Porting to SL6
&
Road to OSI

Cristina Aiftimiei (INFN)

Third EMI All-Hands Meeting

Padova, 17-19 Oct. 2011

EMI is partially funded by the European Commission under Grant Agreement RI-261611

Overview

- Porting to SL6
  - Why/How
  - Details
    - *Project-config*
    - *Repositories*
    - *Nightly-Builds*
    - *Build commands*
  - Deadlines
- Road to OSI

Why/How

- Mandatory platforms for EMI 2
  - SL6 and Debian6, 64bit
- Start work for SL6 within EMI 1, provide support as EMI 1 Updates
  - “infrastructure” ready:
    - *ETICS nodes & client*
    - *EMI 1 repositories, including third-party dependencies*

Details

- Project-configs
  - emi_B_1_dev – to prepare sub-systems configs
  - emi_R_1_rc – to release
- Repositories
  - 1/sl6/{arch,SRPMS}/{base,updates,tgz,third-party}
  - testing/ and deployment/
- NBs enabled for emi_R_1_rc, 12:00 & 22:00
- Build commands:
  - **python etics-client-setup --release 1.5.5-3 --volatile etics_3_5_0 [--skip-plugins]**
  - **etics-checkout --config <*subsys-config*> --project emi --project-config emi_R_1_rc --forcecheckout --verbose –continueonerror <*emi.subsystem*>**
  - **etics-build --config <*subsys-config*> --verbose --target postpublish --continueonerror --repackage=emi-1-sl6-x86_64 <*emi.subsystem*>**
  - Platform: **sl6_x86_64_gcc445EPEL**
  - ETICS Client version: 1.5.5-3: “**client_release=1.5.5-3**”

Deadlines

- **01.Nov.2011**
  - dCache, UNICORE, MPI, LRMS modules
  - VOMS - voms, voms-devel, voms-java-api
  - GliteInformationSystem: - bdii, glue-schema, glite-info-provider-[service|ldap]
  - Glite Security – lcas, lcmaps, trustmanager
- **07.Nov.2011**
  - CESNETSecurity - gridsite-[shared|apache], glite-security-gsoap-plugin, glite-security-gss, glite-px-proxyrenewal
  - gLiteL&B, DGAS, APEL, ARGUS (services +pep-java)
- **11.Nov.2011**
  - DataManagement
  - CREAM+WMS, ARC, StoRM, AMGA

Issues

- Most components should build out-of-the-box
  - using the default platform configuration
- But problems may arise in some cases
  - Differences between packaging/naming for dependencies in SL5 & SL6
- Solutions:
  1) Changes in project-config – platform dependent properties
  2) Changes in subsystems/components configs
- Either solution – discussed in the EMT scope

Road to OSI

- Goal – “binary packages of a high enough level of maturity and maintained by expert open source packagers are distributed via officially endorsed repositories like EPEL, Debian, Fedora, Maven Central”
- Achieve by:

  1) “high level of maturity” ~ 100% EPEL compliance (Rpmlint) & srpms maintained by PTs (packager) → target EMI 2

  2) “expert open source packagers” ~ PTs own “packager” or “trusted” packager maintain/push to endorsed repositories → target EMI 3

“high level of maturity”

- EMI 2:
  - 100% srpms managed by PTs
  - 100% Rpmlint success
- Technical details:
  1) checkout in ETICS a srpm done elsewhere
  2) checkout source code in ETICS and produce a srpm dynamically (specfile)
  - In both cases build will be done using mock

Technical Details

- Need of 2 project-configs:
  - One for EMI 1 SL5/6 – with present sub-system configurations
  - One for EMI 1 SL5/6 where configs are added as soon as they become “compliant”