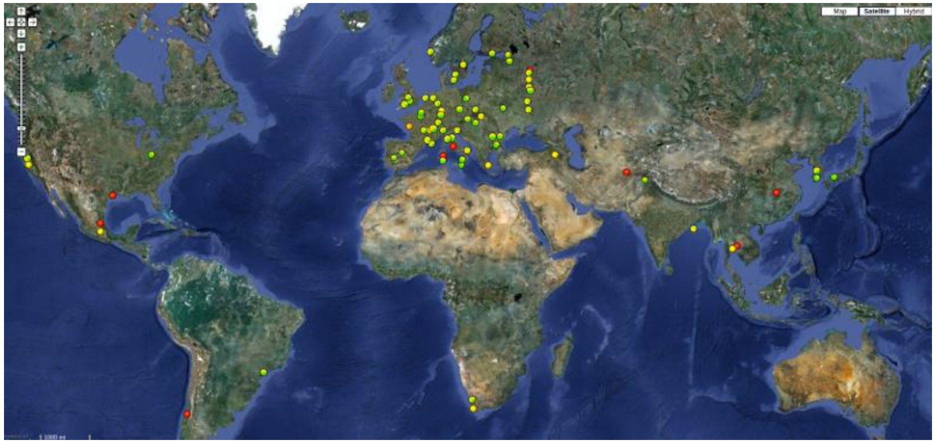
JAliEn

Java AliEn middleware

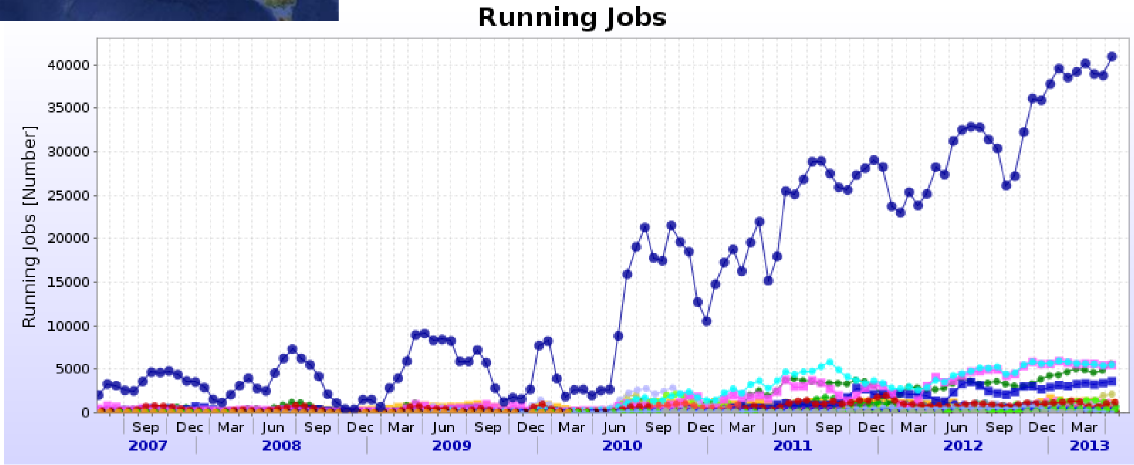
A. Grigoras, C. Grigoras, M. Pedreira
P Saiz, S. Schreiner

ACAT 2013

ALICE Grid Environment



- 100 sites
- up to 60 000 concurrent jobs
- >300 active users

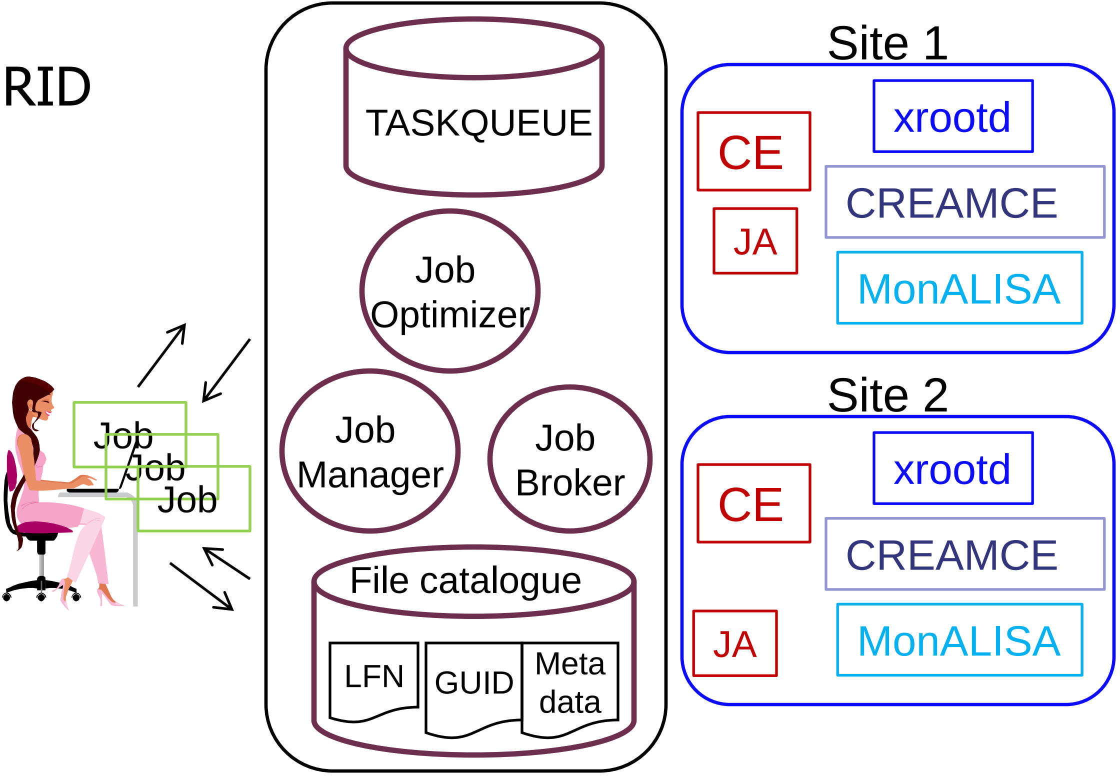


- 14PB of data on tape
- 13PB on disk

AliEn2 @GRID +JAliEn

AliEn2 – Current status

- All components to create a GRID
- File Catalogue
  - UNIX-like file system
  - Mapping to physical files
  - Metadata information
  - SE discovery
- Transfer Model
  - With different plugins
- TaskQueue
  - Job Agent & pull model
  - Automatic installation of software packages
  - Simulation, reconstruction, analysis...



Perl, C/C++, Bash scripting

~ 150 packages to maintain

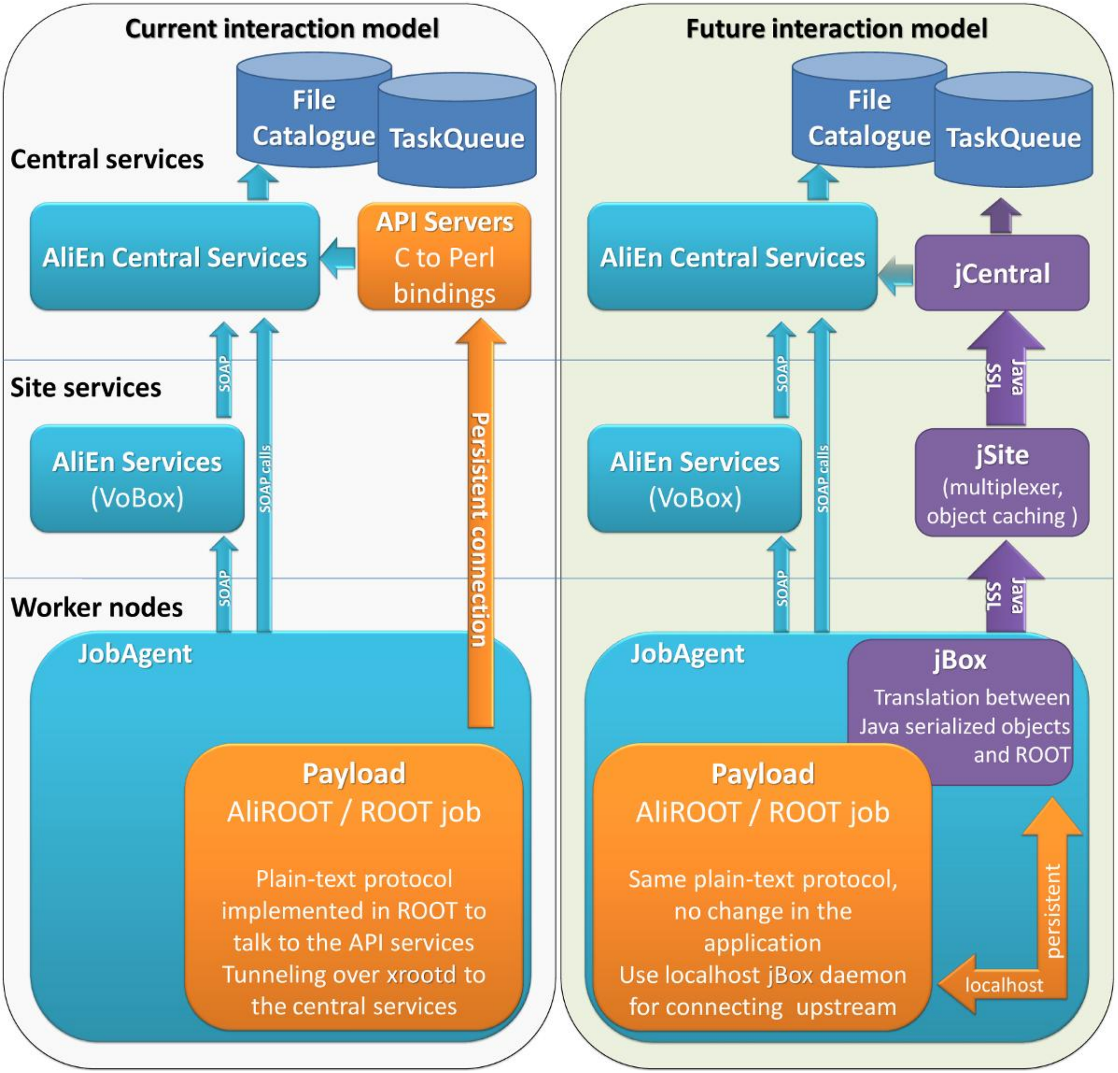
Why JAliEn

- ALICE uses MonALISA for monitoring
  - Jobs, transfers, site services, central services, real-time and accounting data, network topology discovery, storage elements functional tests and many more
- Based on the monitoring information a framework was developed to steer in an automatic way central productions, data registration and replication
- Gradually this framework grew in a full implementation of the AliEn objects and interactions with the central databases
  - File Catalogue: LFN, GUID, PFN, SE, booking table
  - Job, JDL and TaskQueue interactions
  - Users, quotas
    - gLite packages for proxy-based authentication
    - Support for signed JDLs by all parties involved
  - Access envelope generation, storage management, transfer methods
    - Parallel replica uploads and downloads for efficient end-user interaction with the federated SE space
  - and the rest of supporting objects and methods in AliEn

Why JAliEn

- Next step is to provide a full end-to-end Java implementation of AliEn
- More efficient communication infrastructure
    - Persistent, compressed, SSL channels
    - Exchanging Java serialized objects
    - Multi-layered channel multiplexing and object caching
    - Logging and error propagation
- No tokens for authentication, simply use the Grid certificate
- Better real-time monitoring of all components
- Platform-independent, easily maintainable code

AliEn2 vs JAliEn


Current interaction model
Central services
File Catalogue
TaskQueue
AliEn Central Services
API Servers C to Perl bindings
Site services
SOAP
SOAP calls
Persistent connection
AliEn Services (VoBox)
Worker nodes
SOAP
JobAgent
Payload AliROOT / ROOT job
Plain-text protocol implemented in ROOT to talk to the API services Tunneling over xrootd to the central services
Future interaction model
File Catalogue
TaskQueue
AliEn Central Services
jCentral
SOAP
SOAP calls
Java SSL
AliEn Services (VoBox)
jSite (multiplexer, object caching)
SOAP
Java SSL
JobAgent
jBox Translation between Java serialized objects and ROOT
Payload AliROOT / ROOT job
Same plain-text protocol, no change in the application Use localhost jBox daemon for connecting upstream
persistent
localhost

JCentral

- Implement all AliEn objects in Java (LFN, PFN, GUID, Job, SE, access envelopes, users etc)
- Direct connections to the DBs for all operations
  - But for resubmit and kill for the moment
- Delegates presentation to the end client
- Implement the minimal set of commands as API calls via serializable objects
- Encapsulating requester identity

JSite

- Trusted entities that multiplex requests to the central services
- Some of the objects are cacheable and can be quickly returned to the requester
  - For example OCDB file locations …
- Can be cascaded indefinitely in order to reduce the number of sockets per server to a reasonable number
  - Or can be missing altogether, for small sites or opportunistic resources
- Running as another AliEn service on the site VoBox

JBox

- Authentication and authorization
  - No more proxies for the users
- Communication
  - Listing, opening and writing files
  - Submitting and accessing job information
- Components
  - Java daemon responsible for authentication/authorization and communication between user applications and central services
  - ROOT implementation of TGrid functionality
  - Java shell client – same aspect and functionality as previous client
    - Commands implemented as classes using introspection

Current status

- ALICE MonALISA uses it fully
  - LPM and user job management
  - Catalogue browser, file editing, quotas...
- Storage cleanup, all scheduled file transfers
- Communication channels working fine
- Many end-user commands are implemented in Jbox
  - cat, cp, mv, rm, rmdir, find, ls, cd, pwd, mkdir, guid2lfn, lfn2guid, ps, masterjob, packages, submit, kill, whereis, w, uptime, type, time, role, access, commit

Conclusions

- JAliEn is the next version of AliEn Grid middleware
- The transition will be done step by step until all services are replaced by JAliEn services