gLite Release Process

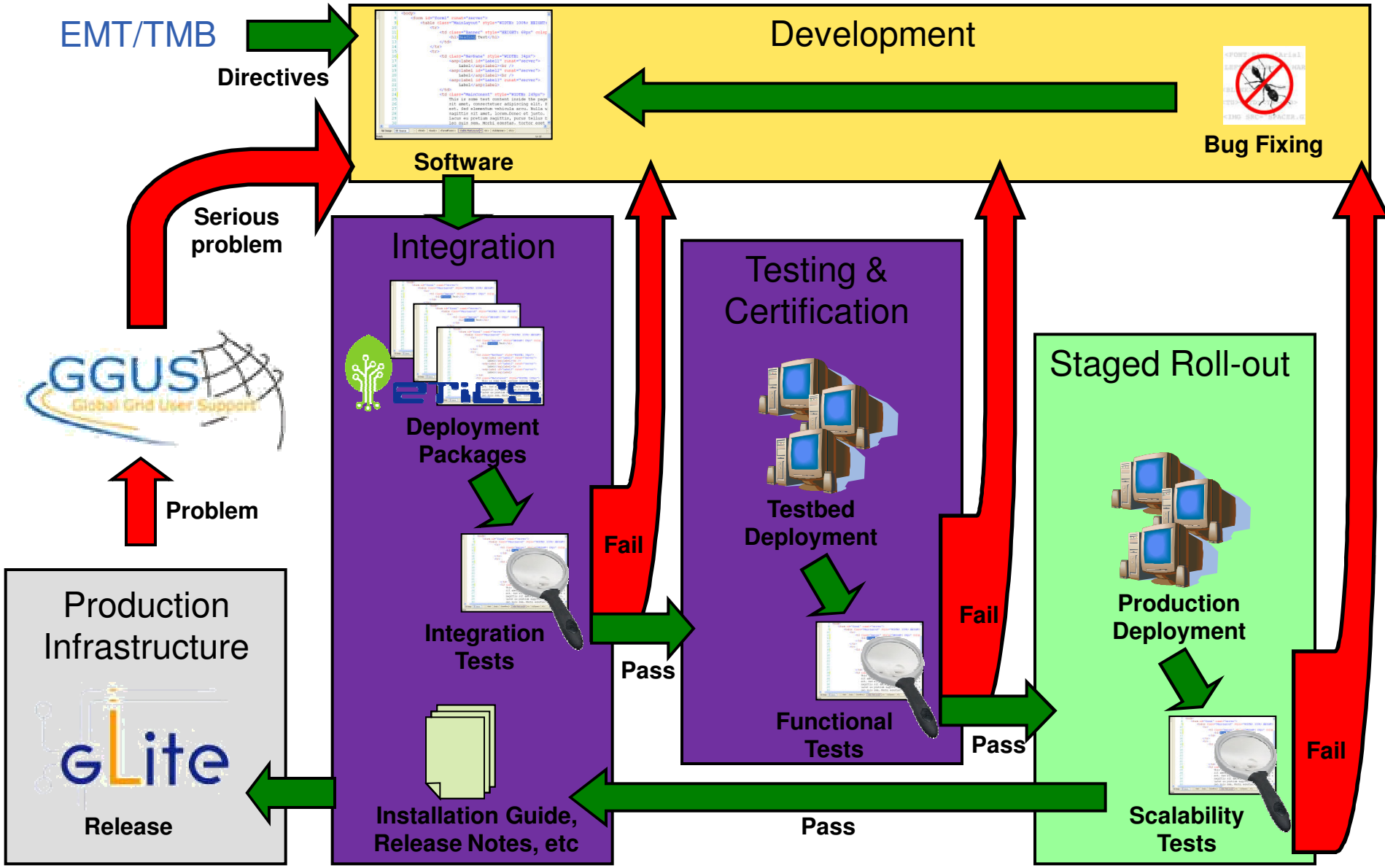
Maria Alandes Pradillo

EMI Workshop 7th April, CERN

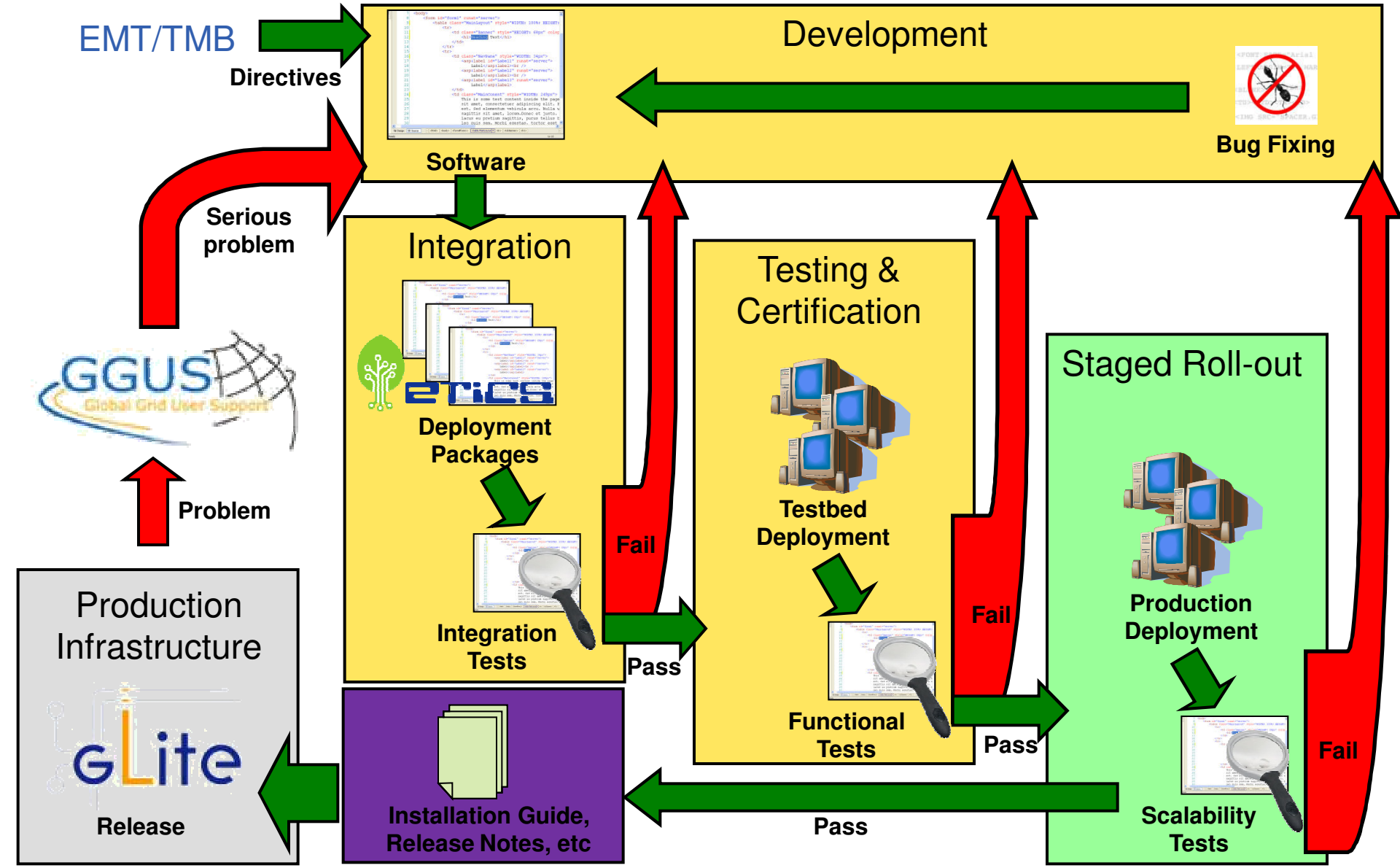
Contents

- Before Product Teams
- With Product Teams
- Basic Conceps
- Before Certification
- During Certification
- After Certification
- Where we are now
- Things to Improve
- Lessons learned

Before Product Teams



With product Teams



Basic Concepts

- https://twiki.cern.ch/twiki/bin/view/EGEE/ProdInt
- **Changes are managed in Savannah**
  - Changes are tracked in bugs
    - One bug -> but many platforms! ☹
  - Patches are the way to release changes
    - One patch -> per platform ☺
- **Software is build in ETICS**
  - One ETICS project configuration per gLite release
    - List of certified component versions that are part of the release
  - One metapackage component per gLite metapackage
    - List of component versions that are part of the metapackage
- **Repositories are now separated**
  - One repository per metapackage

Before Certification

- **Metapackage patch or Internal patch?**
  - Metapackage patch if you have a metapackage
  - Internal patch if you don't. For example:
    - Service Information Provider
    - lcg-infosites or lcg-ManageVOTags
    - APEL packages
- **Build your packages with ETICS**
  - Build and lock against one of the project configurations
  - Create a YUM repository to be used in certification
- **Fill in the Savannah Patch**
  - Add ETICS information: package list + configuration + YUM repo
  - Attach bugs + Patches
  - READY FOR CERTIFICATION!

During Certification

- **Guidelines of what to test in https://twiki.cern.ch/twiki/bin/view/Main/HowToCertify**
- **Internal patches are sometimes difficult to certify**
  - Collaboration between PTs is needed
- **Metapackage patches should be certified as a whole**
  - PTs need to make sure that changes introduced by others are working fine in the final integrated metapackage
- **Detailed and complete test reports are needed**

After Certification

- **Integration Team verifies the patch**
  - Deployment tests OK?
  - Basic tests OK?
  - Bug fixes verified?
- **Release Manager updates the ETICS project configurations**
- **Patches and then ready to be released:**
  - Internal Patch -> Closed
  - Metapackage Patch -> Candidate for Staged rollout
- **Staged rollout/Production cycle happens ~2 weeks**
  - Triggered by Operations
  - Contains a set of patches

Where we are now

- **Not many released patches with the new process**
  - It took a while to fix all build errors in the ETICS project configurations (In 3.1, only a couple of weeks ago!)
  - 5 metapackage patches in 3.2, None in 3.1
  - So the process is not very mature yet
  - Although it is defined since December
- **Metapackage dependency issues in ETICS**
  - Very important to understand the tools we use
- **We have a complete and coherent set of configurations in ETICS**
- **The process is documented and most of the PTs are already familiar with it**

Things to improve

- **Creation of a Savannah patch**
  - Developers need to manually add
    - The list of new packages (and find out which ones they are!)
    - The ETICS configurations
  - They forget packages or put the wrong ETICS configuration
  - This is only detected when preparing the release
- **Monitoring other patches affecting your metapackage**
  - This twiki tries to help:
    https://twiki.cern.ch/twiki/bin/view/EGEE/InternalPatch
  - But it's easy to make errors, forget to update... and it has a maintenance overhead!
  - How can PTs know what other changes have been done?
- **AUTOMATION is crutial to save time and avoid human errors**

Lessons learned

- **Release notes**
  - They are VERY important
  - It's not a negligible work and requires time
  - Expert people should be involved in the process
- **Repositories**
  - They seem to be easy to manage BUT
    - gLite has a lot of packages
    - When automating the creation of repositories you may forget a package
      - *It's worth writing scripts to run deployment tests after the update of a repository*
      - *We didn't test tarballs or rpm lists and bugs were only discovered in production. Maybe it's worth testing them as well.*