Status of the Integration with FTS: ALICE Report

Patricia Méndez CERN (IT-PSS) / INFN(CNAF)
Pablo Saiz CERN (IT-PSS)

FTS workshop for Experiments Integrators

CERN

16 November 2005

Outlook

- ALICE Production: General Points
- Current status of the integration
- FTS Use
- Issues
- Features to add
- Summary

ALICE Production: General Points

- **ALICE has it own Task Queue and related services and use the LCG RB to submit the jobs**
  - Pull Model service: a server holds a master queue of jobs and it is up to the CE that provides the CPU cycles. It asks for the jobs
  - No Information System is included
  - It offers a single interface for ALICE users into the complex, heterogeneous (multiple Grids) and fast-evolving Grid reality
- **Several Grid infrastructures are available during their Data Challenges**
  - LCG INFNGRID possible others in US
  - Lots of resources but different middleware

ALICE Production: General Points

Design Strategy:

- **Use AliEn as a general front-end**
  - The resource is used transparently and independently of the middleware system behind
- **Minimize points of contact between the systems**
  - No need to re-implement services
  - No special requirements to run on remote CE/WNs
- **Make full use of the provided services**
  - Let the Grids do their work
- **Use high-level tools and APIs to access Grid resources**
  - Developers put a lot of abstraction effort into hiding the complexity and shielding the user from implementation changes

ALICE Production: General Points

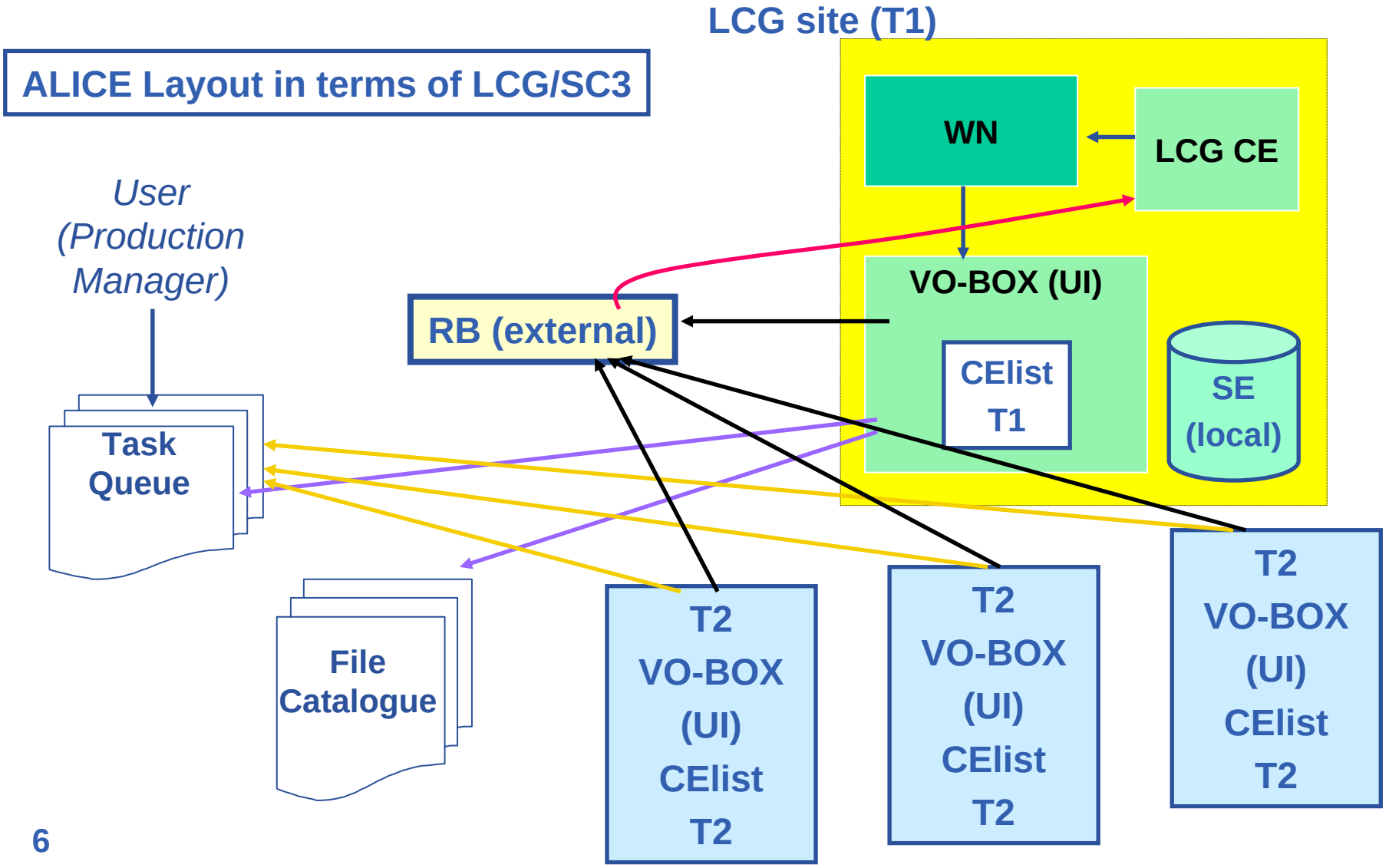
LCG'2 core sites

- CERN, CNAF, FZK, NIKHEF, RAL, Lyon, Taiwan (more than 1000 CPUs)
- Each LCG site with a VO-BOX seen independent
- AliEn services (CE+SE sited in each VO BOX)

INFNGRID

- LNL.INFN, PD.INFN and several smaller ones (400 CPUs not including CNAF)

Interfacing AliEn and LCG



ALICE Production: General Points

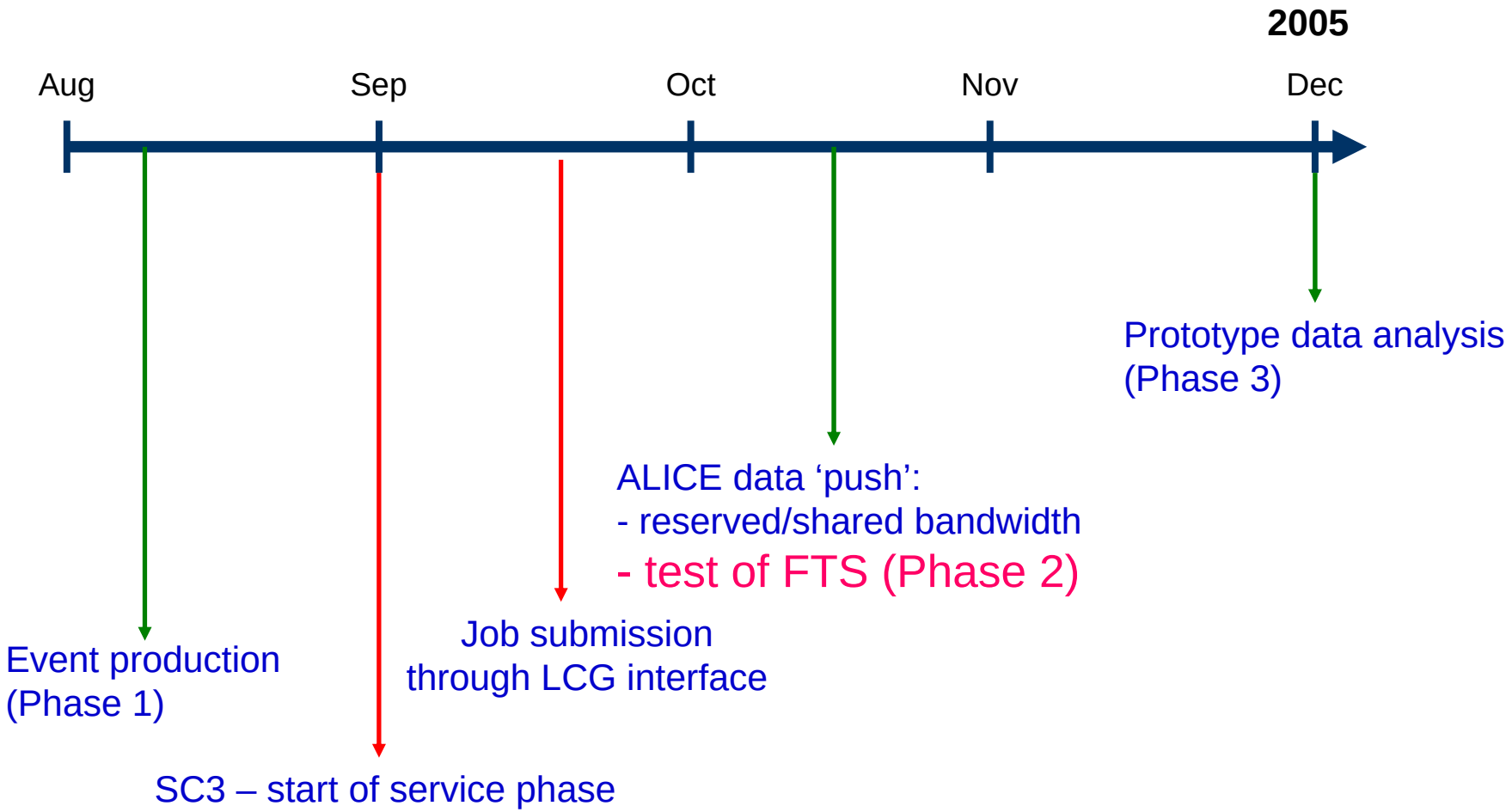
Physics Data Challenge: Phases

- **First Phase: Simulation of Monte Carlo Events in all available resources**
  - Registration of all the outputs in the Alice File Catalog (central catalog) and store them at CERN-CASTOR (for SC3)
  - FTS is not required during this phase
  - It is however tested using the T0-T1 (daily tests) and the T1-T2 ALICE Channels

- **Second Phase: Reconstruction of the raw events stored at CERN**
  - Test of file transfer utilities (FTS)
  - Use of the local catalog at each site (LFC)

- **Third Phase: Analysis phase**
- ALICE will use only those sites involved in SC3

Timeline of PDC05/SC3



L. Betev, F. Carminati. GDB Meeting in Bologna. October 2005

FTS USE among the ALICE Goals

Primary Goals:

- Use of the deployed LCG SC3 infrastructure for the ALICE DC05
- Test of the data transfer and storage services (SC3)
- Test of distributed reconstruction and calibration model (ALICE)
- Integrate the use of LCG resources with other resources available to Alice within one single VO interface for different Grids
- Analysis of reconstructed data

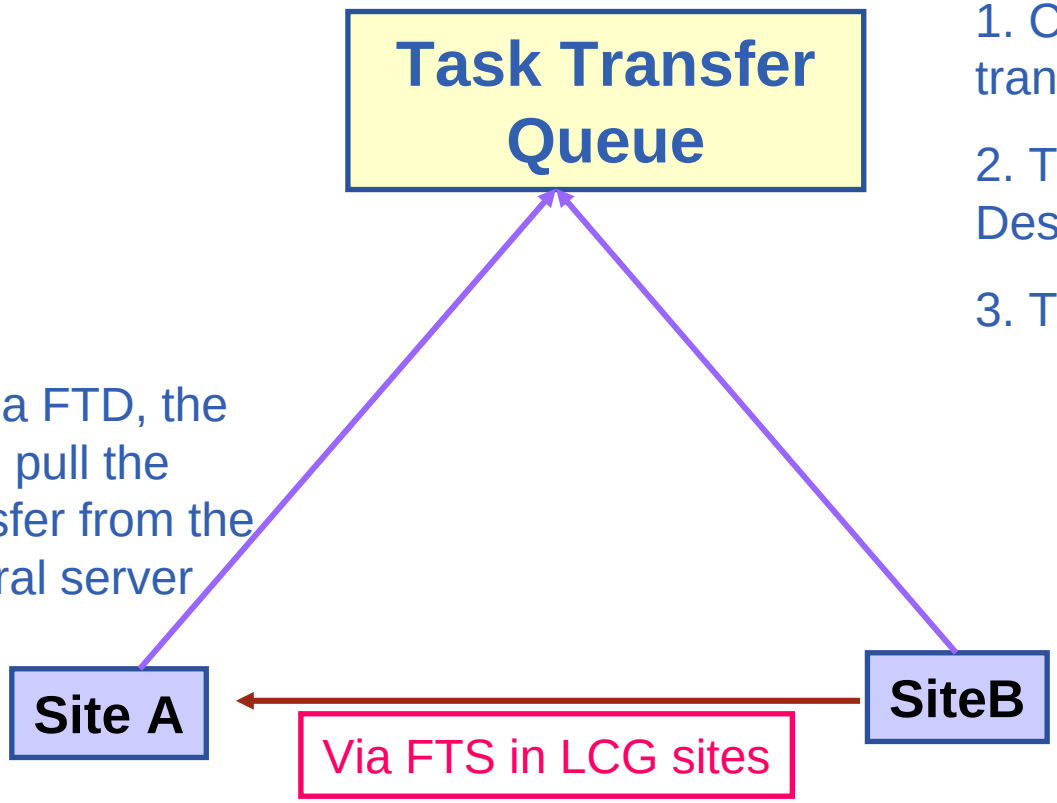
Current Status of the FTS Integration

General Overview:

- ALICE sees FTS layer as service that underlies data placement
- DC04 used FTD (with aiod as protocol)
- FTS used as FTD plug-in
  - FTD is the service implemented by ALICE inside all VO-BOXES
- It is implemented in the AliEn framework through the FTS Perl API provided by developers
- FTS has to be deployed in all VO-BOXES
  - Due to the ALICE Architecture, the “entry door” to the LCG/EGEE is the VO-BOX
  - It is required the full configuration of the VO-BOX as a UI

Current Implementation of FTS

ALICE Layout for the File Transfers



1. Contains the list of transfers
2. The Origin and the Destination (A and B)
3. The SURLs

a) Via FTD, the sites pull the transfer from the central server

b) FTD makes a “get” to obtain information of the request

c) FTS makes the transfer. FTD is an homogeneous layer for all type of sites

How it is being used?

- At this moment FTS is used at test level
- Daily T0-T1 tests are performed
  - Massively test all the T0 <--->T1, T1<--->T2 connections/endpoint involved in SC3 are planning by ALICE
  - Configure/test the script execution on VO-BOXES
- Issue: with or without LFC registration?
  - ALICE has decided to run FTS in the simpler mode: NO automatic update of the catalog is required after the transfer

How is it being used?

- **During the ALICE testing phase, the full integration of the PERL API into AliEn will be performed**
- **First Phase of the testing: T0-T1 tests and integration of FTS inside the VO-BOXES**
- **Second Phase of the testing: T1-T2 connections and deployment in VO BOXES**
- **T1-T1 channels**
  - Not explicitly required
  - Transfers can be performed via CERN as long as the procedure is transparent for ALICE (wait a pair of slides)

ALICE requires

- **ALICE requires the automatic discovery of the FTS Endpoints and the names of the FTS proxies servers through the information system**
- **An upper layer able to hide the transfers among the different SRM**
  - Not all the channels must be predefined
  - Giving as inputs the origin and destination SRM (and the SURL) the system should be able to find the best way to perform the transfer
  - Hide this procedure to users

ALICE requires

- **Simultaneous updates of the VO-BOX and FTS versions**
  - FTS has to be implemented in the VO-BOX
  - At this moment the updates are done by hand
- **Homogeneous configurations in all sites**
- **Tool for FTS traffic monitoring**
  - R-GMA probably

ALICE does NOT require

- **Automatic update of the catalog after the transfer**
  - Their system takes care of this
- **Transfers supporting LFNs are not needed**
  - AliEn defines the LFNs, not LCG
- **Tx-Ty predefined channels not needed**
  - ALICE does not want to care if there is a channel or not

Summary

- **FTS is a mandatory service for ALICE**
- **It has to be deployed in all SC3 sites**
- **It is planned to be used during the 2nd Phase of the DC05**
- **At this moment we are in testing phase**
  - Daily testing their channels
  - Good exercise for SC3 too
- **ALICE requires a transparent way to perform the transfers**
  - Independent of the predefined channels
  - Caring just about the SRM nodes and not about the SRM endpoints