EOSC NORDIC

DRAFT NOT YET APPROVED BY THE EUROPEAN COMMISSION

D5.2 Cross-borders computing through portals

Author(s)	**Kessy Abarenkov (ETAIS/UT), Anne Fouilloux (UiO)**
Status	Draft
Version	1.0
Date	26 February 2021

Document identifier:	
Deliverable lead	ETAIS/UT
Related work package	WP5
Author(s)	Kessy Abarenkov (ETAIS/UT), Anne Fouilloux (UiO)
Contributor(s)	Matthias Obst (UGOT)
Due date	31/03/2021
Actual submission date	26/02/2021
Reviewed by	Adil Hasan (Sigma2), Helmut Neukirchen (UICE), Ebba Þóra Hvannberg (UICE)
Approved by	
Dissemination level	Public
Website	https://www.eosc-nordic.eu/
Call	H2020-INFRAEOSC-2018-3
Project Number	857652
Start date of Project	01/09/2019
Duration	36 months
License	Creative Commons CC-BY 4.0
Keywords	Cross-borders computing, portals, PlutoF, Galaxy

EOSC NORDIC

DRAFT NOT YET APPROVED BY THE EUROPEAN COMMISSION

Abstract:

This deliverable aims at enhancing the cross-border computing solutions of the Nordic and Baltic digital services relevant to EOSC by the example of two community specific portals in the fields of biodiversity and climate. It looks at the availability of HPC resources, packaging of the tools, setup automation, and potential blockers and sustainability issues related to the integration of community specific portals with the Nordic HPC clusters.

Table of Contents

Table of Abbreviations 4

1. Introduction 5

2. Use cases 5

2.1. Biodiversity Pilot 5
2.1.1. Background, scientific objectives and state of the art 5
2.1.2. Resources 6
2.1.3. Technical solutions 7
2.1.3.1. Front-end 7
2.1.3.2. Packaging 8
2.1.3.3. Setup automation 8
2.1.3.4. Potential blockers and sustainability issues 8
2.1.4. Procedures to apply for HPC resourcesing 9
2.1.5. Take-up 11
2.1.6. Next steps 11
2.2. Climate Pilot 12
2.2.1 Background, scientific objectives and state of the art 12
2.2.2 Galaxy Platform for Climate Analysis 13
2.2.3 Resources 13
2.2.4 Technical solutions 14
2.2.4.1. Front-end 14
2.2.4.2. Packaging 15
2.2.4.3. Setup automation 16
2.2.4.4. Potential blockers and sustainability issues 16
2.2.4 Procedures to apply for HPC resourcesing 16
2.2.5 Implemented workflows 17
2.2.5 Take-up 19
2.2.6 Next steps20
3. Conclusions 20
4. Supplementary Items 21
4.1. Supplementary Item S1 21

Table of Abbreviations

Table 1: Abbreviations appearing in the document

Abbreviation	**Explanation**
AMQP	Advanced Message Queuing Protocol
ARMS	Artificial Reef Monitoring Structures program
API	Application Programming Interface
BLAST	Basic Local Alignment Search Tool
CLM	Community Land Model

CMIP	Coupled Model Intercomparison Project
DNA	Deoxyribonucleic acid
eDNA	environmental DNA
EOSC	European Open Science Cloud
ESGF	Earth System Grid Federation
ETAIS	Estonian Scientific Computing Infrastructure
FATES	Functionally Assembled Terrestrial Ecosystem Simulator
GSSAPI	Generic Security Service Application Program Interface
HPC	High Performance Computing
HPC2N	High Performance Computing Center North
ITS	Internal Transcribed Spacer
ITSx	Software for detection and extraction of ITS1 and ITS2 from ribosomal ITS sequences
NICPB	National Institute of Chemical Physics and Biophysics (Estonia)
NIRD	National Infrastructure for Research Data (Norway)
NREC	Norwegian Research and Education Cloud
PI	Primary Investigator
PROTAX-fungi	Tool for taxonomic placement of fungal ITS sequences
SH	Species Hypothesis
SNIC	Swedish National Infrastructure for Computing
SSC	SNIC Science Cloud
UGOT	University of Gothenburg
UT	University of Tartu
VM	Virtual Machine
VRE	Virtual Research Environment

1. Introduction

The aim of subtask T5.2.1 is to facilitate data pre- and post-processing workflows (High Performance Computing (HPC) or High Throughput Computing) on distributed data and computing resources across borders by enabling community specific or thematic portals, such as PlutoF and Galaxy flavours, traditionally designed to submit jobs on local clusters, to allow scheduling of jobs on remote resources.

This deliverable presents generic modules and solutions, i.e. independent from the architecture and the technology of any given portal, designed to support submission of computations from community specific portals to a variety of scheduling systems. These modules allow the integration between the Galaxy and the PlutoF portals with the different schedulers present in Nordic HPC clusters.

2. Use cases

There are two specific pilots where we applied and tested the cross-borders computing in subtask T5.2.1:

1. **Biodiversity Pilot** where digital services developed by the UNITE Community to support global species discovery from environmental DNA (eDNA) are provided by the PlutoF platform with the possibility to submit HPC jobs to the University of Tartu Rocket cluster (UT Rocket). Support for submitting jobs to the other Nordic HPC clusters was added under subtask T5.2.1 and is described in Section 2.1 of this document.

2. **Climate Pilot** where digital services developed by the Nordic Earth System Modelling Community Hub to support Climate Analysis are provided by the Galaxy platform with the possibility to submit jobs on remote resources with Pulsar.

These use cases are discussed in the remainder of this section.

2.1. Biodiversity Pilot

2.1.1. Background, scientific objectives and state of the art

The main scientific goal of this subtask for Biodiversity Pilot is to support researchers in the fields of molecular ecology, taxonomy, biodiversity, etc. with species discovery from their eDNA samples, and unambiguous and traceable communication of these taxa. Analysis tools for a large amount of molecular sequence data can require a significant amount of HPC resources and some knowledge in bioinformatics and/or information technology for the setup in a local HPC cluster by the user.

PlutoF makes UNITE digital services available for the UNITE user community by providing a simple front-end solution as an alternative to a command line interface. PlutoF is an online workbench and computing service provider for biology and related disciplines. It was initially developed for describing biodiversity as well as storing and working with related data. Registered users can enter and manage a wide range of data starting from nature observations and study areas up to molecular systematics, taxonomy and ecology. It also features an analysis module by providing digital services for molecular sequence identification and species discovery from eDNA samples.

PlutoF handles the user management, logging and storing analysis runs and data files, while executing remote jobs in HPC clusters. To be able to provide more resources based on the individual user's needs, the PlutoF platform could integrate HPC resources users have access to, thus allowing more analysis to be submitted by these users. Historically, the UNITE user community includes a high number of researchers from Nordic countries, thus making the EOSC-Nordic project a good opportunity for enhancing the cross-borders computing solution of their services.

The main goals in the Biodiversity Pilot are the following:

1. Package PlutoF analysis services in a way that allows service provider to easily build, transfer and run these services independent of the software available in remote HPC clusters;

2. Allow service provider to send PlutoF analysis jobs to remote EOSC-Nordic HPC clusters given that there is a user community with access to the resources in this cluster;
3. Work out recommended procedure on how users can apply for HPC resources in PlutoF;
4. Work out how to access European Open Science Cloud (EOSC) HPC resources from PlutoF in a standard, consistent, simple and automated way.

Section 2.1.3 covers how goals 1) and 2) are reached, and Section 2.1.4 provides the procedures needed for goal 3). The steps from Section 2.1 as a whole accomplish goal 4).

2.1.2. Resources

Based on the PlutoF platform user community, we identified two potential HPC providers to test our cross-borders computing workflow implementation: 1) Swedish National Infrastructure for Computing (SNIC) for the Swedish user community, specifically services for the marine eDNA collected by the European Artificial Reef Monitoring Structures (ARMS) program , and 2) NICPB HPC cluster for the Estonian user community provided by the Estonian Scientific Computing Infrastructure (ETAIS).

The NICPB HPC cluster was chosen to test the implementation in an environment very similar to the one where PlutoF services were already running (UT Rocket), i.e. a robot account (meaning an authenticated shared user account which is used to submit jobs by a fixed set of users identified at the portal) was available there, as well as the SLURM workload manager and a Singularity module. It was also useful for starting PlutoF front- and back-end developments needed to be able to switch between different HPC clusters at the user level, before access to other EOSC HPC resources was granted.

SNIC was chosen as a resource provider that was accessible to our collaborator Matthias Obst from the University of Gothenburg (UGOT) who also represents users of the Swedish Biodiversity Data Infrastructure (SBDI) as well as the European ARMS program.

As part of EOSC-Nordic, UGOT started a pilot in the field of ecological and biodiversity science to test interoperability between Swedish and Estonian service providers. The goal of the pilot is to direct computational jobs submitted by Swedish users on PlutoF to an HPC resource on SNIC. UGOT currently assembles documentation on how to couple PlutoF with SNIC clusters. Technically, it is possible to seamlessly link HPC resources across both countries by using the existing SLURM scheduling system in the HPC cluster.

There are however two requirements that need to be solved for using robot accounts. First, a log functionality needs to be developed, allowing the activity of any user to be traced on the Swedish side. Second, SNIC does not allow robot accounts at the moment and hence we currently only apply the robot account with a single user. In order to scale up this solution, SNIC needs to consent to the application of robot accounts.

For testing purposes, UGOT set up small projects at the SNIC Science Cloud (SSC) and SNIC High Performance Computing Center North (HPC2N), which are connected from the Estonian PlutoF resource via ssh. The account will be used to test popular PlutoF services (including Basic Local Alignment Search Tool (BLAST) and Species Hypothesis (SH) matching analysis) for a massive data set from the Artificial Reef Monitoring Systems in the Baltic and the North Sea (Obst et al 2020). This data set contains sequence profiles, images, biological measurements, and extensive metadata from a Genomic observatory network deployed in Sweden, Denmark, Finland, and Norway.

It was important to the PlutoF team who implemented the Biodiversity Pilot cross-border solution that potential HPC providers have a user policy in place that is in accordance with the requirements of the

service provided by the PlutoF platform. PlutoF team also had a list of prerequisites for the resource providers: 1) SLURM workload manager, 2) Singularity module, 3) robot account for submitting jobs allowed. Not all candidate HPC providers were able to meet these requirements. For example, the SNIC User policy does not allow submitting jobs as robot user, although such a case is allowed in the SSC. SSC, on the other hand, comes as a Virtual Machine (VM), however without any software installed. In the latter case, the automated process of setting up a Virtual Research Environment (VRE) to install required software was needed.

2.1.3. Technical solutions

2.1.3.1. Front-end

PlutoF handles the user management, logging and storing analysis runs and data files, while executing remote jobs in HPC clusters via ssh. Analysis data files and SLURM scripts are copied (via ssh, scp, rsync through ssh tunnels) to the remote HPC cluster, jobs are started and executed remotely, and analysis results are fetched by PlutoF once jobs are finished. Users are notified upon job completion via email.

As part of this subtask, functionality for sending jobs to different remote HPC clusters instead of one local cluster (UT Rocket) was added. PlutoF developments included support for switching HPC resource providers at the user level (based on the user's preference and availability of HPC providers), and setting proper access parameters when submitting analysis jobs to and receiving analysis results from a remote HPC cluster.

2.1.3.2. Packaging

For easier and more automated building and installation of the software, PlutoF digital services were packaged into Singularity containers with container building code and automated setup scripts available in GitHub (for links of their individual GitHub repositories, see the footnotes in Section 2.1.4).

2.1.3.3. Setup automation

The process of wrapping PlutoF digital services into Singularity containers was documented and published as GitHub repositories, and can be used to automate the installation process cutting down the installation time from several hours to approximately 10 minutes in total. The Biodiversity Pilot includes four digital services to support the eDNA based species discovery:

1. ITSx (detection and extraction of ITS1 and ITS2 from ribosomal ITS sequences),
2. PROTAX-fungi (taxonomic placement of fungal ITS sequences),
3. massBLASTer and
4. SH matching analysis (ver. 1.0).

For setting up VRE in SSC and similar environments, we created a workflow of steps (see Supplementary Item S1 in Section 4.1 of this document) that we are planning to wrap into a Bash script for more convenient use. Here we decided that installing the SLURM scheduler to operate on only one Virtual Machine (VM) would not be practical, and we chose to implement a solution for running scheduled analyses using the task-spooler tool instead. Once the needs for sharing resources within the workgroup increase, it might be worth reconsidering the choice of the scheduling system.

2.1.3.4. Potential blockers and sustainability issues

Potential blockers and sustainability issues identified during the work in this subtask are:

1. An ssh key based authentication was not supported by all resource providers, e.g. SNIC HPC2N supports only Kerberos/GSSAPI based authentication which we implemented in our workflow.
2. HPC service access is normally provided for a certain time period, after which the user either has to a) go through the process of applying for resources again and write to PlutoF support for replacing outdated HPC provider info with new one, or b) write to PlutoF support for replacing outdated HPC provider info with any other HPC resource available in PlutoF as default.
3. Robot user accounts are often not allowed. This was the case with SNIC HPC2N. Together with the SNIC representatives, we came to a solution where using a robot account was allowed for testing purposes in the case where the account was linked to one specific PlutoF user (e.g. the primary investigator (PI) of the ARMS project).
4. Constant maintenance (e.g. software and operating system updates, Singularity container updates, resolving VM service interruptions and unexpected failures) of the VMs (for SSC and similar cases) that the PlutoF team sets up requires additional work and resources from the team.
5. Access to Nordic HPC resources for all PlutoF platform users would be difficult to implement – access in EOSC-Nordic HPC clusters requires belonging to an HPC project which has been given access with limited resource quota. Collaboration with the Puhuri project could be considered for granting external users access/quota in Nordic HPC systems.

2.1.4. Procedures to apply for HPC resourcesing

This section describes the procedures on how users can apply for HPC resources in PlutoF (goal 3) listed at the beginning of Section 2.1).

There are two implementation case scenarios (Figure 1) depending on whether the prerequisites described in Section 2.1.2. are fulfilled or not Accordingly, there are two different procedures on how users can apply for HPC resources in PlutoF:

1. **Case scenario 1 "cloud"** when linking HPC VM (current implementation platform: SSC):
   - A project's PI requests access to create new VMs in the cloud of their choice (e.g. SSC).
   - The PI grants the PlutoF system (robot user account) access to the given VM.
   - The PlutoF team configures new VM (incl. VM setup) for the PlutoF system as a new HPC resource option.
   - In case of other project members who need access to these resources in PlutoF, the PI should email their names to support@plutof.ut.ee.

2. **Case scenario 2 "bare-metal HPC"** when linking an HPC cluster with an existing SLURM scheduling system and Singularity module (current implementation platforms: UT Rocket and NICPB HPC clusters in ETAIS, SNIC HPC2N), then
   - a project's PI grants the PlutoF system (robot user account) access to an existing HPC instance with SLURM scheduling system and Singularity module available.
   - PlutoF team sets up Singularity containers.
   - PlutoF team configures the new instance for the PlutoF system as a new HPC resource option.

- In case of other project members who need access to these resources in PlutoF, the PI should email their names to support@plutof.ut.ee.

An overview of the computing solution for the PlutoF platform before and after the work conducted under subtask T5.2.1 is presented in two diagrams: The diagram in Figure 2 shows that jobs were in the past submitted to only one HPC cluster using ssh key based authentication, HPC environment settings were hardcoded in PlutoF back-end code, tools were not packaged, and needed to be set up and updated manually in the HPC cluster. The diagram in Figure 1 shows that jobs can now be sent to different HPC clusters and HPC VMs using either ssh key based or Kerberos/GSSAPI based authentication, HPC resources can be set at the user level, all tools are packaged into Singularity containers and automated setup can be done using Bash scripts.

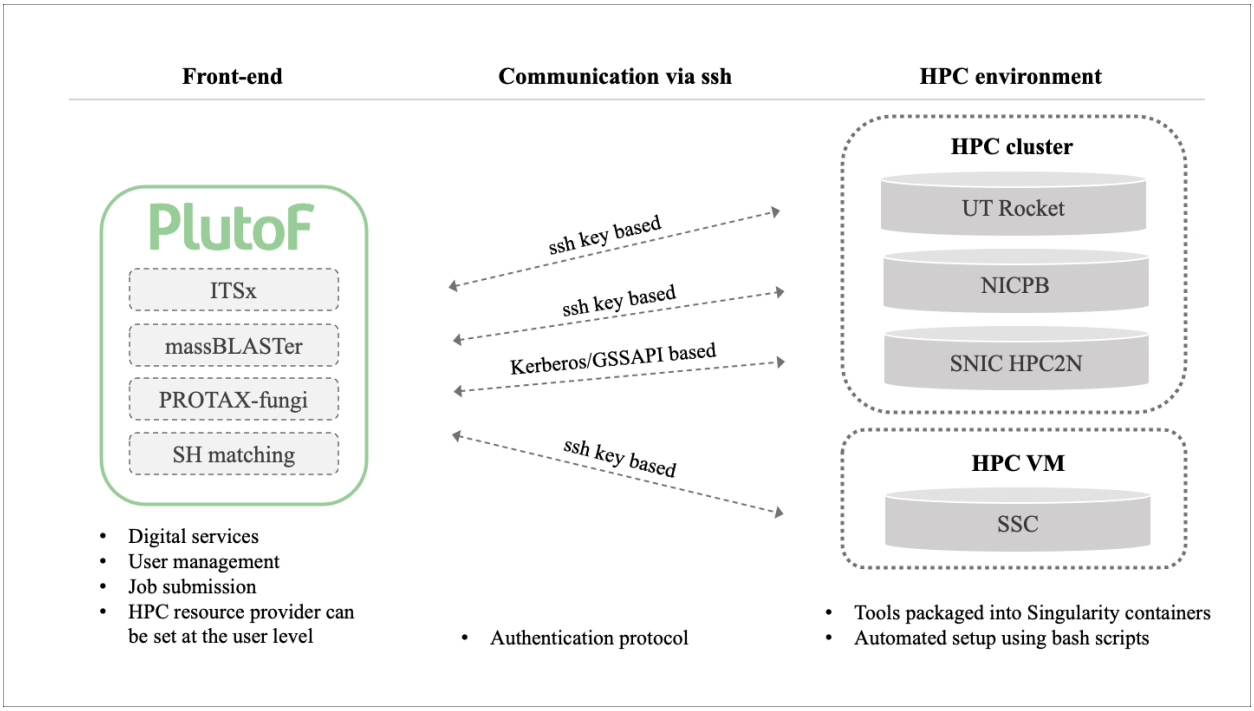

Figure 1. Diagram illustrating the cross-borders computing solution for PlutoF platform.

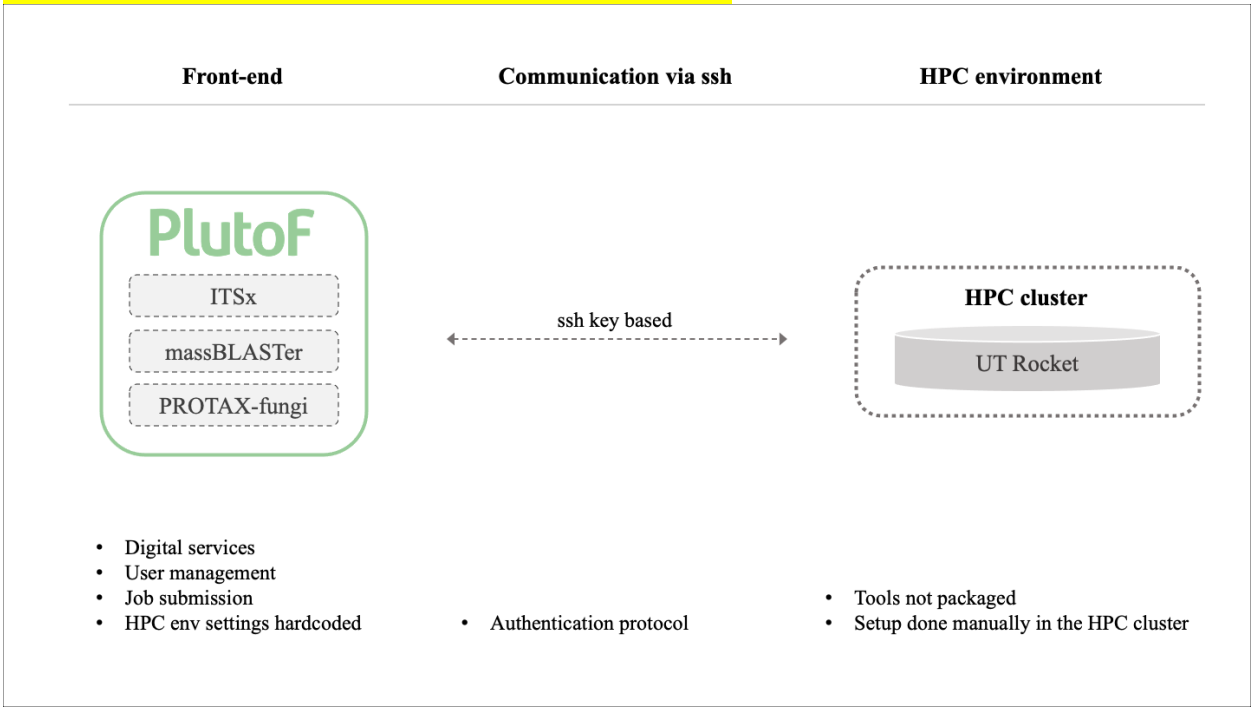

Figure 2. Diagram illustrating the computing solution for PlutoF platform before the work under the subtask T5.2.1 was conducted.

2.1.5. Take-up

Since May, 2020 when UNITE services were moved to Singularity containers, 900 analysis runs by 129 users (data from 13.02.2021) have been started in PlutoF. The new service "SH matching analysis (ver. 1.0)", created as part of the subtask T5.2.1, has been used 56 times by 29 users (data from 13.02.2021).

In April 2021, a PhD course linked to an open workshop about building the forest biodiversity open data services will be organised by the NEFOM network. UNITE digital services will be presented at the workshop and taught during the course. A scientific paper on the UNITE digital services is expected to follow.

2.1.6. Next steps

In this subtask, we successfully integrated remote Nordic HPC resources with the PlutoF platform analysis module thus providing a test bed for the ARMS project to continue with testing the two different solutions (SSC, SNIC HPC2N) for improving the tools and adjusting environment parameters according to the users' needs.

We worked out recommended procedures on how users can apply for HPC resources in PlutoF and how to set up the environment at other similar HPC providers. As a next step, we will advertise the cross-borders solution and the possibility of linking new HPC resources to PlutoF using online media channels and via scientific articles. In addition, we will document and standardise the recommended process of designing and deploying new tools to PlutoF using the same workflow (for packaging and setup scripts) as for the tools included in this subtask.

We plan to organise two PhD courses per year to present and teach the tools made available through PlutoF as EOSC-Nordic services. This also includes writing user manuals and continuous improvement of the tools.

Although we successfully coupled PlutoF platform with the SNIC HPC cluster using a robot account, we have only implemented a test solution where we apply a robot account for one single user. This process needs to be elaborated further to become an accepted and sustainable solution by SNIC.

As part of this subtask, we are also planning to add UNITE digital services provided by the PlutoF platform to the EOSC Nordic service portfolio, and add the PlutoF platform as an EOSC service.

2.2. Climate Pilot

The main technical goals in the Climate Pilot were the following:

1. Package Climate tools following the EOSC-Life tool roadmap and develop the corresponding Galaxy tools;
2. Allow Climate analysis jobs to be sent to remote EOSC-Nordic compute resources, including HPC resources (for the latter, a user community with access to resources in the respective HPC cluster may be necessary);
3. Work out recommended procedures on how EOSC HPC resources can be added in Galaxy.

Section 2.2.4.2 covers how goal 1) is reached while Section 2.2.5.1 covers how goal 2) is reached by explaining how Galaxy sends jobs to remote compute resources with Pulsar. Section 2.2.5.2 provides the procedures needed for goal 3). Finally, Section 2.2.6 shows how scientists made use of the platform to compose scientific workflows for predicting the fate of lowland invading plants in the Nordics and studying how their traits (e.g. height) impact their growth. The results of the simulations are then compared with observations collected by ecologists during dedicated field experiments in Norway.

2.2.1 Background, scientific objectives and state of the art

The main scientific goal of this task is to support field ecologists, environmental scientists, climate modelers and biologists working on improving climate models by better representing terrestrial ecosystems at high latitudes. Running climate models require the usage of a large amount of national HPC resources that favors silo working. Researchers in the Nordics need to have a common platform for running these complex climate models, comparing with observations and more importantly for live-sharing their research work.

Choosing the Galaxy platform has a number of advantages:

- It is a world-wide sustainable project that is driven by community needs
- Any Galaxy tool can be deployed in any Galaxy instance
- The 3 UseGalaxy servers (UseGalaxy.eu, UseGalaxy.org and UseGalaxy.org.au) are freely accessible and implement a common core set of tools
- The European Galaxy server is part of the European Open Science Cloud marketplace
- Cross border computing is already possible: Galaxy can send jobs to a remote Pulsar server but this has not been done with bare-metal HPCs.

Both Finland and Norway are providing cloud computing resources as Pulsar nodes to the European Galaxy instance (http://usegalaxy.eu). Therefore introducing Climate tools in Galaxy will allow cross border computing using cloud computing, including HPC cloud and one of the objectives will be to investigate the procedures to allow climate researchers to use bare-metal HPCs from Galaxy.

2.2.2 Galaxy Platform for Climate Analysis

Galaxy is a platform for data-intensive scientific computations, primarily used via a web-interface but also accessible via command line. It is built with Python and is highly portable and extendable. The usage of Galaxy is based around community-developed tools that can process and work with many different kinds of data. These tools can process data with user-defined parameters and are combined into workflows – a sequence of tasks that further process the input in an automated way. An administrator can customize Galaxy to their needs by installing all the necessary tools via an open repository called “the Galaxy ToolShed” ("ToolShed"). Anyone can register new tools to the ToolShed.

The Galaxy Europe instance is a service (Technology readiness level 9) that is already listed in the EOSC Marketplace and Galaxy Climate Europe is a scientific community flavour of the same instance but with Galaxy tools for Climate Analysis. Many included tools are existing Galaxy tools (machine learning, text manipulation, ecological tools) developed and maintained by existing communities, but NordicESMHub also develops and maintains specific Galaxy Climate tools .

2.2.3 Resources

Pulsar is the Galaxy Project’s remote job running system. It is a Python server application that can accept jobs from a Galaxy server, submit them to a local resource and then send the results back to the originating Galaxy server. This approach is very flexible and makes it easy to add new resources. In the context of climate modelling the following resources are foreseen:

1. **“Cloud computing”**: making climate tools available in Galaxy Europe gives access to available Pulsar nodes connected to the Galaxy Europe instance (shown in Figure 3); for instance, cloud computing resources from CSC cPouta. We can also envisage to add additional cloud computing resources from the Norwegian Research and Education Cloud (NREC), including its HPC-cloud.
2. **“Bare-metal HPC”**: for running large Earth System Model simulations, bare-metal HPC is usually required. However, even though it may not be “technically” difficult to add bare-metal HPC to a Galaxy instance, being able to have proper accounting and to authorize a robot user to access HPC are management challenges that still need to be solved.

Figure 3 shows the countries having currently Pulsar nodes (including both cloud and bare-metal computational resources) deployed on the European Galaxy portal.

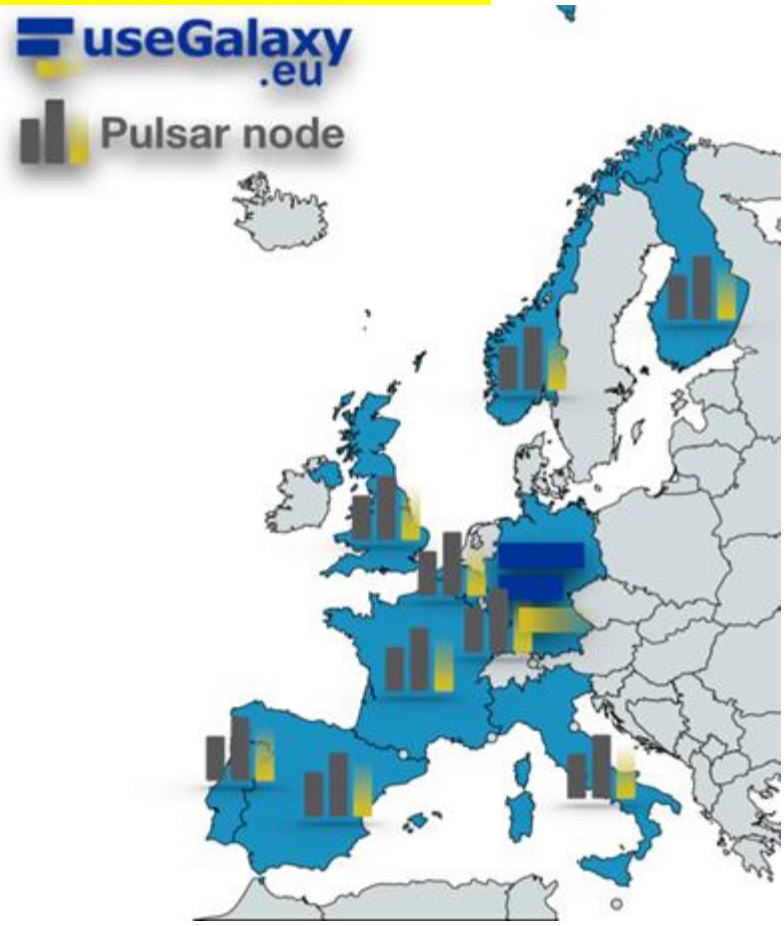

Figure 3. Pulsar nodes deployed on the Galaxy Europe portal (status taken on 01/12/2020).

2.2.4 Technical solutions

The following work was done in this subtask to reach the goals 1) to 2) listed at the beginning of Section 2.2.

2.2.4.1. Front-end

Cross-borders computing through Galaxy

As already mentioned, Galaxy uses Pulsar for executing remote jobs. Pulsar is a server written in Python that allows Galaxy instances to execute jobs remotely (there is no need to have a shared file system). A Galaxy instance sends all the data necessary to execute a job to Pulsar which handles the part of installing and preparing all the tools (also called "staging"), scheduling the jobs, etc. After the computations have been completed, the results are sent back to Galaxy Climate.

In the Galaxy Climate instance, Pulsar uses an AMQP (Advanced Message Queuing Protocol) server for the communication between Galaxy and Pulsar: for a new job to be scheduled, Galaxy publishes a message to a setup-queue with some basic information. Pulsar consumes the messages in the queue one after another. It fetches all the necessary data from Galaxy, keeps Galaxy up-to-date using an additional status

queue and at the end, sends the result files back to the main Galaxy instance. On the backend side, Pulsar (like Galaxy) can use various job-managers (in particular SLURM) and other methods for the actual computation.

2.2.4.2. Packaging

In order to enable FAIR climate data and tools that can be deployed on different infrastructures (personal computer, cloud, HPC) and can reproduce scientific results independently of the target platform, the EOSC-Nordic Climate use case follows EOSC-Life tool development best practices, that is:

- Use of Conda (a cross-platform package and environment manager) to install and manage software and their dependencies: The first step to getting a climate tool deployed into a Galaxy instance is to develop a Conda package for it. Conda is the *de facto* standard in many different communities to deploy software easily and reproducibly. The European Galaxy team is heavily involved in the conda-forge and Bioconda projects: we got technical support from the European Galaxy team to package our climate tool with conda.
- The second step is to create the Galaxy wrapper: A Galaxy wrapper is a formal description of all inputs, outputs, and parameters of a tool, so that Galaxy can generate a GUI out of it and subsequently a command to send to the cluster. All the Galaxy climate tool wrappers are published in the Galaxy toolShed under the "Climate Analysis" category and maintained on Github at https://github.com/NordicESMhub/galaxy-tools.
- Use container technology (Docker and Singularity): containers are automatically built for each available climate tool: a bot automatically creates (Bio)Containers (Docker, rkt and Singularity) by tracking all Galaxy tools to ensure that a container exists for each tool.

At the beginning of the project, experience with these tools was limited within EOSC-Nordic and to fulfill our goal, a scientific collaboration agreement between EOSC-Life and EOSC-Nordic was put in place.

The results of this collaboration are:

- Development of Climate Galaxy tools following the Galaxy IUC standards and best practices.
- Automated management via GitHub Actions configurations in order to continually synchronizing our Climate Galaxy tool GitHub repository with the Galaxy ToolShed components. You can view these in https://github.com/NordicESMhub/galaxy-tools/tree/master/.github/workflows

2.2.4.3. Setup automation

Once a climate tool is available in the Galaxy ToolShed (and corresponding containers created), it can be installed on any Galaxy instance. Request for installation on the European Climate Galaxy instance is done through a Pull Request to the Github repository https://github.com/usegalaxy-eu/usegalaxy-eu-tools: new tools and tool upgrades for the Climate Community are managed in the climate.yaml file.

2.2.4.4. Potential blockers and sustainability issues

Potential blockers and sustainability issues identified during the work in this subtask are:

1. Researchers who were already familiar with numerical modelling, Galaxy, Jupyter and similar tools seemed to find the platform easy to use and were quickly able to exchange workflows with their

colleagues. However for others it is paramount to organize targeted training (with associated high-quality training materials).

2. Exchanging data and histories with colleagues using different Galaxy instances can be cumbersome, especially when data is large (which is typically the case for Climate analysis). Being able to access sufficient storage is therefore necessary (but not the case on the current European Climate Galaxy instance). For this reason, EOSC-Nordic investigates the use of EOSC B2SAFE services.
3. Being able to add bare-metal HPC resources to a Galaxy instance (as a new Pulsar node) has been an obstacle and possible solutions are still under investigation. A long-term and sustainable approach needs to be defined.

2.2.4 Procedures to apply for HPC resourcesing

This section describes the procedures required to add HPC resources to a Galaxy instance:

- Case scenario 1 "HPC-cloud": all national cloud providers in the Nordic require a project's PI to request cloud computing but this is usually a very light weight procedure. The PI needs to briefly state the size (number of processors, memory) and justify the needs and there is usually a cost that needs to be covered by the PI or its institution (bill often sent on a yearly basis). The procedure is very much similar to a commercial cloud provider but depending on the usage the cost associated may be much lower. Once access is granted, the PI is on its own and needs to install and configure himself/herself a Pulsar server on the remote HPC cloud machine. This approach does not seem reasonable as PIs are usually researchers without the required skills to install Pulsar. One possibility could be to allow PIs to directly "order" a pulsar node. The second difficulty is then to ask a Galaxy administrator to configure the Galaxy instance to use this Pulsar node as a job destination (it is very flexible in Galaxy and can be configured for a given tool, a given user or group of users, etc.).
- Case scenario 2 "bare-metal HPC": the procedure to get access to bare-metal HPC is usually stricter with both scientific & technical allocation committees granting HPC access to PIs. In some Nordic countries (for instance in Norway), large national allocations of HPC resources are not free but PIs need to add the cost in their research projects. All climate scientists have (or can easily get) access to HPC resources in their respective country and are usually familiar with the procedure. One remaining obstacle is related to the current allocation process (per PIs and not per portal) and the need to submit jobs to HPC from robot accounts.

For the reasons detailed above, no HPC resources were used for the Climate use case and cross border computing has been achieved with existing cloud computing resources.

2.2.5 Implemented workflows

This section details what has been achieved from a user point of view e.g. what has been offered to scientists.

The European Galaxy portal has been used to run the Functionally Assembled Terrestrial Ecosystem Simulator (FATES) with the Community Land Model (CLM) as a host model. Researchers in Norway (Oslo and Bergen) prepare their simulations (by selecting input data and model parameters) and submit them via the Galaxy portal (see Figure 4).

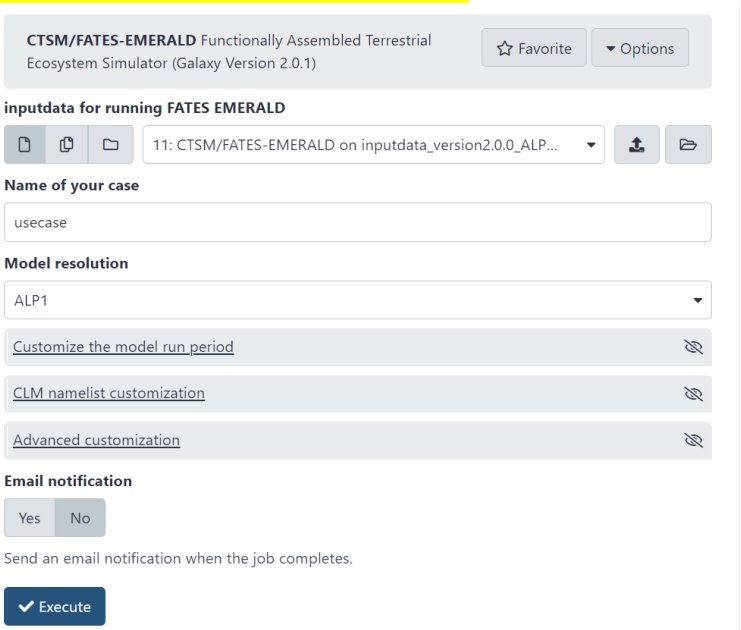

Figure 4. CLM-FATES Galaxy web user interface. Advanced settings (hidden by default) allow users to fully customize their run.

CLM-FATES jobs are dispatched by Galaxy on any Pulsar nodes, and once the results are sent back to Galaxy Europe, users can analyse the results, for instance with Galaxy Pangeo JupyterLab (see Figure 5).

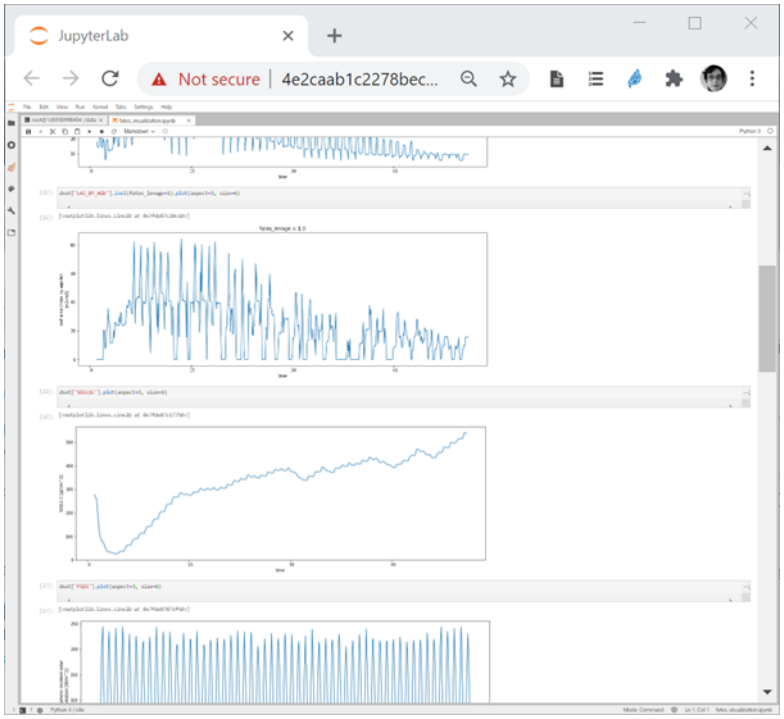

Figure 5. JupyterLab in Galaxy for analysing CLM-FATES model outputs

The user also has the possibility to create more complex workflows (see Figure 6), for instance, by selecting different site locations in Norway, and all the simulations (and data analysis) will be automatically dispatched on available Pulsar nodes.

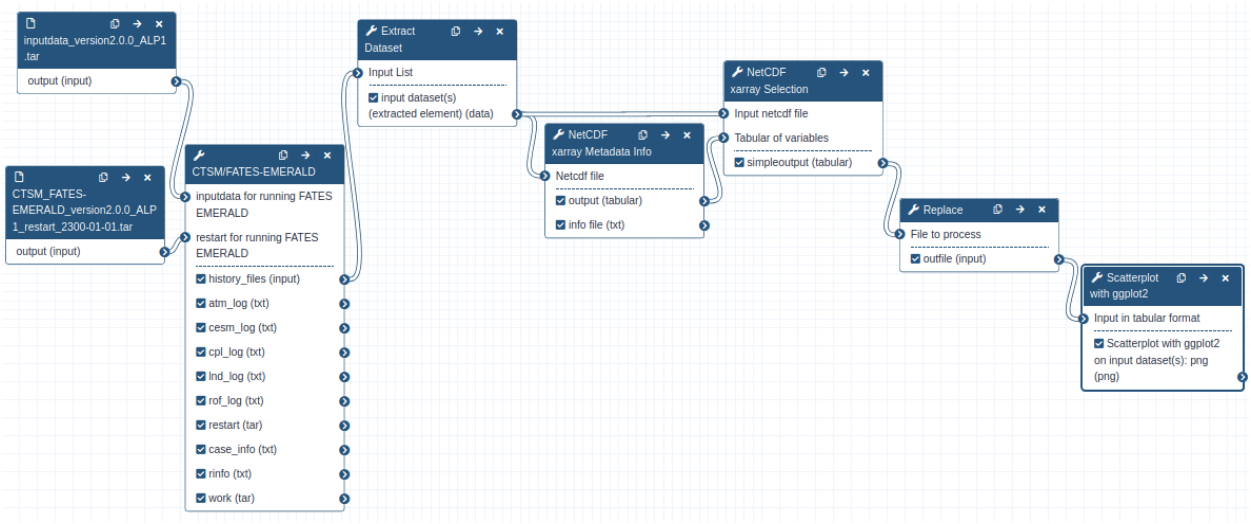

Figure 6. Workflow for running CLM-FATES in Galaxy for a single point location (ALP1) on the Norwegian alpine tundra ecosystem (Latitude: 61.0243N, Longitude: 8.12343E, Elevation: 1208 m).

Finally, the researchers can publish their workflows on WorkflowHub with a unique identifier. An example workflow published there is given in Figure 7.

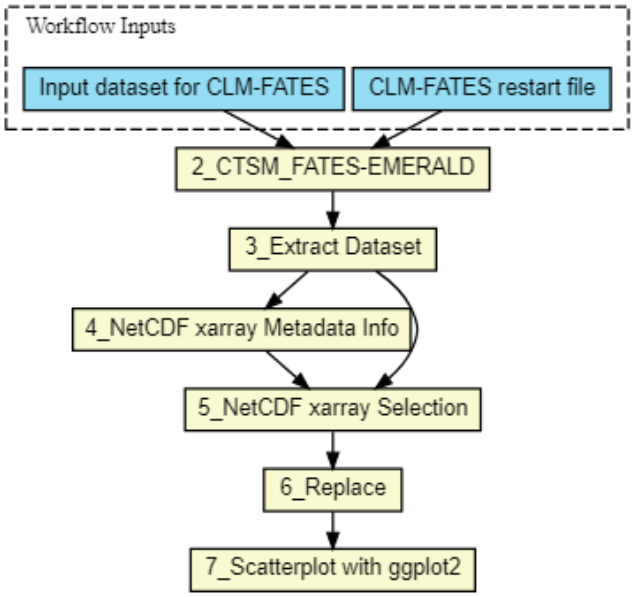

Figure 7: CLM-FATES_ALP1_simulation_5years (https://workflowhub.eu/workflows/65).

2.2.5 Take-up

Dedicated training material for on-boarding users has been developed and is publicly available:

- Introduction to FATES (slides and video),
- CLM-FATES Galaxy tool Hands-on,
- CLM-FATES in Galaxy Climate JupyterLab.

A first online training was organized in October 2020 and 20 researchers participated and were trained. One of the professors involved in the first training has now incorporated the use of the Galaxy (CLM-FATES Galaxy tool) in the course GEO9915 – Ecological Climatology at the University of Oslo. This 10 credit course is for PhD students and provides an overview of the relationships between climate and ecology, with a focus on climate-related feedback within boreal, alpine, and arctic terrestrial ecosystems. About 20 PhD students have signed up and feedback on the usage of the CLM-FATES Galaxy tool will be collected at the end of the course.

2.2.6 Next steps

In this subtask, we successfully added complex tools such as the Functionally Assembled Terrestrial Ecosystem Simulator (FATES) to Galaxy Europe and its Galaxy Climate Workbench, and more than 40 researchers have been trained to use the platform. The preliminary feedback we got from researchers clearly highlights the need to make the following significant improvements:

- Access to bare-metal HPC resources (including LUMI, the European pre-exascale HPC system that will be made available to Nordic researchers) to run larger and longer simulations: for instance, to run global Earth System Models at higher resolution for hundred years of simulations;
- Reduce data movement by improving Pulsar or by simply directing certain climate jobs to specific Pulsar nodes where the corresponding climate data is available (some kind of meta-scheduler). This would significantly improve performance when large amounts of climate data would need to be moved to the Pulsar node.

For implementing these improvements, the best approach would be to deploy a Nordic Galaxy instance and connect new pulsar nodes in each Nordic country and provide “direct” access to:

- Earth System Grid Federation (ESGF) nodes in Sweden, Finland and Norway for processing Coupled Model Intercomparison Project (CMIP) climate data and
- National storage in each country (e.g Swestore in Sweden, National Infrastructure for Research Data (NIRD) storage in Norway).

3. Conclusions

In subtask T5.2.1, two use cases were selected to apply and test cross-borders computing: connecting the Galaxy and the PlutoF portal to Nordic HPC clusters. T5.2.1 successfully managed to: 1) Package analysis services and tools in each respective project according to the best practices, so that they can be deployed and run on different infrastructures independent of the software available on target platforms; 2) Work

out technical solutions and recommended procedures on how their services could be coupled with remote EOSC-Nordic compute resources.

The selected use cases have explored different technical solutions to serve their respective community of users and take-up of the deployed services by researchers has been very good. However, both faced an issue related to the use of robot accounts with bare metal HPCs and while technical solutions have been proposed, their implementation would require changes that need to be handled at a policy level. This would be the role of EOSC-Nordic during the last part of the project.

Another issue faced by both use cases, was the sustainability of the services made available at remote HPC clusters. HPC service access is normally provided for a certain time period, after which the user has to go through the process of applying for resources again. This constant renewing of access and change of the HPC providers' specifics require a number of actions from both sides – the user and the service provider. It also increases the cost of maintaining the tools and services.

With the procedures worked out in subtask T5.2.1 our aim was to provide helpful recommendations accompanied by practical examples on how cross-bordering and integration of remote HPC resources within the EOSC-Nordic can be achieved. We hope that the work conducted under subtask T5.2.1 and described in this deliverable becomes a useful resource for the other community specific and thematic portals with similar objectives for building cross-borders computing solutions in EOSC-Nordic and broader scale.

4. Supplementary Items

4.1. Supplementary Item S1

For setting up a VRE in the SSC (and similar environments) the following steps are necessary:

Create a new VM with Ubuntu base image. Currently tested on Ubuntu 18.04.

Singularity setup (see Singularity Quick Start for reference) consists of the following sub-steps:

Install system dependencies:

```
$ sudo apt-get update && sudo apt-get install -y \
    build-essential \
    libssl-dev \
    uuid-dev \
    libgpgme11-dev \
    squashfs-tools \
    libseccomp-dev \
    wget \
    pkg-config \
    git \
    cryptsetup \
```

```
  unzip
```

Install latest go language support (at the time of writing: version 1.15.5):

```
$ export VERSION=1.13 OS=linux ARCH=amd64 && \  # Replace the values as needed
 wget https://dl.google.com/go/go$VERSION.$OS-$ARCH.tar.gz && \ # Downloads the required Go package
 sudo tar -C /usr/local -xzvf go$VERSION.$OS-$ARCH.tar.gz && \ # Extracts the archive
 rm go$VERSION.$OS-$ARCH.tar.gz    # Deletes the ``tar`` file
$ echo 'export PATH=/usr/local/go/bin:$PATH' >> ~/.bashrc && \
 source ~/.bashrc
```

Install the latest singularity (at the time of writing: version 3.6.4):

```
$ export VERSION=3.6.4 && \ # adjust this as necessary
   wget https://github.com/sylabs/singularity/releases/download/v${VERSION}/singularity-${VERSION}.tar.gz && \
   tar -xzf singularity-${VERSION}.tar.gz && \
   cd singularity
$ ./mconfig && \
   make -C builddir && \
   sudo make -C builddir install
```

PlutoF analysis container setups:

Prepare the needed singularity analysis/plutof packages in the new VM. Setup info should be included with a specific container.

Install task-spooler (see blog posts for reference):

```
$ sudo apt install task-spooler
```

Configure the new VM for use in PlutoF:

Append the public ssh key from PlutoF server into the new VM:

```
$ cat ~/.ssh/rocket_hpc_id_rsa.pub
```

Into:

```
~/.ssh/authorized_keys
```

Add a at PlutoF a new HPC instance with the new VM's info via:

https://api.plutof.ut.ee/admin/analysis/hpccluster/