EGI Federated Cloud for calibrating and analysing Radio-Astronomy data

Susana Sanchez-Expósito [1,*], José Sabater [2], Daniele Lezzi [3], Julian Garrido [1], Lourdes Verdes-Montenegro [1], José Enrique Ruiz [1], Pablo Martín [4]. Raül Sirvent [3], Rosa M. Badia [3], Antonio Ruiz-Falcó [4]

(1) Instituto de Astrofísica de Andalucía – CSIC
(2) University of Edinburgh
(3)Barcelona Supercomputing Center
(4) Fundación de Supercomputación de Castilla y León

(*) sse@iaa.es
http://amiga.iaa.es/p/263-federated-computing.htm

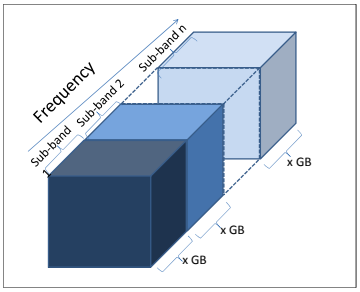
Radio-Astronomy leads the ranking of data generation in a time where all science communities are undergoing their own data deluge. The world's largest radio interferometer by far, the **Square Kilometre Array (SKA)**, will be able to reach **data rates in the exa-scale domain**. Its construction will start in 2018. In the meanwhile SKA pathfinders like **LOFAR** are currently providing astronomers with heavy and complex data volumes which need to be calibrated and analysed. The most suitable tools for doing it usually range from new software developed by the instrument consortium to well known packages developed in some cases several decades ago. The astronomers have to struggle with the short life-cycle of the former and with the difficulties to install the latter in new platforms. In this poster we show to what extent and in which way cloud infrastructures, specifically the **EGI Federated Cloud**, could ease the mentioned issues as well as how the **COMPSs** programming framework facilitates the interaction with the infrastructure and optimizes the execution of the user's code. The work presented is driven by two scientific use cases, the first one represents the first pipeline for calibrating LOFAR data in the cloud, and the second one aims to model the kinematics of galaxies in an automatic way.

Use Case 1: LOFAR data calibration

Fig.1. LOFAR datacube representation

An interferometer as LOFAR correlates the signals from several antennas, generating the so-called measurement sets. They are a kind of **datacubes** (3D data)**:** two Fourier spatial coordinate axes plus a spectral axis.

A datacube can reach several **terabytes,** depending on factors as the amount of involved antennas, the observation time, as well as the amount of observed subbands – i.e. frequency intervals-. LOFAR telescope allows **up to 488 subbands**, which can reach several GBs. **Each subband can be processed independently what allows the parallelization of the whole datacube calibration**.

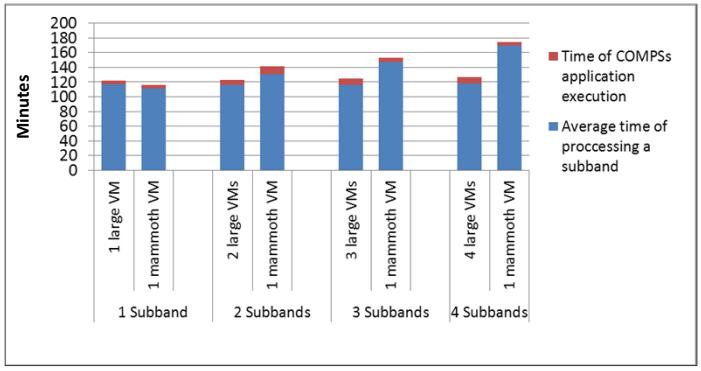



```
import subprocess
import sys
import os
from pycompss.api.task import task
from pycompss.api.parameter import *

@task( script_name = FILE)
def iter_calib(script_name):
    os.chmod(script_name,0744)
    subprocess.call(script_name)
    print "end executiong"
if __name__ == "__main__":
    args = sys.argv[1:]
    DATA_PATH=args[0]
    TEMPLATE_FILE=args[1]
    f=open(TEMPLATE_FILE,'r')
    content=f.read()
    f.close()
    list_f=os.listdir(DATA_PATH)
    for directory in list_f: # Iterate over the data inputs
        if os.path.isdir(DATA_PATH+"/"+directory):
            new_content=content.replace("INPUTDATAPATH",directory)
            script_name="job"+directory+".sh"
            f=open(script_name,"w")
            f.write(new_content)
            f.close()
            iter_calib(script_name)
```

Fig.2. COMPSs application. It iterates over the subbands, executing for each one a COMPSs task that calls the LOFAR software. Through a simple interface for describing the methods, COMPSs is able to analyse the dependencies among them, to match their requirements with the available resources and to orchestrate their execution on VMs.

Fig.3. Execution time. This figure shows the results of different tests in which the application has been configured to calibrate from 1 to 4 sub-bands, and its tasks have been distributed either on a high capacity VM (mammoth=32GB memory + 8 cores) or on several smaller VMs (large=8GB memory + 4 cores). Since each subband is processed in parallel, **the executions for calibrating one subband take approximately the same time than those for calibrating several subbands**. The results also reveal that **the execution time for the whole COMPSs application (in red) is slightly higher than for the tasks (in blue)**. Thus we can state that the time to start and contextualize the VMs is not significant. In addition, the time for the applications running on mammoth is higher than the applications whose tasks have been distributed on smaller VMs. This would mean that **distributing the tasks among several small VMs is more efficient than gathering them in a VM with high memory capacity and amount of cores.**



Use Case 2: Kinematical modelling of galaxies

After the data calibration and reduction process, the next step is to analyse the science-ready data. In particular, in this use case, an input parameter space is explored to get different kinematical models of galaxies.

Unlike the previous use case, this **analysis requires to work on the whole datacube**. Therefore the **application parallelization** can not be achieved by dividing the datacube, but by **dividing the parameter space** to be explored.

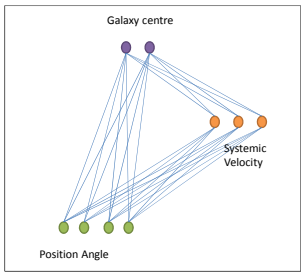
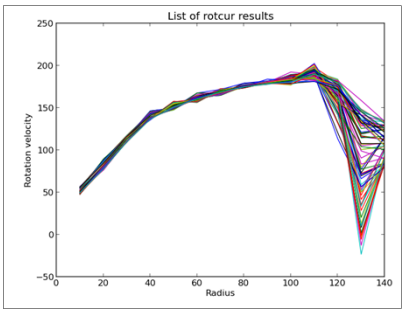


Fig. 4 represents a 3-parameter space with 2,3 and 4 possible values respectively, resulting in 24 modelling tasks.

Fig. 5 shows 81 possible rotation curves of a galaxy, generated by exploring a 4-parameter space with 3 possible values for each one. These rotation would generate 81 different kinematical models of the galaxy.



User Experience

User software deployment	• The AppDB **Virtual Appliances facilitate the installation of the user application on Fedcloud**. The users can directly install their software on a running Fedcloud VM, which has been previously started as an instance of a Virtual Appliance.	• The process of creation and deployment of the Virtual Appliances in the AppDB is not automatic. **There should be a way to let users create snapshots**. • **New software releases imply starting again the deployment process**. Last year LOFAR software team published 8 releases.
Scalability / performance	• The **computing power** provided by the different sites federated in Fedcloud allows to configure virtual machines with different capabilities (memory+cpu) in order to match with the specific needs of the use case.	• **Lack of consolidated solutions for federating storage.** The user data are too large to be stored in the VM images. They should be stored in volumes easily mountable from several VMs and synchronized across different sites.
User interaction	• The COMPSs framework facilitates the porting and deployment (through the PMES service) of the application. COMPSs allows the users to program their applications sequentially, in a transparent way by taking care of their parallelization as well as of the VMs orchestration for their execution.	• **Friendly tools for working as interface for the command line client** would facilitate the first steps in using Fedcloud: to configure the VMs and install the software, manage the proxy certificates and propagate the new releases of Virtual Appliances to the Cloud Marketplace.

Background image credits: skatelescope.org