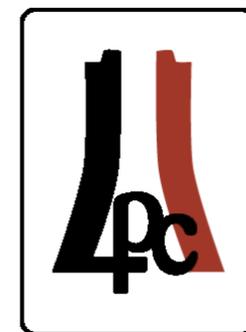# Distributed CMS Analysis on the Open Science Grid

Computing in High Energy and Nuclear Physics
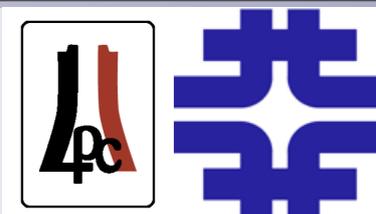13-17 February 2006, T.I.F.R. Mumbai, India

Distributed Data Analysis: DDA-5
02/15/06

Bockjoo Kim (University of Florida)
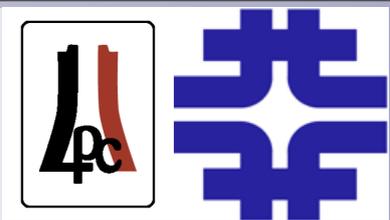Oliver Gutsche (USCMS / Fermilab)

Introduction

- CMS uses a tier-structured GRID based computing fabric to provide computing for the CMS collaboration

- The different computing centers use two GRID middleware implementations:

  - LHC Computing GRID (LCG) / Enabling Grids for E-sciencE (EGEE)

  - OpenScience GRID (OSG)

- The user interacts with the GRID via the CMS Remote Batch Builder (CRAB), which is mainly LCG/EGEE centered

- This talk describes the first integration of OSG capabilities into CRAB for the Service Challenge 3 and summaries further plans
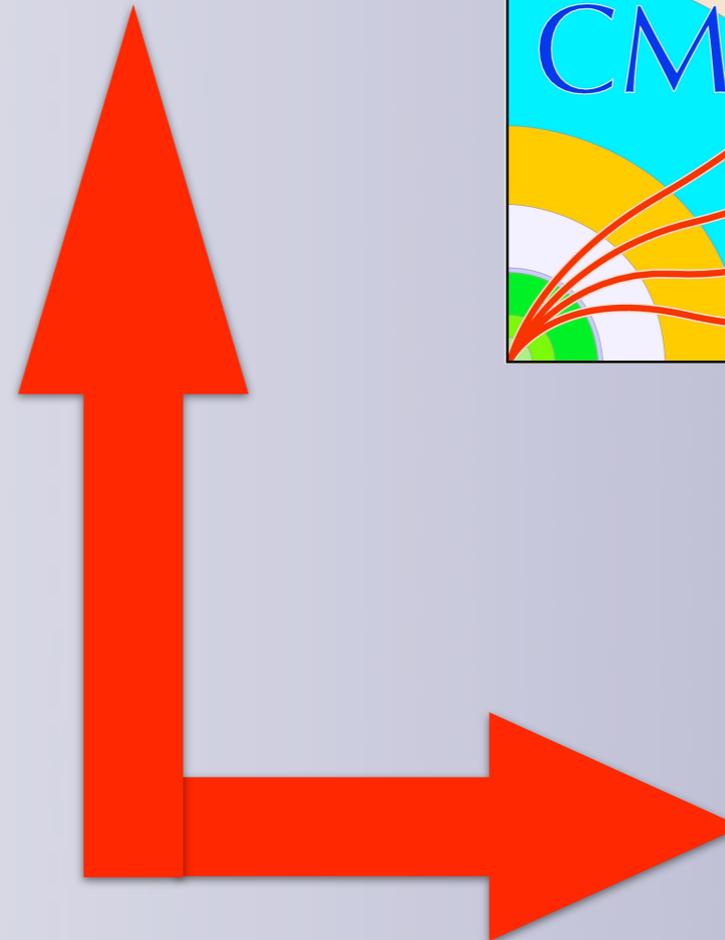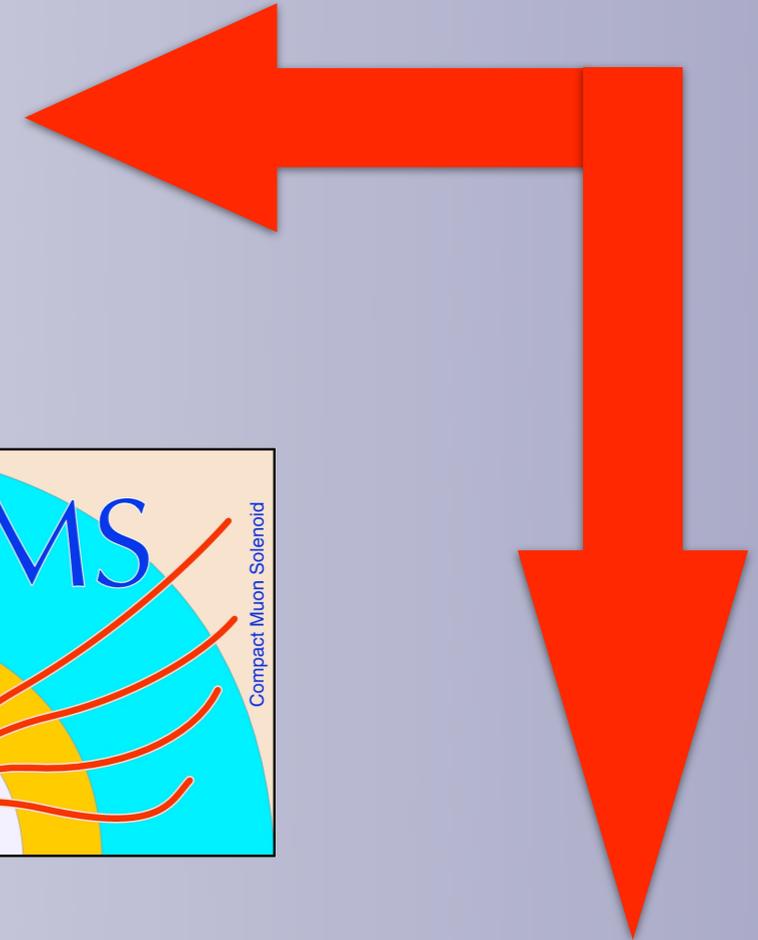
Outline

- CMS computing in the light of LCG/EGEE and OSG

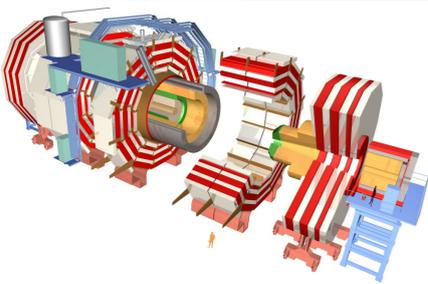- CRAB introduction

- First OSG implementation

  - Condor-G

  - Additions to CRAB

- Service Challenge 3

- Plans, Summary & Outlook

CMS Tier Structure

tier	quantity
T0	1
T1	7
T2	25

- Data is split into ~50 primary datasets (~1 trigger channel)

- Datasets are distributed amongst T1 centers (8-16 per center) and further skimmed to associated T2 centers

- Access to distributed datasets using **CRAB** essential

Middleware within CMS GRID

📌 Europe: **LCG / EGEE**

📌 USA: **OSG**

OSG contribution to CMS Tier structure

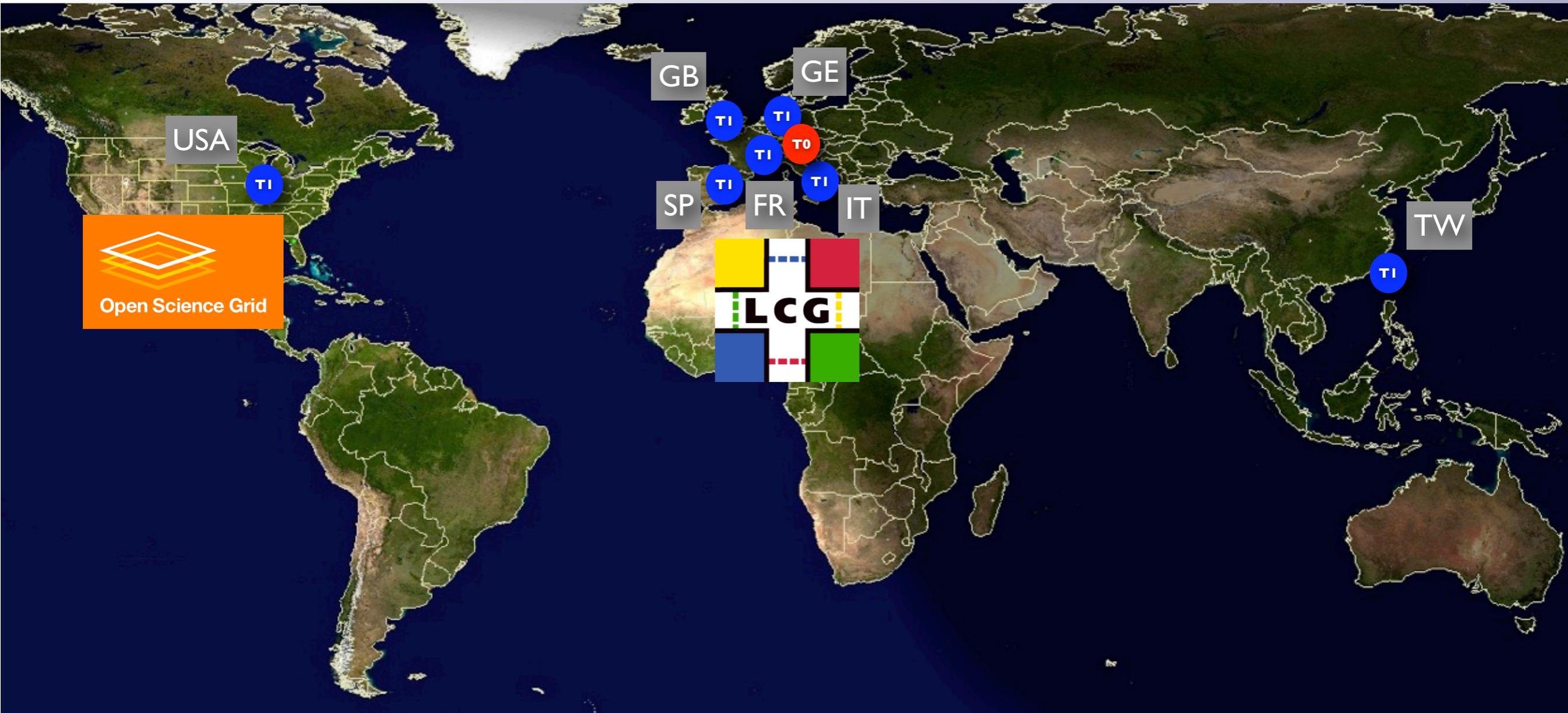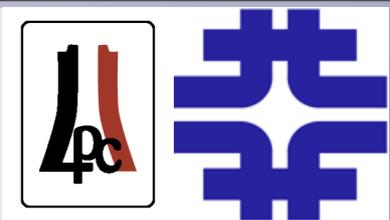- **OSG contribution to CMS tier structure**

  - **T1 at FNAL**

  - **7 attached T2 sites**

Site	Processors	Disk (TB)
Caltech	153	40
Florida	240+	73
MIT	(coming soon)	(coming soon)
Nebraska	256	19
Purdue	228	~25
San Diego	228	44.5
Wisconsin	400	50

LCG/EGEE and OSG

- LCG / EGEE middleware

  - LHC experiment specific approach using more higher level tools

    - access via EGEE tools like **edg-job-submit**

    - utilization of Resource Broker (RB)

      - load balancing

      - sandbox for user file input and output to the remote analysis application

- **OSG** middleware

  - General approach using more lower level tools (experiments have to install own tools at OSG sites)

    - access via GLOBUS tools like **globus-job-submit**

    - no RB

    - missing sandbox functionality
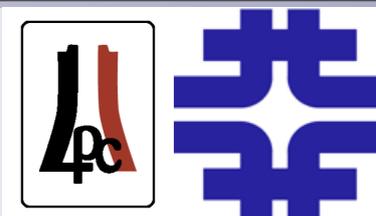
➡ CRAB build to use RB LCG/EGEE infrastructure and cannot be used directly on OSG sites

➡ **Add** functionality to CRAB to be able to also submit to OSG sites

CMS Remote Batch Builder (CRAB)

- **CRAB** provides the user with a framework to submit user analysis jobs to the CMS tier-structure (T1 and T2 centers)

  - **CRAB** takes care of

    - User code packing and submission

    - Execution on the distant site and status information

    - Output retrieval

- User interaction with the GRID infrastructure is limited to CRAB usage

- More information on CRAB:

  - Marco Corvo: CRAB: a tool to enable CMS Distributed Analysis
    [track: Distributed Data Analysis - 273]

  - Daniele Spiga: CRAB usage and jobs-flow Monitoring
    [track: Distributed Data Analysis - poster]

CRAB user interaction

- CRAB splits user interaction into 4 steps

  - Job Creation:

    - Resolve requested dataset / MC sample from global CMS service

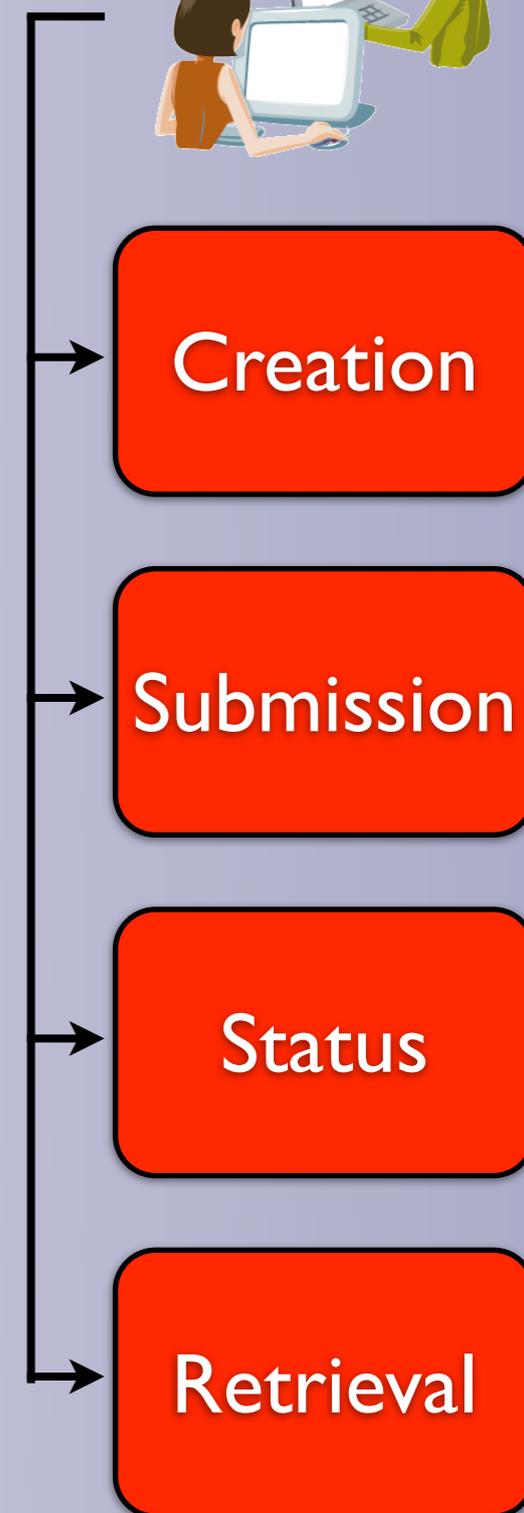   - Generate list of sites providing access

    - Creating jobs in selected job range adding site specific information (jdl, workernode script, POOL catalog retrieval, ...)

  - Job Submission

  - Status Check

  - Job Output Retrieval

Creation

Submission

Status

Retrieval

CONDOR-G

- OSG middleware is based upon VDT toolkit providing

  - GLOBUS

  - Condor-G

- First approach: CONDOR-G provides:

  - GRID submission using GLOBUS toolkit

  - access to OSG sites independent of used local batch system

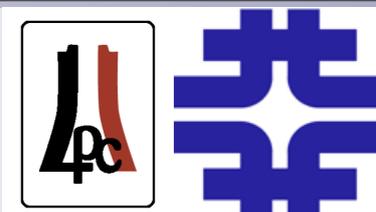 - sandbox for insertion and retrieval of files

- Requirements:

  - OSG site: Submitter:

    - none   local CONDOR installation with activated CONDOR-G

Implementation: Submitter Side

- Main prerequisite:

  - Locally installed and running Condor scheduler with Condor-G capabilities

- First approach: dedicated OSG mode for CRAB

  - Site list generation excludes non OSG sites

  - Load balancing of RB not available

    - first implementation: take first in list

  - Creation of Condor-G JDL

    - Use hardcoded information and later GridCat service for site specific information

  - Changes in workernode script to accommodate OSG implementation

- Replacement of LCG / EGEE commands with Condor commands

"Features" of CRAB OSG addition

- Submission, Status check and Output retrieval have to be executed from the same scheduler

- Submission:
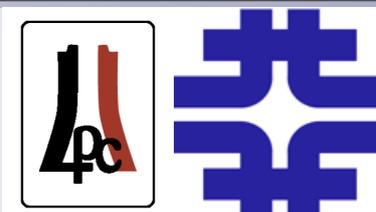
  - Direct submission to site ➡ Order of magnitude faster than RB submission, but: no load balancing, increased load on the Computing Element (CE) of the site

  - Output retrieval

  - Condor-G retrieves output automatically ➡ no need for dedicated CRAB usage step

Service Challenge 3

- Goal of Service Challenge 3 (SC3) in 2005 (11/17/05 - 12/09/05):

  - Exercise realistic CMS startup scenario

    - Transfer data to tier structure and provide access

    - Submit analysis jobs and retrieve output

      - Tool for analysis part: CRAB

- First implementation of OSG additions to CRAB was used during SC3:

  - Enabled participation of OSG sites

  - 6 OSG T2 participated

OSG site SC3 participation

Site Name	Caltech	Florida	Nebraska	Purdue	San Diego	Wisconsin	ALL OSG T2 Sites
All jobs with zero exit code	8	39	2626	360	733	2496	6262
All jobs with non-zero exit code	374	61	1172	148	2128	834	4717
All aborted jobs	409	90	183	629	574	194	2079
All completed jobs	791	190	3981	1137	3435	3524	**13058**

- **13000** analysis jobs have been run against MC samples available at OSG sites transfered during SC3

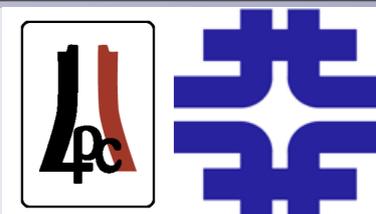- Each job processed 1000 events and finished well below 8 hours

- zero exit code: job finished with success

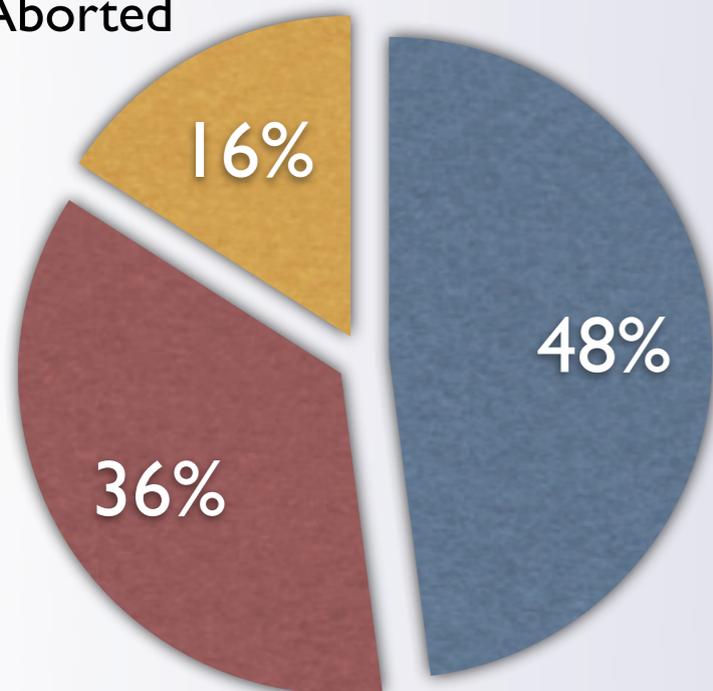- non-zero exit code: specific executable failure exit code

- aborted: problems in GRID submission or output retrieval

Condor-G related failures

exit code zero
exit code non-zero
Aborted

16%

48%

36%

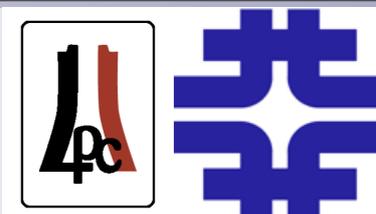**16% aborted jobs**

71% of aborted jobs are related to output retrieval errors

Resulting mostly from application problems

Remaining errors related to communication problems between submitter and site

Globus error	Description	Percentage [%]
7	an authorization operation failed	12.67
10	data transfer to the server failed	0.96
17	the job failed when the job manager attempted to run it	0.43
22	the job manager failed to create an internal script argument file	4.38
24	the job manager detected an invalid script response	0.05
30	the job manager failed to open the user proxy	1.78
43	the job manager failed to stage the executable	2.79
48	the provided RSL could not be properly parsed	0.10
73	the job manager failed to open stdout	0.19
74	the job manager failed to open stderr	0.19
79	connecting to the job manager failed.	0.24
93	the gatekeeper failed to find the requested service	0.10
111	the job manager timed out while waiting for a commit signal	0.14
121	the job state file doesn't exist	0.05
129	the standard output/error size is different	0.10
131	the user proxy expired (job is still running)	1.01
135	the job manager could not stage in a file	3.76
136	the scratch directory could not be created	0.24
156	the job contact string does not match any which the job manager is handling	0.05
158	the job manager could not lock the state lock file	0.05
155	**the job manager could not stage out a file**	**70.71**

Plans

- OSG additions to CRAB will be finalized in latest CRAB Version 1:

  - Two submission possibilities will be implemented:

    - **Condor-G:** the presented implementation will be integrated into the new CRAB Version 1

    - **Resource Broker:** OSG sites will report to the LCG / EGEE Resource Broker to enable submission to OSG sites via the RB

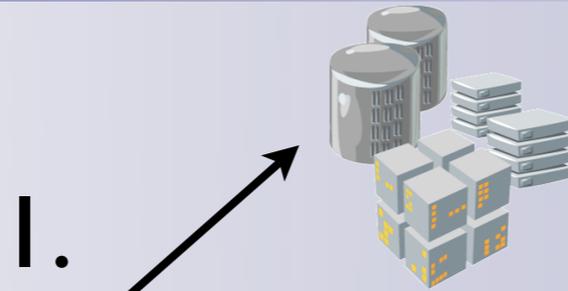- Workernode scripts will be adapted to be able to run both on LCG / EGEE sites and OSG sites with "late discovery"

Summary & Outlook

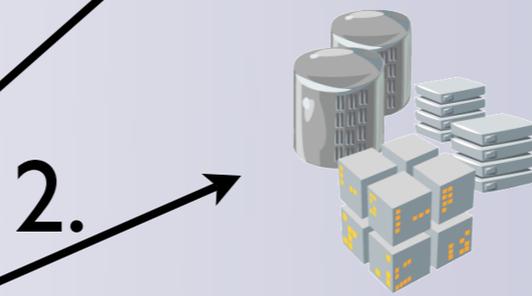- **CRAB** provides user access to datasets and MC samples distributed worldwide within the CMS tier structure

- The two middlewares, LCG / EGEE and OSG, have both been integrated the first time into the common submission tool.

- The OSG enabled CRAB version has been used successfully during the Service Challenge 3

  - The GRID / Condor-G related failure rate is very low

- Plans foresee to finalize the OSG implementation into CRAB to enable job submission from a unique interface to OSG sites via the LCG / EGEE resource broker and Condor-G.

User:
request to analyze
dataset with user code
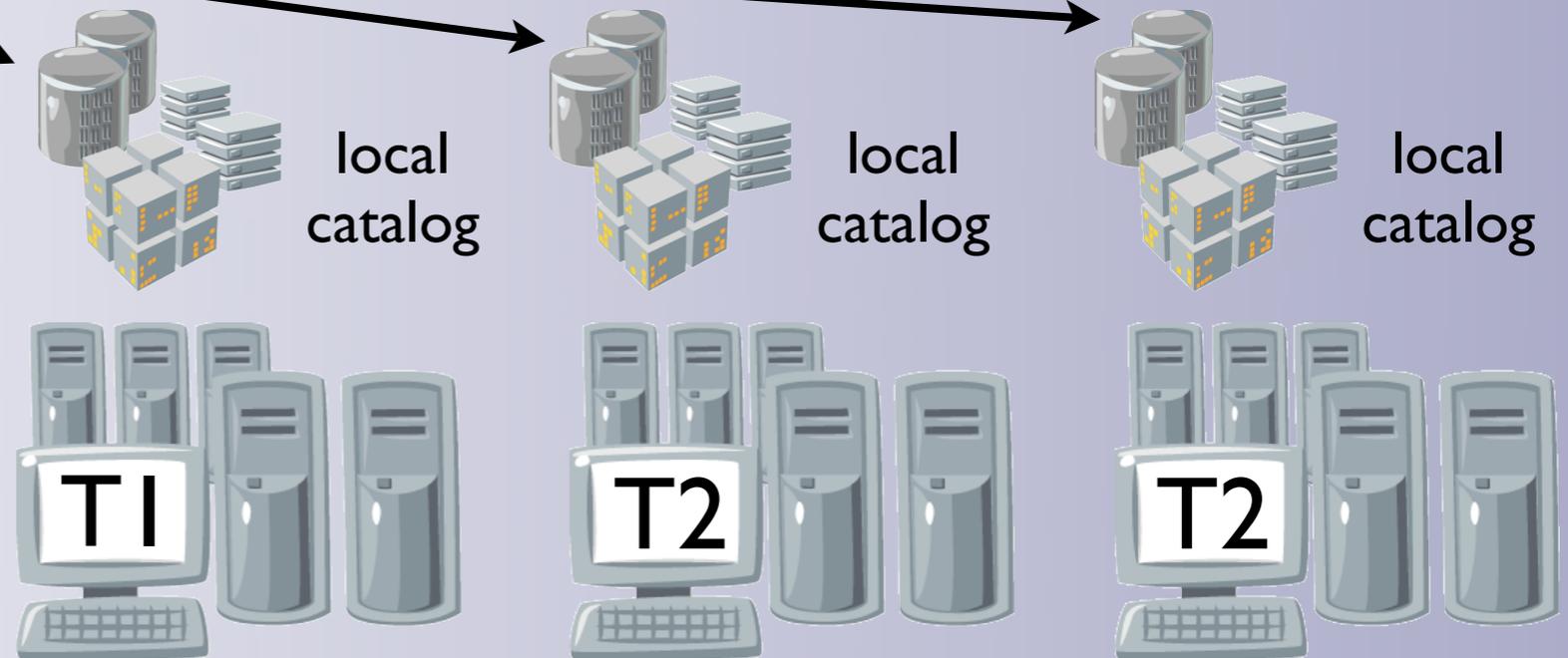
1.

resolve requested dataset
into identifier (RefDB)

2.

inquire which centers
publish requested dataset

3. contact centers and inquire
about dataset locally (PubDB)

local catalog

local catalog

local catalog

T1

T2

T2

Jobs are created locally
- on the User's submission computer
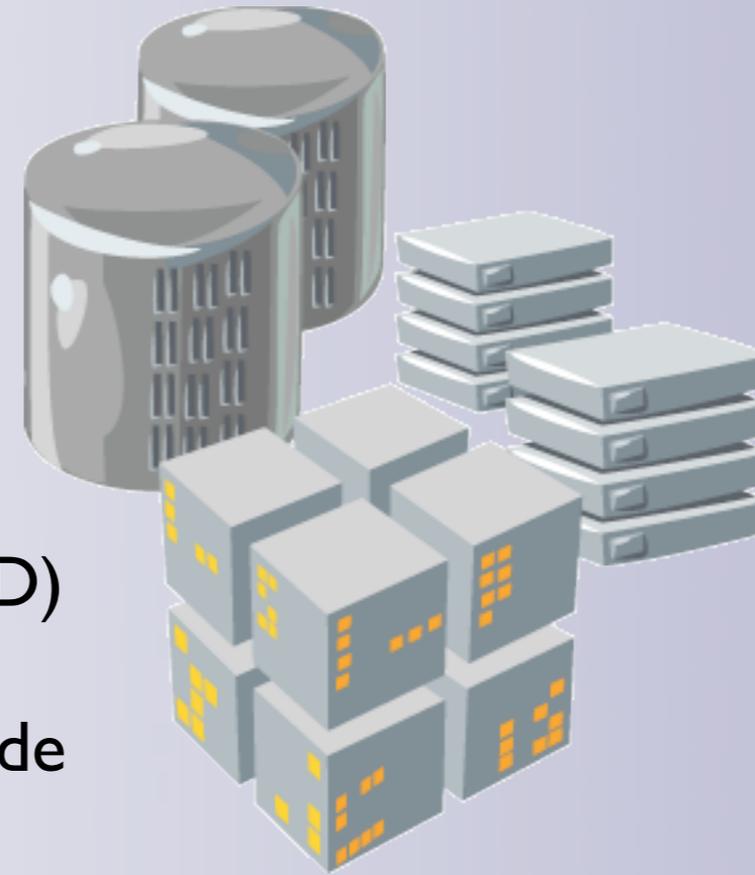- each job is able to run on all centers from the request list

All the user has to know: which datasamples can I use:
http://cmsdoc.cern.ch/cms/production/www/PubDB/GetPublishedCollectionInfoFromRefDB.mod.php

Resource Broker (RB)

- brokers job between requested centers
- provides input and output file handling

User's submitter

- providing created jobs to RB (ship user code to GRID)
- checking status of jobs
- retrieving output (retrieve user code output)

T1 T1 T2