

# CMS Remote Analysis Builder (CRAB) with the ARC grid middleware

Kalle Happonen<sup>\*</sup>, [Erik Edelmann](#)<sup>†</sup>, Tomas Lindén<sup>\*</sup>, Jesper Koivumäki<sup>\*</sup>, Joni Välimäki<sup>\*</sup>

<sup>†</sup>[erik.edelmann@csc.fi](mailto:erik.edelmann@csc.fi)

<sup>\*</sup>Helsinki Institute of Physics


<sup>†</sup>Nordic Data Grid Facility / CSC – IT center for Science

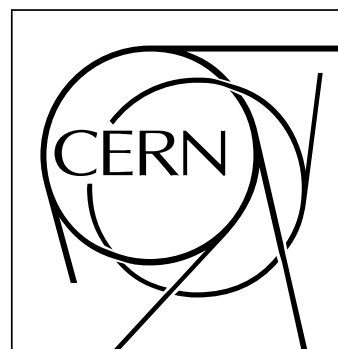
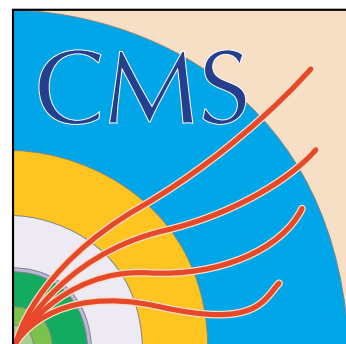


# What is CMS?



# What is CMS?

- CMS = Compact Muon Solenoid.
- One of the LHC experiments at CERN.
- Now with real data! 
  - Running at 3.5 TeV + 3.5 TeV since about two weeks



# CMS computing

- Petabytes of data per year.
- Participating sites organized into a tier based structure.
  - Tier 0: At CERN. Raw data storage with initial calibration and and reconstruction with large storage capacity.
  - Tier 1: At CERN and in international Regional Centers. Further calibration and reconstruction passes with large fraction of simulation and analysis. Large storage capacity and associated support.
  - Tier 2: At national Regional Centers. Significant simulation and analysis.
  - Tier 3: Institute level facilities for analysis and simulation.

# CMS computing: Software

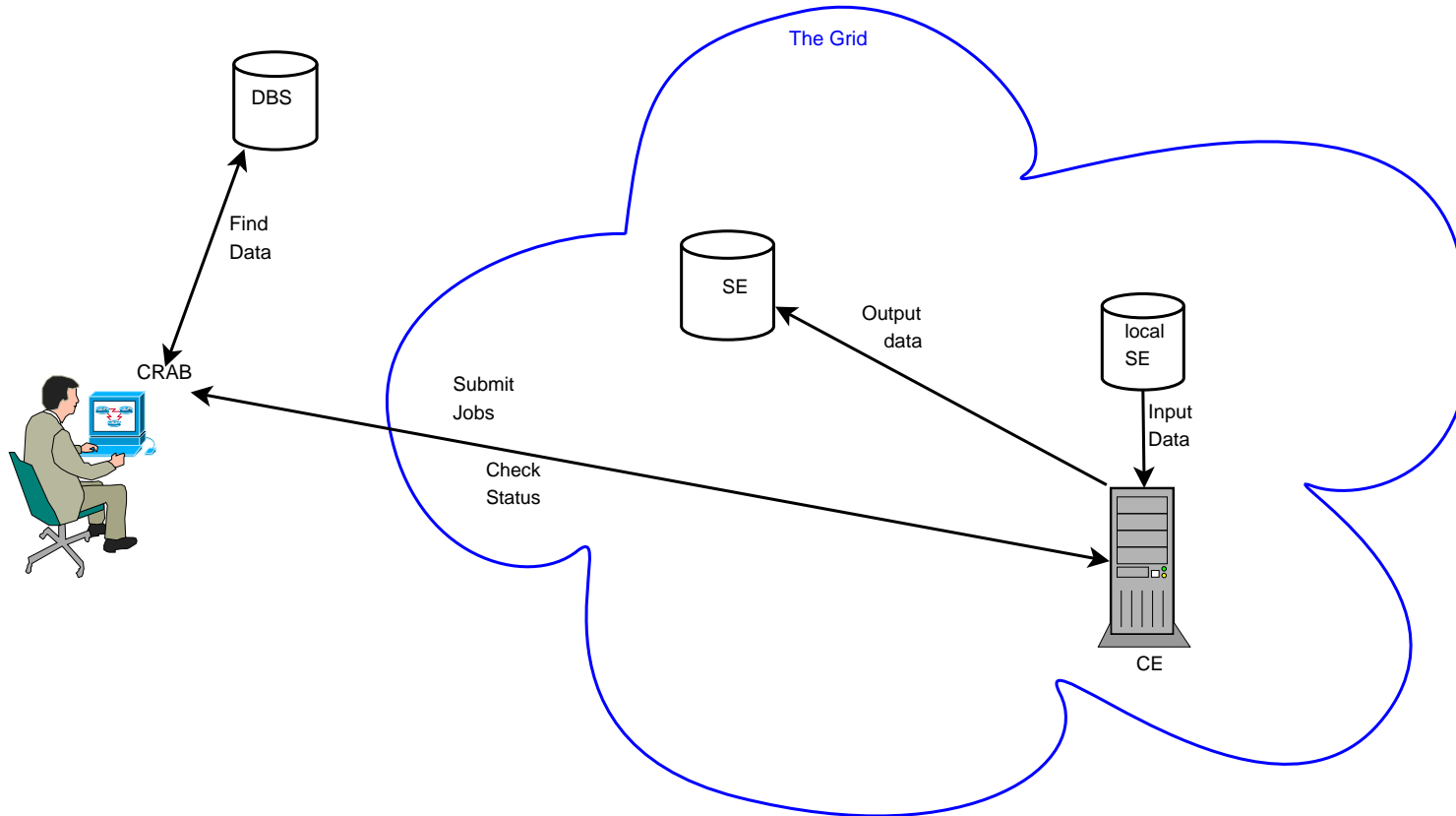
- PhEDEx (Physics Experiment Data Export), for moving data around
- DBS (Dataset Bookkeeping system), for keeping track of where the data is
- CRAB (Cms Remote Analysis Builder), for submitting analysis jobs.
- Others ...

# CRAB

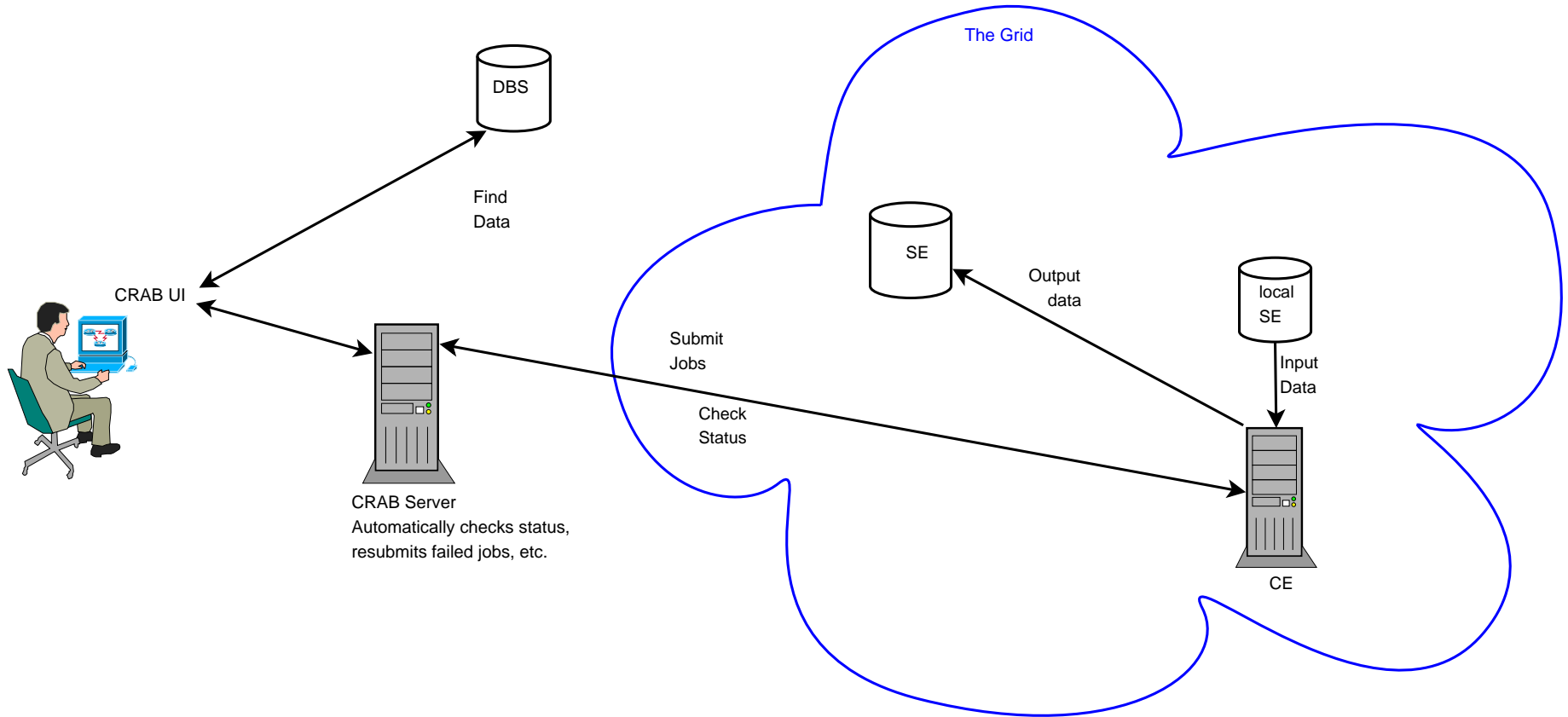
- Provide a simple, grid middleware independent user interface
  - Provides “schedulers” for different backends.
- Makes sure the jobs go to the site where the data is.
- Can be used two ways: Standalone mode, and Server mode.



# CRAB, Standalone mode

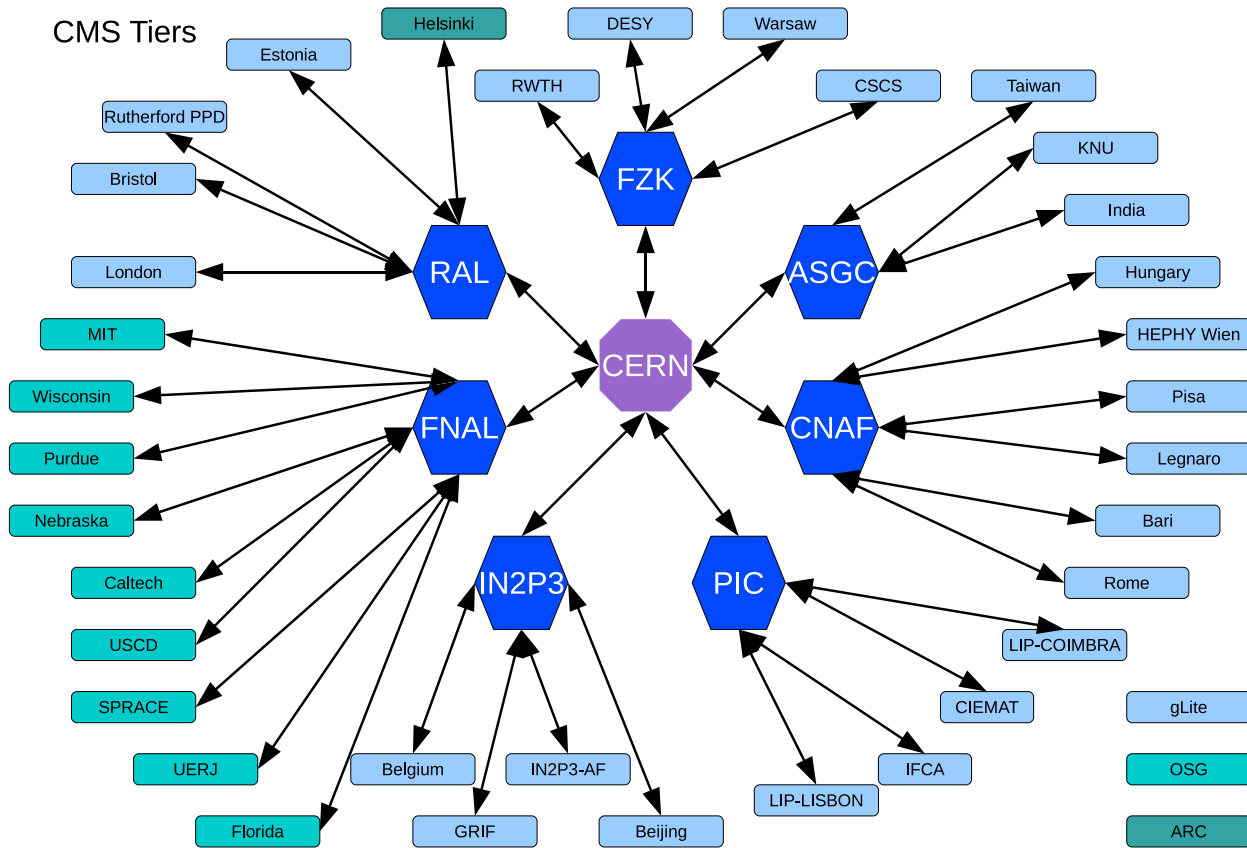


# CRAB, Server mode

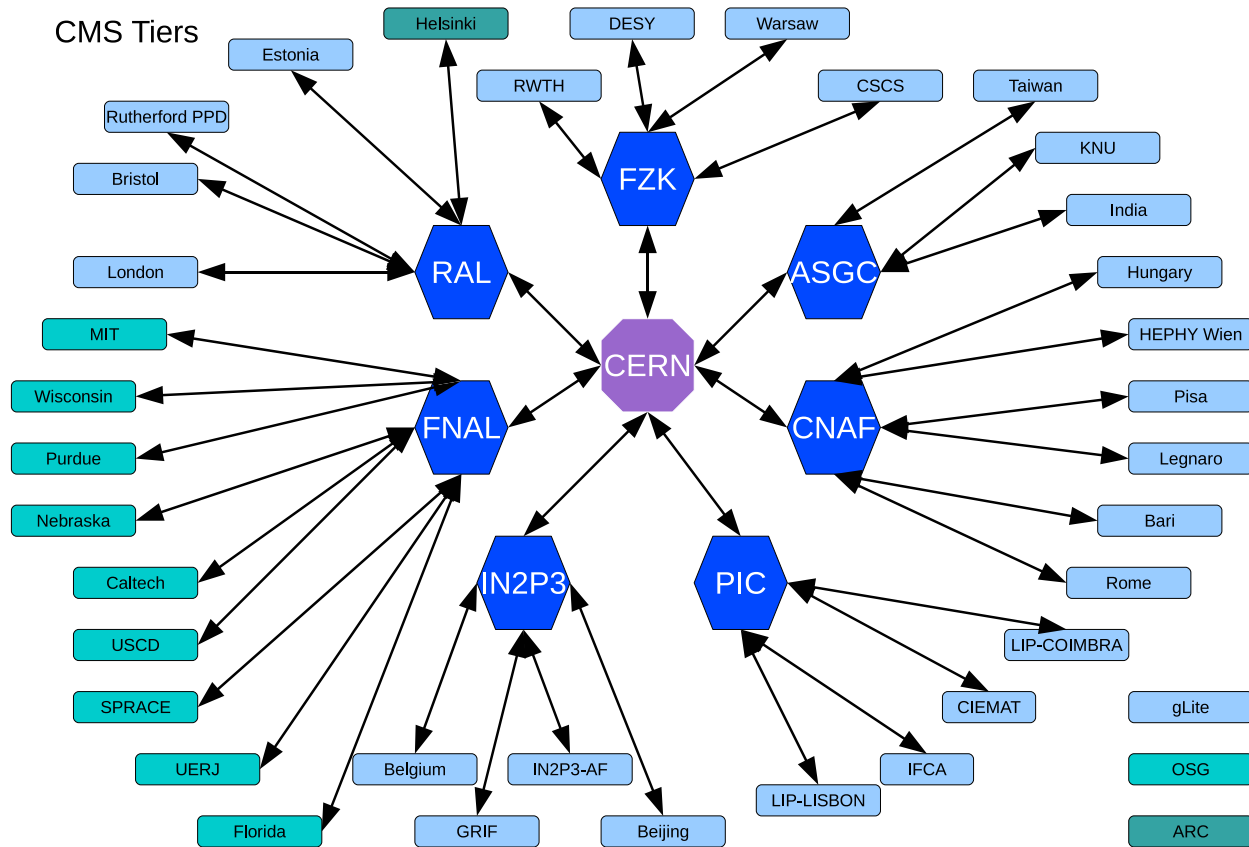




# CMS computing: Tier structure



# CMS computing: Tier structure



ARC used at one site: T2-FI-HIP in Finland



# NorduGrid ARC

- Advanced Resource Connector (ARC) developed and maintained by NorduGrid collaboration
- ARC is designed to be scalable, non intrusive and portable.
- Used in production environment since 2002; latest stable release 0.8.1
- In use by various other projects:
  - Nordic Data Grid Facility (NDGF) for Atlas and Alice Tier 1 and other VOs
  - A few national grid infrastructures, including M-Grid/Finnish Grid Initiative (FGI)
  - others

# Why ARC for CMS?

Adapting CMS software to work with ARC requires some work – why bother for only one Tier-2 site?

# Why ARC for CMS?

Adapting CMS software to work with ARC requires some work – why bother for only one Tier-2 site?

- Easy to set up and operate on a wide range of Linux distributions.

# Why ARC for CMS?

Adapting CMS software to work with ARC requires some work – why bother for only one Tier-2 site?

- Easy to set up and operate on a wide range of Linux distributions.
- ARC is already being used for other projects in Finland, e.g. FGI
  - Some of the resources are shared with other activities, that use ARC
  - We have lots of ARC expertize
  - Initial development effort, but less effort on administration/operation.

# Why ARC for CMS?

Adapting CMS software to work with ARC requires some work – why bother for only one Tier-2 site?

- Easy to set up and operate on a wide range of Linux distributions.
- ARC is already being used for other projects in Finland, e.g. FGI
  - Some of the resources are shared with other activities, that use ARC
  - We have lots of ARC expertize
  - Initial development effort, but less effort on administration/operation.
- NDGF provides development help



# CRAB with ARC

CRAB can submit to ARC in two ways:





# CRAB with ARC

CRAB can submit to ARC in two ways:

- gLite scheduler + gLite/ARC interoperability

# CRAB with ARC

CRAB can submit to ARC in two ways:

- gLite scheduler + gLite/ARC interoperability
- Natively, through the ARC scheduler in CRAB.

# ARC scheduler in CRAB

- ARC scheduler consists roughly 1000 lines of python code.
  - ... most of which in ProdCommon.

# ARC scheduler in CRAB

- ARC scheduler consists roughly 1000 lines of python code.
  - ... most of which in ProdCommon.
- ... + a few bits of runtime environment. (Because in ARC there's no workernode package)

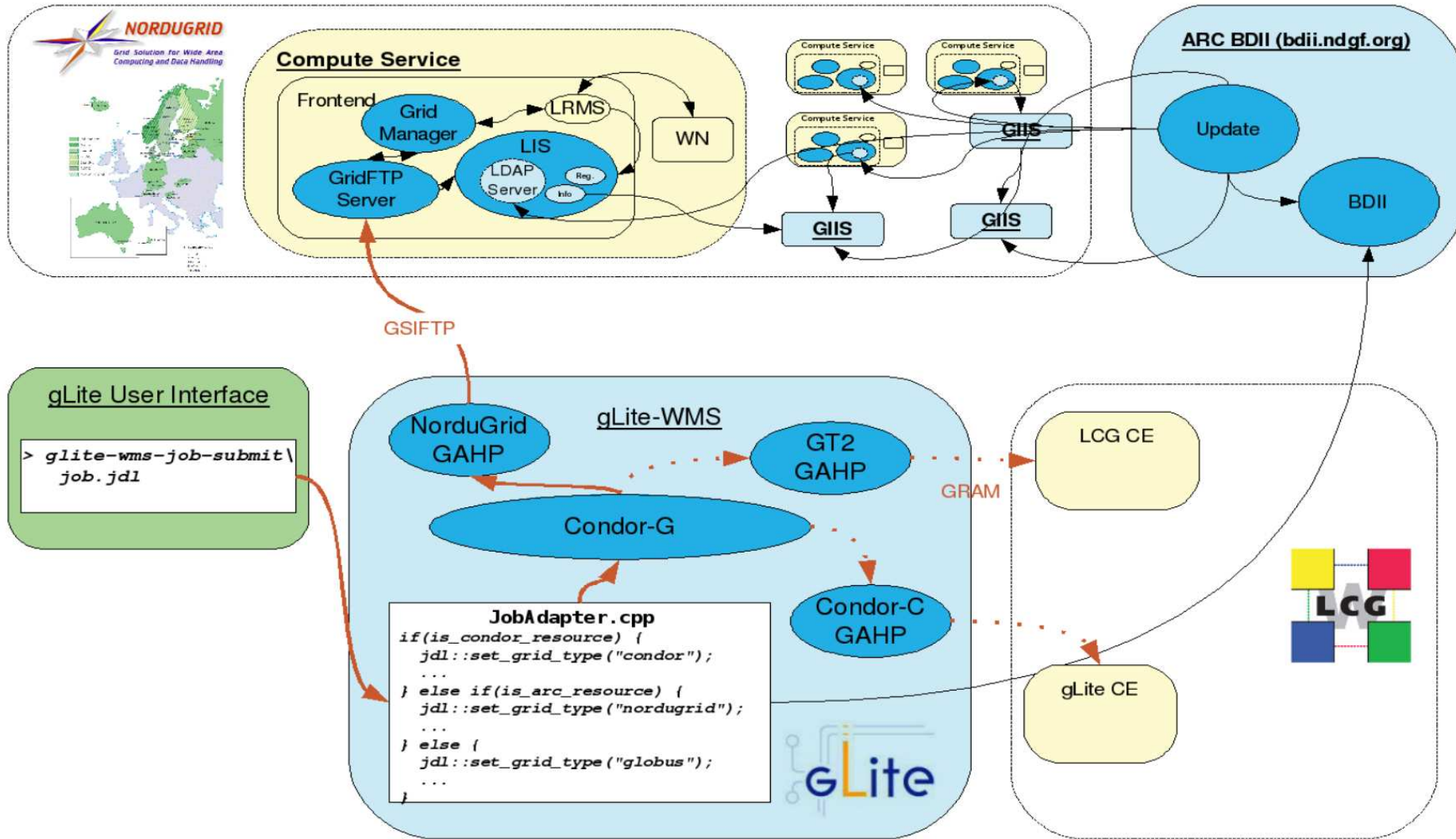
# ARC scheduler in CRAB

- ARC scheduler consists roughly 1000 lines of python code.
  - ... most of which in ProdCommon.
- ... + a few bits of runtime environment. (Because in ARC there's no workernode package)
- Standalone mode:
  - Works, but ...
  - The user is restricted to ARC sites
  - Requires ARC client on user's computer

# ARC scheduler in CRAB

- ARC scheduler consists roughly 1000 lines of python code.
  - ... most of which in ProdCommon.
- ... + a few bits of runtime environment. (Because in ARC there's no workernode package)
- Standalone mode:
  - Works, but ...
  - The user is restricted to ARC sites
  - Requires ARC client on user's computer
- CRAB server with ARC support is being setup.
  - Should fix some of the above mentioned problems

# gLite/ARC interoperability



# gLite/ARC interoperability

- Application independent



# gLite/ARC interoperability

- Application independent
- Drawbacks:
  - ARC site needs the gLite worker node package installed.
  - Various grid operations (submitting, checking status, etc.), is slower than when accessing ARC directly.
  - Requires gLite-UI on the user's computer
  - Increased complexity: twice as many things that can go wrong

# gLite/ARC interoperability

- Application independent
- Drawbacks:
  - ARC site needs the gLite worker node package installed.
  - Various grid operations (submitting, checking status, etc.), is slower than when accessing ARC directly.
  - Requires gLite-UI on the user's computer
  - Increased complexity: twice as many things that can go wrong
- Majority of CRAB jobs in Finland comes this way

# Setting up an ARC site for CRAB

- Install CMSSW + runtime env. scripts.
  - Different scripts for gLite/ARC interop. and native ARC jobs (could be merged?).

# Setting up an ARC site for CRAB

- Install CMSSW + runtime env. scripts.
  - Different scripts for gLite/ARC interop. and native ARC jobs (could be merged?).
- Install gLite WN package + runtime env. scripts (for gLite/ARC interop.)

# Setting up an ARC site for CRAB

- Install CMSSW + runtime env. scripts.
  - Different scripts for gLite/ARC interop. and native ARC jobs (could be merged?).
- Install gLite WN package + runtime env. scripts (for gLite/ARC interop.)
- Be visible in the BDII system (for gLite/ARC interop.)

# Using CRAB with the ARC Scheduler

- Install CRAB (the usual way)

# Using CRAB with the ARC Scheduler

- Install CRAB (the usual way)
- Install the ARC client <http://download.nordugrid.org/> (It's easy!)

# Using CRAB with the ARC Scheduler

- Install CRAB (the usual way)
- Install the ARC client <http://download.nordugrid.org/> (It's easy!)
- Set 'scheduler = arc' in your 'crab.cfg' file



# Using CRAB with the gLite/ARC interop.



# Using CRAB with the gLite/ARC interop.

Just use CRAB as if everyting was gLite.



# Summary

- CRAB w. gLite + gLite/ARC interoperability: Works

# Summary

- CRAB w. gLite + gLite/ARC interoperability: Works
- CRAB standalone + ARC: Works

# Summary

- CRAB w. gLite + gLite/ARC interoperability: Works
- CRAB standalone + ARC: Works
- CRAB server + ARC: Work in progress.

# Summary

- CRAB w. gLite + gLite/ARC interoperability: Works
- CRAB standalone + ARC: Works
- CRAB server + ARC: Work in progress.

Questions?