



Enabling Grids for E-scienceE

# gLite Patch Release Procedure

*Felix Ehm*

*IT-GD*

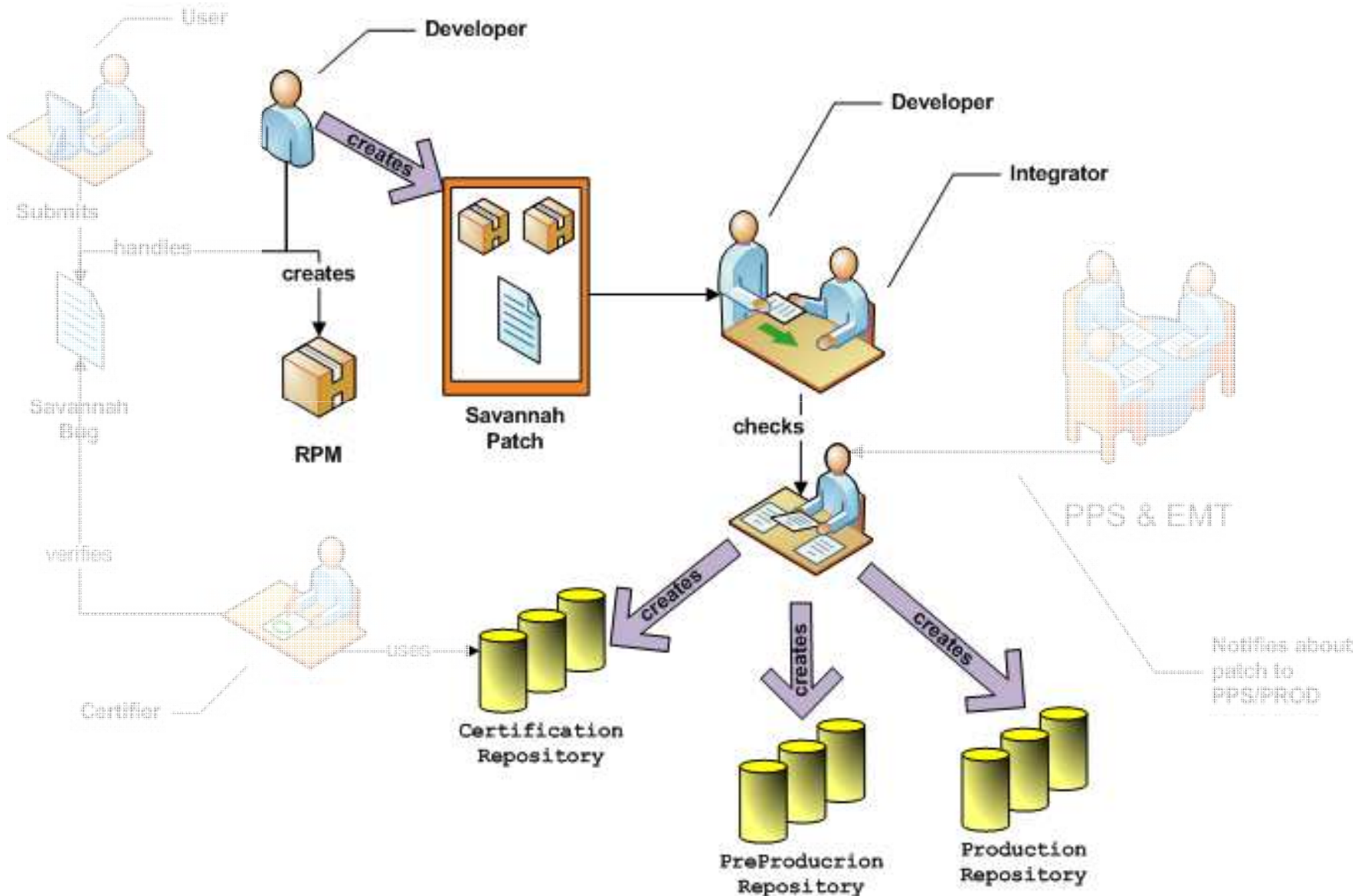
*White Area Lectures*

[www.eu-egee.org](http://www.eu-egee.org)



- **What do we have?**
  - the gLite Middleware **software**
  - components (=metapackages)
  - RPM packages grouped into metapackages
  - **bugs** in parts of the software
  - **fixes** for these bugs provided as new RPMs
  - **patches** introducing fixed RPMs to the Integration Procedure.

- **How does the bug fix go to production ?**
- **Well..**



- **What is a repository ?**
  - Directory with RPM files + some metadata for yum
- **How can I create one ?**

- Create a directory, e.g. :
- Copy your RPMs
- Run createrepo
- Result:

```
mkdir RPMS.release
```

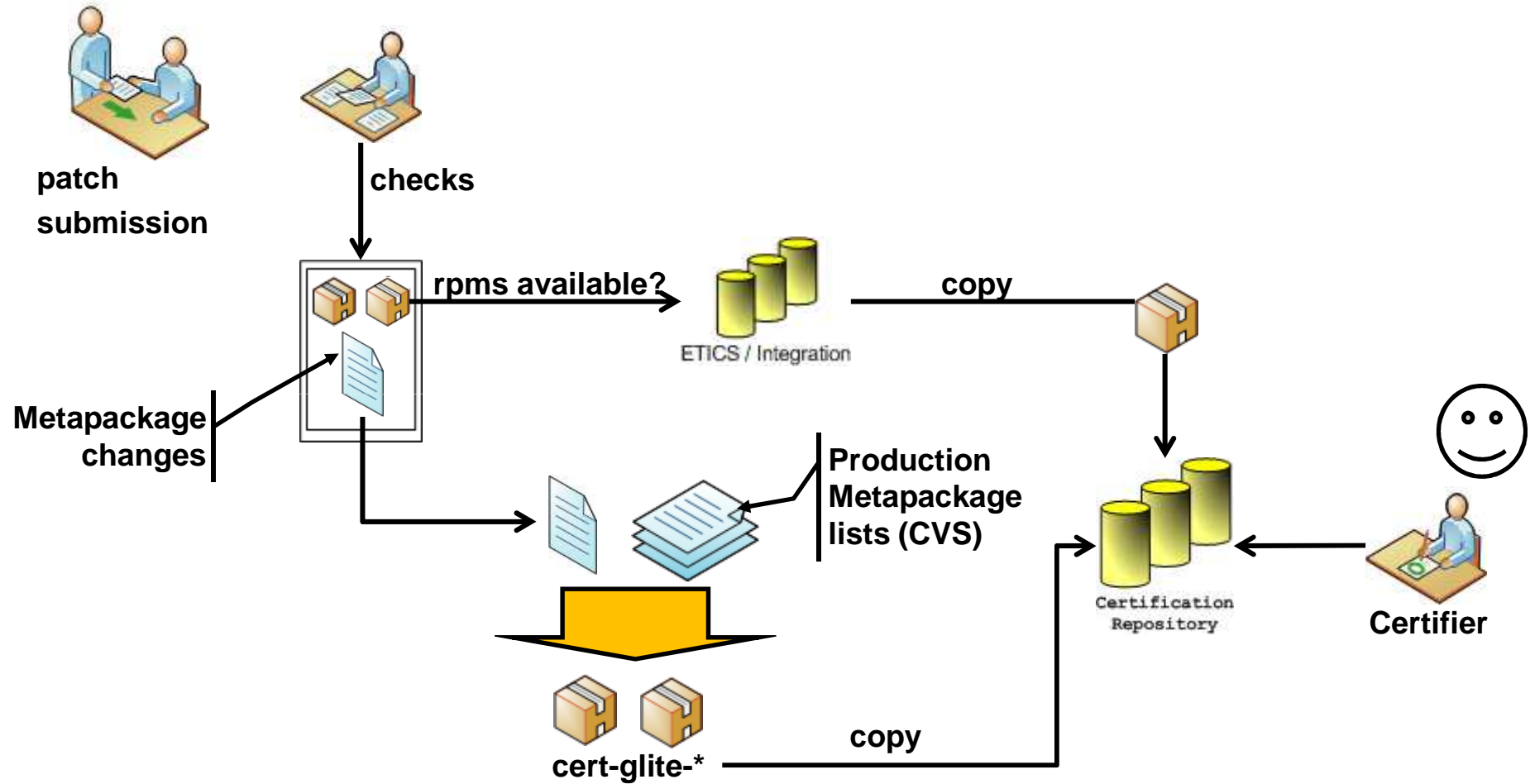
```
cp *.rpm RPMS.release/.
```

```
createrepo RPMS.release
```

```
drwxr-xr-x  2 23834 root 22528 Dec  2 14:42 RPMS.release
drwxr-xr-x  3 23834 root  2048 Dec  4 11:34 repodata
```

- repodata/\* :

```
-rw-r--r--  1 23834 root 291351 Dec  2 14:46 filelists.xml.gz
-rw-r--r--  1 23834 root  55576 Dec  2 14:46 other.xml.gz
-rw-r--r--  1 23834 root 313086 Dec  2 14:46 primary.xml.gz
-rw-r--r--  1 23834 root    951 Dec  2 14:46 repomd.xml
```



- **Integrator**

- Looks for patches in „Ready for Integration“
- Checks patch information
  - Metapackage changes field



- Only used for **add/removals** of packages from/to metapackage
- Strict format, e.g. :

```
R|glite-UI|glite-release||
R|ALL|obsolete-package||
A|ALL|a-new-package|3.1.0-0|
```



- RPM name(s) field
  - Only valid rpm file names must be given (e.g. *glite-release-3.1.-0.noarch.rpm*)
  - If RPMs are external, specify **additionally** the download url in the external packages field
- Subject
  - Must contain **x86\_64** in case of a 64 Bit patch

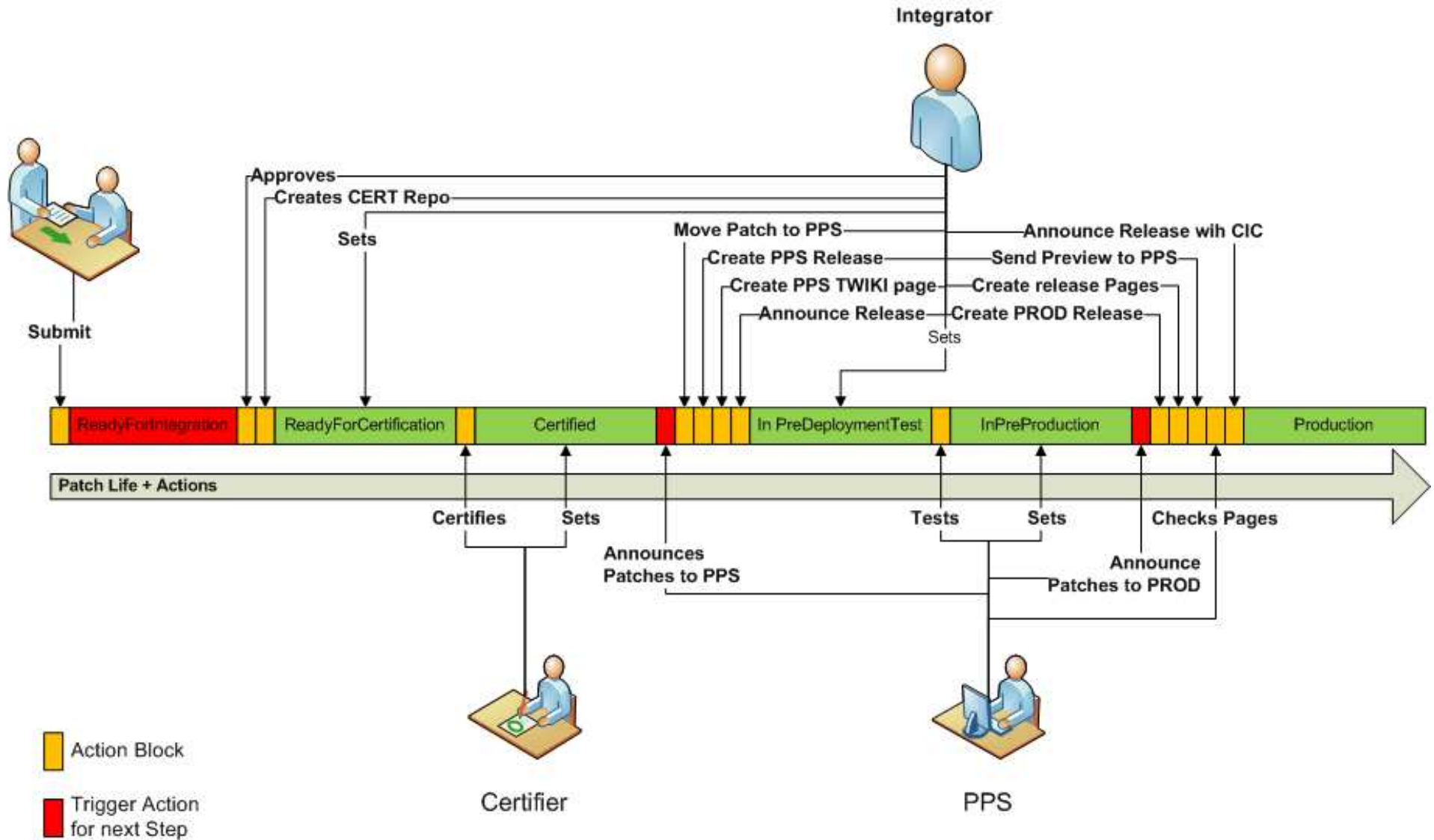
- **Integrator**
  - Provides a RPM repository for the Certifier
    - Extracting metapackage changes from Savannah
    - Extracting RPM names from Savannah
    - Checks if patches are available in ETICS/External repository
    - Copies RPMs to new patch location
    - Creates a **cert-glite-\*** rpm in case of metapackage changes
      - *Prod-lists are basis*
    - Transforms location into a repository
    - Sets the Patch state to “Ready for Certification”



- **Certifier**
  - Starts a new clean SLC machine
  - Enables Node Production repository
  - In case of a patch
    - Installs & configures affected node type (Savannah information)
    - Verifies bug (attached to patch)
    - Enables patch repository
    - Installs patch
    - Verifies bug is fixed
    - Runs set of test
  - In case of a new metapackage or metapackage changes
    - Enables patch repository
    - Installs cert-glite-\* package
    - Checks if node type is ok
  - Sets Savannah patch state to “Certified”

- **PPS**
  - Announces via mail which patches go to PPS
  - Generate GGUS ticket
- **Integrator**
  - Applies the patches to the PPS lists
  - Generates new metapackages depending on the new RPMs
  - Copies new files into a current PPS shadow repository and generates the yum files
  - Generates Twiki release notes page
  - Creates UI/WN TAR ball
  - Sends announcement of new repository

- **Like CERT to PPS, but**
  - Releasenotes are created differently
    - A clean **SL** machine tries to install all nodetypes consecutively
    - Yum output is parsed for packages being pulled in
    - List is compared to previous release list
    - HTML page with differences is generated for each node
  - RSS feed is created with update notification
- **Releasenotes creation gives good opportunity to check for correctness**
  - Installation problems can be spotted early
  - However, takes very long time (~ 1.5h)



- **Past Problems**

- Metapackage changes were done manually
- Documentation
- HowTos
- Release scripts
  - Error handling
  - Stability
  - Able to handle easily new releases (e.g. 3.2, 3.3. , ..)

- **Current Problems**

- Missing Savannah Interface
- Web page extraction for RPM names and changes by sed/awk/tr
- Release notes creation takes too long

- **Past Problems**
  - none
- **Current Problems**
  - Savannah patches are not filled out properly
  - Developers need improved tools/help to do so
    - More use of **checkAffectedMetapackages.sh** script (see Appendix)

- **Future work**
  - gLite web pages should be revised
  - Node update pages should be generated by **repoview**

- **Using Patch Repositories**

- Enable Patch repository

- /etc/yum.repos.d/patch1234.repo

```
[Patch 1234]
name=Patch 1234
baseurl=http://grid-deployment.web.cern.ch/grid-deployment/glite/cert/3.1/patches/1234
enabled=1
```

- Exclude a package from a repository

```
[Patch 1234]
name=Patch 1234
baseurl=http://grid-deployment.web.cern.ch/grid-deployment/glite/cert/3.1/patches/1234
enabled=1
exclude=excluded_package
```

- Protect packages from repository to be upgraded

```
[Patch 1234]
name=Patch 1234
baseurl=http://grid-deployment.web.cern.ch/grid-deployment/glite/cert/3.1/patches/1234
enabled=1
protect=1
```

**This does not work when installing the packages the first time**



	<u>REPO</u>	<u>AFS</u>	<u>WEB</u>
CERT		/afs/cern.ch/project/gd/www/glite/cert/3.1/patches	<a href="http://grid-deployment.web.cern.ch/grid-deployment/glite/cert/3.1/patches/">http://grid-deployment.web.cern.ch/grid-deployment/glite/cert/3.1/patches/</a>
PPS (prepare)		/afs/cern.ch/project/gd/www/glite/prepare/pps/	<a href="http://grid-deployment.web.cern.ch/grid-deployment/glite/prepare/pps/">http://grid-deployment.web.cern.ch/grid-deployment/glite/prepare/pps/</a>
PPS		/afs/cern.ch/project/gd/www/glite/pps/	<a href="http://grid-deployment.web.cern.ch/grid-deployment/glite/pps/">http://grid-deployment.web.cern.ch/grid-deployment/glite/pps/</a>
PROD (prepare)		/afs/cern.ch/project/gd/www/glite/prepare/prod/	<a href="http://grid-deployment.web.cern.ch/grid-deployment/glite/prepare/prod/">http://grid-deployment.web.cern.ch/grid-deployment/glite/prepare/prod/</a>
PROD		/afs/cern.ch/project/egEE/gLite/	<a href="http://linuxsoft.cern.ch/EGEE/gLite/R3.1/">http://linuxsoft.cern.ch/EGEE/gLite/R3.1/</a>
Integration		/afs/cern.ch/project/gd/www/glite/integration/repository/externals	None
ETICS		/afs/cern.ch/project/etics/repository	<a href="http://eticssoft.web.cern.ch/eticssoft/repository/">http://eticssoft.web.cern.ch/eticssoft/repository/</a>

- **General Integration Procedure**
  - <https://twiki.cern.ch/twiki/bin/view/EGEE/IntegrationProcedure>
- **How to fill out a patch**
  - <https://twiki.cern.ch/twiki/bin/view/EGEE/HowToFillAPatch>
  - EGEE MSA 3.2
- **Check affected Metapackages**
  - <http://glite.cvs.cern.ch:8180/cgi-bin/glite.cgi/org.glite.savannah.viewer/cli/check-affected-metapackages.sh?view=log>
- **Savannah client Interface**
  - <https://twiki.cern.ch/twiki/bin/view/EGEE/SavannahCommandLineInterface>
- **EMT**
  - <https://twiki.cern.ch/twiki/bin/view/EGEE/EMTHome>
- **Node Tracker**
  - <https://twiki.cern.ch/twiki/bin/view/EGEE/Glite31NodeTracker>