Enabling Grids for E-sciencE

SL4 Status and Plans February 7th 2007

Markus Schulz

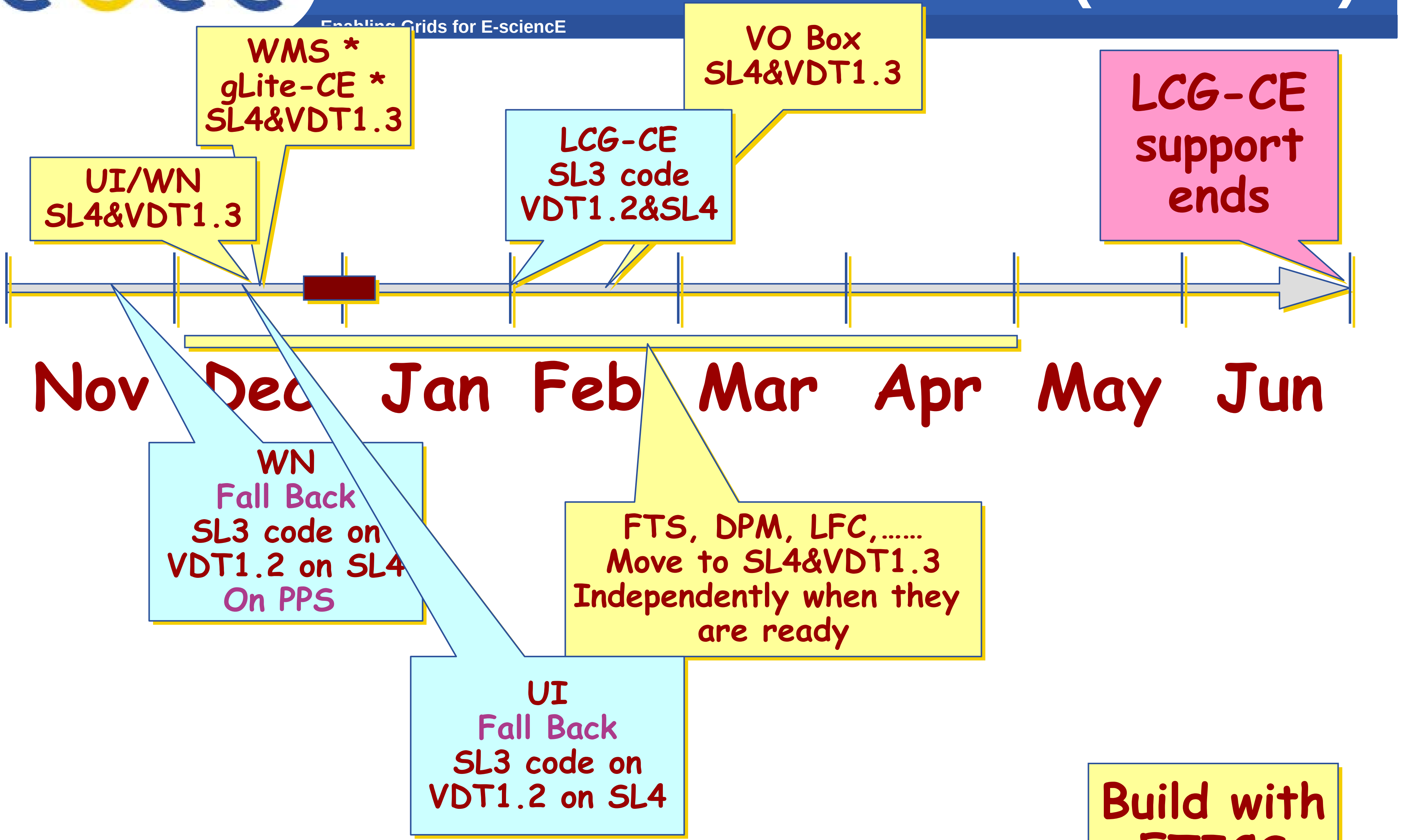
SA3

CERN-IT-GD

www.eu-egee.org

Information Society and Media

Timeline (Planned)



*) The code version depends on test results

Build with ETICS

Status

- **Build with ETICS progressing slowly**
  - “Magic Script” that converts gLite build config. provides only 50% of the magic
- **Refocused on WNs and Uis**
  - UI 32bit SL4 (74%)
  - WN 32 bit SL4 (82%)
- **What fails:**
  - org.glite.security.voms
  - org.glite.security.voms-clients
  - org.glite.security.voms-api
  - org.glite.security.voms-api-c
  - org.glite.security.cgsi-gsoap
  - org.glite.data.srm-api-c
  - org.glite.data.srm-cli
  - org.glite.data.srm-api-perl
  - org.glite.wms.common
  - org.glite.security.proxyrenewal
  - org.glite.jp.ws-interface

Status

- **What fails II :**
  - org.glite.jp.client
  - org.glite.wms.purger
  - edg-replica-location-client-cpp
  - edg-replica-manager-client
  - org.glite.ce.wsdl
  - glite.ce.monitor-client-api-c ▪
  - org.glite.wms.ism
  - edg-replica-metadata-catalog-client-cpp
  - edg-replica-optimization-client-cpp
  - org.glite.wms.rls
  - org.glite.wms.brokerinfo
  - org.glite.wms.classad_plugin
  - org.glite.wms.matchmaking
  - org.glite.wms.broker
  - org.glite.wms.ns-common
  - org.glite.wms.ns-client

Status

- What fails III :
- org.glite.wms-ui.wrap-java
- org.glite.wms-ui.wrap-python
- org.glite.security.voms-api-cpp
- zlib
- org.glite.wms.wmproxy-api-cpp
- org.glite.wms.client
- org.glite.wms-ui.api-cpp
- org.glite.ce.cream-client-api-c
- org.glite.ce.cream-cli
- Dcap
- LCG-DM
- org.glite.data.gfal
- org.glite.data.dm-util
- lcg-dm-common
- lcg-vomscerts

Status

- What fails III :
- gssklog-cern
- Yaim
- globus-config
- edg-gridftp-client
- edg-mkgridmap
- org.glite.data.catalog-api-c
- org.glite.data.catalog-cli
- xalan-c
- org.glite.data.config-service
- org.glite.data.transfer-cli
- org.glite.data.transfer-api-c
- Workload

Status

- **Why do things fail?**
  - Not in the system
  - **Incomplete build config**.
  - Missing externals
  - Core (security packages) fails
  - Magic Script not perfect

- **What do we do about it?**
  - Follow up with developers (EMT)
  - Adding external dependencies (ETICS team)
  - Converting failed builds into Savannah bugs
  - ETICS <-> Developer dialog and support

- **When will it end?**
  - Not within the next 2 weeks
    - Serial nature of the problem
      - *Makes it impossible to give a good estimate*
  - Then 4 weeks of testing AFTER first successful build

Progress?

- **Fallback Solution Status**
- **VDT 1.6 delivered for all flavors requested**
  - Not in ETICS (requires Alberto)

- **SL3 on SL4 UI and WN**

- **UI tarball**
  - Used on PPS
  - Only minor issues related to Python paths and the Python 2.2
    - Libs used by Python etc.
- **WN tarball**
  - Has been tested
  - Passes SAM Tests
  - Same material as RPM based
- **RPM based WN (SL3 on SL4)**
  - Extensively tested by all experiments!!!
    - Minor issues with Python 2.2
  - **1st. production style tests on pps by 3 out of 4 experiments** <--+++++

Progress?

- **RPM based WN (SL3 on SL4)**
- **Why is this not released to production?**

- **Packaging (apt) problem:**
  - One MetaRPM using all SL3 RPMs from SL3 repositories
  - Installs fine ++++
  - Works fine ++++
  - Can’t be upgraded -----------

- **Sites can use the WN tarball**
- **Integration Team is working on a more complex solution**
  - One MetaRPM
  - Extra SL4 repository
    - With hand picked RPMs
    - Requires significant effort to build
    - Requires significant effort to maintain
    - These are the same people that can help with the real SL4 build!! <-------

Dependencies and other Sins

- **SL4 (and other) builds are delayed because:**
  - Many interdependencies between middleware components
  - Structure of the software:
    - Project, subsystem, component
      - *Back references make builds very complex*
      - *No clear split between clients with minimal requirements and services*
        - Component updates require updates of not related nodes
        - Have a look at our release page
  - Structure of the gLite build (which has to be exported to ETICS)
    - The build intelligence in the gLite system tried to achieve consistency
    - This was achieved by the way builds are done
      - *Costly subsystem builds*
    - This can work despite missing explicit information on component level
      - *Which made component based releases very hard*
    - gLite build concept:
      - *Assuming reflective dependencies:*
        - You build WMS
        - You rebuild security too, because this might be affected (shared dependencies)

ETICS

- **The problem that we have is NOT rooted in ETICS**
  - Live without ETICS would be at least as bad it is now
    - And we would loose the support of the ETICS team
- **Moving to a new build system is part of the problem**
  - But this is unavoidable
- **History of SA3 and ETICS:**
  - First trials with ETICS in March 2006
    - Expected to be ready in April
    - No integrated system
  - June basic functionality was there
    - Missing edit functionality (ETICS team had to edit meta data)
  - August edit available
    - First tutorial EGEE-06 in September
  - Since September
    - People got trained
    - User feedback
    - Production bugs
  - Started with monolithic gLite-3.1 build
    - Moving to node type driven build
      - *Not easy because of the software structure*
  - People active in moving gLite to ETICS : ~3