Local replica of JAliEn central services

GSoC 2020

CERN
3 September 2020
Animesh Narayan Dangwal, animesh.leo@gmail.com

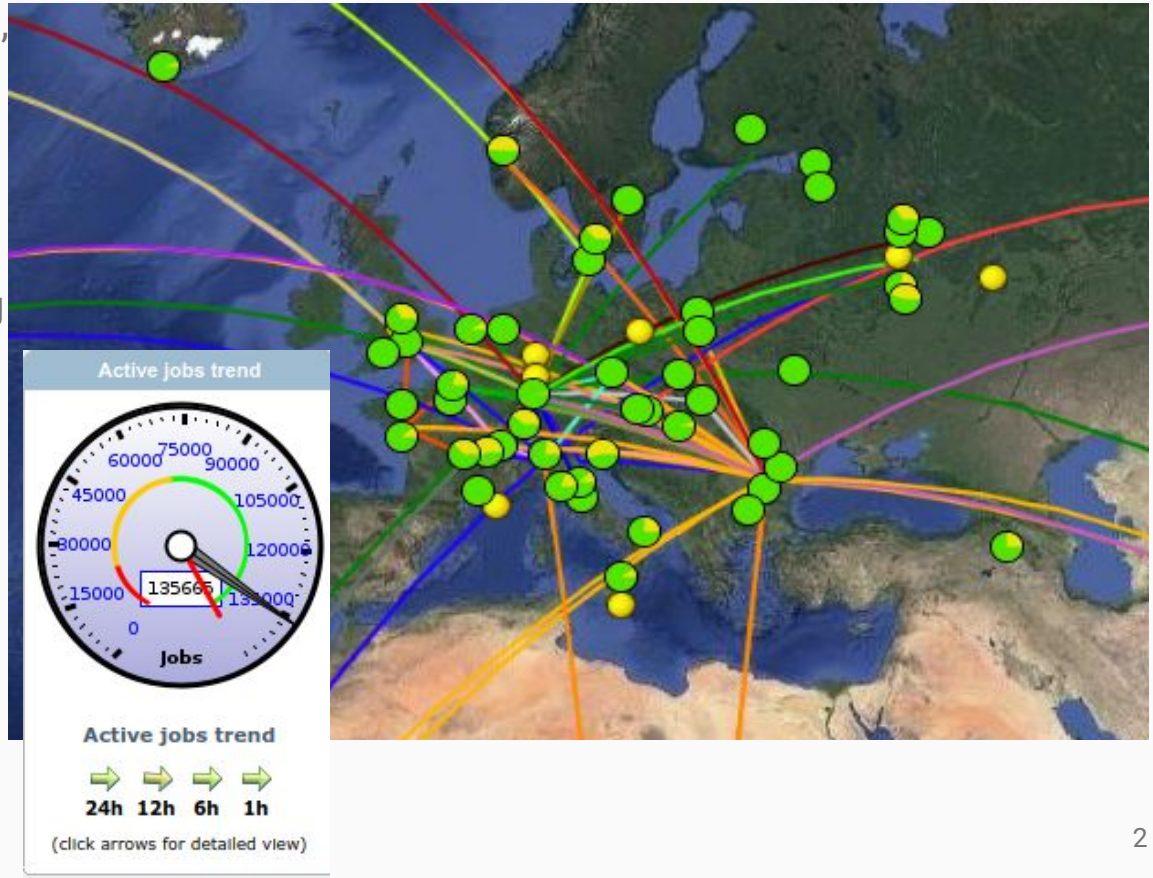
What is JAliEn?

ALICE (A Large Ion Collider Experiment), is one of 4 biggest LHC experiments at CERN. To analyse the recorded events data obtained it uses AliEn, a grid computing middleware.

The Grid is highly distributed computing system. It consists of nodes, specialised for certain tasks (storage, compute) that collaborate with each other. These nodes can be continents apart.

JAliEn is a reimplementation of AliEn in Java. Through which a user can:

- Access the Grid
- Store files on the Grid
- Run jobs on the Grid



JAliEn Components

Central Services (JCentral) keeps track of
- User info and configuration (LDAP)
- Files metadata and job info (MySQL).

But no actual files are stored on JCentral.

Site services:

- **SE** (Storage Element) is a functional element whose purpose is only for storing files. There are many storage services used, for our project we focused only on XRootD. A user must have the right roles to create and read files. This is done by JCentral issuing a token (access envelopes)

- **CE** (Batch Queues+VOBox) is used for running user jobs on a Grid site. New JobAgents are continuously spawned, and they fetch and run user jobs from JCentral. Any ALICE member can submit jobs.

User Interfaces (JSh, alien.py, JAliEn-ROOT, TJAlien) provide the user with the APIs to access files, store files and run jobs on the grid.

Three GSoC phases

1. Central Services: JCentral & alien.py
2. Storage Element: XRootD and file upload
3. Computing Element: HTCondor and Grid jobs

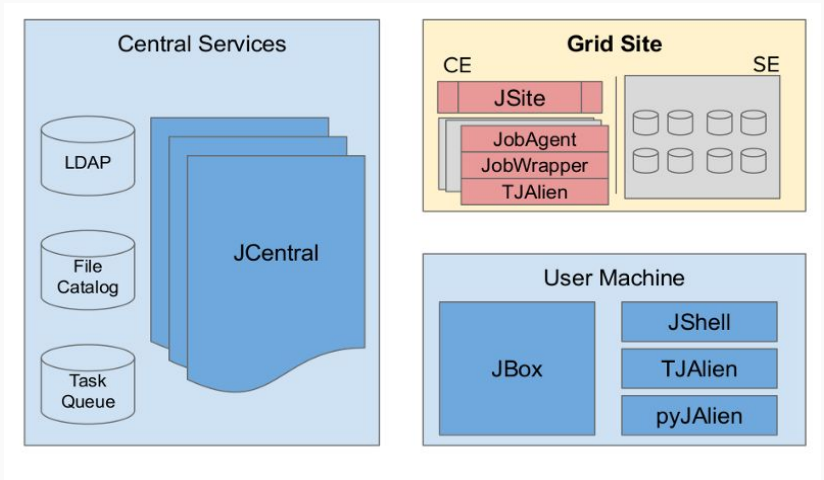


Figure 1: A high level overview of JAliEn. source: https://indico.cern.ch/event/778465/contributions/3378340/attachments/1843679/3023974/nhardi_jalien_security_model.pdf

Example: running a Grid job in JAliEn replica

Sample.jdl:
Executable = "/localhost/localdomain/user/j/jalien/testscript.sh";
Output = {stdout@disk=1};
OutputDir = "/localhost/localdomain/user/j/jalien/output_dir_new/";

Testscript.sh:
#!/bin/bash
echo "it works :)"

```
AliEn[jalien]:/localhost/localdomain/user/j/jalien/ > cp file://sample.jdl alien://sample.jdl
jobID: 1/1 >>> ERRNO/CODE/XRDSTAT 0/0/0 >>> STATUS OK >>> SPEED 6.30 KiB/s MESSAGE: [SUCCESS]

AliEn[jalien]:/localhost/localdomain/user/j/jalien/ > submit sample.jdl
Your new job ID is 1888757065

AliEn[jalien]:/localhost/localdomain/user/j/jalien/ > alien.py ps
jalien 1888757065 0 R testscript.sh

AliEn[jalien]:/localhost/localdomain/user/j/jalien/ >ls
output_dir_new/
sample.jdl
testscript.sh

AliEn[jalien]:/localhost/localdomain/user/j/jalien/ >ls output_dir_new/
stdout
AliEn[jalien]:/localhost/localdomain/user/j/jalien/output_dir_new/ >cat stdout
it works :)
```

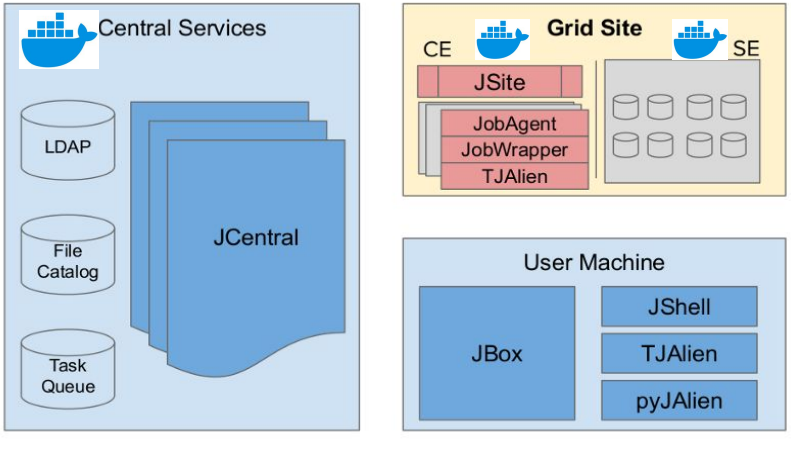
Setting up your local JAliEn replica

Jalien-setup is used to build a local setup environment for a JAliEn deployment. The deployment is automated, with easy setup and teardown via docker-compose. Also makes it easy for CI, debugging and development.

To start

- Clone JAliEn repository and compile the code to get alien.jar
- Clone the jalien-setup repository and run the replica-startup utility `jared` for local setup:
  ```
  jalien-setup/bin/jared --jar /path/to/jalien/jar --volume /path/to/jalien-volume
  ```
- jalien-volumedirectory would be created on success, with all required deployment and setup files such as config files, alien.jar, docker-compose.yml file, crednetials etc.
- Run docker-compose up
- Source env_setup.sh and then connect with any JAliEn client.

	Container Name	Comment
Central Services	JCentral-dev	JCentral, MySQL and LDAP
Storage Element	JCentral-dev-SE	XRootD based storage element
Computing Element	JCentral-dev-CE	The VOBox
	schedd	HTCondor scheduler
	worker	HTCondor worker nodes



Making things user friendly

The code and all the work done can be found here:
https://gitlab.cern.ch/jalien/jalien-setup
Since we'll be aiming to open this for potential external contributors to use. We need to focus on making debugging and development easier.

- Adding a contributor.md to to the jalien-setup repo
- Begin making the config flexible to add users, sites, CEs, SEs to JCentral on the fly.

GSoC Journey

GSoC and CERN made my entire summer. Things started off rocky, with respect to getting comfortable with the code base, and breaking down the project into achievable chunks for each phase.

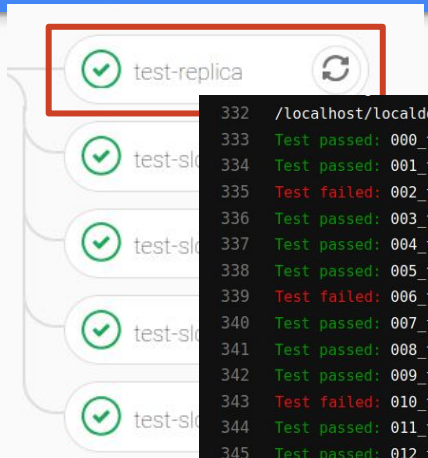
But my mentors and the JAliEn team supported me all the way! With weekly calls, daily chats and remote debugging no problem seemed too great.

Personal Thoughts

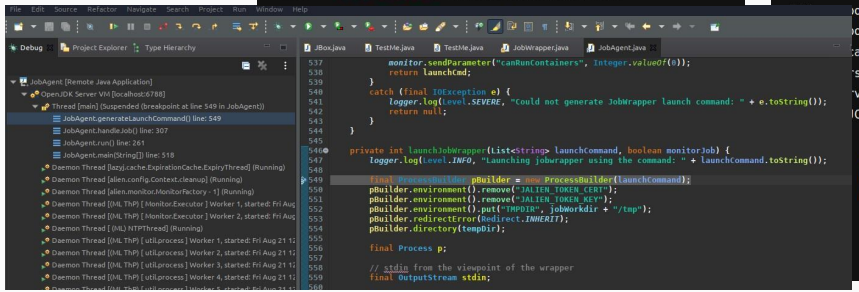
GSoC was an amazing experience, A huge thanks to everyone who was a part of it. Especially Nikola Hardi, Maksim Melnik Storetvedt, Olga Vladimirovna Datskova, Costin Grigoras, Volodymyr Yurchenko, Latchezar Betev and Adrian Sevcenco!

My mentors were always extremely friendly and approachable. They were understanding, and never enforced hard deadlines. This made the work feel comfortable, no matter how daunting it ended up being.

I also plan continuing contributing to the jalien-setup repository.




```
/localhost/localdomain/user/j/jalien/
Test passed: 000_test_first
Test passed: 001_test_TGridCollection
Test failed: 002_test_TFile_Copy
Test passed: 003_test_TJAlienResultRewriter
Test passed: 004_test_pwd_no_newline
Test passed: 005_test_TGrid_GetUser
Test failed: 006_test_TSystem_AccessPathName
Test passed: 007_test_cp_XML_to_grid
Test passed: 008_test_find_touch_xml
Test passed: 009_test_metadata_lag
Test failed: 010_test_find_tgridcollection
Test passed: 011_test_load_talicecollection
Test passed: 012_test_pwd_after_reconnect
Test passed: 013_test_find_turl_attribute
Test passed: 014_test_jbox_only
Test passed: 015_test_cmdline_exit_codes
Test passed: 016_test_cmdline_find
Test failed: 017_test_cmdline_cp
Test passed: 018_test_cmdline_cp_dir
Test failed: 019_test_job_submit
```



```
~/builds/CsZsPfW2/0/nhardi/jalien-ci/workspace/shared/globu
~/builds/CsZsPfW2/0/nhardi/jalien-ci/workspace/shared/globu
lien-ci/workspace/jalien-setup
Can't load /home/ubuntu/.rnd into RNG
140517128319424:error:2406F079:random number generator:RAND
Generating a RSA private key
.....+++++
writing new private key to 'SE_priv.pem'
-----
Signature ok
subject=C = CH, O = JAliEn, CN = TESTSE
Getting CA Private Key
~/builds/CsZsPfW2/0/nhardi/jalien-ci/workspace/shared/globu
~/builds/CsZsPfW2/0/nhardi/jalien-ci/workspace/jalien-setup
CreateConfig done
Computing Element Config done
+ cd /home/ubuntu/builds/CsZsPfW2/0/nhardi/jalien-ci/works
+ cat .env
SHARED_VOLUME=/home/ubuntu/builds/CsZsPfW2/0/nhardi/jalien
JALIEN_SETUP_DIR=/home/ubuntu/builds/CsZsPfW2/0/nhardi/jal
+ cat env_setup.sh
export vol="$(dirname $(realpath $BASH_SOURCE))"
certpath="$vol/globus"
export ALIENPY_JCENTRAL="JCentral-dev"
export X509_CERT_DIR="${certpath}/CA"
export CERT=${certpath}/user/usercert.pem
export KEY=${certpath}/user/userkey.pem
export X509_USER_CERT="${CERT}"
export X509_USER_KEY="${KEY}"
export JALIEN_TOKEN_CERT="${CERT}"
export JALIEN_TOKEN_KEY="${KEY}"
ort JALIEN_HOST="localhost"
ort JALIEN_WSPORT=8097
at docker-compose.yml
sion: "3.3"
vices:
Central-dev:
  image: jalien-base
  command: /jalien-setup/bash-setup/entrypoint.sh
  hostname: JCentral-dev
  ports:
    - 8098:8098 # Java binary serialization
    - 8097:8097 # JSON WebSocket
    - 3307:3307 # MySQL
    - 8389:8389 # LDAP
```

Thank you!