ARC Middleware and its deployment in the distributed Tier1 center by NDGF

Oxana Smirnova
Lund University/NDGF
Grid 2008, July 1 2008, Dubna
Outlook

- ARC “Classic” overview
- NDGF Tier1
- Development of ARC
Advanced Resource Connector (ARC) in a nutshell

- General purpose Open Source European Grid middleware
  - One of the major production grid middlewares
  - Developed & maintained by the NorduGrid Collaboration
  - Deployment support, extensive documentation, available on most of the popular Linux distributions

- Lightweight architecture for a dynamic heterogeneous system following functional design principles
  - start with something simple that works for users and a functionality gradually
  - non-intrusive on the server side
  - Flexible & powerful on the client side

- User- & performance-driven development
  - Production quality software since May 2002
  - First middleware ever to contribute to HEP data challenge

- Strong commitment to standards & interoperability
  - JSDL, GLUE, Active OGF player

- Middleware of choice by many national grid infrastructures due to its technical merits
  - NDGF, SweGrid, SWISS Grid(s), Finnish M–Grid, etc…
  - Majority of ARC users are NOT from the HEP community

Illustrations: “Scandinavian Design beyond the Myth”
www.scandesign.org
ARC "Classic": overview

- Provides reliable implementation of fundamental Grid services:
  - The usual Grid security: SSO, VOMS, GACL
  - Job submission: direct or via matchmaking and brokering
  - Job monitoring & management
  - Information services: resource aggregation, representation, discovery and monitoring
  - Basic data management:
    - Automated seamless input/output data movement
    - Interfacing to Data Indexing (e.g. LFC), client-side data movement
    - Storage Elements
  - Logging service

- Builds upon standard open source solutions and protocols
  - Globus Toolkit® pre-WS API and libraries (no services!)
  - OpenLDAP, OpenSSL, SASL, SOAP, GridFTP, GSI
Almost all Monte Carlo and data processing today is done via Grid

There are 20+ Grid flavors out there
  - Almost all are tailored for a specific application and/or specific hardware

LHC experiments make use of 3 Grid middleware flavors:
  - gLite
  - ARC
  - OSG

All experiments develop own higher-level Grid middleware layers
Provides a unique distributed “Tier1” center via NorduGrid/ARC

Involves 7 largest Nordic academic HPC centers

...plus a handful of University centers (Tier2 service)

Connected to CERN directly with GEANT 10GBit fiber

Inter-Nordic shared 10Gbit network from NORDUnet

Budget: staff only, 2 MEUR/year, by Nordic research councils

2008-07-01
## ARC Compute elements

**Distributed over wide area**

**Operating systems**
- RedHat, Ubuntu, Scientific Linux, CentOS, Rocks

**Batch systems**
- LoadLeveler, Torque, Condor, SGE

**Compute element flavors**
- ARC CEs
- ALICE VOBoxes
- No LCG/glite CEs

**No single entry point for job submission**
- Not actually needed by the VOs

**WLCG monitoring and accounting procedures**
- SAM tests (modified for ARC-CE)
- Aggregation via SGAS and via ATLAS “dashboard”
- No aggregation (yet) in ALICE monitoring system
Worker Nodes

Operating systems

- Typically, same as CEs: RedHat, Ubuntu, Scientific Linux, CentOS

Managed entirely by local batch systems

- No Grid middleware installed
  - Files staged in/out by the CE
  - Optimizes data movement => better overall efficiency

Connectivity

- Normally no inbound
- Outbound connectivity is sometimes limited to certain IP ranges by sysadmins.
Storage elements

Distributed storage
- Pools distributed over wide area (from Norway to Slovenia)
- Single entry point via dCache SRM at srm1.ndgf.org
- We use tapes as well (NDGF introduced ngstage tool in ARC)

Middleware: dCache 1.8
- Extended by NDGF for distributed storage
- Uses GridFTPv2

Operating systems
- dCache-admin: Ubuntu
- pools – all the abovementioned systems (RHEL, CentOS, ...)

WLCG monitoring
- Same way as any conventional dCache SE (own monitoring tools, SAM tests)
NDGF Tier1 services summary

Level 1: central services, 99% up-time
- dCache: admin, doors: gridftp, xrootd, srm
- Data indexing: RLS and LFC
- WLCG monitoring: site BDII

Level 2: distributed, 98% up-time
- Storage pools
- ARC gatekeepers
- VOBoxes (ALICE)
- FTS: with GridFTPv2 patches
- WLCG 3D database
- VOMS

Level 3: distributed, 95% up-time
- Worker nodes
- ARC GILses
- Monitors
Relation to Tier2s

- Ljubljana Tier2 is affiliated via ATLAS
  - Practically, extends Tier1
  - W.r.t. data retention, has a “normal” Tier2 policy
- Several Swiss sites extend Tier1 computing-wise
- Several Nordic Tier2s are being formed
  - Largely extending Tier1 w.r.t. computing power, but not charged with data storing duties
ARC status and plans

- ARC is a mature software which has proven its strengths in numerous areas
- Production release 0.6.3 is out
  - Stability improvements, bug fixes
  - LFC (file catalogue) support
- ARC faces a scalability challenge posed by "ten-thousand-core" clusters
  - File cache redesign is necessary (ongoing)
  - Uploader/downloaders load on frontends needs to be optimized
  - Local information system needs to be improved (BDII is being tested)
- Release plans based on pre-WS ARC "Classic"
  - 0.6.x stable releases will be periodically released
  - Preliminary planning for an 0.8 release incorporating new major features
- Also in development:
  - A Web-service based solution (working prototypes exist)
  - Main goal: achieve interoperability via standards conformance (e.g. BES, JSDL, GLUE, SRM, GridFTP, X509, SAML)
  - Migration plan: gradually replace ARC components with new modules, possibly co-deploying both initially
  - Supported by EU via the KnowARC project; proposed to be a part of EGI middleware toolkit
ARC "Classic" and WS-based ARC

- **ARC "Classic"**
  - Robust solution
  - Non-standard interfaces
  - Not modular
  - Limited portability
  - Difficult to ride

- **WS-based ARC prototype**
  - No-crazy design (not an extreme middleware)
  - Standard interfaces
  - Modular structure
  - Supports MS Windows, Mac OS
  - **Improved performance**
    - Natural continuation of ARC Classic
  - **Enhanced user experience**
Service decomposition for ARC

A look inside the WS–based ARC prototype

- Already inside the box
  - Hosting Environment Daemon
    - Flexible service hosting and development framework
    - Takes care all the networking-layer (e.g. SOAP)
    - Available on MS Windows and Mac OS as well
    - Offers Python and Java language bindings in addition to the native C++
  - Test service (echo)
  - A–REX (computing element)
  - Central information indexing service
  - Basic storage system services
  - Security framework (including delegation)
  - ARCLIB and powerful command–line tools
    - Including plugins for ARC "Classic", gLite
- Soon to be added to the box
  - Monitoring and Accounting service (MARS)
  - broker service supporting pull operation as well (scheduler)
  - P2P–based information system backbone
  - Consistent storage system solution
  - some sort of support for dynamic application management and job sandboxing using virtualization techniques
  - Sysadmin–friendly, functionality–centric configuration
  - Technical documentation

Warning: not a production level!
The flagship service: A-REX

- **ARC** Resource–coupled **EX**ecution service
- Job execution service implemented within the Hosting Environment Daemon
- Clever Grid layer on top of a computing resource
- Based on the Grid Manager module of ARC
  - Numerous enhancement in code base
- Standard interface:
  - BES–compliant service (only the BES–Factory supported)
  - JSDL
  - NorduGrid extensions for both BES and JSDL
  - WSRF for information queries
- Demonstrated standard–based interoperability with large number of prototypes including gLite, Unicore
  - Supercomputing 2007 HPCP interoperability fest
  - ARC/gLite/Unicore Tutorials at OGF3, Barcelona
On ARC prototype backward compatibility

- On the server side, almost everything has new interface/format
  - Information model
  - Information query
  - Job management
  - Security language
  - Configuration

*OLD clients do not work with new servers, nevertheless we provide co-deployable services*

- On the client side, ARCLIB offers a smooth transition, all the client utilities support old servers

*OLD servers are accessible from new clients*
On interoperability

- Strategy: interoperability via open standards
  - BES, JSDL, GLUE, SRM, GridFTP, X509, SAML
  - Agreed extensions are critical (profiles)
- For a short term (?) transitional period gateway-like solutions are necessary
  - ARCLIB addresses the ARC -> other middleware direction
  - ng* commands will offer transparent access capability to 3rd party services
- Target platforms: gLite, Unicore
Further information

- ARC "white paper":
- WS–based ARC design:
- Code:
  - svn.nordugrid.org -> arc0 and arc1 directories of the nordugrid repository
  - download.nordugrid.org -> official source and binary packages (mostly for arc classic), external software
- Join the community!
  - Check out, sing up for the nordugrid-discuss mailing list
  - Attend some of the Technical Meetings or Conferences