

EMI Testbed Improvements and Lessons Learned from the EMI 3 Release

CAPANNINI, Fabio (INFN)

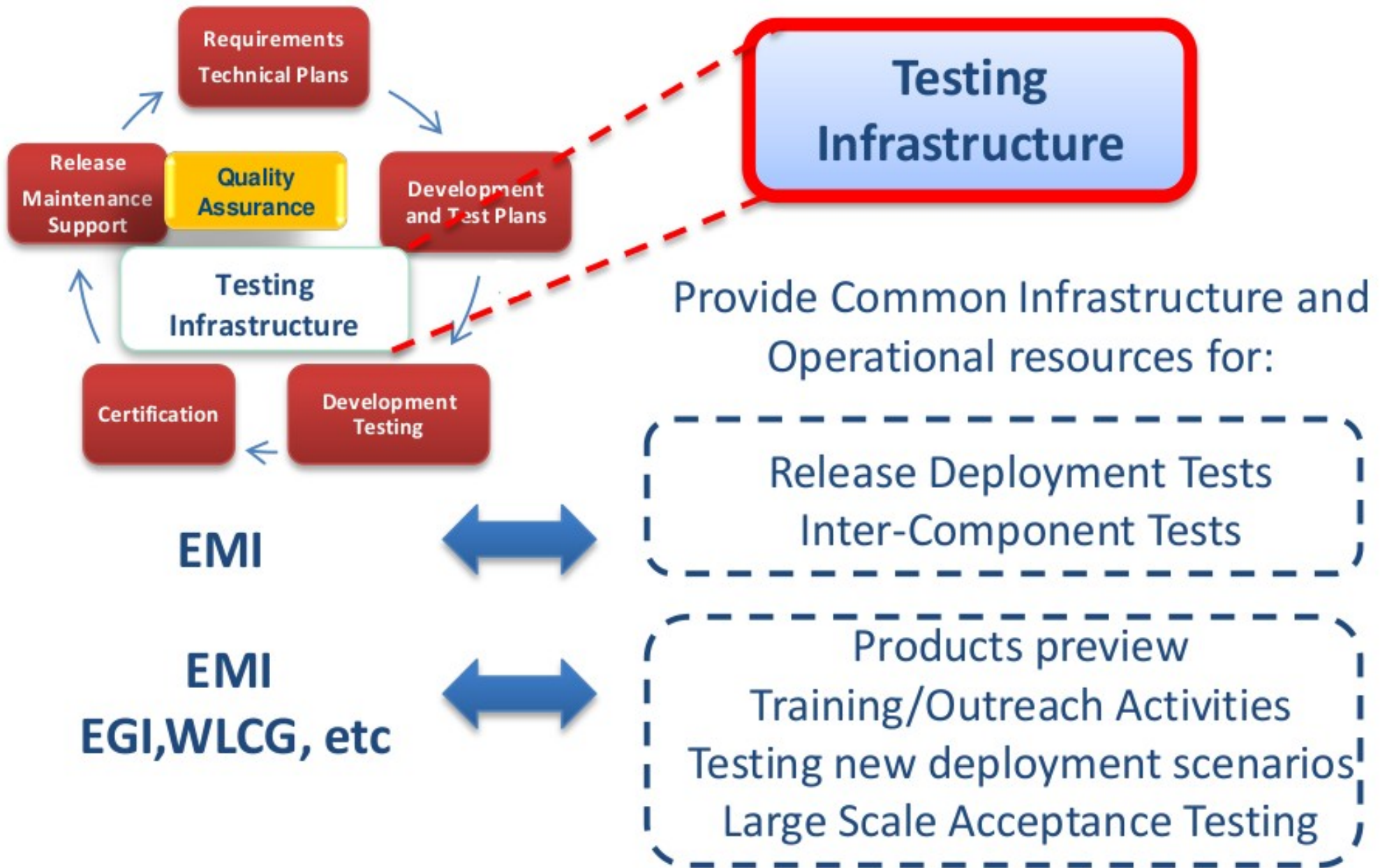
Manchester, 10-04-2013

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

- Context:
 - Role of Testing Infrastructure as part of EMI Quality Assurance
- EMI Internal Release Testing
 - Release Deployment Tests
 - Inter-Component Testing
 - Automation: Testsuites
- Activity with EMI Partners
 - Large Scale Testbed
- Lesson learnt and EMI 3 deployment strategy

- Quality Assurance Policies
 - definition and monitoring
- Metrics and KPIs
 - definition and collection
- Quality Control
 - verification and reporting
- Build/Test Tools and Repositories
 - selection, maintenance and integration
- Certification / Integration / Large Scale Testbeds
 - setup, maintenance and deployment

Role of Testing Infrastructure



Inter-Component testing infrastructure numbers:



- Resources: 250 instances, a snapshot of pre-EMI, EMI-1, EMI-2 RC, EMI-3 RC, tools testing
- Sites involved: CERN, CESNET, CNAF, DESY, JUELICH, KOSICE, NIIF
- Testbed Views: 3 information system services for production versions + 3 for RC versions
- Monitoring: SAM-NAGIOS / Nagios, service availability, service probes
- Testers VO: testers.eu-emi.eu, testers2.eu-emi.eu, testers3.eu-emi.eu
- Support: GGUS, with EMI-Testbed Support Unit
- Documentation:
<https://twiki.cern.ch/twiki/bin/view/EMI/TestBed>

- Increased testbed dimension

Releases	pre-EMI	EMI-1	EMI-2-3
Platform	SL4/SL5, debian	SL5/64	SL5/64, SL6/64, Debian6

- Increased scope of activities
 - Deployment Tests (Updates + EMI3 RC started)
 - Feedback on documentation, deployment logbooks
 - Migration to EMI Tests + Dissemination: support to WLCG
 - Release level Bug Fix / Workaround verification
 - SAM-NAGIOS Probes testing (EGI-EMI)

- **Changes in Integration Testing approach: from Continuous Integration (not ready for that yet)...**
 - Too much manual intervention needed in the process
 - High number of (Products, Platforms, Releases) combinations
 - Schedule coordination and specific setup needed across PTs
- **...to Integration Tests campaigns**
 - Created an Integration Testing Task Force
 - Defined list of integration tests (26 tests)
 - Changed Release cycle to have serial deployment/testing campaign of release for each platform (1 week / platform)

- **Product Testing tools overview**
 - No homogeneous approach, but ongoing effort to increase

	ARC	dCache	gLite	Unicore
Automated testing	<p>Automated Revision tests /code integrity (link)</p> <p>Automated functionality tests (link)</p> <p>Performance testing framework (link)</p>	<p>Jenkins framework (unit test, static code, automated g2 and s2 tests for srm)</p>	<p>Not exhaustive list: DPM, LFC, FTS (SAKET) gLite Computing (Robot Framework) Various PT-Made (ex. STORM, VOMS (etics), CaNL, gridsite, LB, proxyrenewal</p>	<p>Atlassian Bamboo</p>

- GOAL: Provide an infrastructure for large scale acceptance, scalability and interoperability testing of EMI components.
- WHAT IS IN PLACE: 16 Instances across 4 sites (documentation)
 1. Pre-Deployment across voluntary sites of RC versions
 2. Demand (from EMI developers to test specific scenario) and Supply (from volunteers)
- Less interest in approach (1.): sites go for roll-out
- Better to focus on early testing of specific scenarios (2.) but DEMAND also coming from users bringing real use cases experience.

- New services deployed on the testbed:
 - **EMIR** - A Service Endpoint Registry conceived during the EMI project. Its main goal is to discover all the Service Endpoints that exist
 - **HYDRA** - file encryption/decryption service
 - **Pseudonymity** - meant for providing its users a way to hide their true identity behind a pseudonymous identity
- Inter component integration testing campaign covering 26 tests
- EMI1, EMI2, EMI3 run in parallel on the testbed
- New VOMS VO: testers3.eu-emi.eu

- Issues identified only during deployment in EMI testbed phase
- Integration testing for EMI2 not successful: delays in having the testbed in shape due to products not deployable
- **Need for:**
 - more efficient and complete testing activity
 - Increased product quality and control over the process
 - Shorter delays in the release process
 - Early identification of the problems occurring at the interface between development and testing
 - Better coupling of development and testing phases

Two-step deployment:

- In the **first phase** the different product teams are responsible for providing and maintaining an instance of their service where the initial testing validation is carried out.
 - During this phase the SA2 people responsible for the testbed would take care of hosting and configuring a set of "core" services, which lay the foundation of the integration testbed (i.e. voms, top-bdii) that other services will reference during the interoperability and integration testing campaigns.
- In a **second phase** of the testing process the services are deployed on the central testbed and a new set of tests is performed centrally as a second independent check.

- **EMI3** testbed deployment proved more straightforward and less prone to deployment issues than the preceding release deployments
- EMI testbed proved to be a fundamental tool for interoperability tests and of great help in spotting issues in products that would have been otherwise overlooked
- Large scale testbeds: focus is now on users real case tests and early testing of specific scenarios
- More testing and integration of latest nagios probes is needed
- Continuous integration testing not feasible yet within EMI context, maybe room for other integration testing campaigns
- EMI testbed already grown to considerable dimension

HAGEMEIER, Bjoern (JUELICH)

ELWELL, Andrew (CERN)

BERNARDT, Christian (DESY IT)

KOCAN, Marek (Kosice)

DVOŘÁK, František (CESNET)

DONGIOVANNI, Danilo (INFN)

AIFTIMIEI, Cristina (INFN)

CECCANTI, Andrea (INFN)

CRISTOFORI, Andrea (INFN)