

# ARC Client Status and Plans Nordugrid – Uppsala 2012

**Martin Skou Andersen** ([skou@nbi.ku.dk](mailto:skou@nbi.ku.dk))

University of Copenhagen

Also on behalf of Zsombor Nagy,

Qiang Weizhong and David Cameron

- ARC Client Tools
- What's new in 12.05
- Improved Resource Discovery
- Improved Brokering
- Improved Submission
- Improved Job Management
- Future Plans and Tasks

## Credential management

arcproxy arcslcs

arc renew

# ARC Client Tools



## Credential management

arcproxy arcsiics  
arc renew

## Job submission

arcsub arcresub  
arctest arc migrate

## Credential management

arcproxy arcsiocs  
arc renew

## Job submission

arcsub arcresub  
arctest arc migrate

## Job management

arcstat arcget arccat  
arc resume arcclean  
arckill

## Credential management

arcproxy arclsics  
arc renew

## Job submission

arcsub arcresub  
arctest arc migrate

## Data management

arcls arcrm arccp  
arc mkdir arc acl

## Job management

arcstat arcet arccat  
arc resume arcclean  
arckill

# ARC Client Tools



## Credential management

arcproxy arcsiocs  
arc renew

## Job submission

arcsub arcresub  
arctest arc migrate

## Data management

arcls arcrm arccp  
arc mkdir arc acl

## Job management

arcstat arcet arccat  
arc resume arcclean  
arckill

## Resource discovery

arcinfo arcsync

# ARC Client Tools



## Credential management

arcproxy arcsiics  
arc renew

## Job submission

arcsub arcresub  
arctest arc migrate

## Data management

arcls arcrm arccp  
arc mkdir arc acl

## Job management

arcstat arcet arccat  
arc resume arc clean  
arc kill

## Devel tools

arcecho arcsrmping  
arcwsrf

## Resource discovery

arcinfo arcsync



# What's new in 12.05 — Security

- Support for PKCS#12 certificates
- Support credential source from Firefox's NSS internal PKCS#11 module
  - Invoke with: `arcproxy --nssdb`
  - Default NSS DB location: `~/.mozilla/firefox/profiles.in`



- `arcmkdir` – New command
  - Allows for directory creation on grid storage elements and catalogs
  - Supported protocols:
    - *file*
    - *gsiftp*
    - *srm*
    - *lfc*
    - *gfal*



- Added support for:
  - LDAP GLUE2
  - EMI ES interface + EMI ADL (job description)
  - EMI Registry - resource discovery
  - Bulk job submission and management
- libarcclient library restructured to accommodate EMI ES and GLUE2 support
  - Resource discovery more seamless and flexible
  - Also removed dependency on job list file
  - Encapsulated plugin loading functionality
  - More Python friendly bindings
- Most of libarcclient API modified



# What's new in 12.05 — Compute



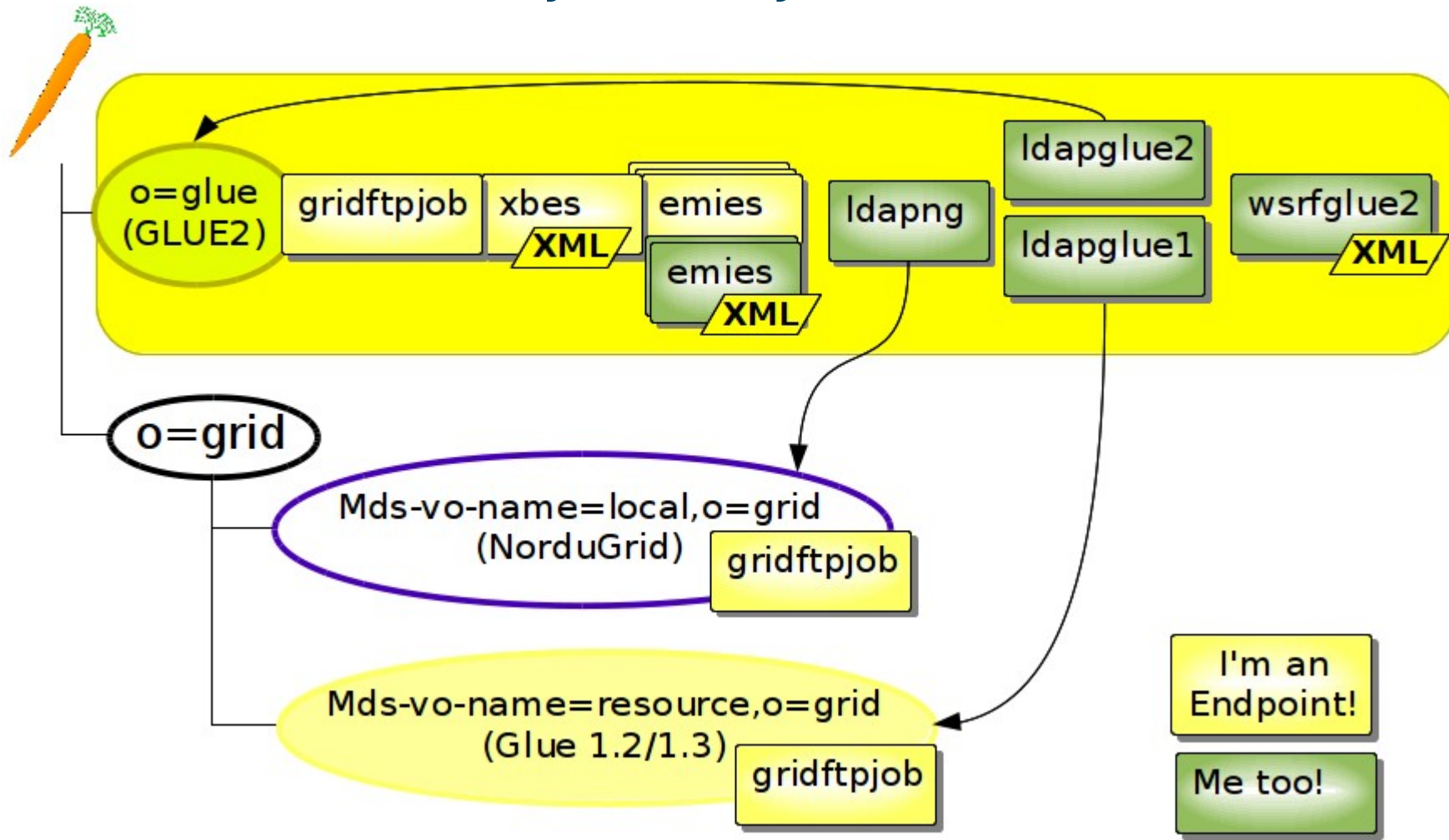
- Added support for:
  - LDAP GLUE2
  - EMI ES interface + EMI ADL (job description)
  - EMI Registry - resource discovery
  - Bulk job submission and management
- libarcclient library restructured to accommodate EMI ES and GLUE2 support
  - Resource discovery more seamless and flexible
  - Also removed dependency on job list file
  - Encapsulated plugin loading functionality
  - More Python friendly bindings
- Most of libarcclient API modified



- Old approach
  - Couldn't handle:
    - *Multiple endpoints per resource*
    - *Discovery through multiple interfaces (resource duplication)*
  - Covered three functionalities in one class – Not very flexible
  - Target information kept internal
  - Threading split between convenience and specialised classes
  - Blocking once querying was initiated

# Improved Resource Discovery

- Local information system layout – Current situation



## class EntityRetriever

- New approach
  - Handles multiple endpoints per resource
  - Possible to merge duplicate resources
  - Builds on general template class  
3 different instances available:
    - *Index/registry querying (ServiceEndpointRetriever)*
    - *Local infosys querying (TargetInformationRetriever)*
    - *Job list querying (JobListRetriever)*
  - Can easily be linked to provide extended functionality
    - *Index/registry + local infosys querying (ComputingServiceRetriever)*
  - Fetched information added to provided container

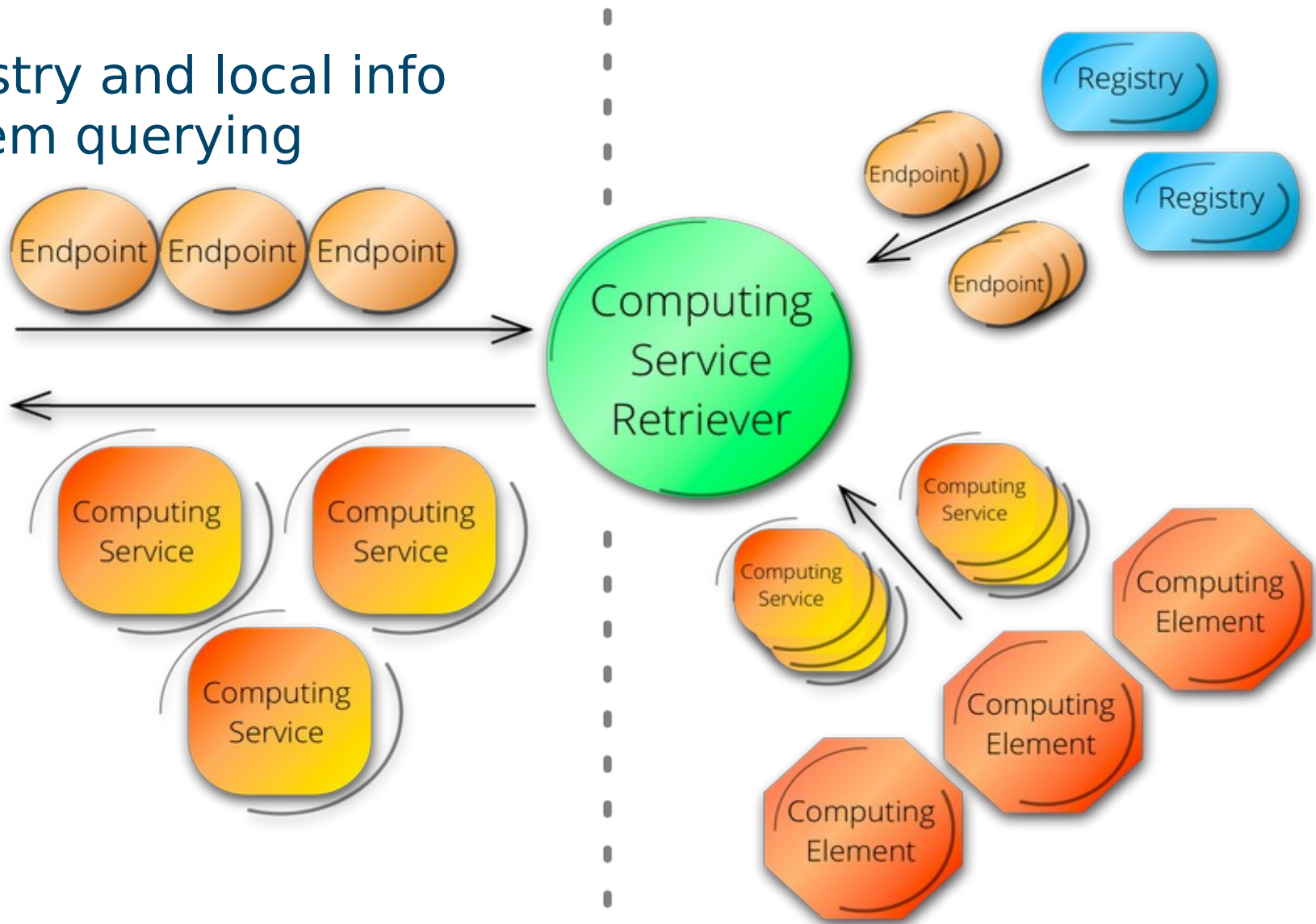
## `class EntityRetriever`

- New approach cont.
  - Threading code kept within template class
  - Specialised classes should just be thread safe
  - Initiating querying doesn't block – Information added to container when fetched
  - Querying preferred interfaces takes precedence



# Improved Resource Discovery

- Registry and local info system querying



## class Broker

- Old approach
  - Matching and sorting kept internal
  - Custom methods for looping over sorted targets
  - Not possible to add new targets to existing sorting
  - Plugin loading not encapsulated – Explicit loader instance needed
- New approach
  - Matching and sorting methods public available
  - External container used for holding matching and sorted targets
  - Plugin loading encapsulated – No need for loader instance
  - Sorting compares one against one

`class Submitter`

- Encapsulated plugin loading – No loader instance needed
- Bulk job submission support through library
  - Currently not implemented at plugin level
  - Brokering doesn't yet take advantages of this
- Test jobs can easily be added (arctest)
  - Test job descriptions are read from file

## `class JobSupervisor`

- Dependency on job list file in library removed  
(`~/ .arc/jobs.xml`)
- Plugin handling fully encapsulated
- More flexible usage of member methods
- Support for bulk job management operations
  - Currently not implemented at plugin level

- Complete GLUE2 support
  - Currently only ~80% done
  - Synchronizing jobs
  - Merge duplicates
  - Update existing objects
  - Support for additional entities (e.g. `AdminDomain`)

- Complete EMI ES support
  - Create archoId command (PauseActivity operation)
    - *Pending feature request exist*
  - Handle state attributes properly
  - Settle on how to publish endpoints (service related)
    - *Discussion ongoing*
  - arcresume support (ResumeActivity and RestartActivity operations)
  - Make submission less error prone
    - *Handle error responses from service better*

- Enhance documentation
  - API documentation – Classes needs to be documented
  - Technical description of how to code against ARC libraries
    - *C++ examples – CLI code*
    - *Python examples – Some exist already – Not up to date*
    - *Java examples – Non existing*
    - *Examples on usage of client, credential and data libraries*
  - Tutorial – Programming against ARC libraries
  - ...

# Future Plans and Tasks



- Bugfixing – Current status as of 31-05-12

## Severity

|                             | major    | normal    | minor    | trivial  | enhancement | feature request | Total     |
|-----------------------------|----------|-----------|----------|----------|-------------|-----------------|-----------|
| aleks@fys.uio.no            | .        | <u>7</u>  | .        | <u>1</u> | <u>1</u>    | .               | <u>9</u>  |
| balazs.konya@hep.lu.se      | .        | .         | .        | .        | <u>1</u>    | .               | <u>1</u>  |
| david.cameron@cern.ch       | .        | <u>1</u>  | .        | .        | <u>1</u>    | <u>1</u>        | <u>3</u>  |
| martoni@niif.hu             | .        | <u>3</u>  | <u>1</u> | .        | <u>2</u>    | .               | <u>6</u>  |
| mattias.ellert@fysast.uu.se | .        | <u>7</u>  | .        | <u>2</u> | <u>2</u>    | <u>2</u>        | <u>13</u> |
| skou@nbi.dk                 | <u>3</u> | <u>24</u> | <u>2</u> | <u>1</u> | <u>3</u>    | <u>1</u>        | <u>34</u> |
| weizhongqiang@gmail.com     | .        | <u>5</u>  | .        | .        | <u>1</u>    | <u>2</u>        | <u>8</u>  |
| Total                       | <u>3</u> | <u>47</u> | <u>3</u> | <u>4</u> | <u>11</u>   | <u>6</u>        | <u>74</u> |



- Performance analyse
- Bulk job submission and management
  - Implement operations in plugins
  - Take advantage of bulk submission in broker
- Direct submission to CE
  - Don't do resource discovery and brokering
- Support RESTful access interface provided by VOMS servers
  - Already started



**Thank you!**

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