ARC Feedback from VOs / sites

ALICE

- ARC modules for AliEn were implemented using the features of ARC in 2014 and successfully worked until 2017. It was not too difficult to implement these modules. The only trouble with job databases on user far in 2015 and 2016 when SL6.X showed troubles with BDB formats, thus, migration to XML helped.

- ARC does not provide useful additional functionality to ALICE over CREAM.

- In principle it is easy for ALICE to migrate job submissions from CREAM to ARC, but in practice we have needed to get each site to fix what is published in GLUE v1;

- I have recently started looking into the situation for GLUE v2 and found only 1 site (KIT) publishing realistic numbers there... :-(

- ALICE have not yet provided recommendations w.r.t. ARC vs. HTCondor CE: both have pros and cons.

- A useful feature ALICE would like to see in ARC: information system contents that are OK out of the box, i.e. make it easier for an admin to get them right…

ARC Feedback from VOs / sites

CMS

- We rely on ARC CEs quite a bit in the CMS distributed computing grid and are happy with their overall performance

- Setting up an ARC/condor is trivial. 15 min work.

- Migrating from CREAMCE to ARC? Also trivial. ARC demands one configuration file. CREAM? Well it demanded yaim. Just getting rid of yaim was a supreme advantage.

- ~~HTCondorCE has a problem now: Apel publishing. There is a group trying to work it out.~~

- As far as I know neither CMS nor Atlas are pushing for one of them. Even if one of the CMS guys (Brian B) is involved with HTCondorCE.

- CMS use essentially condor factories to submit jobs. They don't care if sites are using Arc/Condor or HTCondorCE.

- There are things that could be improved:

  - My ARC CEs still block from time to time on what I perceive as some sort of race condition involving a-rex or grid-info-soft-register. That needs improvement.

  - There is a push to replace BDII with JSON files. Anyway, BDII in ARC has always been poor.

  - LHCb keep complaining that Arc sites have problems to publish OS version/release. I have a hack to force publish number of jobs running, waiting, etc. also to publish SI00.

  - Need better docs on stuff like draining the whole CE, or block just one VO, banning one user, blocking one VO except a set of DNs.

  - jura has improved accounting, but the documentation seems obsolete.

ARC Feedback from VOs / sites

LHCb

- The only major extra functionality that ARC provides (over Cream) is the specific resource request / site customisation (RAM, #cores, extra environment variables, ...). This is not (yet) used by LHCb significantly as the sites that support LHCb already give us the resource in exactly the way we require from the VO-card. Some of these features however are used by other VOs that use DIRAC.

- Given the DIRAC plugin nature, it is very easy / seamless for any experiment using DIRAC to deploy an ARC CE.

- LHCb submission uses use the ARC api (code on github: https://github.com/DIRACGrid/DIRAC/blob/integration/Resources/Computing/ARCComputingElement.py)

- LHCb is fairly agnostic and is equally able to use ARC or HTCondorCE. Choosing which is a decision left to the site depending on their preference.

- LHCb/DIRAC currently uses a fairly simple setup with resources expected to be approximately static at a given site. These requirements are already met by ARC.

- Problem with ARC as used by DIRAC : DIRAC polls the CEs regularly for the resources used by the VO. This is not a clear number in ARC5 and (without hacks) gives the total usage for the given site, rather than just for the VO at the site, to avoid overloading the sites. Various sites have implemented different fixes to resolve this. I understand that this is fixed automatically in ARC6, but we need to see the results from sites in production before we are able to update DIRAC appropriately (if needed).