Data Management in the Advanced Resource Connector

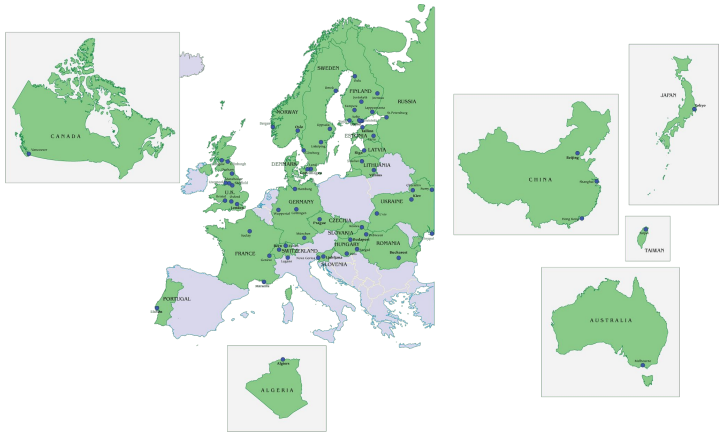
David Cameron (University of Oslo)
on behalf of ARC development team

ARC Overview

- What is ARC/NorduGrid/NDGF?
  - NorduGrid is a collaboration of institutes participating in ARC development
  - ARC (Advanced Resource Connector) is the software
  - NDGF is an infrastructure which uses ARC in its WLCG T1 computing sites

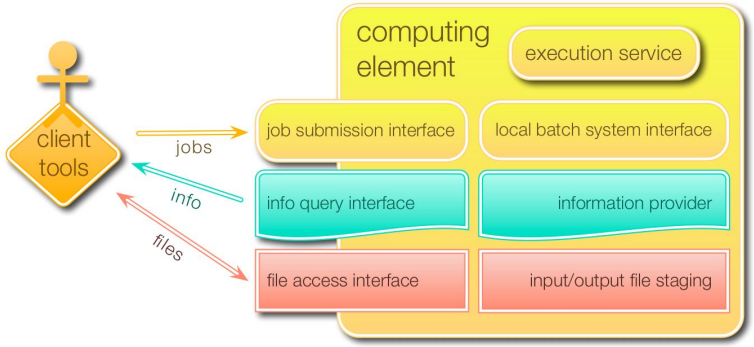
ARC Computing Element

- A gateway from Grid to batch system
- In use since 2001
- Widely deployed on European Grid sites
- Well-suited to HPC environment

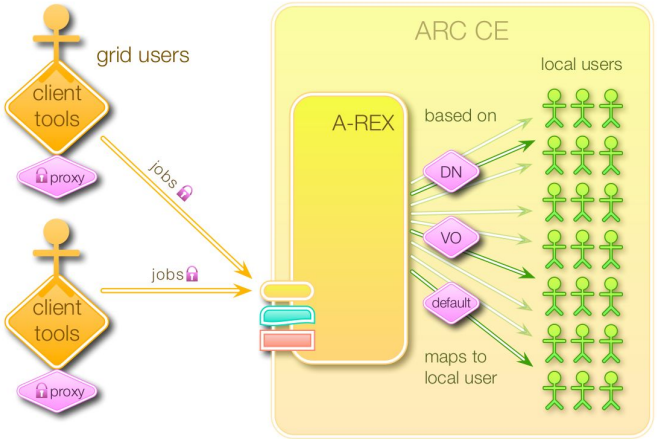


Worldwide ARC CE deployment

Components of the Computing Element

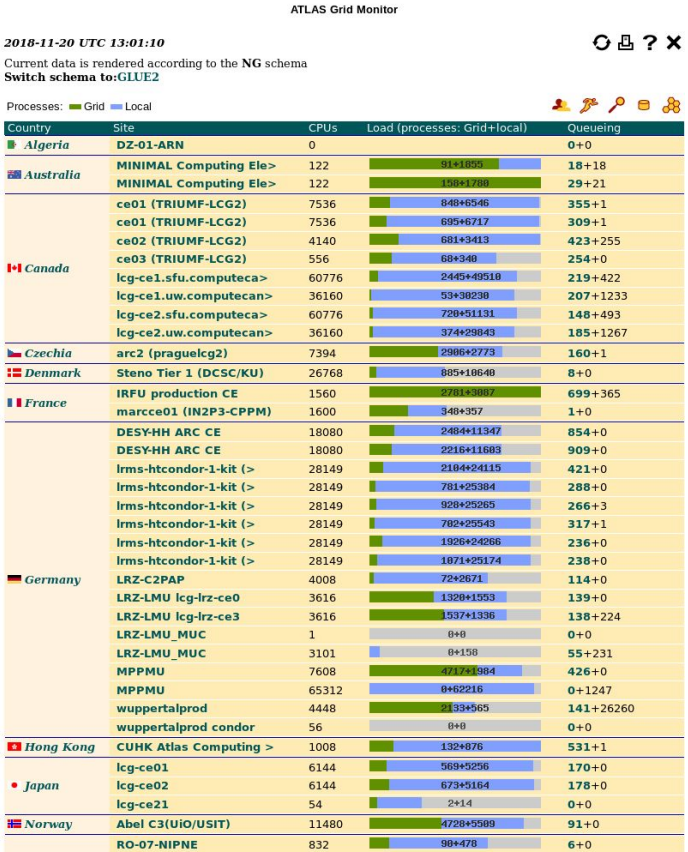


Mapping from grid users (certificates) to local users



In addition to the CE

- SDK for client applications (C++ and python)
- CLI for job and data management (arcsub, arcstat, arccp, arcls, ...)
- Information index: LDAP-based EGIIS soon to be replaced by DNS-based ARCHERY
- Cache indexing service (ACIX)
- Grid monitor (ATLAS ARC CEs)
- Accounting publishing to APEL and SGAS



ATLAS Grid Monitor

2018-11-20 UTC 13:01:10

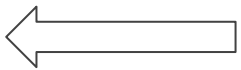
Current data is rendered according to the **NG** schema
Switch schema to:GLUE2

Processes: Grid Local

Country	Site	CPUs	Load (processes: Grid+local)	Queueing
Algeria	DZ-01-ARN	0		0+0
Australia	MINIMAL Computing Ele>	122	91+1855	18+18
	MINIMAL Computing Ele>	122	158+1788	29+21
Canada	ce01 (TRIUMF-LCG2)	7536	848+6546	355+1
	ce01 (TRIUMF-LCG2)	7536	695+6717	309+1
	ce02 (TRIUMF-LCG2)	4140	681+3413	423+255
	ce03 (TRIUMF-LCG2)	556	68+340	254+0
	lcg-ce1.sfu.computeca>	60776	2445+49510	219+422
	lcg-ce1.uw.computecan>	36160	53+30230	207+1233
	lcg-ce2.sfu.computeca>	60776	720+51131	148+493
	lcg-ce2.uw.computecan>	36160	374+29843	185+1267
Czechia	arc2 (praguelcg2)	7394	2906+2773	160+1
Denmark	Steno Tier 1 (DCSC/KU)	26768	885+10640	8+0
France	IRFU production CE	1560	2781+3087	699+365
	marcce01 (IN2P3-CPPM)	1600	348+357	1+0
Germany	DESY-HH ARC CE	18080	2484+11347	854+0
	DESY-HH ARC CE	18080	2216+11603	909+0
	lrms-htcondor-1-kit (>	28149	2104+24115	421+0
	lrms-htcondor-1-kit (>	28149	781+25384	288+0
	lrms-htcondor-1-kit (>	28149	928+25265	266+3
	lrms-htcondor-1-kit (>	28149	702+25543	317+1
	lrms-htcondor-1-kit (>	28149	1926+24266	236+0
	lrms-htcondor-1-kit (>	28149	1071+25174	238+0
	LRZ-C2PAP	4008	72+2671	114+0
	LRZ-LMU lcg-lrz-ce0	3616	1320+1553	139+0
	LRZ-LMU lcg-lrz-ce3	3616	1537+1336	138+224
	LRZ-LMU_MUC	1	0+0	0+0
	LRZ-LMU_MUC	3101	0+158	55+231
	MPPMU	7608	4717+1984	426+0
	MPPMU	65312	0+62216	0+1247
	wuppertalprod	4448	2133+565	141+26260
	wuppertalprod condor	56	0+0	0+0
Hong Kong	CUHK Atlas Computing >	1008	132+876	531+1
Japan	lcg-ce01	6144	569+5256	170+0
	lcg-ce02	6144	673+5164	178+0
	lcg-ce21	54	2+14	0+0
Norway	Abel C3(UiO/USIT)	11480	4728+5589	91+0
	RO-07-NIPNE	832	90+478	6+0

Modes of using ARC CE

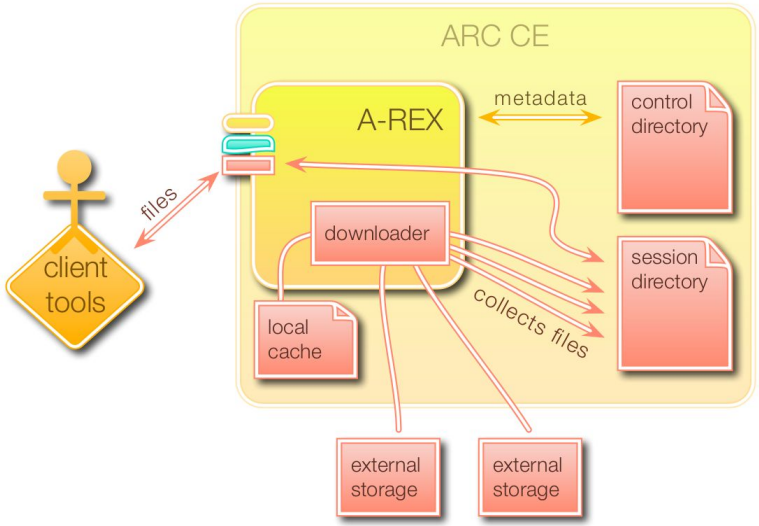
- Pilot gateway (ATLAS, CMS, LHCb)
  - Pilot factory submits wrappers through Condor-G
  - Pilot starts on worker node and pulls real payload
- NorduGrid (ATLAS)
  - ARC Control Tower pulls payload from Panda and pushes to ARC CE with correct requirements
  - ARC CE does staging of input/output
  - ARC Control Tower handles communication with Panda
- Truepilot (ATLAS)
  - ARC Control Tower pulls payload from Panda and pushes to ARC CE with correct requirements
  - Pilot on WN uses pre-placed payload
  - Pilot takes care of data staging and Panda communication just like pilot pull



Data management: the focus of this talk!

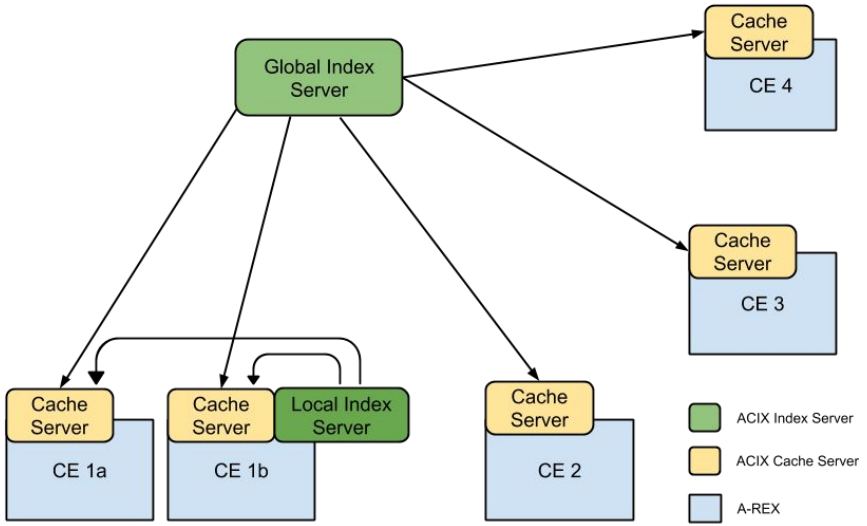
Data staging

- For HPC and many Grid sites ARC CE performs the critical role of transferring input and output data for jobs
  - Generally copying data between a shared file system and Grid storage
- ARC keeps a cache of input data on the shared file system
  - Jobs requiring already cached files do not need to re-download them
  - Cache is self-managing using LRU



More on cache, ACIX and Candypond

- Caching of remote input files is a very powerful feature for workloads which require the same input data for many jobs
- Several related services also exist:
  - CandyPond: extension of A-REX service allowing on-demand caching of files by a running job
  - CacheAccess: extension of A-REX service allowing the cache to be exposed to the outside
  - ACIX: A catalog of cache content - useful for brokering jobs to CEs where data is already cached
  - Whistleblower: Publication of cache content to an external service through message queues



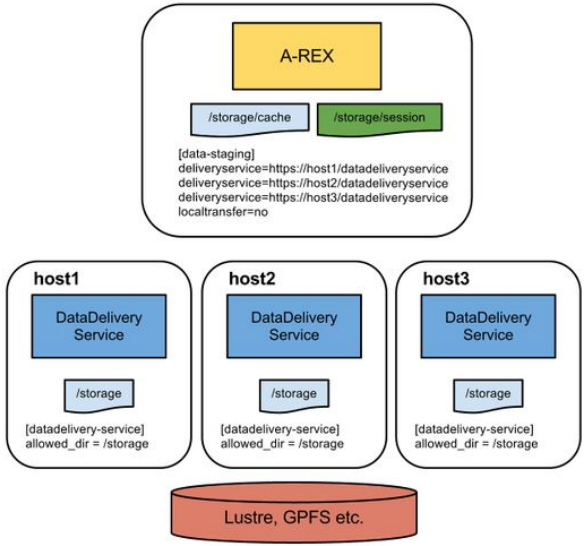
Possible ACIX deployment, with one global Index Server and a local Index Server for CE 1a and CE 1b

Data staging protocols

- Largely influenced by WLCG evolution, the current data transfer protocols supported by ARC are:
  - ACIX (ARC Cache Index)
  - File
  - GridFTP
  - HTTP(S)
  - LDAP
  - **Rucio** (ATLAS data management system)
  - SRM (Meta-protocol for access to WLCG storage, now deprecated)
  - S3
  - Xrootd (Native protocol to access files stored in ROOT format)
  - LFC, dcap, rfio, ... (legacy WLCG protocols supported through gfal2 library)
- Note that ARC CE does not do 3rd party transfer, all data is transferred to or from a local file system

Scaling up data staging

- Data transfer capability can be scaled up by adding extra data staging hosts as *delivery servers*
- The master CE hosts delegates data transfer to the delivery servers
  - All intelligence, scheduling etc is in A-REX, the delivery servers simply to a point-to-point transfer



Multiple hosts with one large shared FS

ARC and Rucio

- Rucio was added in 2014 just before Rucio went into production for ATLAS
- Implemented using REST calls with native ARC HTTP client

```
> arcls -L rucio://rucio-lb-prod.cern.ch/replicas/mc16_13TeV/EVNT.12714678._001433.pool.root.1
        srm://srmv2.ific.uv.es:8443/srm/managerv2?SFN=/lustre/ific.uv.es/grid/atlas/atlasdatadisk/rucio/mc16_13TeV/e5/fd/EVNT.12714678._001433.pool.
root.1
srm://grid002.ft.uam.es:8443/srm/managerv2?SFN=/pnfs/ft.uam.es/data/atlas/atlasdatadisk/rucio/mc16_13TeV/e5/fd/EVNT.12714678._001433.pool.root.1
srm://lapp-se01.in2p3.fr:8446/srm/managerv2?SFN=/dpm/in2p3.fr/home/atlas/atlasdatadisk/rucio/mc16_13TeV/e5/fd/EVNT.12714678._001433.pool.root.1
srm://sdrm.t1.grid.kiae.ru:8443/srm/managerv2?SFN=/t1.grid.kiae.ru/data/atlas/atlasdatadisk/rucio/mc16_13TeV/e5/fd/EVNT.12714678._001433.pool.root.1
srm://dcsrm.usatlas.bnl.gov:8443/srm/managerv2?SFN=/pnfs/usatlas.bnl.gov/BNLT0D1/rucio/mc16_13TeV/e5/fd/EVNT.12714678._001433.pool.root.1
```

- Rucio as an Object store proxy

```
> arccp myfile rucio://rucio-lb-prod.cern.ch/objectstores/s3+rucio://oshost:port/bucket/scope:lfn/RSE/write
```

ARC Cache Integration with Rucio

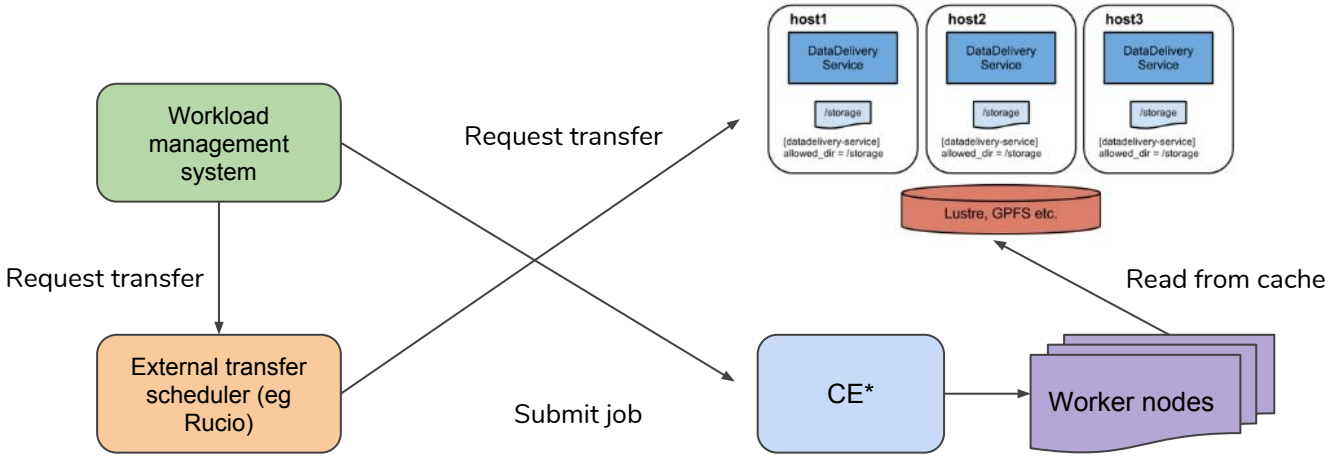
- Cache data can be registered in Rucio on a volatile RSE
  - i.e. Rucio does not manage the data on the RSE but can index it
- Useful for brokering jobs to where data is already cached
  - It is not guaranteed that the data is still in the cache when the job gets there, but not a problem - ARC will download it again
- A probe on the CE (the “whistle-blower”) periodically sends lists of files added to and deleted from the cache

```
# python whistle-blower.py
usage: whistle-blower.py [-h] --cache-dir CACHE_DIR --rse RSE
                         [--broker BROKER] [--port PORT] [--topic TOPIC]
                         [--timeout TIMEOUT] [--chunk-size CHUNK_SIZE]
                         --username USERNAME --password PASSWORD
```

  - ARC CE is configured to periodically dump the cache content to files
  - The whistle blower compares the dumps and looks at the difference
  - ActiveMQ messages with add/delete replica are sent to the Rucio message brokers
- The mechanism is not ARC-specific, can be used for any cache or non-Rucio-managed storage

ARC as a Data Transfer Service

- A stand-alone delivery server could provide a mechanism for pre-placing data in the cache without the need for ARC CE
- I.e. alternative transfer tool to FTS in Rucio



* Any kind of gateway for scheduling payloads, vaccuum model etc

Conclusion

- Integrating the Rucio catalog into an existing system is rather straightforward
  - An HTTP client is all that is needed
  - The volatile RSE concept allows caching systems to be integrated natively in Rucio
- ARC provides several data management features coupled to Rucio which could be useful for any community
- Close collaboration between ARC and Rucio developers keeps the two products well connected