SA3 Feedback and the Future

Oliver Keeble - oliver.keeble@cern.ch

JRA1 All-hands, Helsinki

Information Society and Media

Summary

- **SA3 – a reminder**
- **gLite 3.1**
  - Release process
  - Integration
- **Reassessment of building**
- **Dependency Challenge**
- **Feedback**

SA3 responsibilities

- **Mission statement from the Technical Annex**
  - "The goal of the SA3 activity is to manage the process of building deployable and documented middleware distributions, starting by integrating middleware packages and components from a variety of sources"
- **We are the ones in between you and deployment**
- **~30 FTEs**
  - test writing, testbed management, information system, configuration, testing, certification, integration
- **Assuming 60 Changers (JRA1 + LCG, VDT etc)**
  - **Changers/Testers ~ 5**
  - **Changers/Integrators ~ 10**

gLite 3.1 - status

- **Worker Node is in production**
- **User Interface is in PPS**
- **glite-CE is being tested in SA3**
- **WMS is being handled separately, but a full gLite 3.1 version is available**
  - Effort on workload should be transferred to the 'final product' as soon as practical
- **APEL build issue**
- **Many nodes can be tested – take them!**
- **What potential issues can we look out for**
  - proxy format
  - python2.2
  - grid ftp services

- **Current Status**
- https://grid-deployment.web.cern.ch/grid-deployment//cgi-bin/reports.cgi?action=package

Node	slc3_ia32_gcc323	slc4_ia32_gcc346	slc4_x86_64_gcc346
glite-WN	X	O	X
glite-UI	X	O	X
glite-WMS	X	O	X
glite-LB	O	O	X
glite-MON	X	X	X
glite-BDII	O	O	O
glite-PX	O	O	O
glite-CE	X	O	X
glite-TORQUE_utils	X	O	X
glite-TORQUE_client	X	O	X
glite-TORQUE_server	X	O	X
glite-VOMS_oracle	O	O	O
glite-VOMS_mysql	O	O	O
glite-SE_dpm_disk	O	O	X
glite-SE_dpm_mysql	O	O	X
glite-SE_dpm_oracle	O	O	X
glite-AMGA_server	X	X	X
glite-AMGA_client	X	X	X
glite-SE_classic	N/A	N/A	N/A

Drilling down

EGEE-II: SA3 Activity

glite-WN_slc4_ia32_gcc346

Package	Version	Status
a1_grid_env	2.0.0-1	Found
c-ares	1.3.0-4	Found
CGSI_gSOAP_2.6	1.1.15-6	Found
classads	0.9.8-2	Found
cleanup-grid-accounts	1.0.2-1	Found
dcache-client	1.7.0-28	Found
DPM-client	1.6.3-1	Found
edg-gridftp-client	1.2.6-1	Found
fetch-crl	2.6.3-1	Found
GFAL-client	1.9.0-2	Found
glite-data-api-perl	1.1.1-1	Found
glite-data-catalog-api-c	2.0.0-5	Found
glite-data-catalog-api-perl	2.0.0-3	Found
glite-data-catalog-cli	1.7.3-1	Found
glite-data-catalog-interface	2.0.0-8	Found
glite-data-srm-api-c	1.1.0-3	Found
glite-data-srm-api-perl	1.1.0-2	Found
glite-data-srm-cli	1.2.8-2	Found
glite-data-transfer-api-c	3.1.0-2	Found
glite-data-transfer-api-perl	3.1.0-1	Found
glite-data-transfer-cli	3.3.0-2	Found
glite-data-transfer-interface	3.1.0-4	Found

Please find your favourite node type and check the lists

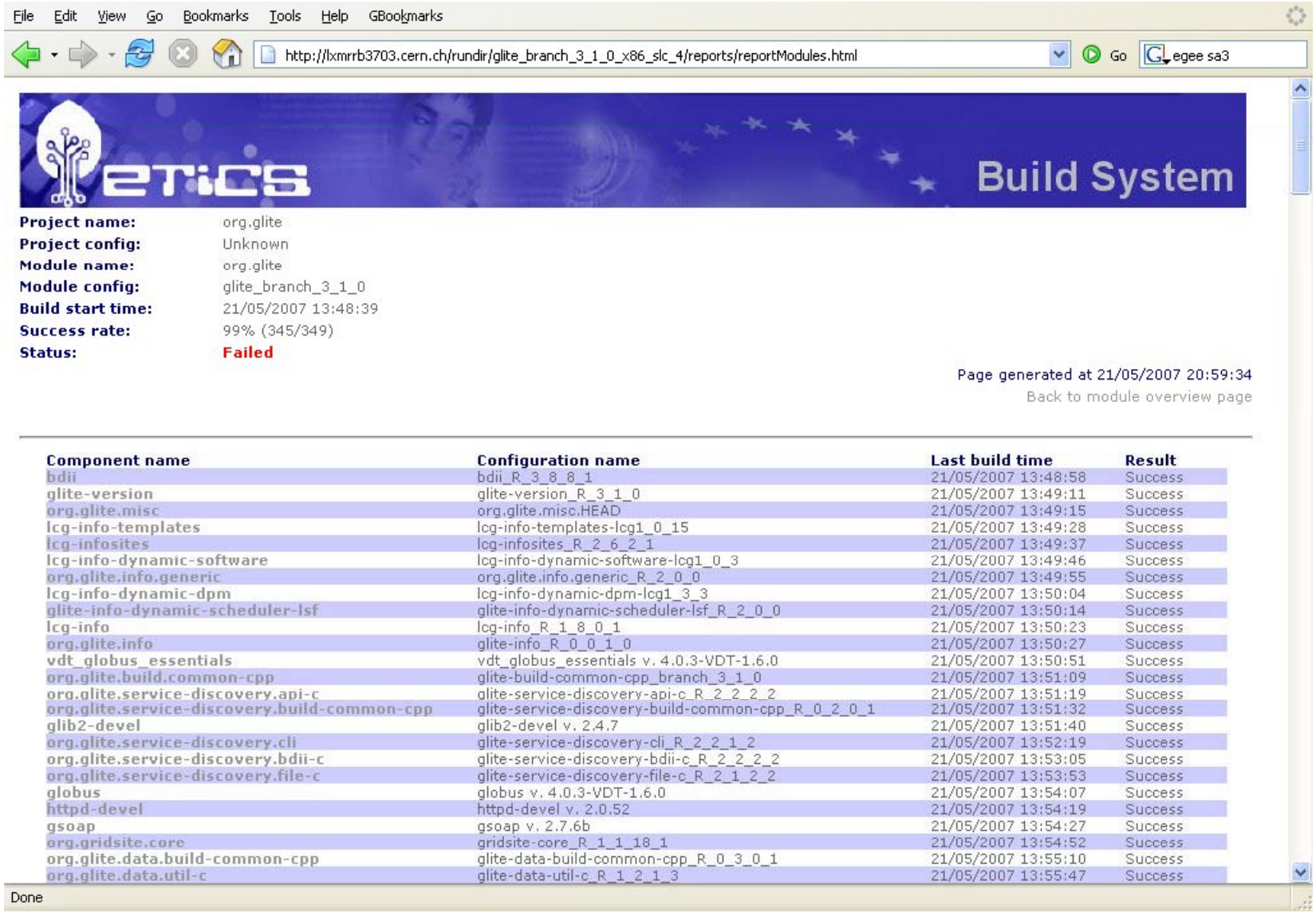
Impact of 3.1 on release process

- **Patch for each platform**
  - glite_OS_arch
    - gLite 3.0 / SL3 / i386
    - gLite 3.1 / SL4 / i386
    - gLite 3.1 / SL4 / x86_64
    - Maybe....
      - *gLite 3.1 / SL3 (already building)*
      - *gLite 3.0 / SL3 from ETICS?*
    - Don't forget Debian
- **In the beginning, a patch may be an entire new node type**
- **Maintain patch-free turnaround for unreleased services**
- **Developers – we can make repositories available for you to try the installation and fix problems**
  - How can we make this as quick as possible?

gLite 3.1 integration process

- **Creation of meta-packages and repositories has been decoupled from ETICS.**
  - Integration requires package name and version
    - ETICS uses configuration names and component names
      - *No easy mapping*
    - Require bulk operations on many lists
    - Would like to query across multiple lists
  - Large speed advantage
- **ETICS is the 'build factory', ETICS repository is the interface**
  - Integration is then done at the package level
  - Metapackages are maintained as simple text files with module and version information.
- **Metapackages are decoupled from configuration**
  - Config is not shipped in metapackages
  - Yaim will allow developers to take over more config if they wish

List Management



Component name	Configuration name	Last build time	Result
bdii	bdii_R_3_8_8_1	21/05/2007 13:48:58	Success
glite-version	glite-version_R_3_1_0	21/05/2007 13:49:11	Success
org.glite.misc	org.glite.misc.HEAD	21/05/2007 13:49:15	Success
lcg-info-templates	lcg-info-templates-lcg1_0_15	21/05/2007 13:49:28	Success
lcg-infosites	lcg-infosites_R_2_6_2_1	21/05/2007 13:49:37	Success
lcg-info-dynamic-software	lcg-info-dynamic-software-lcg1_0_3	21/05/2007 13:49:46	Success
org.glite.info.generic	org.glite.info.generic_R_2_0_0	21/05/2007 13:49:55	Success
lcg-info-dynamic-dpm	lcg-info-dynamic-dpm-lcg1_3_3	21/05/2007 13:50:04	Success
glite-info-dynamic-scheduler-lsf	glite-info-dynamic-scheduler-lsf_R_2_0_0	21/05/2007 13:50:14	Success
lcg-info	lcg-info_R_1_8_0_1	21/05/2007 13:50:23	Success
org.glite.info	glite-info_R_0_0_1_0	21/05/2007 13:50:27	Success
vdt_globus_essentials	vdt_globus_essentials v. 4.0.3-VDT-1.6.0	21/05/2007 13:50:51	Success
org.glite.build.common-cpp	glite-build-common-cpp_branch_3_1_0	21/05/2007 13:51:09	Success
org.glite.service-discovery.api-c	glite-service-discovery-api-c_R_2_2_2_2	21/05/2007 13:51:19	Success
org.glite.service-discovery.build-common-cpp	glite-service-discovery-build-common-cpp_R_0_2_0_1	21/05/2007 13:51:32	Success
glib2-devel	glib2-devel v. 2.4.7	21/05/2007 13:51:40	Success
org.glite.service-discovery.cli	glite-service-discovery-cli_R_2_2_1_2	21/05/2007 13:52:19	Success
org.glite.service-discovery.bdii-c	glite-service-discovery-bdii-c_R_2_2_2_2	21/05/2007 13:53:05	Success
org.glite.service-discovery.file-c	glite-service-discovery-file-c_R_2_1_2_2	21/05/2007 13:53:53	Success
globus	globus v. 4.0.3-VDT-1.6.0	21/05/2007 13:54:07	Success
httpd-devel	httpd-devel v. 2.0.52	21/05/2007 13:54:19	Success
gsoap	gsoap v. 2.7.6b	21/05/2007 13:54:27	Success
org.gridsite.core	gridsite-core_R_1_1_18_1	21/05/2007 13:54:52	Success
org.glite.data.build-common-cpp	glite-data-build-common-cpp_R_0_3_0_1	21/05/2007 13:55:10	Success
org.glite.data.util-c	glite-data-util-c_R_1_2_1_3	21/05/2007 13:55:47	Success

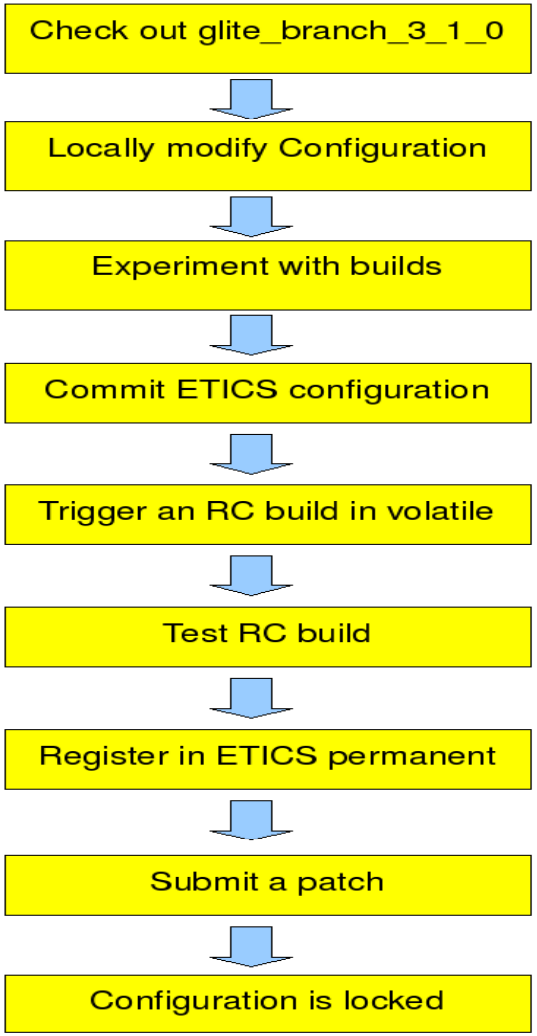
New approach to the build - 1

- **Problem with multiple builds**
  - Inefficient
  - Does not maximise testing
  - Final 'release build' often problematic
  - Slows everything down
- **The primary goal is to allow a developer to perform a build, in a controlled environment, whose products cannot be changed and whose build log is archived with the artifacts.**
- **Why? Because this means a developer can produce a release candidate.**
- **New activity demarcation line will be patch level integration**

New approach to the build - 2

- **SA3**
  - enforces build acceptance criteria (eg the build was remote)
  - freezes current glite_branch_3_1_0 and then maintains baseline (NOT necessarily what is in the release, but tracking API extensions and changes)
- **JRA1 developer**
  - experiments with a particular tag, producing local builds until s/he is happy
  - updates configuration and triggers a 'release candidate' build (this will be remote)
  - tests the RC build
  - registers it if tests pass
  - submits a patch, referencing the artifact in the ETICS repo. At this point the config is locked

Build Workflow



New approach to the build - 3

- **ETICS requirements**
  - Registration of build artifacts on fulfilment of project defined conditions (Policy)
  - Archiving of build arguments and build log with artifact and build information (R1.1 or even before)
  - Configuration locking on successful build of the artifact or store of the build information in a reusable form (R1.2)
  - Local manipulation of build tree (not affecting ETICS server) for experimentation (R1.1)
  - Two stage build then register
  - ...what else? Is there a problem with an update which crosses subsystems? Does the developer still have to synchronise with a subsystem manager?

Dependency Challenge

- **Dependency Challenge**
  - PMB endorsed plan
    - https://edms.cern.ch/file/840168/1/EGEE-II-NA1-TEC-gLiteRestructuring-v0.4.doc
  - Driven by Markus and Claudio
- **Progress so far**
  - ALL metapackages have been reviewed, reduced rpm lists are now available
- **Stage One**
  - Teams will be set up to review the dependencies
  - Publish acceptance criteria for external/internal dependencies (Joachim, reviewed by JRA1)
  - Both deps on the OS and fully external deps will be considered as multiplatform support is affected in both cases
    - Support level (platform, 64bit)
    - Commonly used in project
    - Not available via another package
    - Maintenance status of package
  - Duration: 1 week

Dependency Challenge

- **Stage Two**
  - Understand what can be removed, what can be handled by a few extra lines or changes
  - SA3 and JRA1 (Markus and Claudio) David, Maarten , Steve Fisher (Alastair)
  - Can progress in parallel for different components
  - Duration: Potentially infinite ----> limit to not more than 2 weeks
- **Stage Three**
  - Implement those changes that can be implemented within 1 week (according to the developers)
  - Duration: Developers 2 weeks, testing 2 weeks

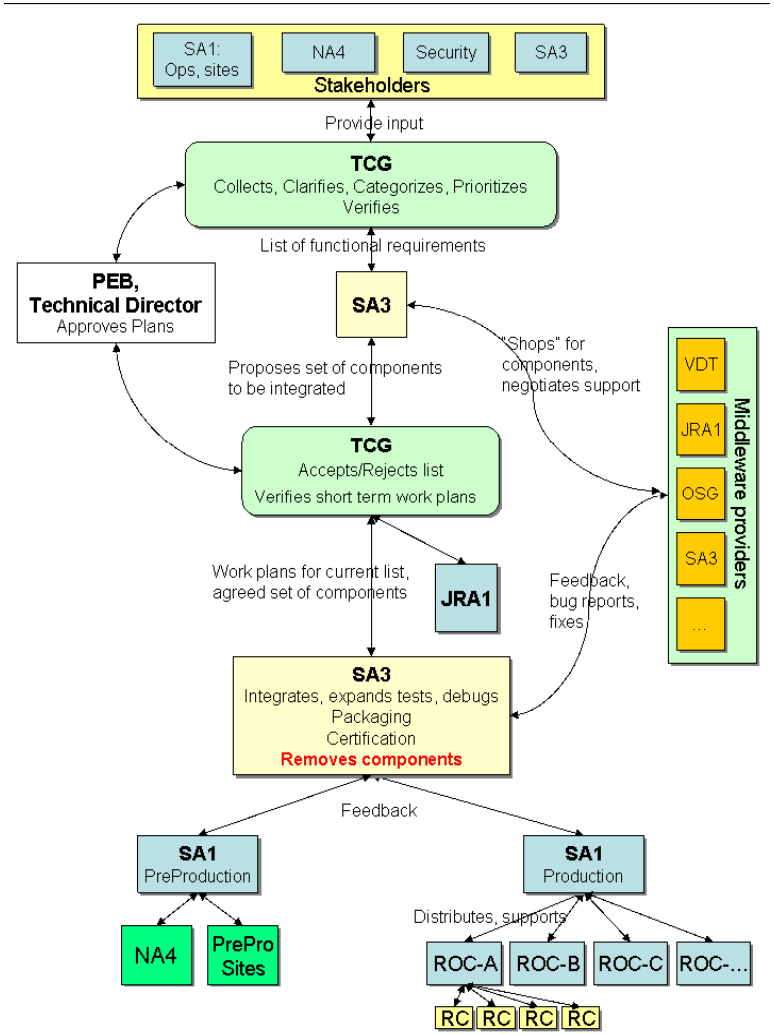
Feedback

- **Documentation**
  - The lack of detailed documentation of the architecture and design of the WMS has impacted the development of interoperation with ARC.
- **Reproducibility – IC experience**
  - A developer running a service is never reproducible
- **Reactivity**
  - Can wait a week trying to find someone
- Standardisation on the use of /etc/init.d/ startup scripts, via an attempt to move all supplied packages to adhere to the basic Linux Standard Base specification
- All RPMS provided by the project should aim to follow the guidelines from the fedora project and should be split into subpackages where appropriate with a general categorisation of:
  - **common, client, server, documentation.**
  - **maximal client separation**
  - **keep the fat on the server**

Summary

- **Please find me to discuss this stuff!**

- **Extra slides...**


SA1: Ops, sites
NA4
Security
SA3
Stakeholders
Provide input
TCG
Collects, Clarifies, Categorizes, Prioritizes
Verifies
List of functional requirements
PEB,
Technical Director
Approves Plans
SA3
"Shops" for components, negotiates support
Proposes set of components to be integrated
TCG
Accepts/Rejects list
Verifies short term work plans
VDT
JRA1
OSG
SA3
...
Middleware providers
Work plans for current list, agreed set of components
JRA1
Feedback, bug reports, fixes
SA3
Integrates, expands tests, debugs
Packaging
Certification
Removes components
Feedback
SA1
PreProduction
SA1
Production
Distributes, supports
NA4
PrePro Sites
ROC-A
ROC-B
ROC-C
ROC-...
RC
RC
RC
RC

Current status – gLite 3.0

- **What is a distribution?**
  - Different from a codebase
  - Please understand what gLite 3.0 means to the Service Activities!
- **gLite 3.0 distro**
  - Bunch of tags
  - Built against an Operating System (SL3)
  - With a bunch of externals (VDT 1.2, ...)
- **Release process**
  - I'm here to answer questions...
- **According to the GIIS monitor;**
  - Over 200 sites (not all active)
  - Over 30000 CPUs
  - 14TB of storage

So what's the value?

- **Why don't we pass tags to production?**
- **SA3 ensures that software**
  - is built in a controlled environment
  - is what is required in production
  - can be installed
  - is made available with as part of a complete service
  - can be configured through a generic interface
  - has been tested
  - is released and advertised in a known location
  - has appropriate documentation available