CMS connect

Kenyi Hurtado Anampa[9], Justas Balcas[1], Brian Bockelman[8], Jadir Marra Da Silva[5], Robert Gardner Jr[6], Bo Jayatilaka[3], Farrukh Aftab Khan[4], Kevin Lannon[9], Krista Larson[3], James Letts[7], Marco Mascheroni[3], David Mason[3], Anthony Tiradani[3], Antonio Perez-Calero Yzquierdo[2]

[1] Caltech (US)
[2] CIEMAT (ES)
[3] Fermilab (US)
[4] National Centre for Physics (PK)
[5] Universidade Estadual Paulista (BR)
[6] University of Chicago (US)
[7] University of California, San Diego (US)
[8] University of Nebraska (US)
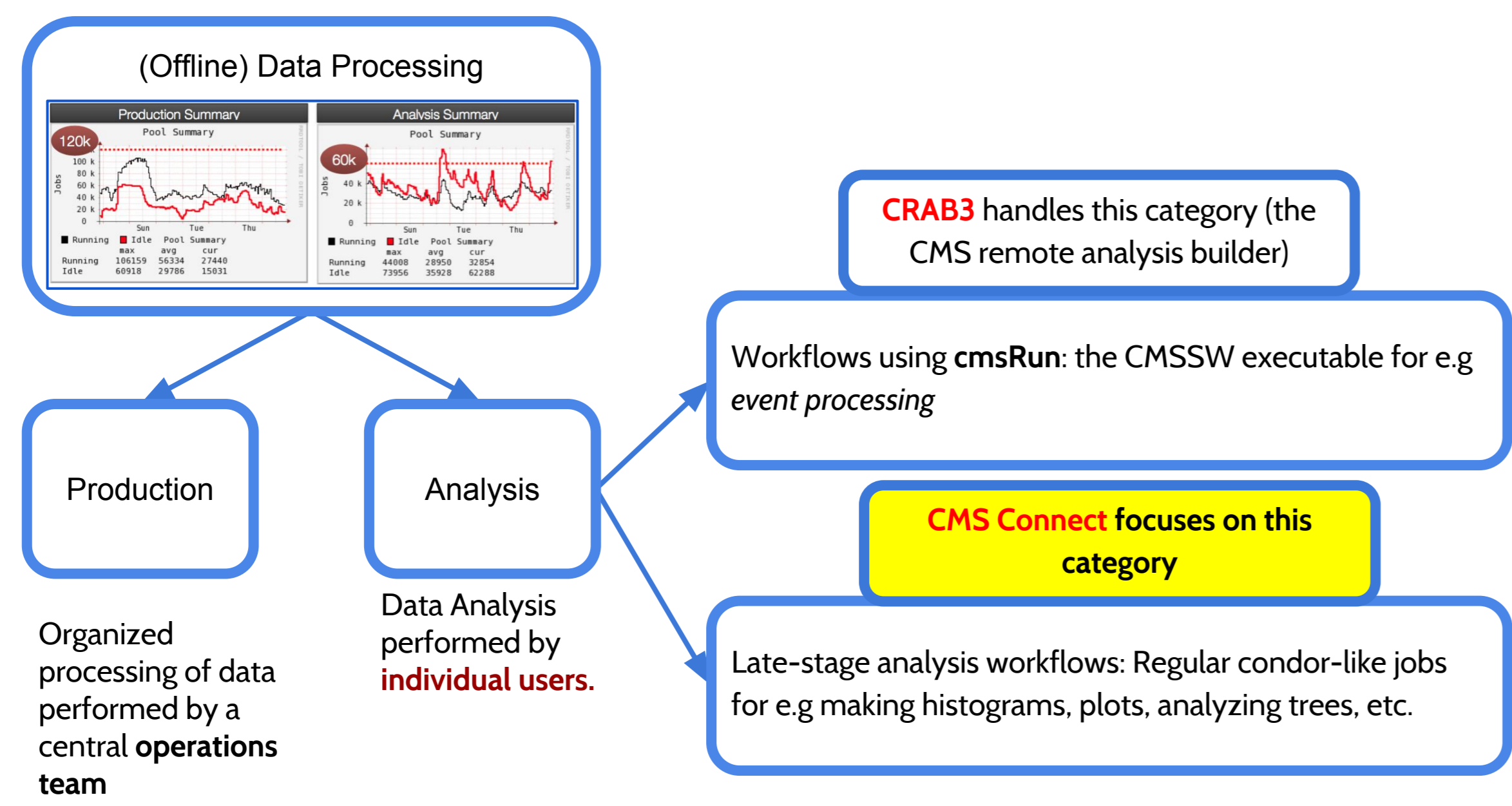[9] University of Notre Dame (US)

A service to provide Tier 3-Like capability to Institutions for **non-CRAB** workflow jobs using all resources coming from the CMS Global Pool

Motivation

The Compact Muon Solenoid (CMS) is a multipurpose experiment collecting and analyzing data from proton-proton and heavy ion collisions at the Large Hadron Collider (LHC).

From the computing point of view, this simply translates into tons of data and simulation processes that need to be handled with a large amount of computing resources. The CMS Global Pool provides over 100K cores globally distributed for the experiment but users need an easy way to access such pool. While analysis jobs based on cmsRun (the standard CMS executable) can be easily submitted via CRAB3, late-stage analysis jobs are often easier to submit via e.g regular HTCondor jobs like the ones that are submitted to the LPC CAF at Fermilab or Tier 3 centers.

CMS Connect provides a service with a Tier 3-Like interface where users can submit condor jobs to the CMS Global Pool.

CRAB3 handles this category (the CMS remote analysis builder)

Workflows using **cmsRun**: the CMSSW executable for e.g *event processing*

CMS Connect focuses on this category

Late-stage analysis workflows: Regular condor-like jobs for e.g making histograms, plots, analyzing trees, etc.

Production: Organized processing of data performed by a central **operations team**

Analysis: Data Analysis performed by **individual users.**
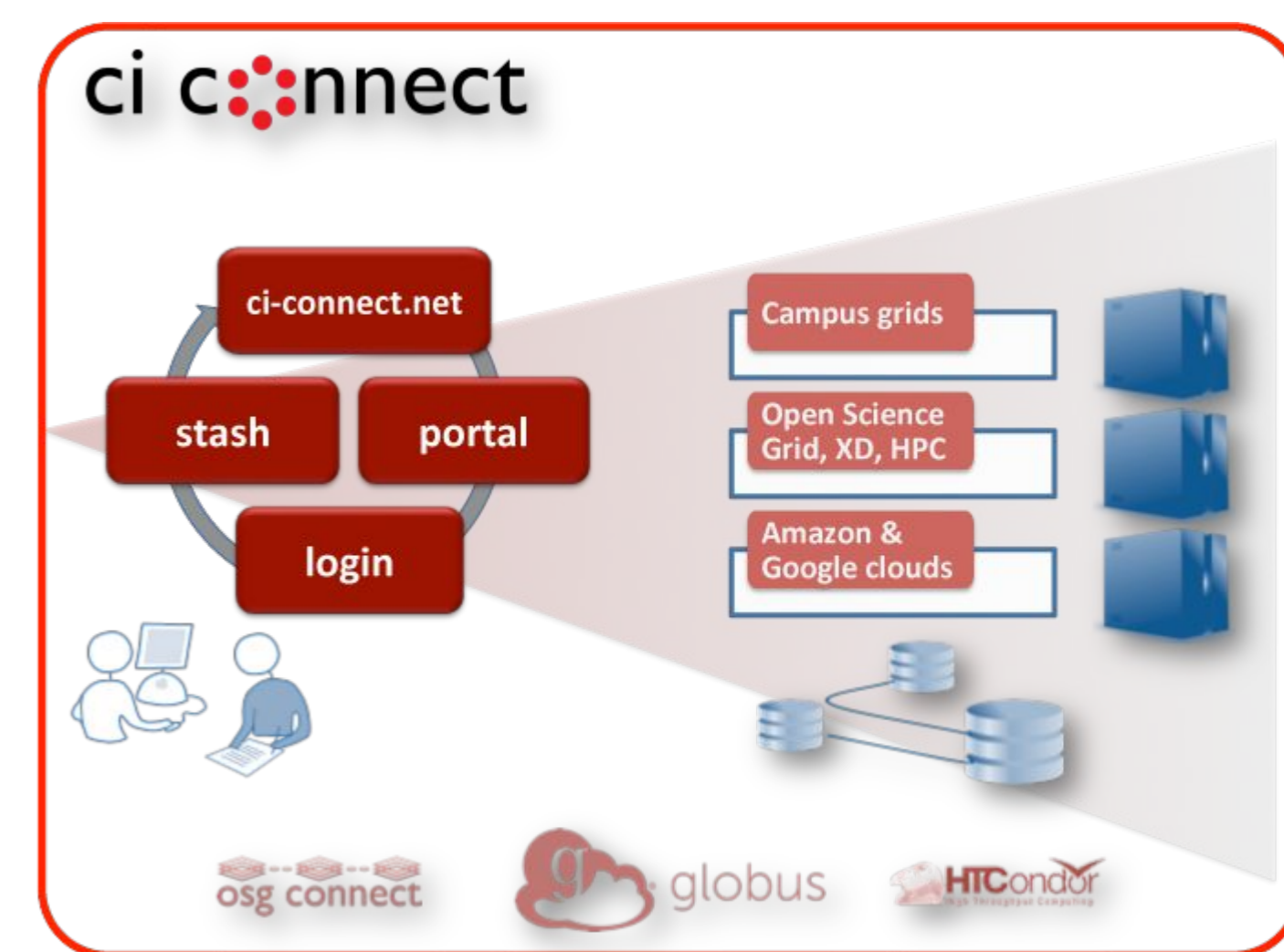
Offline Data Processing categories

Base Platform

CMS Connect is based on the CI Connect platform, developed by the OSG Area coordinator of Campus Infrastructures Rob Gardner from the University of Chicago and his team.

The platform is composed of different well supported technologies to provide a set of computing services focused on batch-like analysis and it has been adopted by different projects like OSG Connect and ATLAS Connect user.
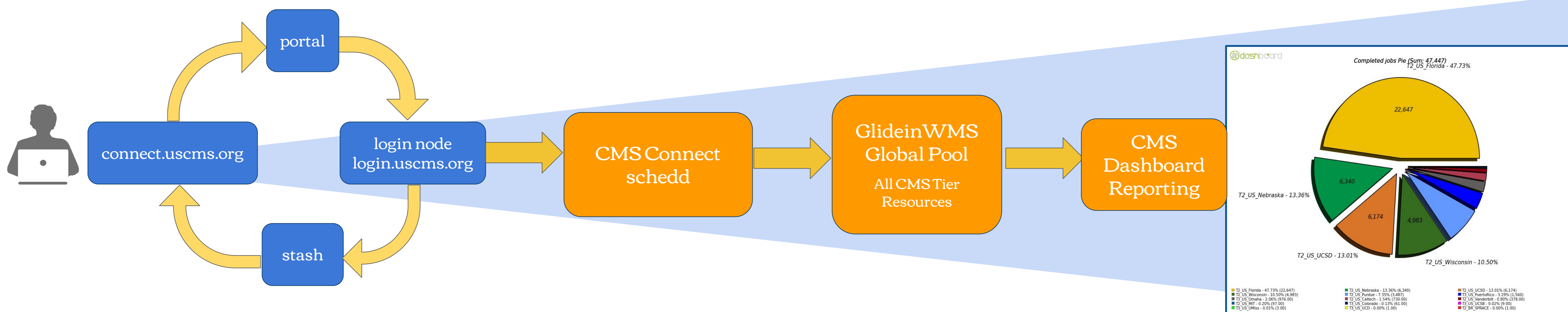
Components
- Portal Authentication and Group Management
  - Leverages the Nexus service from **Globus**
    - Provides identity management (globus id)
    - Users can be grouped in different projects.
- Login Host
  - Auto provisioning of user accounts
- Connecting CPU resources
  - HTCondor
- Distributed Data Access
  - XRootD, Globus access, http
- Distributed Software
  - cvmfs

CI-Connect schematic

CMS Connect in action

connect.uscms.org → portal → login node login.uscms.org → CMS Connect schedd → GlideinWMS Global Pool (All CMS Tier Resources) → CMS Dashboard Reporting

stash

CMS Connect components

Signing up

This procedure requires creating a Globus ID in order to handle the identity and group management per user.

connect.uscms.org
↓ http://connect.uscms.org/signup
portal.uscms.org
↓
Globus ID creation
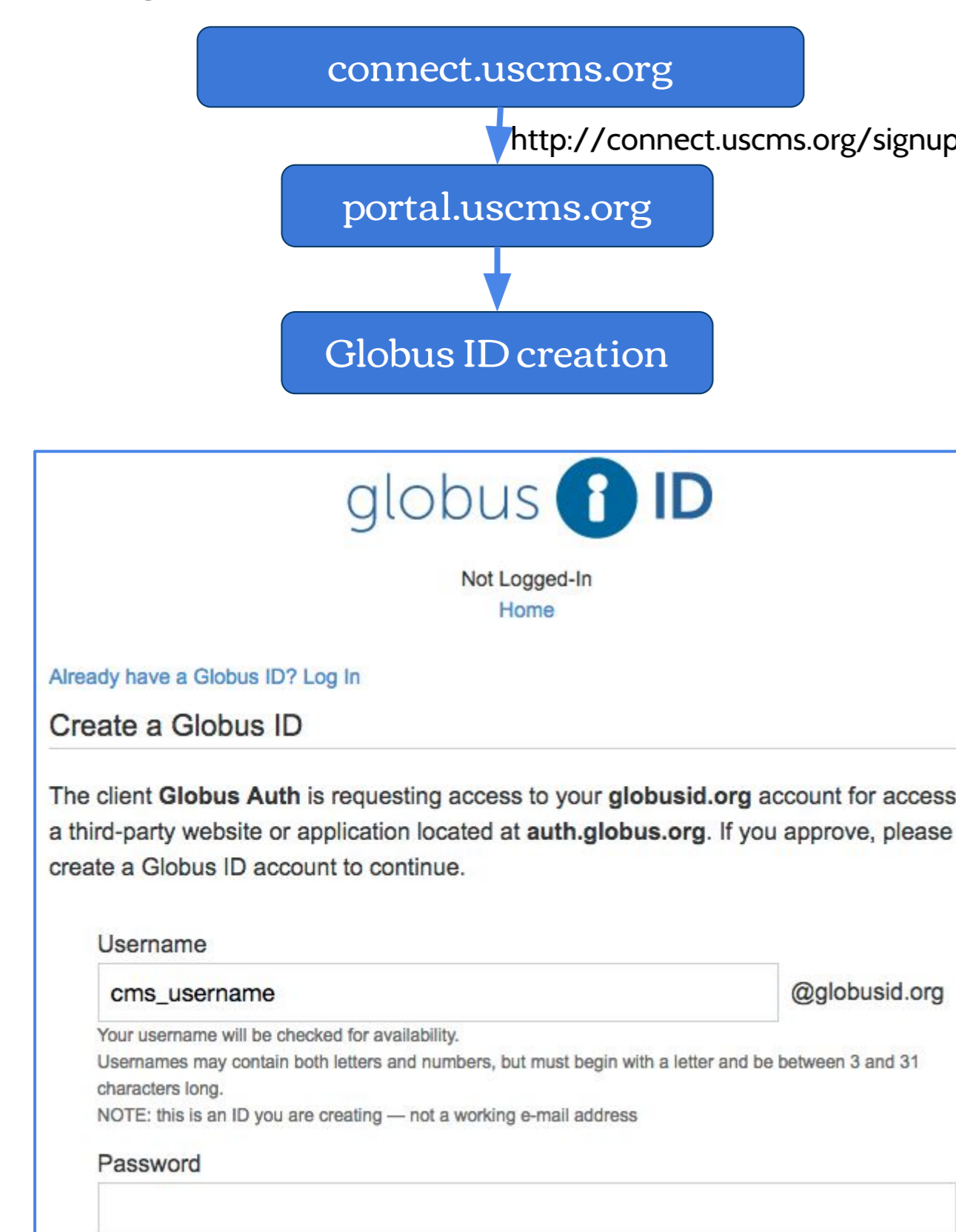
Are you a CMS member?
Check this QR-codes:

Sign up   Twiki   Docs   Tutorial

Group Management

Once signed-up, the service gives you the ability to sort users per Institution. Each Institution manager can create e.g their own analysis groups.
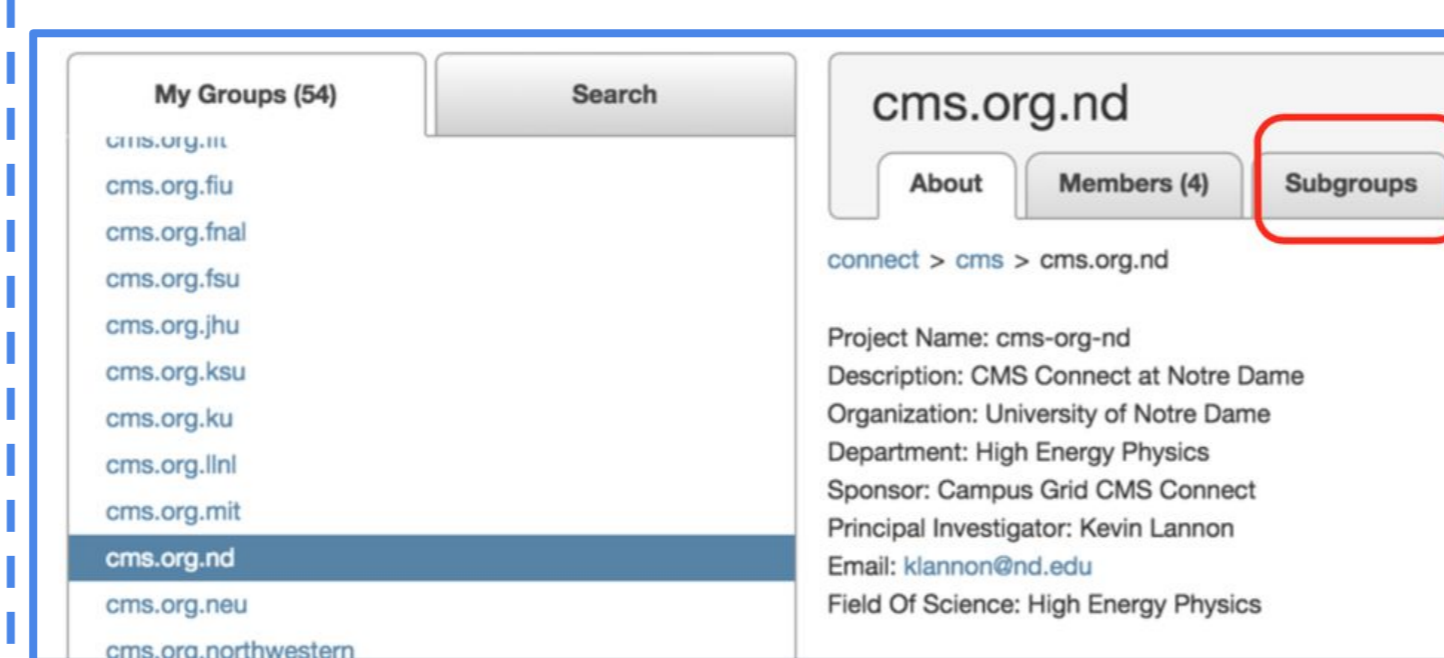
Stash Area

Users also have access to a temporary data storage provided by OSG to e.g assist with pre-stage job input files like sandboxes, etc. This area can be accessed via a fuse, XRootD or http.

Submitting jobs

The submit host (login.uscms.org) allows you to submit HTCondor-like jobs. The service actually runs a wrapper on top of HTCondor that lets the user control the Sites jobs will be submitted to, as well as verifying valid CMS proxy certificates are submitted with the job and such jobs are reported to the CMS Dashboard.

Selecting Sites for submission

In order to specify what Sites to submit to, a condor classAd needs to be added to the submit file:

```
+DESIRED_Sites="T2_US_Purdue,T2_US_UCSD"
```

Or it can be set in general by invoking the set_condor_sites submission script. For example, to use all Tier sites available in the US:
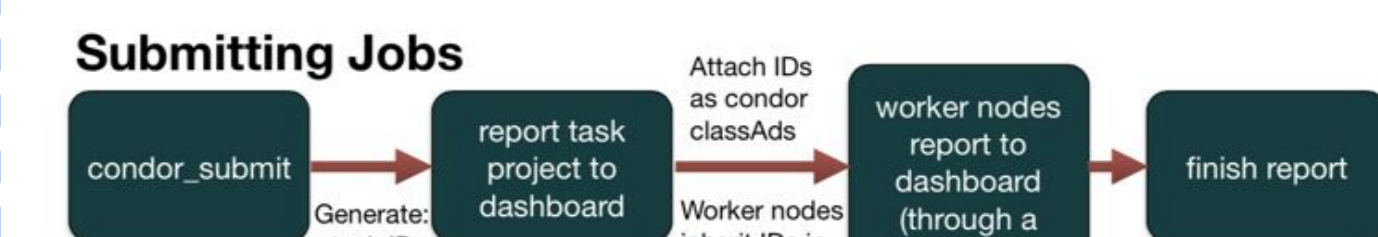
```
source /etc/ciconnect/set_condor_sites.sh T?_US_*
```

Project Accounting

Jobs are also reported by projects, where each project corresponds to your Institution name or analysis subgroup and is specified in the submission file.
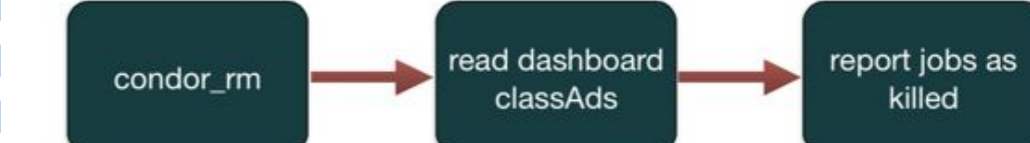
```
+ProjectName = "cms-org-nd"
```
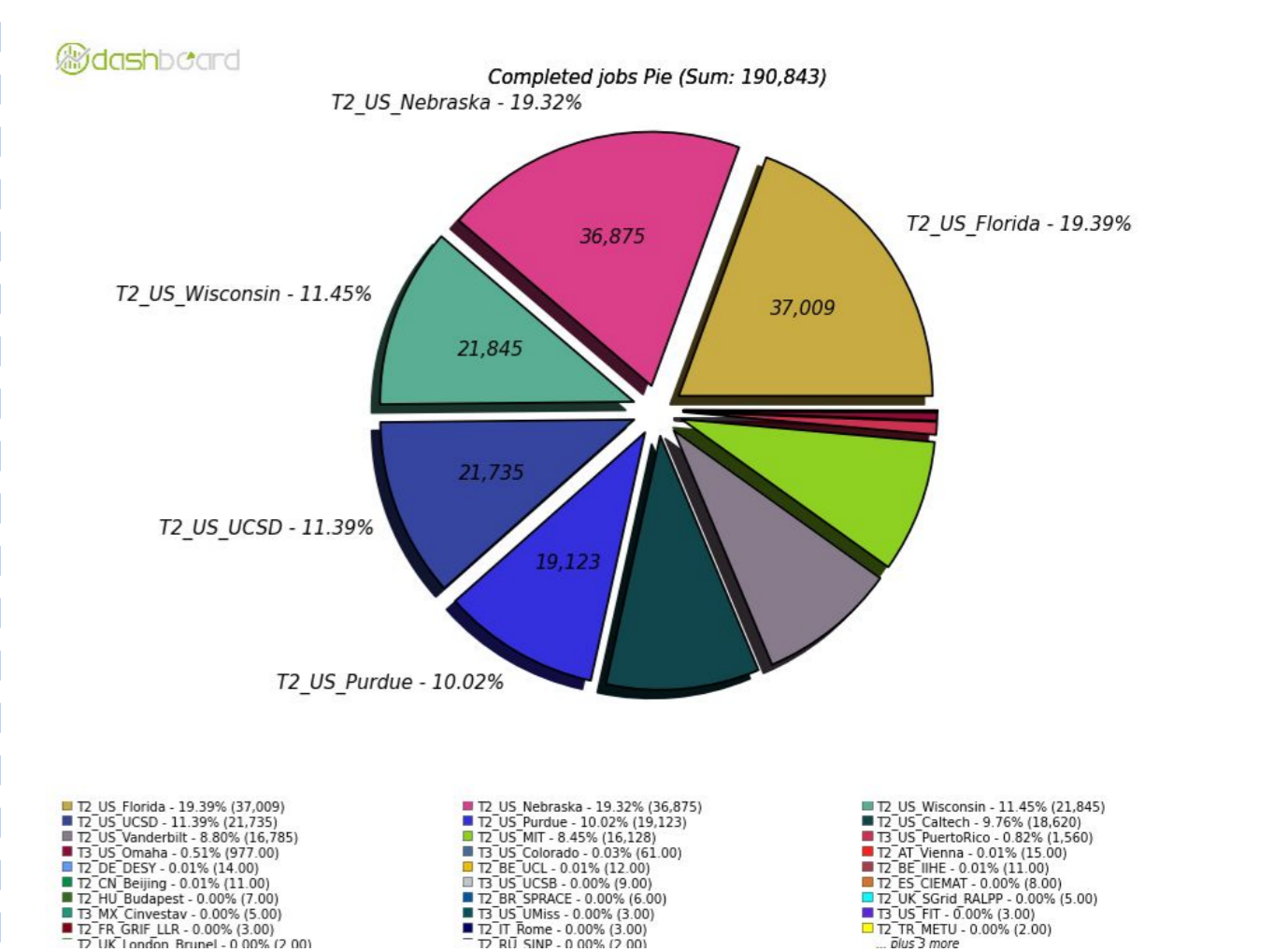
CMS Dashboard

Both WMAgent and CRAB report activity to the CMS Dashboard for Production and cmsRun Analysis jobs respectively. CMS Connect does the same for condor jobs submitted from this service.

Submitting Jobs

condor_submit → report task project to dashboard (Generate: - task ID - Monitor ID - Scheduler ID) → Attach IDs as condor classAds (Worker nodes inherit IDs in their environment) → worker nodes report to dashboard (through a wrapper) → finish report

Deleting Jobs

condor_rm → read dashboard classAds → report jobs as killed

Dashboard Reporting on CMS Connect

Example of condor jobs than ran through CMS Connect in the dashboard

UNIVERSITY OF NOTRE DAME