25
4.2.4 User test.....	26
5 TESTBED INSTALLATION.....	28
5.1 CURRENT STATUS OF THE TESTBED INSTALLATION.....	28
6 LIMITATIONS AND KNOWN PROBLEMS.....	30
7 SUMMARY.....	31
REFERENCES.....	32



GLOSSARY

CE	Computing Element
DGAS	Distributed Grid Accounting System
DHCP	Dynamic Host Configuration Protocol
DN	Distinguished Name
DNS	Domain Name Service
EGEE	Enabling Grids for E-science
FQAN	Fully Qualified Attribute Name
GID	Group Identifier
GRAM	Grid Resource Allocation and Management
GSI	Grid Security Infrastructure
HLR	Home Location Registers
IP	Internet Protocol. Also IP address.
JDL	Job Description Language
LCAS	Local Center Authorization Service
LCMAPS	Local Credential MAPping Service
LRMS	Local Resource Management System
LVM	Logical Volume Management
NAT	Network Address Translation
QoS	Quality of Service
R-GMA	Relational Grid Monitoring Architecture
RPM	Red Hat Package Manager
RSL	Resource Specification Language
SSH	Secure Shell
SLA	Service Level Agreement
SLS	Service Location Server
Torque	An open source local resource management system providing control over batch jobs and distributed compute nodes.
SOAP	Simple Object Access Protocol
TCP	Transmission Control Protocol
UID	User Identifier
VM	Virtual Machine



ACCOUNT MANAGEMENT PRODUCTIZED
Description of DJRA1.4 deliverable

VO	Virtual Organization
VOMS	Virtual Organization Membership Service
VPN	Virtual Private Network
VUS	Virtual User System
WM	Workload Manager
WMS	Workload Management System
WN	Worker Node



1 EXECUTIVE SUMMARY

The report describes the current state of development and deployment of the account management software developed within the research activity (JRA1) of the BalticGrid project [3]. At this stage, the software achieved production quality and it is ready for installation in the real environment.

The report describes also accounting system deployed in the testbed –Distributed Grid Accounting System [13]. DGAS is going to replace the currently used accounting system in the project –Apel, as it better fits to the accounting requirements [2]. DGAS is a production quality software developed by and deployed in DataGrid, EGEE and INFN Grid.

The following document aims to be a help for administrators that are going to deploy our account management software and DGAS.

The former *Report on implementation of JRA1 prototypes* [5] (released in May 2007) summarized work on JRA1 software; both SLA one and user management one. The report contained description of the software, installation instructions and configuration manual. The report was formally part of deliverables **DJRA1.2** and **DJRA1.3**.

Deliverable **DSA1.7** “Report on implementation of logging system and accounting policies” was prepared by SA1 activity in collaboration with JRA1.

The first DGAS HLR server was installed and connected with Computing Element client. This work was done in collaboration with SA1. Practical tests with this accounting system started. Contacts with DGAS developers (INFN) were established in order to get support for the installation, configuration and further integration.

The software developed within the JRA1 activity was presented at the **Third AHM meeting** in Tallinn (May 2007). Also a schedule of work for JRA covering the time period until the end of 2007 was defined there. Major actions planned at the Third AHM (May 2007) regarding accounting and user account management were as follow:

1. DGAS (June 2007)
 - Established contacts with DGAS developers (INFN)
 - Solving problems with DGAS integration
 - Final integration of DGAS and VUS
 - Installation on JRA1 testbed
3. Further work on VUS (Virtual User Account System, responsible: PSNC)
 - The Installation Guide will be prepared until the end of May 2007
 - All JRA1 sites will start the installation
 - Report (deliverable) to be ready until end of May
4. Integration of DGAS and VUS (July-August)
5. Deployment in SA1
 - Discussion about the intermediate state, i.e. how to make available JRA1 results on wider testbed installation
 - Will make available RPMs, scripts and installation guides on internal web area
 - To be used later by SA1
 - First common discussions between JRA1 and SA1 will start at the meeting in Stockholm (Dec. 2007).



ACCOUNT MANAGEMENT PRODUCTIZED

Description of DJRA1.4 deliverable

The process of building JRA1 testbed was started in July 2007. During the installation on the remaining JRA1 sites we find out some problems, which required an update of the earlier defined schedule:

The VUS system was implemented on the gLite computing element (gLite CE). At the installation on other sites we detected serious problems of incompatibility between gLite CE and LCG CE. It was not expected during the development. Mainly all BalticGrid sites updated the software on computing elements with LCG CE. Therefore we decided in JRA1 to adopt the VUS system for LCG CE. The development was finished in October. An updated version of the installation guide was prepared by PSNC and is available at the internal BG web site.

The structure of the document is as follows. The document reminds objectives and motivation that are behind the account management task of JRA1 (chapter 2). Especially useful for administrators is chapter 3 containing installation and configuration guide, followed by description of test procedures (chapter 4). Then, the current testbed is described (chapter 5). Chapter 6 deals with problems and limitations we have identified. Finally, the document is summarized in chapter 7.



2 OBJECTIVES

The Joint Research Activity of BalticGrid project (JRA1) works on SLA security and enforcement, account management and accounting and will deploy these services in the BG in co-operation with SA1 and with support of SA2. In this document we focus on account management and accounting task. The main objective of JRA1 concerning this task is as follows 3:

- to design, develop and deploy, a non-intrusive, standards based user account management system.

BalticGrid project focuses on extending EGEE infrastructure to the Baltic states, so that using EGEE software –gLite/LCG is assumed. Another assumption is reusing existing solutions as much as possible. To fulfil these assumptions, the proposed systems are based on existing tools, previously developed by the project partners and the EGEE middleware.

The main aim of user management system is controlled, secure access to grid resources. The scope of features offered by such a system may vary significantly, depending on the needs of users and administrators. The user management system should fulfil, well defined BalticGrid requirements. In order to identify these requirements (described in detail in the report [4]) we performed a survey for administrators and application developers. We made a number of assumptions for our system, based on the survey and our experience:

- It could not interfere in the sites that are not going to use the system.
- The proper level of security, job isolation, automation of the administrative work and compliance with standards must be achieved.
- Support for accounting and easy integration with the accounting system to be deployed in BalticGrid.

We found, that in order to follow the objective, fulfil the requirements and assumptions we should extend the existing gLite tools as they are not satisfactory from that point of view. We have designed and implemented a system based on Virtual User System [12] of PSNC, integrated with gLite LCMAPS module.

The survey and good practice implied the following requirements for the accounting and logging system 2:

- It must be possible to identify the user and context (VO, role, capabilities, time) of any action that is subject to local logging or accounting. Thus it should be possible to fetch all the accounting and logging information from the standard system mechanisms (such as e.g. Unix accounting). In many applications it may be a sufficient amount of data.
- The system should be capable of storing non-standard accounting data. This is especially useful when the node provides access to non-standard resources (e.g. equipment in a virtual laboratory). The data may be put by any external application through a special API.
- The accounting data must be easily accessible for all parties involved in a particular action. The resource provider wants to access all information connected with resources it provided to a grid. A virtual organization would like to see all information on resources used by its members so that VO can make internal decisions according to that information. The end user who submitted some jobs would like to access all information on resources consumed by those jobs.
- The accounting mechanism must be able to gather the accounting data for the VO as well as for the groups of resources.



ACCOUNT MANAGEMENT PRODUCTIZED

Description of DJRA1.4 deliverable

The above assumptions are fulfilled to a great extent by combination of DGAS and VUS software (the limitations are described in the chapter 6).

The DGAS and VUS software is currently working on a few testbed sites (described in the chapter 5). The next step is to deploy this product on the production sites of the BalticGrid. This must be done in cooperation with SA1 activity, that is responsible for administrating infrastructure in the BalticGrid project.



3 INSTALLATION AND CONFIGURATION GUIDE

3.1 VUS

Introduction

The gLite WMS System [7] is responsible for the distribution and management of tasks across Grid resources. In the WMS user management on local resources is realised by modules called Local Center Authorization Service (LCAS), Local Credential MAPPING Service (LCMAPS) [9] and underlying Globus Gatekeeper and GridFTP services [6]. Typically, these services are located on Computing Element (in gLite architecture).

VUS extends functionality of LCMAPS and it is implemented as set of LCMAPS plugins. The plugins follow the original LCMAPS pattern of authorisations, including VOMS [32,32] support, so the authorisation configuration remains the same, but they offer an additional VUS features.

Prerequisites

1. Installation of Computing Element glite-CE or lcg-CE.
2. MySQL (other SQL DBMS possible, but not tested).
3. STDC++ library (libstdc++ RPM).
4. UNIX ODBC with driver for MySQL (may be installed and configured by VUS configuration script if not done before).

Installation for glite-CE

From the location <http://www.balticgrid.org/Internal/JRA1/vus4glite1> one should download the following:

1. **glite-security-lcmaps-plugins-vus-basic-0.9.1-1.i386.rpm** – the VUS enabled LCMAPS plugins.
2. **glite-security-vus-common-0.9.1-0.i386.rpm** – the base library for VUS.
3. **vus-config.tar.gz** – configuration scripts and instructions

As root, install the VUS RPMs:

```
rpm -i glite-security-vus-common-0.9.1-0.i386.rpm  
rpm -i glite-security-lcmaps-plugins-vus-basic-0.9.1-1.i386.rpm
```

Installation for lcg-CE

From the location <http://www.balticgrid.org/Internal/JRA1/vus1.0.1> one should download the following:

1. **lcg-ce-lcmaps-plugins-vus_gcc3_2_2-1.0.1-1.i386.rpm** – the VUS enabled LCMAPS plugins.
2. **lcg-ce-vus-common_gcc3_2_2-1.0.1-1.i386.rpm** – the base library for VUS.
3. **vus-config-1.0.1.tar.gz** – configuration scripts and instructions

As root, install the VUS RPMs:

```
rpm -i lcg-ce-vus-common_gcc3_2_2-1.0.1-1.i386.rpm  
rpm -i lcg-ce-lcmaps-plugins-vus_gcc3_2_2-1.0.1-1.i386.rpm
```



Initial Configuration

Unpack the archive `vus-config.tar.gz` and change to the unpacked directory.

Become root and check if `$GLITE_LOCATION` is set correctly.

Run `vus-config.sh`. The interactive script will ask you set of questions, the default values (displayed in brackets) you may set by pressing Enter. The script performs the following actions (each action may be skipped if already performed, the script may be stopped at any moment by Ctrl+C):

1. Creating VUS database user and database.
2. Creating empty tables in the database.
3. Installing ODBC database access tools (UnixODBC and MySQL driver).
4. Configuring ODBC database access (definition of ODBC data source).
5. Configuring VUS -the configuration file `$GLITE_LOCATION/etc/vus.conf` is created, the old configuration is removed.
6. Configuring LCMAPS (`$GLITE_LOCATION/etc/lcmaps`) -this affects how the authorisation is performed. Your current configuration will be completely overwritten, so you may have to perform this step manually (see next section). The old configuration is stored respectively in `lcmaps.db.2` and `lcmaps.db.gridftp.2`.

Advanced Configuration

ODBC

See <http://www.unixodbc.org> for more details.

VUS

The configuration file is `$GLITE_LOCATION/etc/vus.conf`. Configuration parameters are presented in Tab. 1.

VUSDataBase	VUS database name (ODBC data source name)
VUSDataBaseUser	database user name
VUSDataBaseAuth	password to the database
VUSMapTimeout	the minimal user mapping time in seconds
ClearAccountScript	path to the script, that is run on the account while unmapping user, may be used for removing any personal data left by the user.

Tab. 1 VUS Configuration Parameters

IMPORTANT: it is strictly recommended to set ownership of `vus.conf` to root and access rights to 600 for the security reasons.

LCMAPS

File `$GLITE_LOCATION/etc/lcmaps/lcmaps.db` defines authorisation policies for Gatekeeper.

File `$GLITE_LOCATION/ETC/LCMAPS/LCMAPS.DB.GRIDFTP` defines authorisation policies for GridFTP.

The standard LCMAPS plugins in the above files may be replaced by corresponding VUS-enabled ones Tab. 2 presents the plugins.



Standard LCMAPS plugin	Corresponding VUS enabled plugin	User mapped to ...
lcmaps_localaccount.mod	lcmaps_vus_localaccount.mod	local -personal account, exactly as in gridmap-file
lcmaps_poolaccount.mod	lcmaps_vus_poolaccount.mod	one of the pool accounts – VUS virtual accounts
lcmaps_voms_localaccount.mod	lcmaps_vus_voms_localaccount.mod	local account, authorization is based on VOMS attributes, users of matching VOMS attributes are mapped to the same account
lcmaps_voms_poolaccount.mod	lcmaps_vus_voms_poolaccount.mod	one of the pool accounts, authorization is based on VOMS attributes – VUS virtual accounts

Tab. 2 LCMAPS plugins

The VUS-enabled plugins use the same arguments as their standard counterparts, except `-override_inconsistency`.

The simplest way to enable VUS on your site without modifying the authorisation policy is to do the above replacement and removing the unsupported argument.

More details on: <http://www.nikhef.nl/grid/lcaslcmaps>

Further Information and Help

- Look at References section.
- Developers support:
 - Michał Jankowski <jankowsk@man.poznan.pl>

3.2 DGAS

Definition of concepts

The **Home Location Register** [14](HLR) service is the part of DGAS that is responsible for keeping the accounting information for both grid users (**User HLR**) and grid resources (**Resource HLR**). It receives the accounting information, the so called Usage Records from the grid resources, and stores them for later retrieval. These usage records are the basis for the job cost computation¹, the phase in which the HLR computes the cost for a given job. The job cost can then be debited to the grid user and credited to the grid resource, thus implementing an economic accounting for the the grid activities of the single users. Information can be gathered from the HLR service on a per user, per resource, per job basis.

The **URCollector** (previously **GIANDUIA**[15]) daemon is installed on a Computing Element (or a generic grid resource) in order to collect the usage records of the executed user jobs by analysing LRMS logs. The **DGAS Pushd** daemon is then responsible for sending them to the DGAS HLR services.

The **Price Authority** (PA) is a key component of the DGAS toolkit, providing the features necessary for Economic Accounting. In a few words, a PA server is an entity that assigns the prices to the subset of grid resources within its administrative domain. The prices, that are kept in a historic price database, can be assigned manually or using different dynamic pricing algorithms. The price of a resource is



used to compute the cost for a job. The given cost can then be charged to the user that submitted the job.

DGAS2APEL tool may be run on Resource HLR in order to provide APEL accounting system with data. In that case, the standard way APEL retrieves the accounting from CEs should be switched off.

Deployment Strategy

One Resource HLR per one resource provider is intended to manage accounting information. So that, it should gather accounting from one or few sites (CEs). In the simplest case, it may be located on CE, but one should take into account the performance aspects. In BalticGrid we plan to deploy these services for all providers (institutions) and clients for all CEs in the first step. This document describes this step in the current version. Then, the accounting from these HLRs may be forwarded to APEL using DGAS2APEL.

The so called second level HLR, one per the whole project, will be deployed in the next step. The second level HLR stores aggregated accounting data from many resource providers.

The optional steps are:

- Deploying User HLRs one per VO in order to provide VO managers and single users with control over their resource usage. The HLR client must be installed on UI in order to give the user tools for querying the HLR service. No configuration on UI is required.
- Deploying PA services necessary to enable economic accounting.

3.2.1 CE CLIENT

Requirements

The software requirements are Scientific Linux v.3, Java sdk 1.4.2, .

The standard port used by DGAS server is 56568, so it must be open on firewalls for out coming connections.

Installation

Create /etc/apt/sources.list.d/dgas.list containing:

```
rpm http://grid-it.cnaf.infn.it/apt ig_sl3-i386 3_0_0_test
rpm http://grid-it.cnaf.infn.it/apt ig_sl3-i386 3_0_0
```

```
apt-get update
apt-get install glite-dgas-hlr-clients
```

Configuration

Copy the templates of configuration files and then modify required parameters:

```
cd /opt/glite/etc
cp dgas_atmClient.conf.template dgas_atmClient.conf
cp dgas_gianduia.conf.template dgas_gianduia.conf
cp dgas_ce_pushd.conf.template dgas_ce_pushd.conf
```

Parameters to be modified:

File dgas_atmClient.conf



```
res_acct_bank_id = "hostname:portnumber:X509CertSubject"
```

This should be set to the contact string of the DGAS HLR server that manages the CE's account(s).
The default portnumber is 56568

The host certificate subject doesn't need to be specified, but if it is not specified, the trailing ":" after the port number is essential!

The other parameters in this configuration file can be left as they are (they are meaningful only if there is a price authority server).

File dgas_gianduia.conf:

```
siteName = "ALOE-TEST.PSNC"
```

or whatever is your official site name within baltic grid.

```
lrmsType = "pbs"
```

(if you use PBS/Torque).

```
pbsAcctLogDir = "/var/spool/pbs/server_priv/accounting/"
```

Please verify that this default value is correct. The parameter has to point to the directory that contains your PBS accounting log files (usually names after the date of their generation, e.g. 20070314). For CEs with LSF the equivalent parameter lsfAcctLogDir has to be configured instead.

```
ignoreJobsLoggedBefore = "2006-01-01"
```

This parameter allows you to specify beginning from when jobs should be accounted. The default value is "2006-01-01", but you can set this parameter at the beginning of this month, or even for today.

for glite-CE:

```
useCEJobMap = "yes"
```

```
ceJobMapLog = "/var/log/glite/accounting/blahp.log"
```

For details see the comments in the config file and parameter BLAHPD_ACCOUNTING_INFO_LOG in the /opt/glite/etc/blah.config If the above parameter was not set, it should be: BLAHPD_ACCOUNTING_INFO_LOG=/var/log/glite/accounting/blahp.log

for lcg-CE:

```
useCEJobMap = "yes"
```

```
ceJobMapLog = "/opt/edg/var/gatekeeper/grid-jobmap"
```

These two parameters allow to the urcollector to find grid-related information for the jobs in the log files that are written by the LCG CE with patch #898, please read on the web page <https://savannah.cern.ch/patch/?898> about this patch for the CE (in case you haven't installed it yet). Using that patch is strongly recommended, since it allows to map such information like the user's FQAN to the local accounting information (e.g. only local user ID) from the LRMS log files.



In case you don't want to use the patch (and accept having only local accounting information, except the VO that is determined as described above), you can set `useCEJobMap = "no"`.

for glite-CE:

```
glueLdifFile = "/opt/lcg/var/gip/ldif/lcg-info-static-cluster.ldif"
```

for lcg-CE:

```
glueLdifFile = "/opt/glite/etc/glite-ce-ce-plugin/out.ldif"
```

The other parameters of this configuration file can be left as they are.

File `dgas_ce_pushd.conf`

```
#defaultVOUserHLRList = "/opt/glite/etc/dgas_VOUserHLR.conf"  
#defaultUserHLR = "<userHLR_hostname>:<port>:<host_cert_subject>"  
forceLocalOnly = "yes"
```

`forceLocalOnly = "yes"` means that usage records will be sent only to the HLR server that manages the CE's account(s) (Resource HLR). The User HLR will be used if users and their VOs are interested in accounting data.

```
siteName = "ALOE-TEST.PSNC"
```

The same as in the `dgas_giaundia.conf`

All other parameters can be left as they are.

Starting the Deamons

```
cd /etc/init.d
```

```
ln -s /opt/glite/sbin/glite-dgas-pushd  
chkconfig --add glite-dgas-pushd  
service glite-dgas-pushd start
```

```
ln -s /opt/glite/sbin/glite-dgas-urcollector  
chkconfig --add glite-dgas-urcollector  
service glite-dgas-urcollector start
```

3.2.2 HLR SERVER

Requirements

The software requirements are Scientific Linux v.3, Java sdk 1.4.2, .

The machine must have a public IP and domain addresses.



The standard port used by DGAS HLR server is 56568, so it must be open on firewalls for all CEs.

Installation

Create `/etc/apt/sources.list.d/dgas.list` containing:

```
rpm http://grid-it.cnaf.infn.it/apt ig_sl3-i386 3_0_0_test
rpm http://grid-it.cnaf.infn.it/apt ig_sl3-i386 3_0
```

```
apt-get update
apt-get install glite-dgas-hlr-service
```

It may occur a problem while `apt-get install glite-dgas-hlr-service`:

The following extra packages will be installed:

mysql (3.23.58-16.RHEL3.1)

The following packages will be REMOVED:

MySQL-server (4.1.11-0)

The following NEW packages will be installed:

mysql (3.23.58-16.RHEL3.1)

perl-DBD-MySQL (2.1021-4.EL3)

If so, before installing `glite-dgas-hlr-service` do the following:

```
apt-get install MySQL-server
apt-get install MySQL-client
apt-get install MySQL-shared-standard
rpm -ivh
ftp://ftp.nluug.nl/pub/sunsite/distributions/smeserver/releases/obsolete/7.
1/smeos/i386/CentOS/RPMS/perl-DBD-MySQL-2.9004-3.1.i386.rpm

apt-get install glite-dgas-hlr-service
```

Request for Host Certificate

Before you request you need globus postinstall configuration:

```
/opt/gpt/sbin/gpt-postinstall
/opt/globus/setup/globus/setup-gsi
```

And CA configured, e.g.:

```
wget http://grid.eenet.ee/CA/ca_Estonia-certrequest-config-1.0-1.noarch.rpm
```



```
rpm -i ca_Estonia-certrequest-config-1.0-1.noarch.rpm
```

or manually copy globus-host-ssl.conf.*, globus-user-ssl.conf.*, grid-security.conf.* to /etc/grid-security/certificates

and create links /etc/grid-security/globus-host-ssl.conf, /etc/grid-security/globus-user-ssl.conf, /etc/grid-security/grid-security.conf to these files

Request and mail request e.g.:

```
grid-cert-request -host lcg2-ce.hep.kbfi.ee -force  
cat /etc/grid-security/hostcert_request.pem | mail ca@grid.eenet.ee
```

Save the signed cert to /etc/grid-security/hostcert.pem

Configuration

```
cd /opt/glite/etc  
cp dgas_hlr.conf.template dgas_hlr.conf
```

Modify parameters:

```
hlr_sql_user  
hlr_sql_password
```

RECOMENDED:

Configure MySQL not to listen for remote connections, if installing a dedicated DGAS HLR for which only local access to the database is required - make sure that the line "skip-networking" is specified under the configuration group "[mysqld]" in the configuration file /etc/my.cnf. This is especially important if no password is used for MySQL (see parameters 'hlr_sql_user' and 'hlr_sql_password' of the HLR configuration file).

Add subjects of certificates of the authorized CEs to /etc/grid-security/gridmap-file, e.g.

```
"/C=PL/O=GRID/O=PSNC/CN=host/aloe-test.man.poznan.pl" nobody
```

Preparing HLR Database

Create the HLR database:

```
$GLITE_LOCATION/sbin/glite-dgas-hlrd-dbcreate <mysqlRootPassword>
```

Add VO (in case of Resource HLR, VO refers to resource provider), resource group and resources, e.g.

```
/opt/glite/sbin/glite-dgas-hlr-addvo -f "PSNC"
```

```
/opt/glite/sbin/glite-dgas-hlr-addgroup -f "PSNC" -g "ALOE-TEST.PSNC"
```



```
/opt/glite/sbin/glite-dgas-hlr-addressource -r "aloe-  
test_man_poznan_pl_balticgrid" -S "/C=PL/O=GRID/O=PSNC/CN=host/aloe-  
test.man.poznan.pl" -g "ALOE-TEST.PSNC" -f "PSNC" -c "aloe-  
test.man.poznan.pl:2119:/jobmanager-pbs-balticgrid"
```

```
/opt/glite/sbin/glite-dgas-hlr-addressource -r "aloe-  
test_man_poznan_pl_balticgrid_local" -S "/C=PL/O=GRID/O=PSNC/CN=host/aloe-  
test.man.poznan.pl" -g "ALOE-TEST.PSNC" -f "PSNC" -c "aloe-  
test.man.poznan.pl:balticgrid"
```

OPTIONS:

-f --vo_id <voID>	Specifies the VO to which the group is associated in the HLR database (your institution name, as set in dgas_ce)
-g --gid <groupID>	Specifies the group to which the resource is associated in the HLR database (site name in dgas_ce_pushd.conf)
-r --rid <resID>	String that identifies the resource in the database. It can be set up freely by the HLR administrator, It must be unique per every resource added.
-c --ceId <ceID>	The global grid ID of the resource (CE contact string for grid jobs, <CE name>:<queue name> for local jobs)
-S --certSubject <cert>	The subject of the resource's host certificate.

Starting the service

```
cd /etc/init.d  
ln -s /opt/glite/sbin/glite-dgas-hlrd  
chkconfig --add glite-dgas-hlrd  
service glite-dgas-hlrd start
```

Further Information and Help

- Look at References section.
- Developers support:
 - Andrea Guarise <guarise@to.infn.it>
 - Rosario Piro <rosario.piro@to.infn.it>
 - Giuseppe Patania <patania@to.infn.it>
- People from BalticGrid with some DGAS experience:
 - Michał Jankowski <jankowsk@man.poznan.pl>
 - Ilja Livenson <ilja_l@tudeng.ut.ee>



4 TEST PROCEDURES

4.1 VUS

4.1.1 TESTING DIFFERENT AUTHORIZATION PLUGINS

There are 4 of them, but the most common are local account and VOMS pool account.

Assumptions:

- /opt/lcg/etc/lcmaps/lcmaps.db and /opt/lcg/etc/lcmaps/lcmaps.db.gridftp are configured exactly as they are in lcg-vus-config.tar.gz from the installation.
- job executable is /usr/bin/whoami, so you can easily check to which account the user was mapped.
- Use `edg-job-submit -r <your_site>` on your UI for submitting the job and `edg-job-get-output` for getting the result.

Tests of plugins:

a) local account

- comment (using #) "voms" policies in lcmaps.db* files
- /etc/grid-security/gridmapfile should contain ONE LINE with your personal DN and local account name, e.g.: "/C=PL/O=GRID/O=PSNC/CN=Michal Jankowski" jankowsk
- submit a job and check the output as it is finished - it should be the account name (e.g. jankowsk)
- gridftp a file to and from the CE

b) pool account

- comment (using #) "voms" policies in lcmaps.db* files
- /etc/grid-security/gridmapfile should contain ONE LINE with your personal DN and .<pool of accounts>, e.g.: "/C=PL/O=GRID/O=PSNC/CN=Michal Jankowski" .balticgrid
- submit a job and check the output as it is finished - it should be an account name out from the given pool (e.g. balticgrid01)
- gridftp a file to and from the CE

c) VOMS local account

- comment (using #) "non-voms" policies in lcmaps.db* files
- /etc/grid-security/gridmapfile should contain ONE LINE with some FQAN matching your VOMS proxy and local account name, e.g.:
"/balticgrid/Role=NULL/Capability=NULL" balticgrid02
- submit a job and check the output as it is finished - it should be the account name (e.g. balticgrid02)
- gridftp a file to and from the CE

d) VOMS pool account



- comment (using #) "voms" policies in lmaps.db* files
- /etc/grid-security/gridmapfile should contain ONE LINE with some FQAN matching your VOMS proxy and .<pool of accounts>, e.g.: "/balticgrid/Role=NULL/Capability=NULL" .balticgrid
- submit a job and check the output as it is finished - it should be an account name out from the given pool (e.g. balticgrid01)
- gridftp a file to and from the CE

4.1.2 TESTING THE DYNAMIC MAPPING/UNMAPPING

The test touches pool and VOMS pool plugin (see 4.1.1 b and 4.1.1 d) You need n+1 (at least 2) different certificates, where n is number of accounts in the pool. Names of accounts of pool <pool of accounts> are names of files: /etc/grid-security/gridmapdir/<pool of accounts>*. For this test you will want a small pool, possibly of one account, so you may either temporary rename some files of existing pool or prepare a special pool for tests. E.g. for "balticgrid pool from the above examples:

```
[root]# ls /etc/grid-security/gridmapdir/balticgrid*  
/etc/grid-security/gridmapdir/balticgrid01
```

Submit jobs subsequentially using proxies generated from different certificates. If the previous job(s) are running on n accounts, the submission should fail. If some accounts are idle for at least VUSMapTimeout seconds (a parameter from \$LCG_LOCATION/etc/vus.conf), the job should be successfully submitted on one of these accounts.

4.2 DGAS

There are several possibilities to test the DGAS installations. The easiest way is to use DGAS command line interfaces – glite-dgas-*. There are also several point of views: ressource administrator, ressource user, VO administrator etc. In this section we describe the ressource administrator tests, ressource user tests and DGAS built-in tools called “HRL regression test suite”.

For describing the tests we use:

- Ressource HRL = lcg2-ce.hep.kbfi.ee
- Site = kriit.eenet.ee

For using the tests on other sites these values can be changed.

4.2.1 HLR SERVER SIDE GENERAL TESTS

The first thing to test is **if the server is up and running**:

```
$ glite-dgas-ping -s lcg2-ce.hep.kbfi.ee  
Server alive.
```



For getting more information about site:

```
$ glite-dgas-hlr-resinfo -s lcg2-ce.hep.kbfi.ee -c  
"kriit.eenet.ee:2119/jobmanager-pbs-balticgrid"  
  
|kriit_eenet_ee_balticgrid|kriit.eenet.ee:2119/jobmanager-pbs-  
balticgrid|kriit.eenet.ee:2119/jobmanager-pbs-  
balticgrid|eenet.ee_EENet|0|wall_time=0,cpu_time=0,job_number=0|
```

Testing DGAS site HRL:

```
$ glite-dgas-hlr-query -H lcg2-ce.hep.kbfi.ee -Q resourceAggregate -  
A "count(dgJobId)" -r "%kriit.eenet.ee%"  
  
MIN(date) |MAX(date) |count(dgJobId) |  
2006-06-16 15:49:41|2008-01-07 17:15:10| 39515|
```

Then run a grid job in the cluster and run the same command. The count of grid jobs must be increased by 1.

glite-dgas-hlr-query is a very powerful tool and for more information read the manual.

4.2.2 HRL REGRESSION TEST SUITE

The content of this paragraph is from DGAS new documentation draft. It is available via DGAS developers: Andrea Guarise <andrea.guarise@to.infn.it>

The regression test suite is available on the HLR that allows to perform a set of functional tests on an HLR service in order to determine if there are problem in is normal duty cycle.

The test suite is located on the server under the directory:

```
/${GLITE_LOCATION}/libexec/dgastests  
or  
/${GLITE_LOCATION}/libexec/dgas-hlr-service-tests
```

depending on the version of the software.

It is composed by a test runner command:

```
[root@grid003 dgas-hlr-service-tests]#./glite-dgas-testrun.pl
```

You must specify the configuration file!

USAGE:

```
testRun <OPTIONS>
```

OPTIONS are:

```
--force      Force exiting with exit status 0  
--verbose    Verbose output  
--debug      Adds debug information  
--conf       MANDATORY, used to specify the main configuration file  
--logdir     Used to specify where to log files
```



The basic usage is:

```
./glite-dgas-testrun.pl --conf test.conf --logdir /tmp/ full.test
```

Which runs the full set of test and put the logs in the /tmp/ directory.

The command performs a set of tests specified with a proper syntax in the .test file specified on the command line and produces an output like this:

```
[root@t2-hlr-01 dgastests]# ./glite-dgas-testrun.pl --conf test.conf --
logdir /tmp/ full.test
Check environment
0) Check GLITE_LOCATION                success
Check host certificate
1) host certificate file exists         success
2) host certificate file is a valid certifcate  success
3) host cert has enough validity time    success
Check MySQL daemon
4) MySQL daemon status is OK           failure (1),expected: 0
5) MySQL daemon is alive               failure (1),expected: 0
Check HLR daemon
6) hlr conf file exists                success
7) hlr database exists                 failure (1),expected: 0
8) hlr_tmp database exists             failure (1),expected: 0
9) glite-dgas-hlrd status              success
10) crontab runs translateDb           failure (1),expected: 0
11) hlrServer daemon is alive          success
12) hlrTqd daemon is alive             success
13) hlrHAD daemon is alive             success
14) glite-dgas-hlrd in /etc/rc.d/init.d/ success
error reading info for service glite_dgas_hlrd: No such file or directory
15) glite-dgas-hlrd in chkconfig       success
16) glite-dgas-hlrd service compatiblity success
Check client-server communication
17) HLR answers to pingClient          success
Check account management
18) Add resource VO metagroup          success
19) Add resource group                 success
20) Add resource                       success
21) Fail to add duplicate metagroup     success
22) Fail to add duplicate group        success
```



ACCOUNT MANAGEMENT PRODUCTIZED
Description of DJRA1.4 deliverable

23) Fail to add duplicate resource	success
24) Query resource by rid	success
25) Query resource by ceId	success
26) Add user	success
27) Fail to add duplicate user	success
28) Query user by DN	success
29) Delete user	success
30) Delete resource	success
31) Delete resource group	success
32) Delete resource VO metagroup	success
33) Fail to delete non existent user	success
34) Fail to query non existent user	success
35) Fail to delete not existent resource	success
36) Fail to query non existent resource	success
37) Fail to delete not existent group	success
38) Fail to delete not existent VO metagroup	success
Check ATM client- ENGINE communication	
39) Call ATM Client Route 3	success
40) Call ATM Client Route 2	success
41) Call ATM Client on duplicate job (69)	success
42) Insert Job with cpuTime < 0, start = 0	success
43) Insert Job with userVO!=fqan, start = 0	success
44) Check that route 3 job is in trans_queue	success
45) Check that route 2 job is in trans_queue	success
test qMgrd - HLR communication (can take several minutes)	
46) Wait for route 3 job to be accountd	success
47) Wait for route 2 job to be accountd	success
48) Running hlrTranslateDb	success
49) Call ATM Client on duplicate job (70)	success
50) Call ATM Client on duplicate job (71)	success
test hlrQueryUser hlrQueryResource	
Unable to receive data	
51) hlrQueryResource -r	failure (1),expected: 0
52) hlrQueryResource Fails to get non existent res	success
53) hlrQueryUser -u	success
54) hlrQueryUser fails to get non existent user	success
55) Call ATM Client to test resubmission	success
56) Clean up resub job	failure (1),expected: 0
57) Insert Job with start = 0, qtime = 0	success
58) Clean start = 0 qtime =0 job	failure (1),expected: 0
59) Check voOrigin propagation	success



ACCOUNT MANAGEMENT PRODUCTIZED
Description of DJRA1.4 deliverable

60) Clean voOrigin propagation job	failure (1), expected: 0
61) Check hlr-query -o option	success
62) Clean hlr-query -o job	failure (1), expected: 0
63) Check hlr-query -a option	success
64) Clean hlr-query -a job	failure (1), expected: 0
test hlrAdvancedQuery interface	
65) Test resourceJobList (route 3 job)	success
66) Test resourceJobList (route 3 wrongcpu job)	success
67) Test resourceJobList (route 2 job)	success
68) Test HLR reports VO from 'fqan'	success
69) Test userJobList -u option	success
70) Test resourceJobList -r option	success
71) Test resourceAggregate	success
72) Test resourceAggregate -r option	success
73) Test resourceAggregate -g option	success
74) Test resourceAggregate -G option	success
75) Test userAggregate simple	success
76) Test userAggregate -u option	success
Test hlrAdmin	
77) Add hlrAdmin	success
78) Query hlrAdmin	success
79) Error adding already existent hlrAdmin	success
80) Deleting hlrAdmin	success
Test hlrAdvancedQueryClient	
81) Test resourceJobList (no priviledges)	success
82) Test userJobList (unauthorised user)	success
83) Test resourceAggregate	success
84) Test resourceAggregate -r option	success
85) Test resourceAggregate -g option	success
86) Test resourceAggregate -G option	success
87) Test resourceJobList	success
88) Test resourceJobList (lrmsId)	failure (1), expected: 0
89) Test resourceJobList -r option	success
90) Test userJobList	success
91) Test userJobList -u option	success
92) Check "unknown query type"	success
Test clean up	
93) Clean up route 3 job	failure (1), expected: 0
94) Clean up route 2 job	failure (1), expected: 0
Executed N: 95 tests.	
Failures: 14	
Success: 81	



Failed tests:

:4:5:7:8:10:51:56:58:60:62:64:88:93:94

A complete log is available in the file `/tmp//full.test.log`

For each of the tests performed, a line is printed. Success is reported if everything is fine, otherwise a Failure is reported with the exit status returned by the test within round parenthesis and the exit status expected for the test to be successful.

A complete log file, with the output of the test commands is also written in the directory specified with the `-logdir` option. In this case `/tmp/full.test.log`

For more information about writing the test is available in DGAS new documentation draft which is available via DGAS developers: Andrea Guarise <andrea.guarise@to.infn.it>

4.2.3 CE CLIENT SIDE TESTS

Steps:

1. CE should run two demons (check, if such processes are run):

- `glite-dgas-urcollector`
- `glite-dgas-pushd`

The demons should be start from `inittab`.

2. Check if the accounting is collected from the LRMS:

- Stop the `pushd` demon
- Submit a job to your CE, wait until it is completed
- After some time, the `urcollector` should process the accounting and the usage record corresponding to the completed job should be located in the URBox (typically it is `/var/glite/dgasURBox` directory or any other specified in the `dgas_gianduia.conf`). Look into the file with usage record and check if the usage information is correct.
- Look at the `dgas_gianduia_urCollector.log` (typically it is located in `/var/log/glite` directory or any other specified in the `dgas_gianduia.conf`) to trace problems if any.

3. Check if the accounting is pushed to the HLR:

- Start the `pushd`
- After some time, the `pushd` should send usage record to the HLR. If the record was accepted by HLR, it disappears from the URBox (this is the expected result of the test), else the `pushd` attempts to resend it after some time. If it fails for some longer time, the `pushd` stops the attempts and moves the record to `ERR` subdirectory.
- Look at the `dgas_ce_pushd.log` (typically it is located in `/var/log/glite` directory or any other specified in the `dgas_gianduia.conf`) to trace problems if any.

4. Check if the accounting is available on the HLR. For this test, you must be a resource admin on HLR. The test may be performed on CE or any other machine with DGAS client installed.

- Create your proxy certificate (`grid/voms-proxy-init`)
- Run the following command, it should return accounting data of your job:



```
glite-dgas-hlr-query -H <your HLR server> -Q resourceJobList -u <dn  
of the user that submitted the task>|grep <global job id>
```

4.2.4 USER TEST

The first test for users is also **if the server is up and running:**

```
$ glite-dgas-ping -s lcg2-ce.hep.kbfi.ee  
Server alive.
```

If the server is answering the you can **check your user data:**

```
$ glite-dgas-hlr-userinfo -slcg2-ce.hep.kbfi.ee -c  
"/DC=org/DC=balticgrid/OU=eenet.ee/CN=Hardi Teder" -o human  
HLR user id: HardiTeder  
email: Unknown  
description: Hardi Teder  
certifcate subject: /DC=org/DC=balticgrid/OU=eenet.ee/CN=Hardi Teder  
HLR group id: balticgrid_group_not_specified  
Assigned credits: 0  
Spent credits: 0  
Number of executed jobs: 0  
Aggregate cpuTime: 0  
Aggregate wallTime: 0
```

Testing DGAS user HRL:

```
$ glite-dgas-hlr-query -H "lcg2-ce.hep.kbfi.ee:56568:" -Q  
resourceAggregate -A "count(dgJobId)" -u "%Hardi Teder%" -t 2007-12-  
20_
```

```
MIN(date) | MAX(date) | count(dgJobId) |  
2008-01-03 11:54:44 | 2008-01-07 16:10:30 | 4 |
```

Then run a grid job in the cluster and run the same command. The count of grid jobs must be increased by 1.

```
$ glite-dgas-hlr-query -H "lcg2-ce.hep.kbfi.ee:56568:" -Q  
resourceAggregate -A "count(dgJobId)" -u "%Hardi Teder%" -t 2007-12-  
20_
```

```
MIN(date) | MAX(date) | count(dgJobId) |  
2008-01-03 11:54:44 | 2008-01-07 16:10:30 | 5 |
```

It is possibel to get a job based information from HRL:

```
$ glite-dgas-hlr-query -H "lcg2-ce.hep.kbfi.ee:56568:" -Q resourceJobList -A "count(dgJobId)" -u  
"%Hardi Teder%" -t today
```



ACCOUNT MANAGEMENT PRODUCTIZED
Description of DJRA1.4 deliverable

*1|date=2008-01-07

16:10:30|edg.JobId=https://broker.eenet.ee:9000/_LVvRdS6LGVL2BGORFDoug|resourceGridId=kriit.eenet.ee:balticgrid|accountType=resource|userGridId=/DC=org/DC=balticgrid/OU=eenet.ee/CN=Hardi
Teder|remoteHlr=NULL|cpuTime=8|wallTime=311|mem=466120|vmem=1465876|cost=0|fqan=/baltic
grid/Role=NULL/Capability=NULL|userVo=balticgrid|si2k=1000|sf2k=1000|start=1199715030|end=
1199715341|localUserId=balticgrid034|lrmsId=52473.kriit.eenet.ee|siteName=EENet|hlrGroup=eenet.
ee_EENet|urOrigin=|



5 TESTBED INSTALLATION

5.1 Current Status of the testbed installation

Site	Partner	Service	Hardware and Operating System	JRA1 Software version	Notes
aloe-test.man.poznan.pl	PSNC	gLite-CE + VUS WN Torqe DGAS HLR	1 CPU: Intel Celeron 635 MHz 380 MB RAM Scientific Linux CERN 3.0x	gLite VUS v.0.9.1 DGAS HLR clients & service v. 3.1.4	Resource HLR server for PSNC.
lcg2-ce.hep.kbfi.ee	NICPB, PSNC	DGAS HLR DGAS2Apel	1 CPU: AMD Athlon 64 3500+, 2210 Mhz 1 GB RAM Scientific Linux CERN 3.0x	DGAS HLR service v. 3.1.4	Resource HLR server for NICPB and EENet
kriit.eenet.ee	EENet	lcg-CE+VUS, DGAS client	1 CPU: Intel Pentium 4 3GHz 1.5GB RAM Scientific Linux CERN 3.0x	lcg VUS v.1.0.1 DGAS HLR clients & service v. 3.1.4	
dgas.balticgrid.org	EENet	DGAS HLR	1 CPU: Dual Core AMD Opteron, 2210 Mhz 512 MB RAM Scientific Linux 3.0x	DGAS HLR clients & service v. 3.1.4	DGAS user HRL for balticgrid VO users
karklas.mif.vu.lt	VU	lcg-CE + VUS	1 CPU: Intel Celeron 1200 MHz	lcg VUS v.1.0.1 DGAS HLR clients &	Planned installation, currently CE is not installed



ACCOUNT MANAGEMENT PRODUCTIZED
Description of DJRA1.4 deliverable

		WN	380 MB RAM Scientific Linux 3.0x	service v. 3.1.4	yet.
test1.mif.vu.lt	VU	DGAS HLR	1 CPU: Intel Celeron 900 MHz 320 MB RAM Scientific Linux 3.0x	DGAS HLR service v. 3.1.4	Resource HLR server for VU. Installed, but not tested with CE client yet.



6 LIMITATIONS AND KNOWN PROBLEMS

The current DGAS implementation doesn't follow the Usage Record standard in full. The consequence of this is that some accounting metrics, especially non standard ones are not supported by DGAS. So that, the non standard accounting, that may be collected by VUS database cannot be forwarded to DGAS HLR service. However it is still possible to recover this data directly from the VUS database, in the full context of global user identity.

The current testbed installation include a user HLRs. User HLRs collect the same accounting data as resource HLRs, but from the user's and his VO's point of view. The user HLR is strictly connected to VO, so it is reasonable (for organizational reasons), that it would be managed by the same entity as VOMS. VOMSes are located outside our small testbed, but the user HLR for BalticGrid VO is inside the testbed.

The current testbed installation doesn't include a "second level" resource HLR. This kind of HLR collects some aggregated data (statistics). For that purpose, we use APEL and dgas2apel tool. APEL is more suitable for statistics at the moment, as it is compatible with EGEE infrastructure.



7 SUMMARY

Virtual User System was integrated with gLite in order to fulfil BalticGrid requirements on the area of the user management. The current status of deployment was discussed. Installation and configuration procedures were described in detail, as well as the test procedures as the document aims to help administrators to work with the system. The document also lists problems, that were identified from the experience gained while installing and running the system in the testbed.

The system was installed and tested in JRA1 testbed environment. This deployment proved, that it is matured enough to start deployment on sites in production state. So such a deployment will be the next step.

We have started a discussion with SA1 activity to deploy and connect both installations, i.e. the production SA1 testbed with the JRA pilot installation what would be the final stage and prove of concept to have a heterogeneous environment. RTU is the first partner from outside JRA, where the installation will start as a further step.

The experience of BalticGrid work was described in a conference paper and presented by Michal Jankowski (PSNC) at the eChallenges conference (24 - 26 October 2007, The Hague, The Netherlands):

Michal Jankowski, Norbert Meyer "**Dynamic User Management in the BalticGrid Project**", In: Paul Cunningham and Miriam Cunningham (Eds) "Expanding the Knowledge Economy: Issues, Applications, Case Studies", vol.4 of book series "Information and Communication Technologies and the Knowledge Economy", IOS Press, 2007 Amsterdam, ISBN 978--1--58603--801-4, pp. 1401-1406.



REFERENCES

1. *gLite Installation and Configuration Guide* v. 3.0 (rev.2), http://glite.web.cern.ch/glite/packages/R3.0/R20060502/doc/installation_guide_3.0-2.pdf, May 8 (2006)
2. *Implementation of Logging System and Accounting Policies*, BalticGrid deliverable BG-DSA1.7, Nov. 2006.
3. *BalticGrid, EU FP6, Contract 026715, Annex I - "Description of Work"*, Sep. 15 (2005)
4. *Account Management Report on Integration with Workload Management System (WMS) Environment*, BalticGrid deliverable DJRA1.1., June 2006.
5. *Report on Implementation of JRA1 Prototypes*, BalticGrid deliverables DJRA1.2 and DJRA1.3., June 2007.
6. <http://www.globus.org/toolkit/>
7. *EGEE User's Guide, WMS Service*, <https://edms.cern.ch/document/572489/>, May 3, 2006.
8. <http://www.unixodbc.org>
9. <http://www.nikhef.nl/grid/lcaslcmaps/>
10. R.Alfieri, R.Cecchini, V.Ciaschini, L.Dell'Agnello, A.Frohner, A.Gianoli, K.Lentey, F.Spataro, *VOMS: an Authorization System for Virtual Organizations*, 1st European Across Grids Conference, Santiago de Compostela, February 13-14, 2003.
11. V.Ciaschini, *A VOMS Attribute Certificate Profile for Authorization*, <http://infnforge.cnaf.infn.it>, October 2004.
12. M.Jankowski, P.Wolniewicz, N.Meyer, *Virtual User System for Globus based grids*, Cracow Grid Workshop '04 Proceedings, Cracow 2004.
13. <http://www.to.infn.it/grid/accounting/main.html>
14. A. Guarise, G. Patania and R.M. Piro. "DGAS – Home Location Register. User's Guide". Technical Report EGEE-JRA1-TEC-571271-HLR. September 23, 2005
<https://edms.cern.ch/document/571271/1>
15. A. Guarise, G. Patania and R.M. Piro. "DGAS - GIANDUIA. User's Guide". Technical Report EGEE-JRA1-TEC-571271-GIANDUIA. September 23, 2005
<https://edms.cern.ch/document/571271/1>
16. Michal Jankowski, Norbert Meyer *Dynamic User Management in the BalticGrid Project*, In: Paul Cunningham and Miriam Cunningham (Eds) *Expanding the Knowledge Economy: Issues, Applications, Case Studies*, vol.4 of book series *Information and Communication Technologies and the Knowledge Economy*, IOS Press, 2007 Amsterdam, ISBN 978--1--58603--801-4, pp. 1401-1406.