Integration of BSC CPU resources in CMS

A. Delgado, J. Flix, J.M. Hernández, A. Pérez-Calero, C. Acosta

28/29 April 2021
I Workshop de Computing y Software de la Red Española de LHC

GOBIERNO DE ESPAÑA
MINISTERIO DE CIENCIA E INNOVACIÓN
Ciemat Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas
cfp CIEMAT física de partículas
PIC port d'informació científica

The context

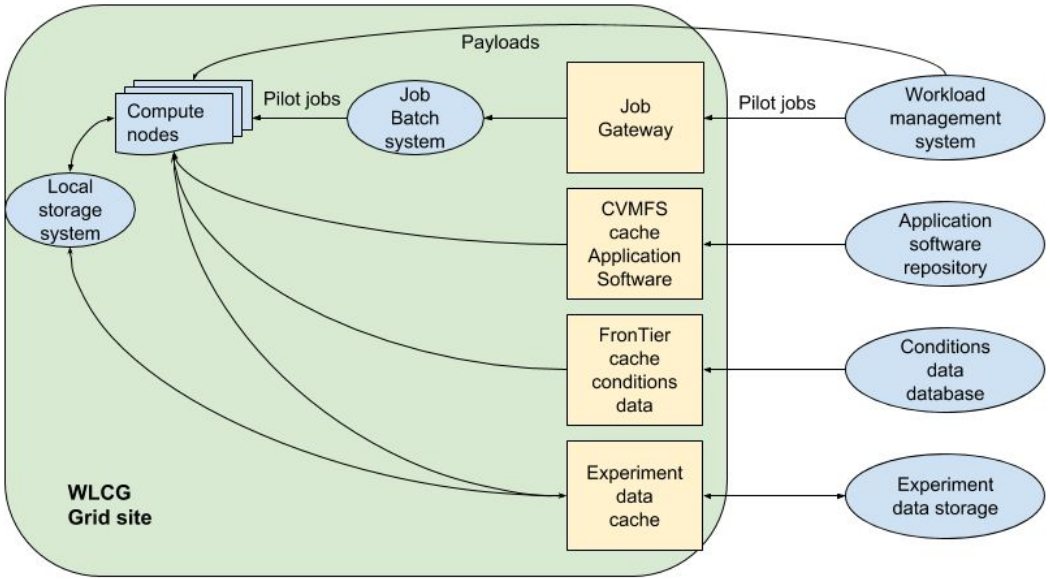
- Pressure to use HPC facilities for WLCG computing
  - Flat funding for WLCG resources, lot of public funding went to HPCs

- In 2020 BSC has designated LHC computing as a **strategic project**
  - Agreement promoted by WLCG-ES community
  - Guaranteed use of up to 7% of MareNostrum4 (~95M coreHours/year max.)
  - WLCG-ES request: all CPU time required for LHC simulation in Spain
    - ~55 Mhours in 2021
      (ATLAS 50%, CMS 30%, LHCb 20%)
  - Submission of proposals for time allocation every 4 months
  - MareNostrum5 ~ 17xMN4, available in 2022

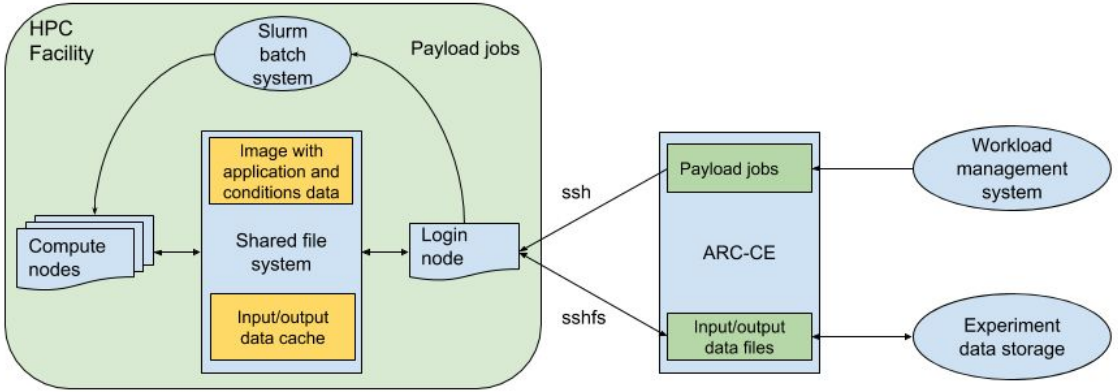
The problem

Using MN4 resources is very challenging for CMS

- **No Internet connectivity in compute nodes**!
- A show stopper for CMS
  - Pilot with late binding model execution of payloads
  - Pilots and payloads need access to external services
- Experiment edge services not allowed inside BSC
- Only communication through the login node (ssh, sshfs)



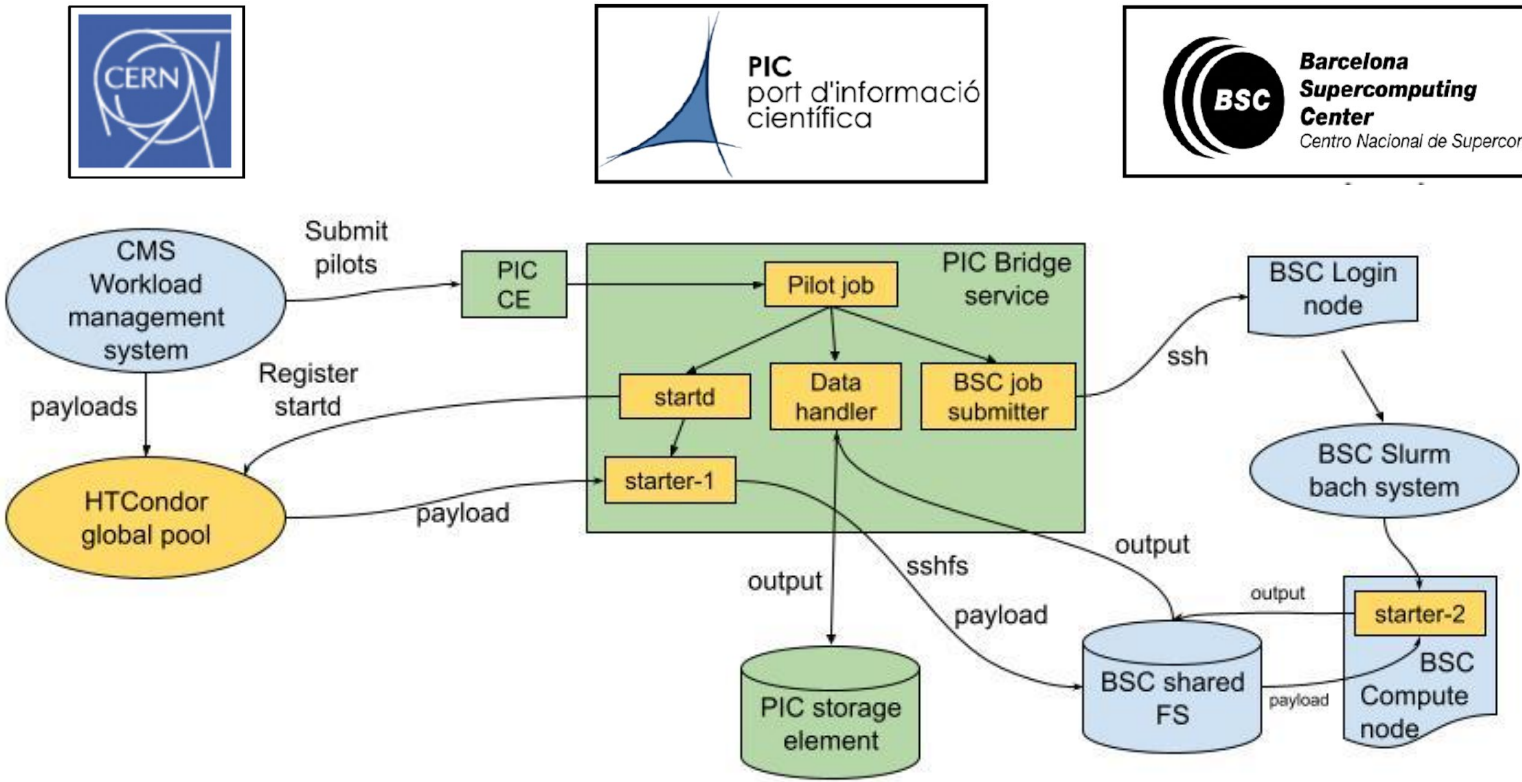
The solution



- Follow ATLAS model to push self-contained jobs and collect results from a shared file system through a BSC login node
  - Application software packaged in a container image (singularity)
  - However, early job binding / ARC-CE not in use in CMS

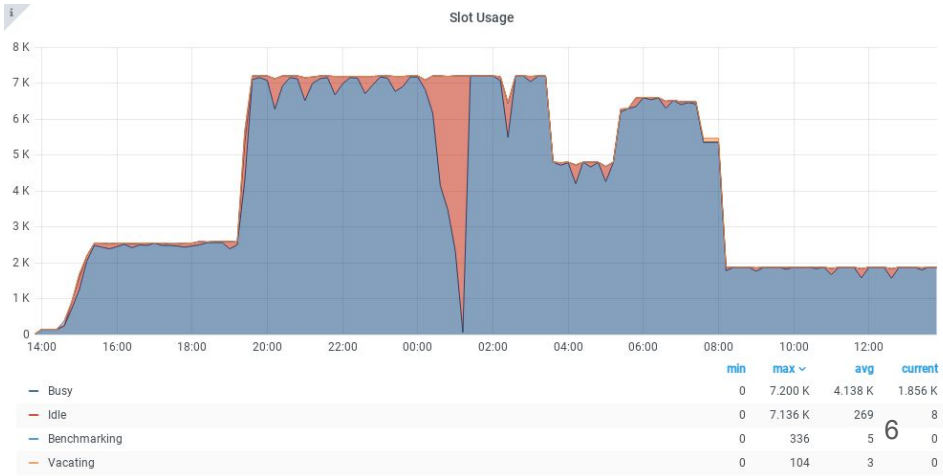
- **Major development in CMS**
  - Modify CMS resource provisioning, job scheduling and execution framework (HTCondor) to use a shared file system as communication layer
  - Install bridge service at PIC to connect CMS WMS and BSC
  - Encapsulate experimental software and conditions data in a container image
- Concept presented at CHEP19

HTCondor split-starter mode



Proof of concept

- Integration done using the Nov 2020 - Feb 2021 allocation (1M CPUHrs)
- HTCondor workflow successfully tested at scale
- CMS simulation workflow tested successfully
  - Custom-built singularity image
  - Custom-generated sqlite conditions data file
  - Manually pre-placed input dataset and manual stageout of output files
  - No CMS WM layer involved yet (payloads manually submitted)
- Very successful! CMS can run simulation workflows at scale at BSC!



Towards fully automation and integration in production

- New allocation granted: Mar 2021 - Jun 2021 of **6M CPUHrs**
- Delivery of **application software** (CMSSW)
  - Initial solution, based on Singularity images, lacks of automation in CMS
  - cvmfs/cms.cern.ch being replicated to BSC and synced via cvmfs_preload
    - 12.6 TB - 183M files (~2 weeks @ 10 MB/s)
- Access to **conditions data**
  - Access via local files implemented in CMSSW
  - CMS plans to centrally produce and distributed (possibly via cvmfs) these files
- Connection to the production **CMS WMS**
  - Get central pilots and payloads matching BSC features
  - Optimize scalability and resource usage efficiency of bridge service
- Handling of **input and output** datasets
  - Copy files from/to PIC storage as pre/pro job steps

Summary

- Big effort invested in integrating BSC CPU resources in CMS
  - HTCondor team: development of split-starter mode
  - CIEMAT/PIC team: interface with CMS and BSC, bridge service deployment, configuration, testing, handling of input/output datasets, etc
  - CMS: Handling of conditions data via files
- Proof of concept tested at scale
- Last developments ongoing towards automation and integration in production
  - Accepted contribution to CHEP21
- Consume current allocation with CMS production workflows