PY-MESOS

A Galaxy job runner to launch dockerized tools on top of a Mesos cluster

M. Antonacci [1], G. Donvito [1], R. Gallitelli [2], M. A. Tangaro [2], M. Tomaiuolo [1]

[1] National Institute for Nuclear Physics - Bari Via E. Orabona 4 70125 Bari Italy

[2] National Research Council - Institute of Biomembrane and Bioenergetics Via G. Amendola 165/A 70126 Bari Italy

ELIXIR All Hands 2017, 21-23 March, Rome, Italy

The large amount of biological data generated by high-throughput bioinformatic experiments requires powerful computational resources and high level software to easily access and analyse data.

Galaxy allows data analysis by integrating multiple tools through an easy to use web-based environment. By default it is designed to run jobs exploiting local resources, but it can be configured to make use of remote clusters through different plugins (named job runners).

We have developed a python job runner, named Py-Mesos, able to run Galaxy on Mesos clusters, using Chronos as job scheduler to run dockerized tools (e.g. from Docker Hub). Apache Mesos is an open-source resource manager able to run applications in distributed environments, supporting multiple computing frameworks, exploiting the largest possible collection of heterogeneous resources, being provided either by public and/or private vendors.

Apache Mesos

Apache Mesos (mesos.apache.org) is a Resource Manager used to enable different frameworks to efficiently share clusters. Then to isolate resources it leverages on modern OS container technologies, like linux cgroups and Solaris zones, for instance to limit CPU, memory and network bandwidth. A web UI allows cluster state monitoring.

Job orchestration is performed using Chronos as job scheduler, running on top of Mesos. Chronos natively supports job scheduling inside Docker containers, input data fetching, dependencies handling and job rescheduling in case of failure.

Galaxy

Galaxy is an open-source framework for bioinformatics analysis. It allows to run command line tools and complex workflows over large dataset collection through an easy-to-use web-UI, to reproduce, share and publish analysis and results.

It can be deployed on local servers, but, if properly configured, it is able to run tools on remote clusters: the Galaxy web front-end runs on a single server as usual, while tools run on cluster nodes. Currently Galaxy supports various resource managers (e.g. SLURM, TORQUE, Condor, Pulsar, etc.) using different job runners and leaving to the end-user the choice among them. Each job runner is a mediator between Galaxy and the external executor framework.

INDIGO-Datacloud

The H2020 INDIGO-Datacloud project (indigo-datacloud.eu) aims to develop an open-source computing and data platform supporting multi-disciplinary scientific communities, provisioned over public and private e-infrastructures. The first release of INDIGO software catalogue has been released in August 2016 (www.indigo-datacloud.eu/service-component).

In particular, the INDIGO platform allows to automatically instantiate a Mesos cluster using Docker containers. Moreover, the INDIGO service component CLUES is able to elastically scale cluster resources up and down, powering on/off new nodes depending on the cluster workload, making resources available only if needed.

INDIGO - DataCloud

Py-Mesos Job Runner

A new Galaxy job runner, named Py-Mesos, has been developed starting from the default one and implementing all the needed functions to retrieve cluster information and the external runner configuration, to submit, stop and recover jobs and update their status, through Chronos.

Py-Mesos interacts with Chronos via API calls, using a JSON file to setup all the required resources. In particular it integrates the information about the Docker images of the tools to be used and the Mesos cluster where the job will run. Each job is executed within a docker container. Indeed, Chronos is able to fetch the dockerized tool images automatically from Docker Hub, avoiding any tools dependency resolution issue.

Finally, we are implementing data stage in/out to copy the job's input datasets to the node and the output results back from the node, avoiding any approach requiring modifications to the remote MESOS cluster nodes (e.g. shared file system), which would be allowed only to the cluster administrator(s).

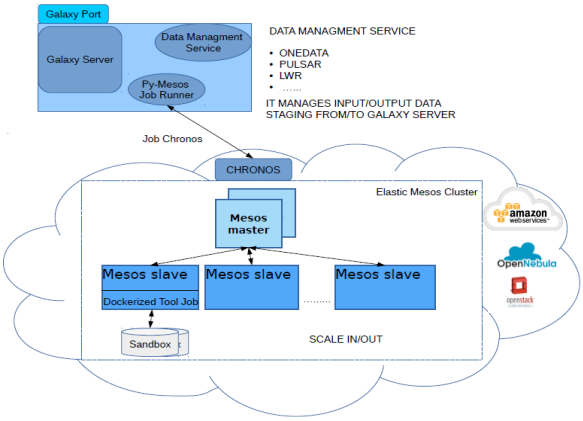

Figure 1. Py-Mesos Architecture.

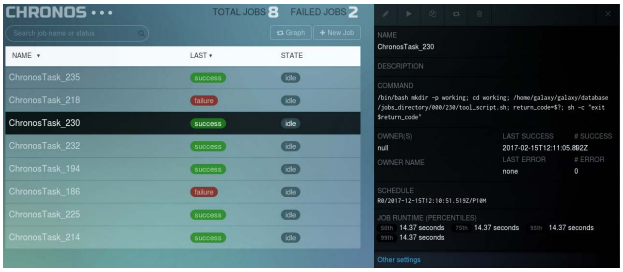

Figure2. What you see on Chronos.

Submitting Galaxy jobs to Mesos through Py-Mesos plugin allows to build a scalable, fault tolerant system for bioinformatic analyses, based on Galaxy. Since Mesos is supporting different frameworks, this solution enables Galaxy to run analyses using different cloud providers, avoiding to depend on particular configurations, e.g. OpenStack, OpenNebula public e-infrastructures or other private cloud solution like Amazon or Google.

Moreover, leveraging on the INDIGO-Datacloud platform capabilities, it is possible to deploy the whole stack (from the Galaxy framework to the Mesos cluster) in an easy and completely automatized way on top of mixed resources, bare-metal and/or public/private cloud environments.

Contact:

G. Donvito, giacinto.donvito@ba.infn.it

R. Gallitelli, rosanna.gallitelli@ba.infn.it

INFN-Bari, Via Orabona, 4, 70125, Bari (Italy)

IBBE Istituto di Biomembrane e Bioenergetica

INFN Istituto Nazionale di Fisica Nucleare

RECAS BARI