EUROPEAN MIDDLEWARE INITIATIVE

Lessons Learned in the NorduGrid Federation

David Cameron
University of Oslo

With input from Gerd Behrmann, Oxana Smirnova
and Mattias Wadenstein

NORDUGRID
Grid Solution for Wide Area Computing and Data Handling

ARC

NDGF
NORDIC DATAGRID FACILITY

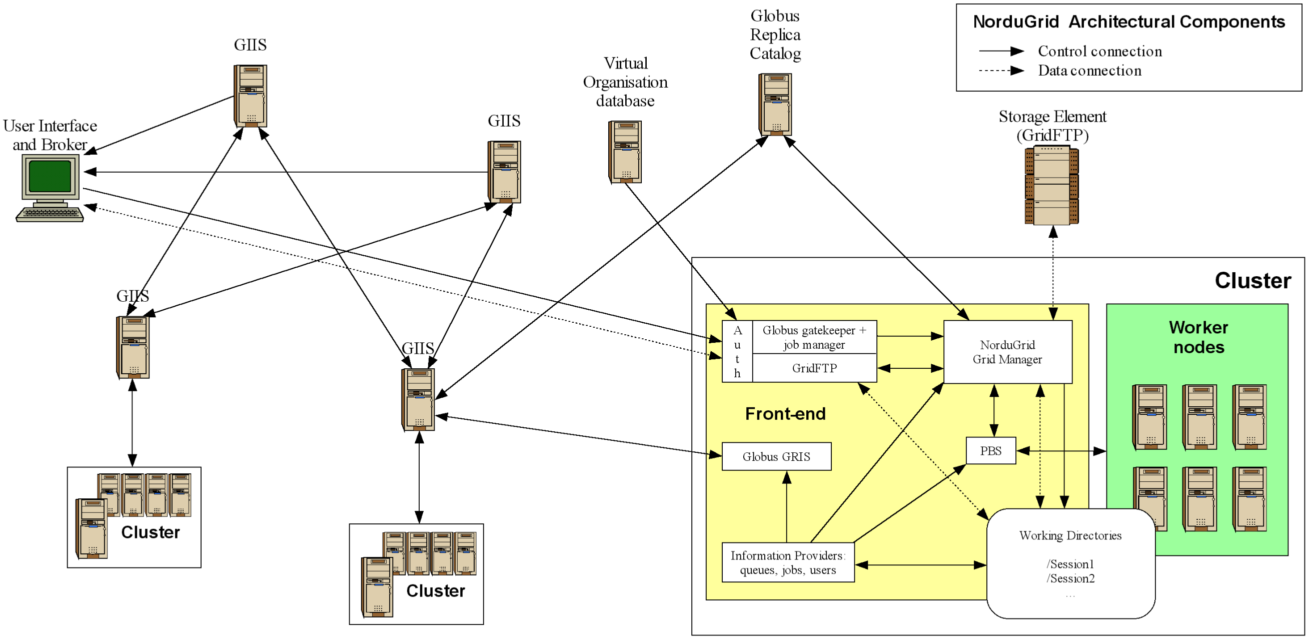
Creating Federated Data Stores For The LHC
14.9.12, Lyon, France

EMI is partially funded by the European Commission under Grant Agreement RI-261611

History Lesson

- 2001
  - NorduGrid collaboration formed by Scandinavian universities
  - Grid computing for LHC physicists
  - Resources provided by institutes
  - Grid middleware:
    - *Globus*
      - GridFTP SE, MDS info-system, RLS catalog
    - *Advanced Resource Connector (ARC)*
      - CE interface, batch system interaction, data staging, client tools, VOs, accounting etc.

History Lesson



October 2002

NorduGrid architecture, 2002

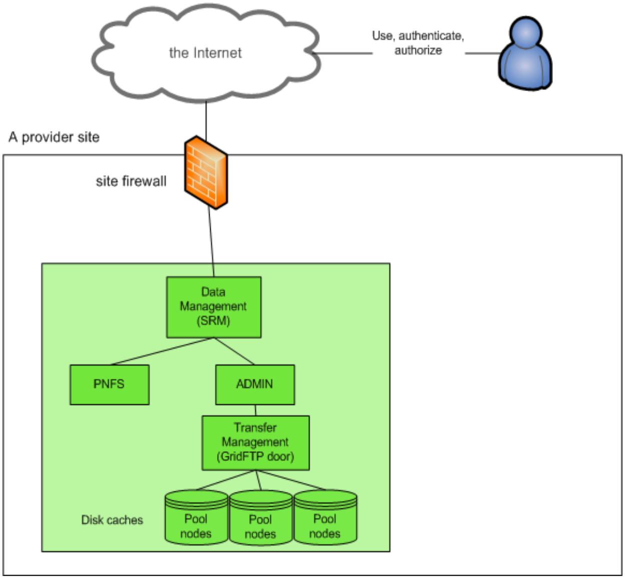
Moving Forward

- 2006
  - Nordic DataGrid Federation (NDGF)
  - Nordic Tier 1 centre for WLCG
    - *ARC CEs + Distributed dCache SE + ARC CE caches*
  - Distributed resources presented as single entity
    - *Resources still owned by institutes*
    - *NDGF provides connecting glue*

NDGF-T1

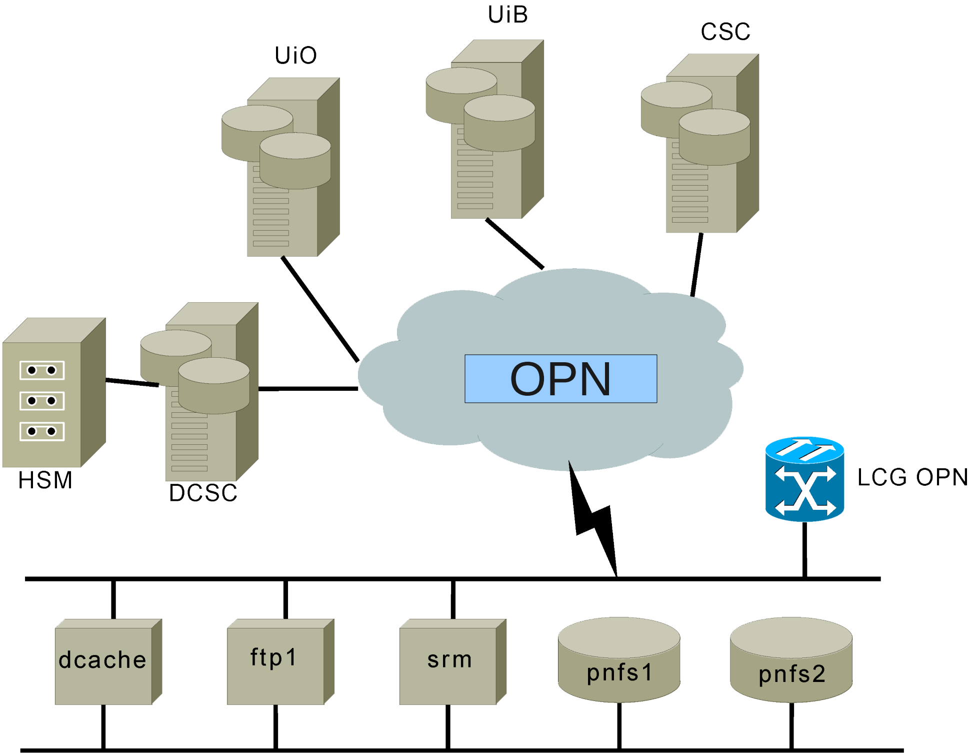
- Distributed Centre?
  - No one country large enough to host T1 itself
  - Nordic culture of cooperation
  - Blurry Tier concept
    - *No one site dominates*
    - *Good enough network (+ local caching) so that computing not tied to storage*
  - ➔ Distributed Computing (easy)
  - ➔ Distributed Storage (not so)

dCache



- Transparent access to data on mass storage systems under a single namespace
- Interaction with HSM
- “Doors” provide access via various protocols eg SRM, GridFTP

Distributed dCache



Distributed dCache

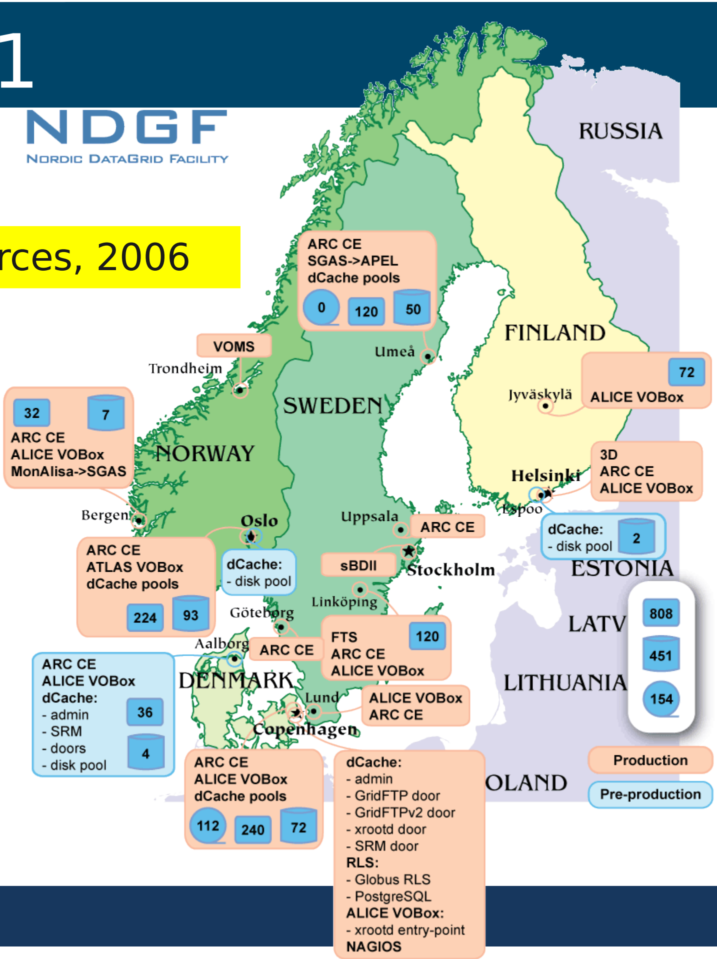
- Front-end nodes near Copenhagen (next to OPN switch)
    - *srm.ndgf.org, ftp1.ndgf.org, namespace DB, ...*
- Pool nodes scattered from Ljubljana to Umeå
    - 10Gb/s links
- Designed for maximum availability for acceptance of T0 data
- Front-end is central point of failure (as with any other site)
    - But failure of one pool/site only leads to some data unavailable
        - *Internal replication of recent data minimises effect*

NDGF-T1

EUROPEAN MIDDLEWARE INITIATIVE

NDGF
NORDIC DATAGRID FACILITY

NDGF T1 resources, 2006



ARC CE
SGAS->APEL
dCache pools

VOMS

ALICE VOBox

ARC CE
ALICE VOBox
MonAlisa->SGAS

3D
ARC CE
ALICE VOBox

ARC CE

dCache:
- disk pool

ARC CE
ATLAS VOBox
dCache pools

dCache:
- disk pool

sBDII

FTS
ARC CE
ALICE VOBox

ARC CE

ARC CE
ALICE VOBox
dCache:
- admin
- SRM
- doors
- disk pool

ALICE VOBox
ARC CE

ARC CE
ALICE VOBox
dCache pools

dCache:
- admin
- GridFTP door
- GridFTPv2 door
- xrootd door
- SRM door
RLS:
- Globus RLS
- PostgreSQL
ALICE VOBox:
- xrootd entry-point
NAGIOS

Production

Pre-production

NDGF-related dCache improvements

- Critical factor - NDGF developer (Gerd) became dCache developer
- GridFTPv2
  - Control channel via head node, data channel directly via pools
- New namespace implementation
- New SRM service container
- WebDAV support
- xrootd support

- Current protocols supported (dCache doors)
  - SRM, GridFTP, HTTP, WebDAV, DCAP, GSIDCAP, Xrootd, NFS 4.1

Other Tiers

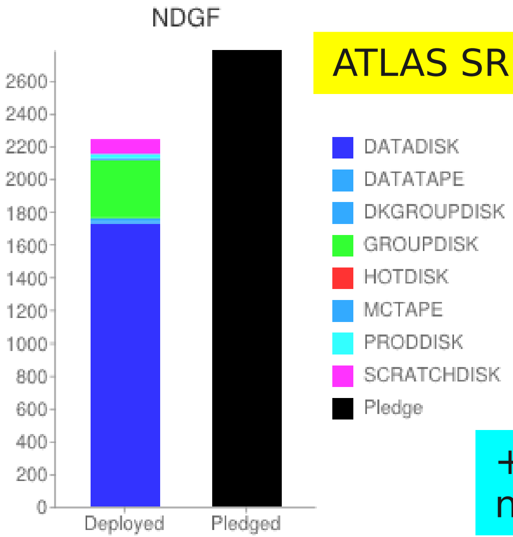
- Several Tier2/3 sites are associated with NDGF-T1
  - Some are independent – with own SRM endpoint (Swegrid, Ljubljana, Bern, ...)
  - Some are simply separate pools - with same endpoint but separate SRM space tokens (Norway T2, Copenhagen, Geneva, ...)

Operations

- Distributed storage – distributed people
- Operator on Duty rotates among 4 countries weekly
  - Sysadmin sitting at their institute
  - Deal with GGUS, downtimes, operations meetings etc
- Chatroom for communication
  - Weekly chat meeting (more efficient than voice!)
- Wiki, JIRA task tracking etc.

Current Status

- NDGF T1 stores ~3PB and 2M files (ATLAS + ALICE)



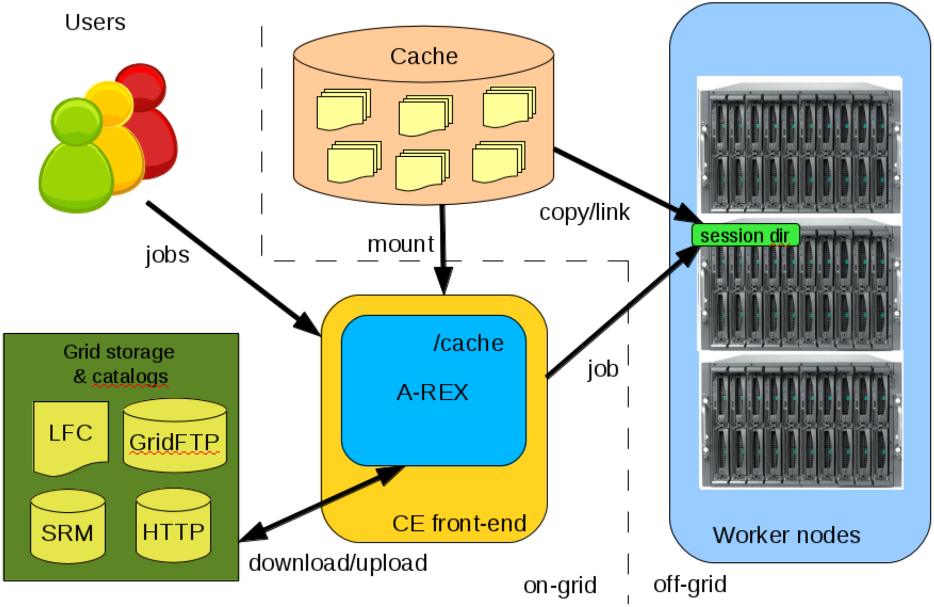
ATLAS SRM space tokens

+500TB on tape not in space token

Second level storage

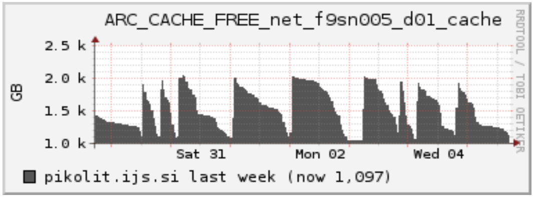
- NDGF T1 dCache provides persistent reliable mass storage for managed data transfers
- On-demand replication and unmanaged storage is provided by ARC caches

ARC Data Management Architecture

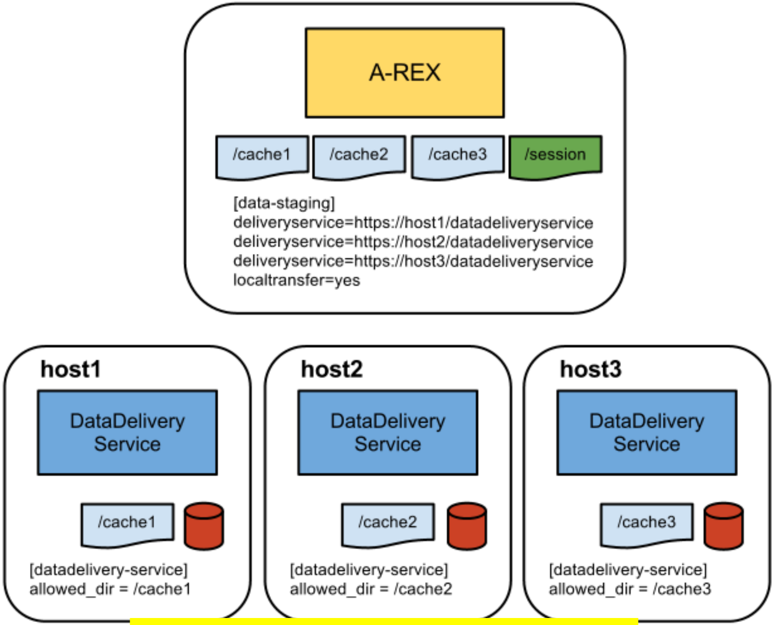


ARC CE Cache

- Local file system (NFS, GFPS, Lustre etc) mounted on CE front-end
- Cached files soft-linked or copied to job's working dir
- Authorisation checked against original source (and cached)
- Files always cached unless disabled in job description
- Space managed automatically using LRA
- No administration required
- Not accessible from outside



Multiple Caches



```
[data-staging]
deliveryservice=https://host1/datadeliveryservice
deliveryservice=https://host2/datadeliveryservice
deliveryservice=https://host3/datadeliveryservice
localtransfer=yes
```

```
[datadelivery-service]
allowed_dir = /cache1
```

```
[datadelivery-service]
allowed_dir = /cache2
```

```
[datadelivery-service]
allowed_dir = /cache3
```

Data is always written locally

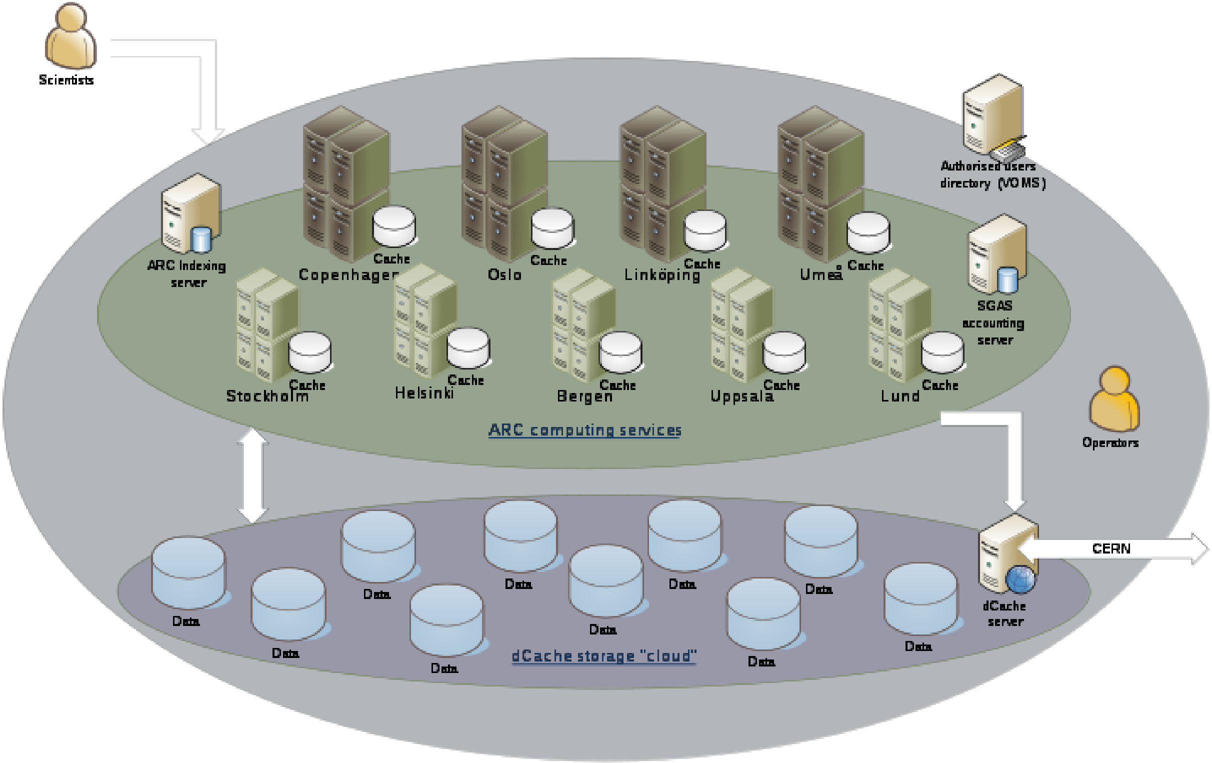
ARC CE Cache

- Counted as pledged storage but not accounted...
  - Depends on country (T2 in Sweden)
- Recommended size 100TB for 2000-core site running ATLAS production/analysis
  - Estimate ~1PB cache space in all NorduGrid
- Cache filesystem must have very good performance!
  - ARC CE writing + jobs reading
  - Can be scaled by adding more CE staging nodes and more caches

ARC Cache Index Service

- Caches publish their content periodically to a central index
    - Using Bloom filters for efficiency – which leads to false positives
    - Very simple web service – http query returns JSON dictionary of url:sites
- Jobs can be brokered to sites where files are cached
    - If false positive or file was deleted, it doesn't matter! ARC can download it again
    - No need for enforced consistency

NDGF


Scientists
ARC Indexing server
Copenhagen
Cache
Oslo
Cache
Linköping
Cache
Umeå
Cache
Authorised users directory (VOMS)
SGAS accounting server
Stockholm
Cache
Helsinki
Cache
Bergen
Cache
Uppsala
Cache
Lund
Cache
ARC computing services
Operators
CERN
dCache server
Data
Data
Data
Data
Data
Data
Data
Data
Data
Data
Data
dCache storage "cloud"

Advantages

- The combination of distributed persistent managed storage and caching gives many advantages
  - Pool downtime does not have to block jobs
  - No administration is required for the caches
  - No consistency requirement
  - Automatic replication on demand of popular data
  - No replication of unused data
  - Reduced load on managed storage
  - Managed storage does not need to be fast (for direct random data reading)

What next?

- Read from dCache via HTTPS instead of GridFTP
  - Solves network problems with multi-homed machines and OPN/public network
  - Writing of large files via HTTP still problematic
- Access data from ARC caches on other sites
  - Back-up replicas if main storage is down
  - But we don't want another SE
- Options:
  - Make xrootd federation of caches with loose consistency
    - *Security?*
  - Make own federation using ACIX + ARC CE HTTP interface

Lessons Learned

- NDGF staff core developers in dCache and ARC
  - Rapid availability of required new features
  - Influence over development strategy
- And involved in user communities (ATLAS, ALICE, etc)
- Automatic internal dCache replication and caching saves us in pool downtimes
- Distributed coordination takes a lot of close communication and learning
  - Some people more experienced than others
  - Automatic well-documented procedures for everything
- Users change requirements all the time and like control
  - Hardly any traditional middleware is used these days for job management, will data/storage management follow?
- Availability != happiness
- Impossible to make general system to suit everyone
  - Even if that's not what funding agencies want to hear...

*Disclaimer: presenter is funded by EMI, which funds ARC and dCache