



Enabling Grids for E-science

The Road to Production: SGE Integration Process with CREAM-CE

E. Freire, J. López, A. Simón, C. Fernández, R. Díez, P. Rey, S. Díaz (CESGA)
G. Borges (LIP)

4th Iberian Grid Infrastructure Conference
Braga, May 25, 2010

www.eu-egee.org



- **Intro: GridEngine and CREAM-CE**
- **Development**
- **Certification**
- **Staged Rollout**



Intro: GridEngine and CREAM-CE

- **GridEngine advanced features:**
 - Support for up to 10.000 nodes per master server
 - Shadow master hosts for failover purposes
 - Complete administration GUI
 - Integrated scheduler
- **CREAM CE**
 - New Computing Element that will replace LCG-CE
 - Webservice-based interface

Release life-cycle from development to production:

Phase	EGEE Project Activity
Development	JRA1
Certification	SA3
Staged Roll-out	SA1

Patch #3179:

- ***“Release 1.6 of CREAM CE for sl5_x86_64”***
- **Already includes GridEngine Blah scripts**
- **All the necessary scripts to use GridEngine are included in CREAM CE under /opt/glite/bin directory:**
 - BUdaterSGE
 - sge_cancel.sh
 - sge_filestaging
 - sge_helper
 - sge_hold.sh
 - sge_resume.sh
 - sge_status.sh
 - sge_submit.sh

Patch #3764:

- “*glite-SGE_utils support for CreamCE V1.6 release SL5/x86_64*”
- **It includes all necessary functions to configure GridEngine in CREAM-CE**

- **Each change in the software is documented in the change management system by the developers in a document called patch using Savannah tool**
- **Everything starts as a new patch**
- **The new packages created with the patch are uploaded into glite CVS repository**
- **Later, the new packages are grouped in a meta-package built using the ETICS web tool**

- **Once the code is ready it is uploaded in the gLite CVS:**
 - glite-info-dynamic-sge
 - lcg-jobmanager-sge
 - org.glite.yaim.sge-client
 - org.glite.yaim.sge-server
 - org.glite.yaim.sge-utils
- **Each component includes a Makefile and Spec file following ETICS guidelines**
- **ETICS is used to automatically build the RPMs**

- 1. Commit the code to glite CVS**
 - Each component in glite CVS requires an equivalent ETICS component (request through EMT)
- 2. Create ETICS components configurations follow a template so that ETICS knows where to fetch the software in glite CVS**
- 3. Build glite-SGE_utils metapackage defining the proper RPM dependencies**
- 4. The patch is created following the directives of the Product Team Integration Guide**
 - It is here where external dependencies can be added
- 5. Sge-utils subsystem build is triggered in ETICS**
- 6. Request to generate YUM repository**
- 7. Patch submitted and ready for certification**

 **gLite Middleware - Patches: patch #3764, glite-SGE_utils support for...**

- Group
- [Main](#)
- [Homepage](#)
- [Download](#)
- [Docs](#)
- [Mailing Lists](#)
- [Source Code](#)
- [Bugs](#)
- [Tasks](#)
- [Patches](#)
- [News](#)

You are both [technician](#) and [manager](#) for this tracker.

patch #3764: glite-SGE_utils support for CreamCE V1.6 release SL5/x86_64

Submitted by: [Goncalo <goncalo>](#)
 Submitted on: 2010-02-05 15:51

Submit Changes and Browse Items

Submit Changes and Return to this Item

Status:	<input type="text" value="Ready for Certification"/>	Open/Closed:	<input type="text" value="Open"/>
Priority:	<input type="text" value="7 - High"/>	Assigned to:	<input type="text" value="None"/>
Discussion Lock:	<input type="text" value="Unlocked"/>	Release Path: *	<input type="text" value="Standard"/>
Summary: *	<input type="text" value="glite-SGE_utils support for CreamCE V1.6 release SL5/x86_64"/>		
glite release: *	<input type="text" value="gLite 3.2"/>	Operating System: *	<input type="text" value="SL5"/>
Architecture: *	<input type="text" value="x86_64"/>		
YUM repo file URL:	<input type="text" value="http://etics-repository.cern.ch:8080/repository/pm/registered/repomd/name/3764"/>		
gLite subsystem tag(s) / ETICS configuration:	<input type="text" value="sge-utils_3_2_1
glite-info-dynamic-sge_R_4_0_903_3
glite-yaim-sge-utils_R_4_1_1_5"/>		

- 1. Developer changes patch status to “Ready for certification”**
- 2. Certifiers change patch status to “In certification” as soon as they start working on it**
- 3. Install the patch**
 1. Use the corresponding YUM repository
 2. Read “Release Notes” carefully
- 4. Check if the packages name and version are the same as the ones listed in the patch**
- 5. Check fields “Metapackages to be reconfigured” or “Metapackages to be restarted” and act accordingly**
- 6. Check if the bugs listed are fixed or if the new functionalities are working fine**
- 7. Patch status is changed to “Certified” or “Rejected”**

If the patch has bug dependencies, bug status must be updated:

- **Ready for Review:**
 - Patch solves the bug and certifier gives it to the original submitter for final check
- **Fix Certified:**
 - Patch solves the bug, no feedback from original submitter needed
- **Fix not Certified:**
 - Certifier could not verify if bug is fixed for any reason
- **Bug is not fixed:**
 - Comments must be reported



Patch Structure and Release Notes

<u>Labels</u>		
<u>packages:</u>		
<u>Metapackages to be reconfigured:</u> *	glite-SGE_utils	<u>Metapackages to be restarted:</u> *
		glite-SGE_utils
<u>Documentation location:</u>	https://twiki.cern.ch/twiki/bin/view/LCG/YaimGuide400 https://twiki.cern.ch/twiki/bin/view/LCG/SGE_configuration_variables	
<u>Configuration changes:</u> *	<p>1) The service needs to be configured the first time <u>CREAMCE</u> is installed together with <u>glite-SGE_utils</u>. It introduces the support for <u>CREAMCE V1.6</u> (see bug #63047).</p> <p>2) Some sites use <u>SGE</u> installations shared via NFS or equivalent (see bug #59060). In order to prevent any changes in that <u>SGE SHARED</u> setup, a new <u>yaim</u> variable, called <u>SGE_SHARED_INSTALL</u>, is introduced. Its default value is "no" informing <u>YAIM</u> that NO SHARED installation is being used, and that the <u>SGE</u> software must be configured. If a <u>SGE</u> installation shared via NFS (or equivalent) is being used, and you do NOT want to change it with <u>YAIM</u>, set <u>SGE_SHARED_INSTALL=yes</u> in your <u>site-info.def</u> file.</p>	
<u>Release notes:</u> *	<p>1) This new <u>glite-SGE_utils</u> release integrates <u>SGE LRMS</u> with <u>CREAMCE</u> (Version 1.6) in SL5, x86_64. The <u>CREAMCE</u> integration consists in setting up the <u>BLAH</u> configuration to interoperate with <u>SGE</u>. For <u>BLAH</u> to work, the following set of <u>SGE</u> scripts and binaries should be available under <u>/opt/glite/bin</u>: <u>BUpdaterSGE</u>, <u>sgc_helper</u>, <u>sgc_hold.sh</u>, <u>sgc_submit.sh</u>, <u>sgc_resume.sh</u>, <u>sgc_status.sh</u> and <u>sgc_cancel.sh</u>. These scripts are installed by <u>glite-ce-blahp rpm</u> but the same files can also be downloaded from http://www.egee.cesga.es/cream/releases/0.60/bin_0.60.tar.gz</p> <p>2) The <u>CREAMCE</u> must be installed in a separate node from the <u>SGE QMASTER</u>, and the same <u>SGE</u> software version should be used in both cases. After installation</p>	
<u>Pre-production release:</u>		<u>Release:</u>

* Mandatory Fields

- Final step before reaching Production
- Software and release documentation are validated through the operation in real conditions
- Early detection of possible bugs in the new middleware before it is released in Production
- A critical step to guarantee the quality of new packages

1. **Verification of the patch by the gLite Release Team**
 - If everything is fine status is changed to “Ready for Rollout”
2. **gLite Release Team uploads the corresponding RPMs to the *Delta* repository (aka *Beta* repository)**
 - Patch status is changed to “Rolling out”
3. **Operation Managers for the Stage Rollout create the rollout tasks and assign them to sites participating in the process**
4. **Site starts testing the patch**
 1. Status changed to “In Progress”
 2. Outcome: Success/Fail/Warning
 3. Quarantine period of a week
 4. Status changed to “Done”
5. **Operation Managers check task reports**
6. **Patch status is changed to “Ready for Production” or rejected**
7. **gLite Release Team moves RPMs to the Production repository**
8. **gLite Release Team updates Production Release pages**
9. **Update is announced via CIC portal broadcast tool**

- **The road from development to production is complex**
- **The release life-cycle is composed by three phases that try to guarantee the quality of new packages**
 - Development: blah scripts and yaim functions are developed
 - Certification: the patches are certified in laboratory
 - Stage Rollout: patches are tested in real conditions
- **The support of GridEngine in CREAM CE is the result of a joint effort between CESGA, LIP and IC**
- **GridEngine provides an alternative to torque/maui for sites wanting a modern batch system**

