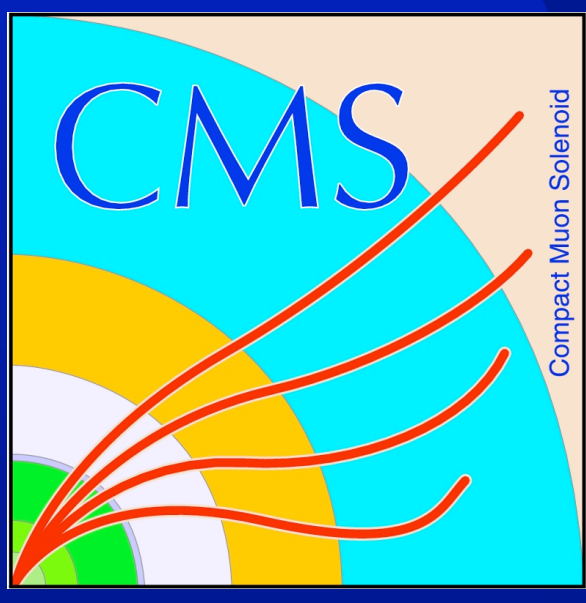




Evolution of the pilot infrastructure of CMS: towards a single glideinWMS pool

Alison McCrea, Krista Majewski, Stefano Belforte, James Letts, Oliver Gutsche
On behalf of the CMS collaboration

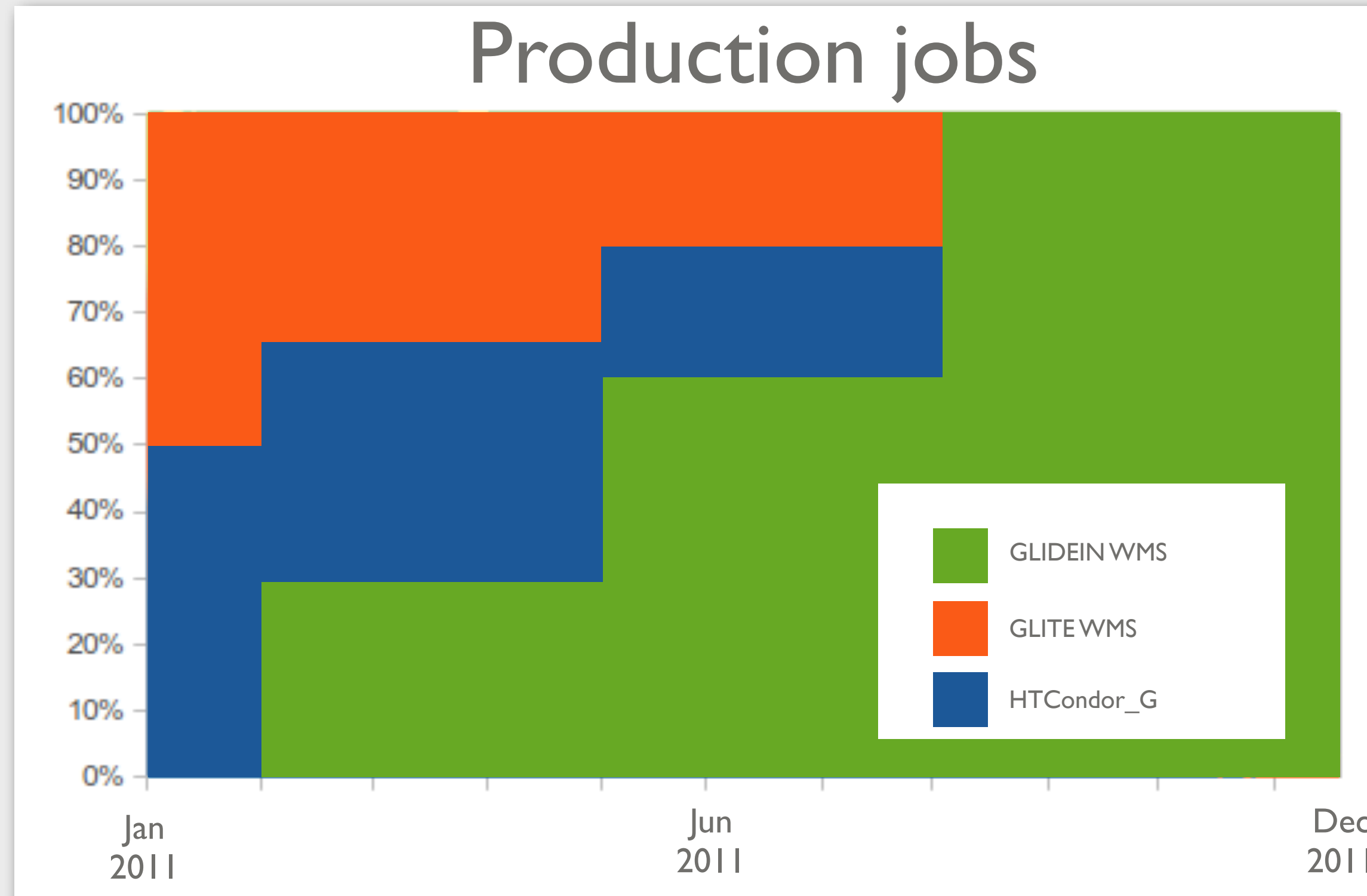


Timeline of submission infrastructures usage - Transition to glideinWMS

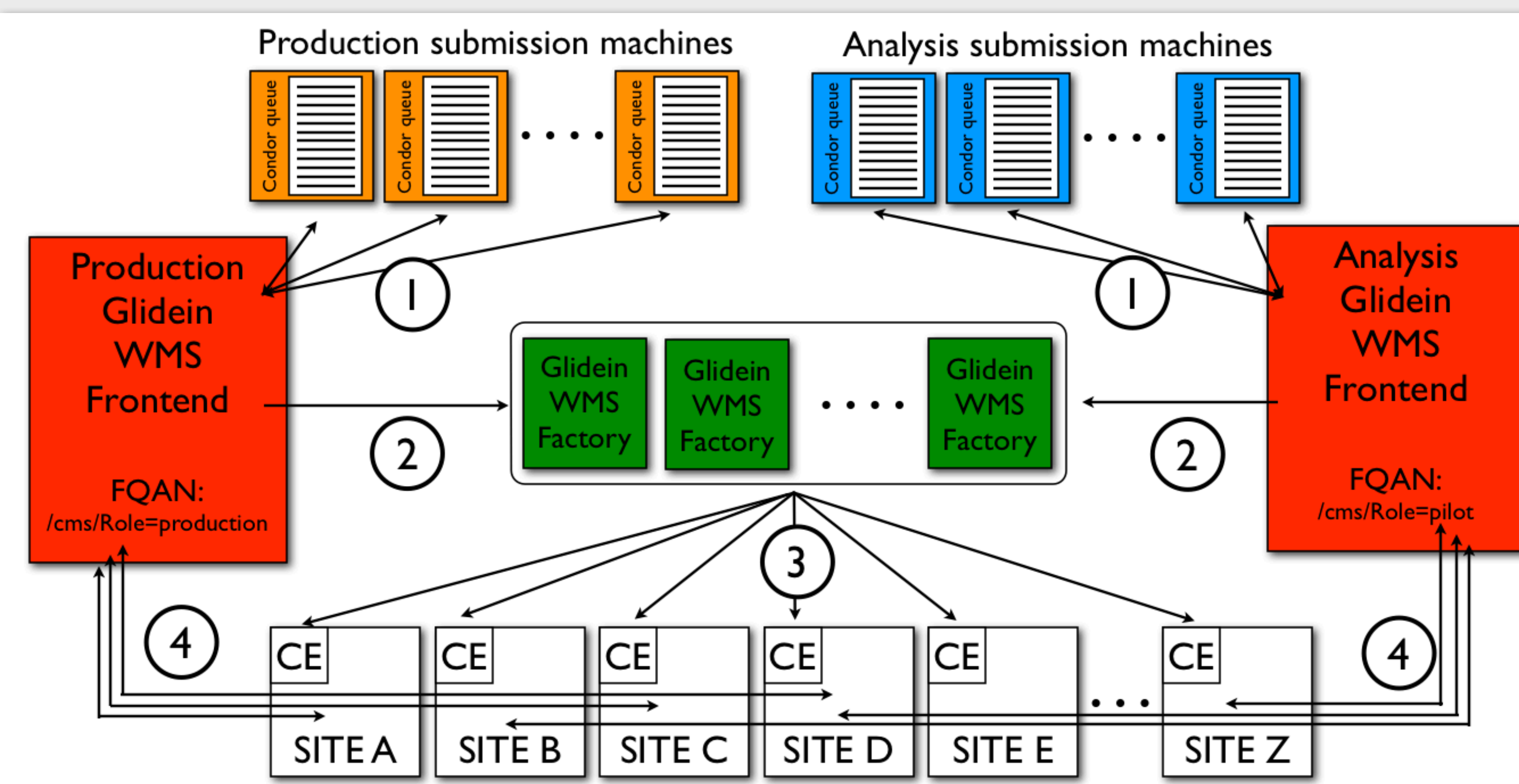
- CMS started LHC run I (2010-2012) using mainly two "direct" submission infrastructures:
 - gLite WMS
 - HTCondor_G

- Central **production** activities transitioned to the pilot-based **glideinWMS** submission system in **2011**

- Analysis** jobs started the transition to **glideinWMS** in October 2012 and recently reached **> 95% adoption**



Current glideinWMS setup: separate systems for production and analysis

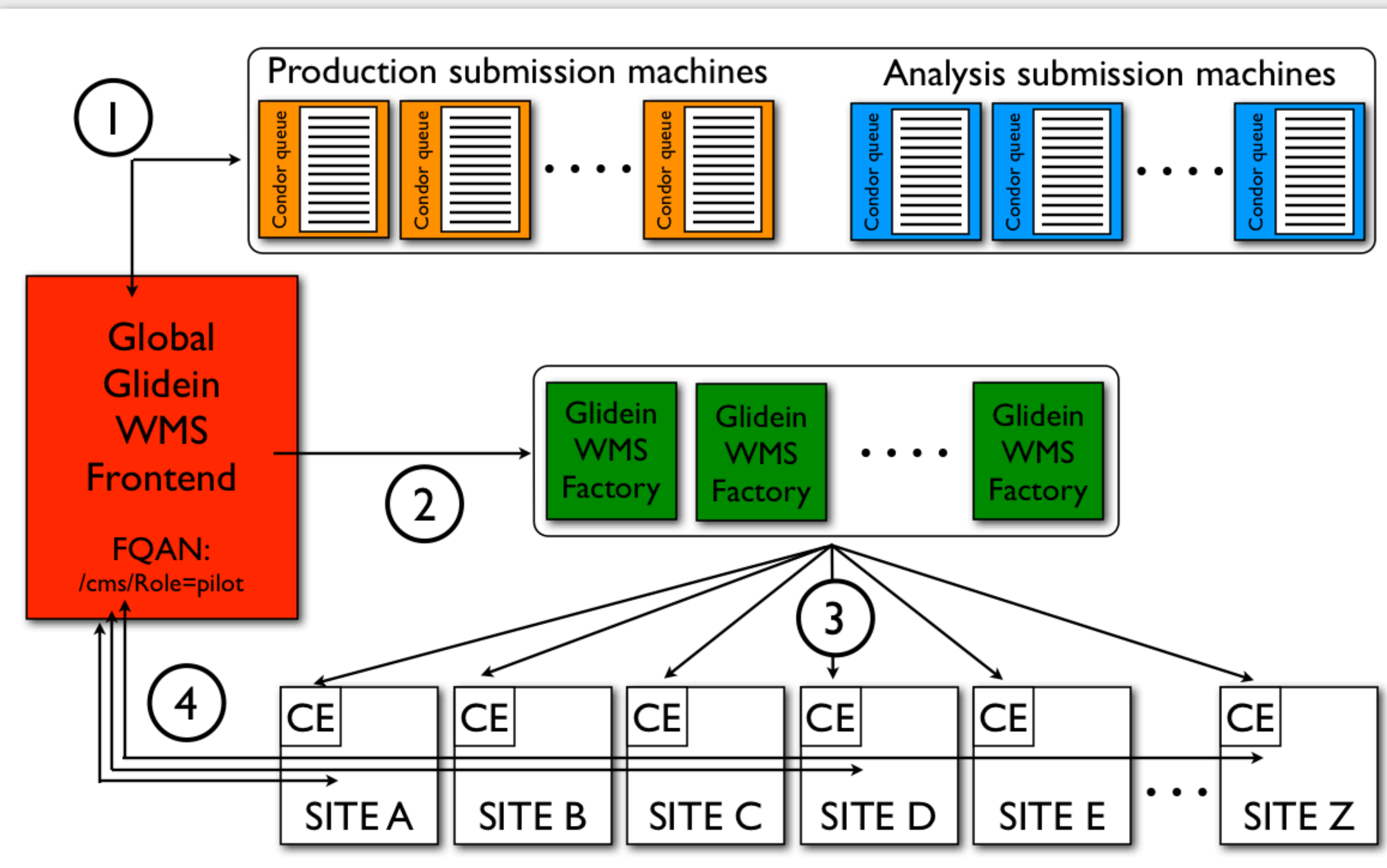


Two glideinWMS systems to initially support mixed direct/pilot submissions

- Each frontend monitors separate set of submission machines
 - Both frontends request pilots from all factories
- Factories submit pilots using the DNs and VOMS role from the respective frontend
- 1st prioritization level: CEs at the sites**
- Prioritization between production and analysis pilots
- Support for mixed direct/pilot submissions:
- multiple DNs in analysis frontend**
- Pilots can run analysis jobs from many different users
 - Because of CE level prioritization: analysis frontend uses several DNs round-robin to even out shares between direct/pilot submission
- Both frontends match jobs to pilots separately, jobs mainly run within the context of the pilot proxy
- 2nd prioritization level: frontend pool**
- Within one frontend, rules can be applied to prioritize some jobs/workflows over others

Disadvantages causing inefficiencies: **two separate prioritization levels and complicated DN setup for analysis**

Global glideinWMS setup: one system with global prioritization



One glideinWMS system for both analysis and production activities

- Global frontend monitors all production and analysis submission machines
 - Global frontend requests pilots from all factories
 - Factories submit pilots exclusively with pilot VOMS role
- No CE level prioritization**
- Global frontend matches jobs to pilots
- Global prioritization level: frontend pool**
- Prioritize production vs. analysis usage at a site (e.g. 50% production, 50% analysis)
 - Prioritize analysis jobs on user and analysis project level (fair share prioritization and possibility to increase priorities for selected users/projects)
 - Allow for dynamic re-prioritization
 - Allow to steer only certain jobs from certain users, regional groups to resources reserved for them (example: regional resources of a country for their users)
- Jobs use glexec to switch context inside pilot**
- Use either analysis permissions or production credentials
 - Proxy handling done transparently by HTCondor underneath glideinWMS

Acknowledgements

We would like to thank the United States Department of Energy (DOE), the United States National Science Foundation (NSF), the Italian Istituto Nazionale di Fisica Nucleare and the CMS computing project for their support and help.

Related presentations/posters

- Parallel Talk: "CMS Computing Operations During Run I"
 - Track: "Distributed Processing and Data Handling B: Experiment Data Processing, Data Handling and Computing Models"
- Parallel Talk: "Using ssh as portal - The CMS CRAB over glideinWMS experience"
 - Track: "Distributed Processing and Data Handling B: Experiment Data Processing, Data Handling and Computing Models"
- Poster: "CMS experience of running glideinWMS in High Availability mode"