Secure deployments of Galaxy Servers for analysing personal and Health data leveraging the Laniakea service

Tangaro MA, Foggetti N, Donvito G, Antonacci M and Zambelli F

International Symposium on Grids & Clouds (ISGC) 2023
March 19-31, 2023

Outline

- Legal issues concerning genomic and genetic data
- Tools, data, compute and … GDPR
- Galaxy
- Laniakea
- Dashboard
- Encryption
- VPN
- Conclusions

Legal issues concerning genomic and genetic data in Laniakea

- Application of GDPR: art. 9: Particular category of Data (Genetic, genomic and other typology of data)

- The Convention on Human Rights and Biomedicine (article 10), the Universal Declaration on the Human Genome and Human Rights (article 5, letter c), and UNESCO´s International Declaration on Human Genetic Data (article 10)

- ELSI (Ethical, Legal, Social implication document)

- Authorization n. 8/2016 – General authorization concerning genetic data processing 15 December 2016:
  - Pseudonymization
  - Use of SSH protocol for sharing data
  - Specific measures of identification
  - High level of security measures in order to protect and to prevent cyber attack
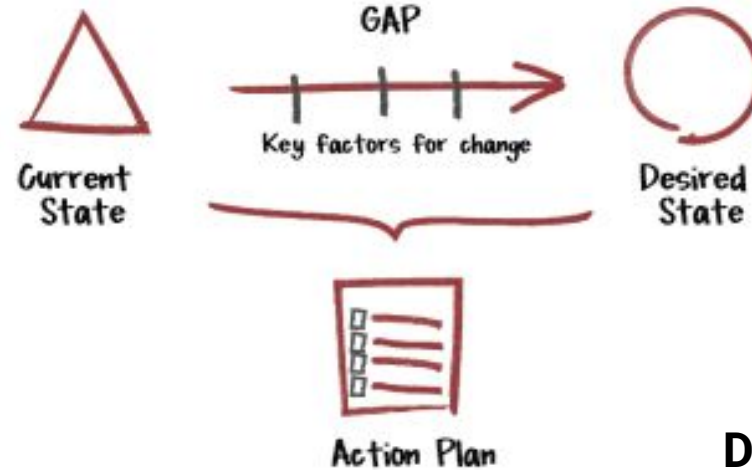
Use case application

The use case of Lanikea has been analysed also in INFN Cloud

- Definition of Policy for services (mandatory in EU)
- definition of ToU and AUP harmonized
- Possible application of ISO standard in order to improve the legal compliance

4

Gap Analysis

Fragmentation of applicable law and jurisdiction

IPR Exception and Limitations: blinding list and not harmonized

Differences between different law at EU Level concerning specific sectors (e.g., health, genetics)

Data protection: Transnational data processing: conflict of laws and sectoral law

Discrimination defined by the national application of EU Law

Common standards and interoperability

Harmonization of **pseudonymization e anonymization procedures**

Guidelines

Objectives

To Identify the legal framework applicable in the field of data protection and IPR

To Increase the level of awareness in the context of processing data and resources protected by IPR

To mange DMP/OA/OS

Target: Reserarchers

The Checklist

- See "Legal Compliance Guidelines for Researchers: a Checklist"
  - Both digital (with interactive checkboxes, https://doi.org/10.5281/zenodo.632766), and printable (https://doi.org/10.5281/zenodo.6327691)
- To guide researchers in management of research outputs vis-à-vis IP and data

- protection laws

- To promote best practices to achieve FAIR ecosystem, removing unnecessary restrictions to reuse and access + facilitating convergences of national solutions

The Checklist

RESEARCH PROPOSAL 1	RESEARCH IMPLEMENTATION 2	RESEARCH REVIEW 3
Background information, IPR, Exploitation, TTO, FAIR, DPIA.	IP Management Plan, IP FLEXIBILITY, GDPR, DMP, Findable, Accessible, Re-Usable, Interoperable.	IP Management Plan, Licences and FAIR/OS/OA, Re-use of Data.

Laniakea architecture
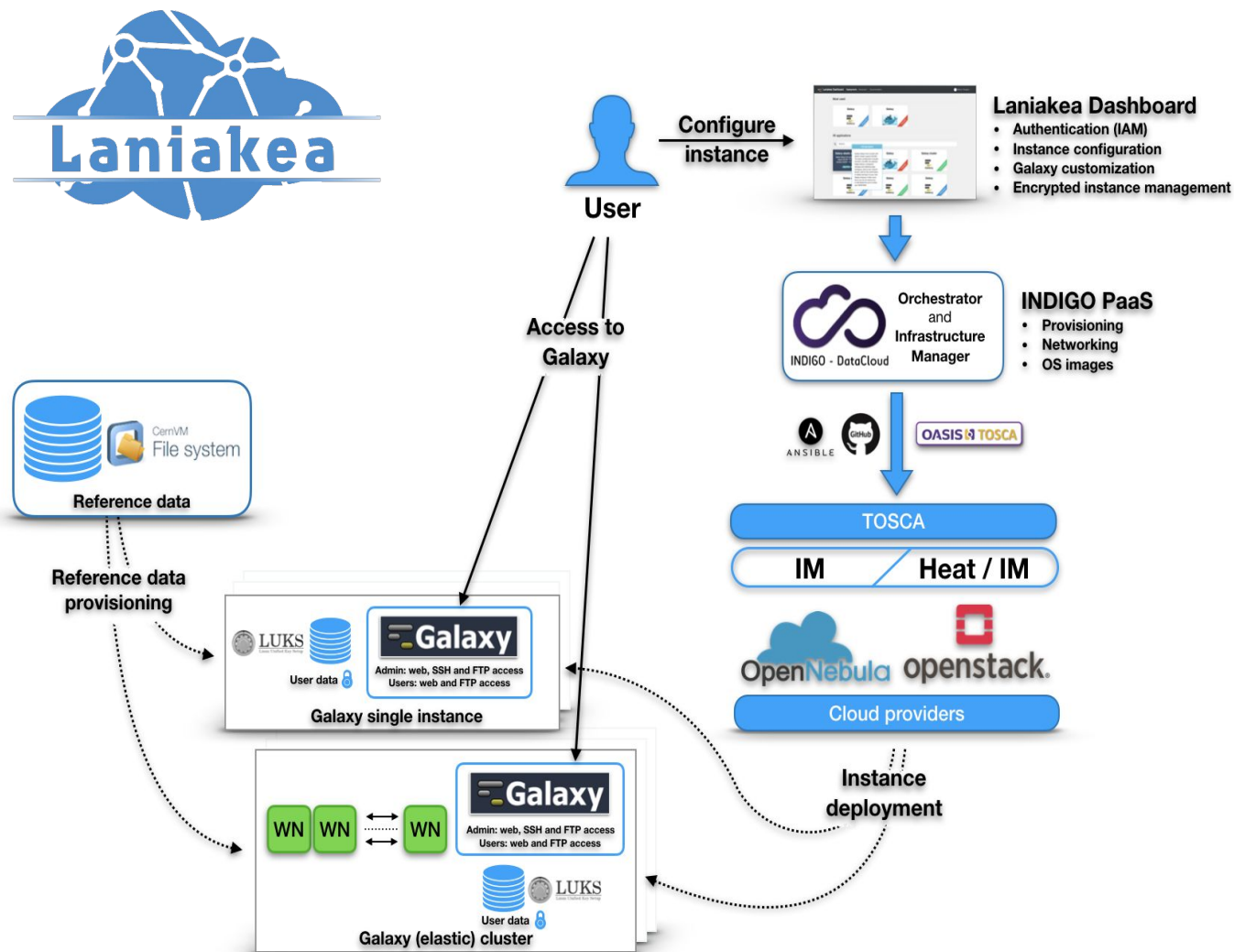
- **Dashboard -** User friendly access to configuration and and launch of a Galaxy instance.

- **IAM -** Authentication and Authorization system.

- **INDIGO PaaS -** Galaxy automatic deployment.

- **Cloud Providers -** (INFN) ReCaS-Bari and others.

- **Persistent storage** - With/without encryption.

- **Reference data availability** - With CERN-VM FileSystem.

- **CLUES -** Elasticity manager.

Legal issue and the application of Checklist

Identify the conditions / and data protection based on internal and international regulations

France LS vs Italian LS
The pseudonimization requirements:
IT-> simple pseudonimization
FR-> Entity for pseudonimization

Authentication system legal Compliant

Connection with DPO and TTO

Measure needed to use genetic and genomic data

Data retention conditions
Process of reusability of data (pesudonimization e/o aninimization)

Plan for the use of data compliance with EU and National law IPR

Repository choice: compliance with National and International Legal Framework

Legal Framework for the use and re-use of health data for scientific purposes
DOI: 10.5281/zenodo.6334878

Data

Data growth at EMBL-EBI by data resource

Data volume growing not only in quantity but also on variety!

Data growth at EMBL-EBI Source: Charles E. Cook et al. Nucl. Acids Res. 2020; Volume 48, Issue D1, Pages D17-D23
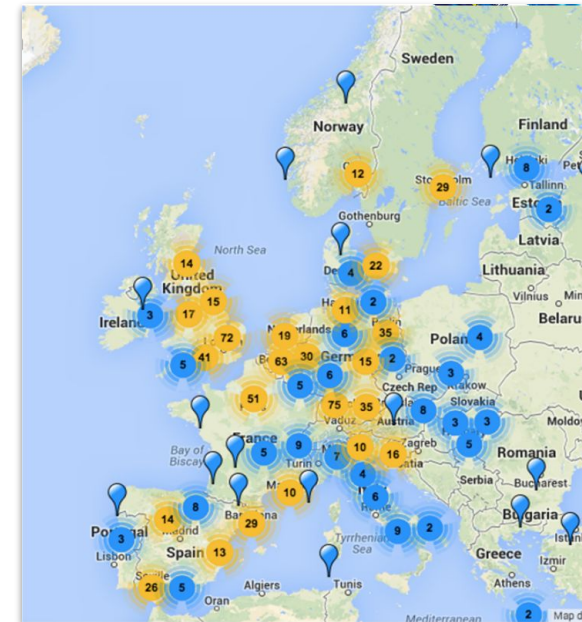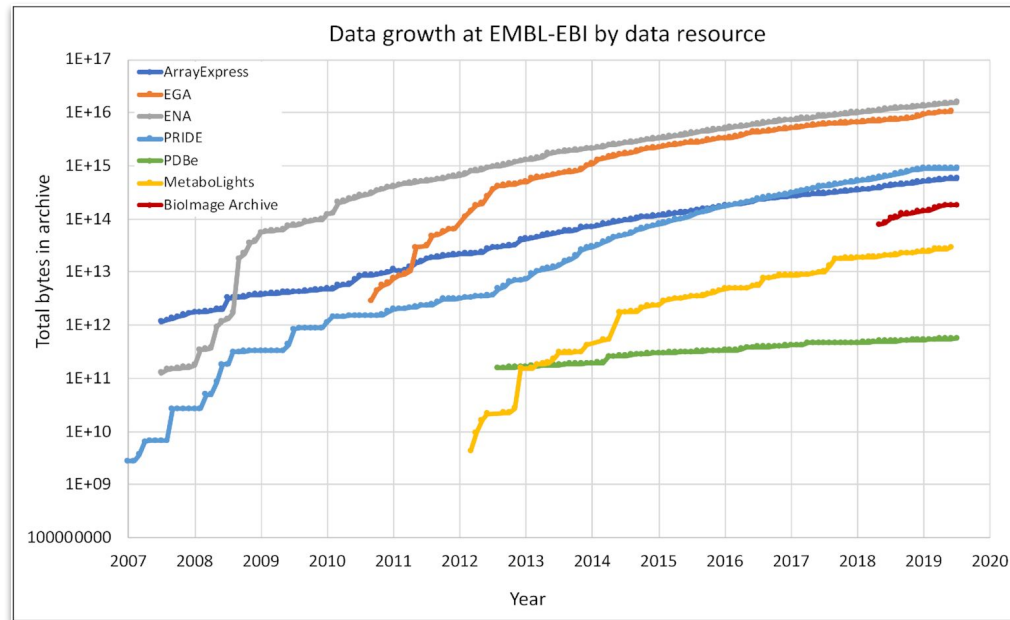
Genomic data are distributed across several sequencing centres and/or IT infrastructures

Discipline	Data size	# devices
HEP-LHC	15PB/year	1
Astronomy	15PB/year	several
Genomics	0.4TB/genome	>1000

11

Tools

The command line is the standard way to use most bioinformatics tools:

- Plenty of parameters
- Multiple input and output data and formats
- Reference data
- Need to run multiple times
- Need to change the parameters for each run

Workflows involve more than one tools!

Tools are usually manually installed.

GDPR

The GDPR explicitly recognizes genetic data as "Special Categories of Personal data".

Sensitive genetic data processing for scientific research purposes is possible, provided this is allowed by EU or Member States law framework and appropriate safeguards measures are in place.

Motivation

DATA SOURCES

DATA STORAGE
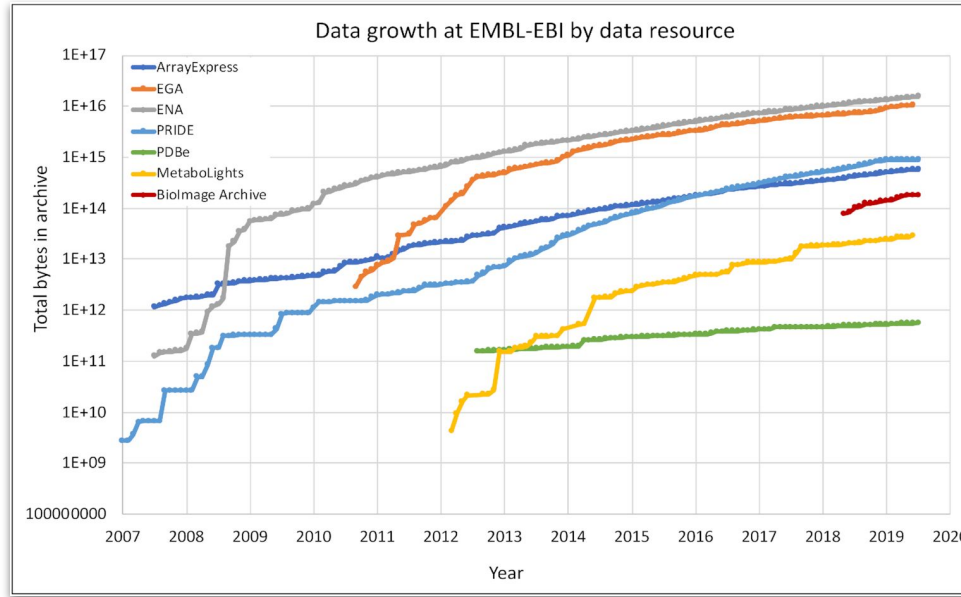
Data growth at EMBL-EBI by data resource
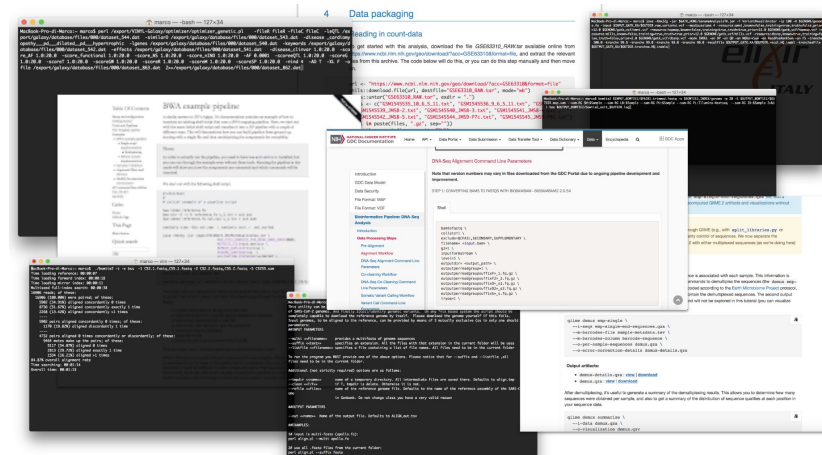
DATA ANALYSIS TOOLS

DATA PROTECTION (GDPR)
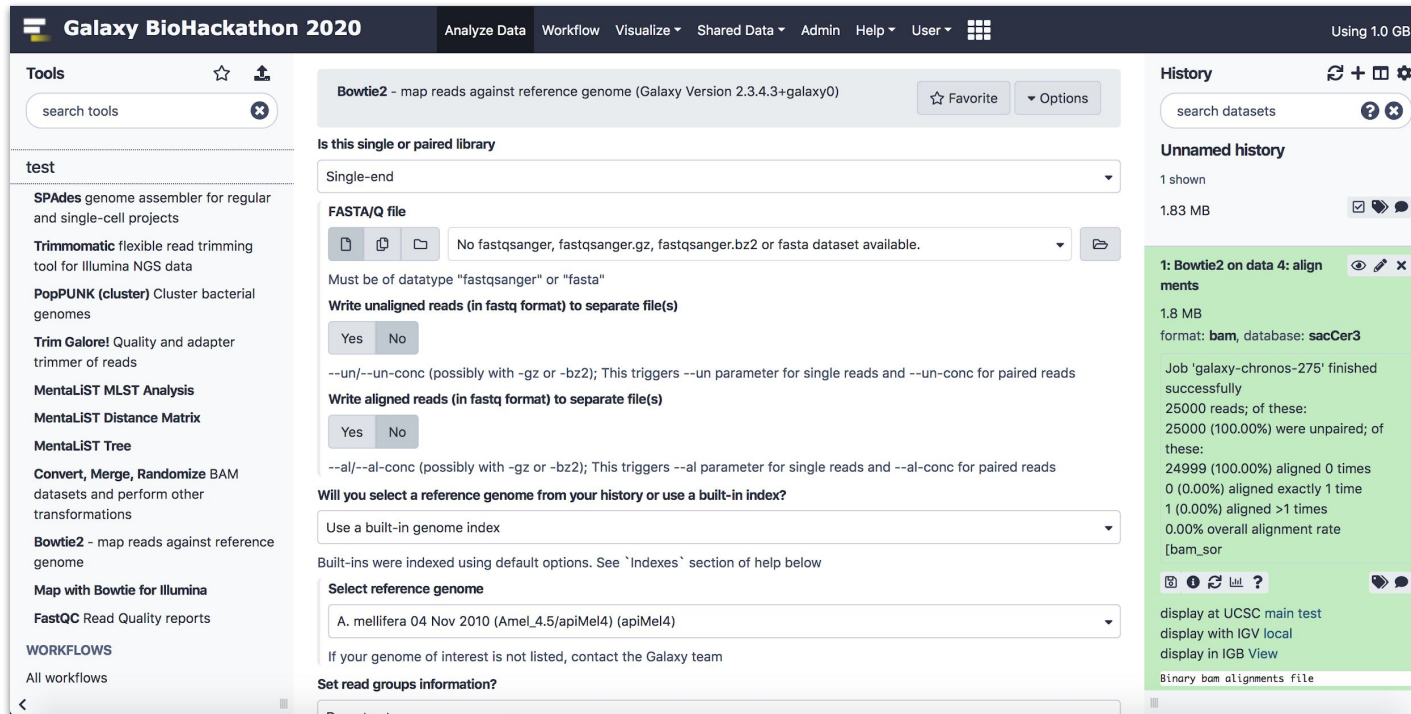
Galaxy
PROJECT

Galaxy is a workflow manager adopted in many life science research environments in order to facilitate the interaction with bioinformatics tools and the handling of large quantities of biological data.

Through a coherent work environment and an **user-friendly web interface** it organizes data, tools and workflows providing **reproducibility, transparency** and **simple data sharing** functionalities to users.

galaxyproject.org
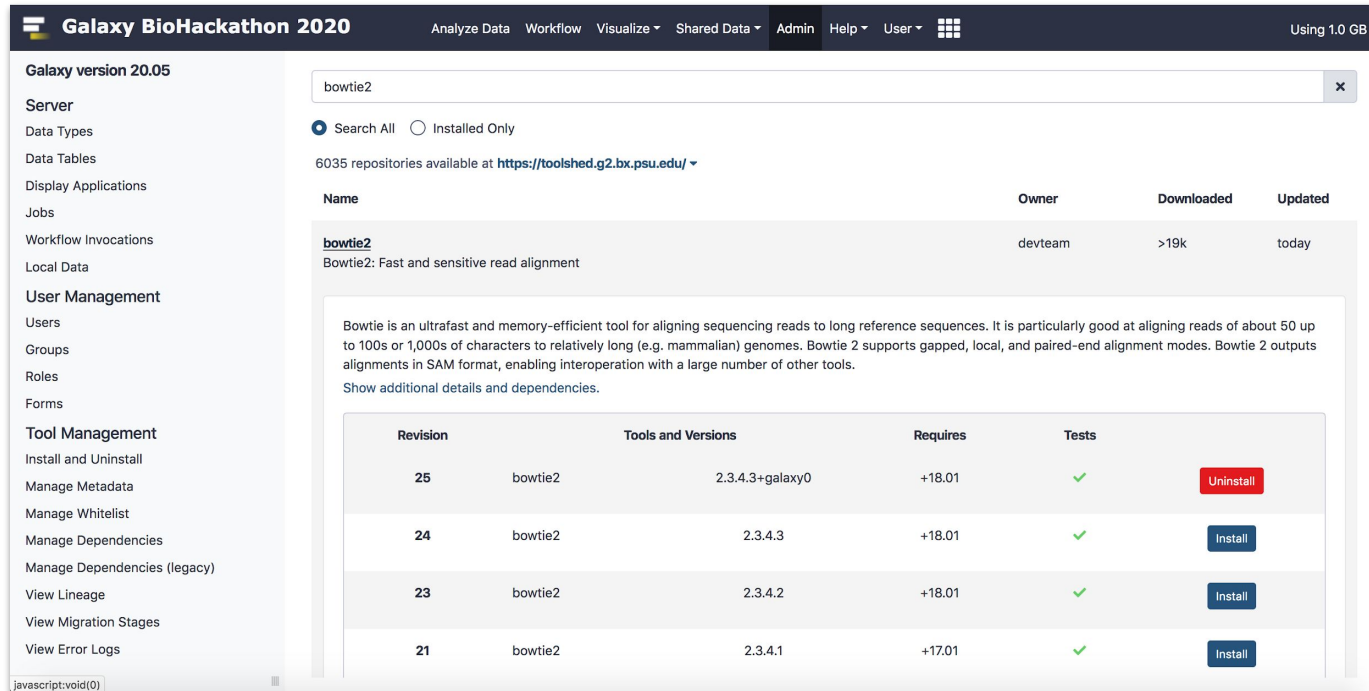
- Tools graphical user interface.
- Input and output data management.
- Output visualization.
- Data and analysis parameters sharing.
- Used tools and parameters configuration always available **-> analysis reproducibility**.
- Reference data already available for many tools.
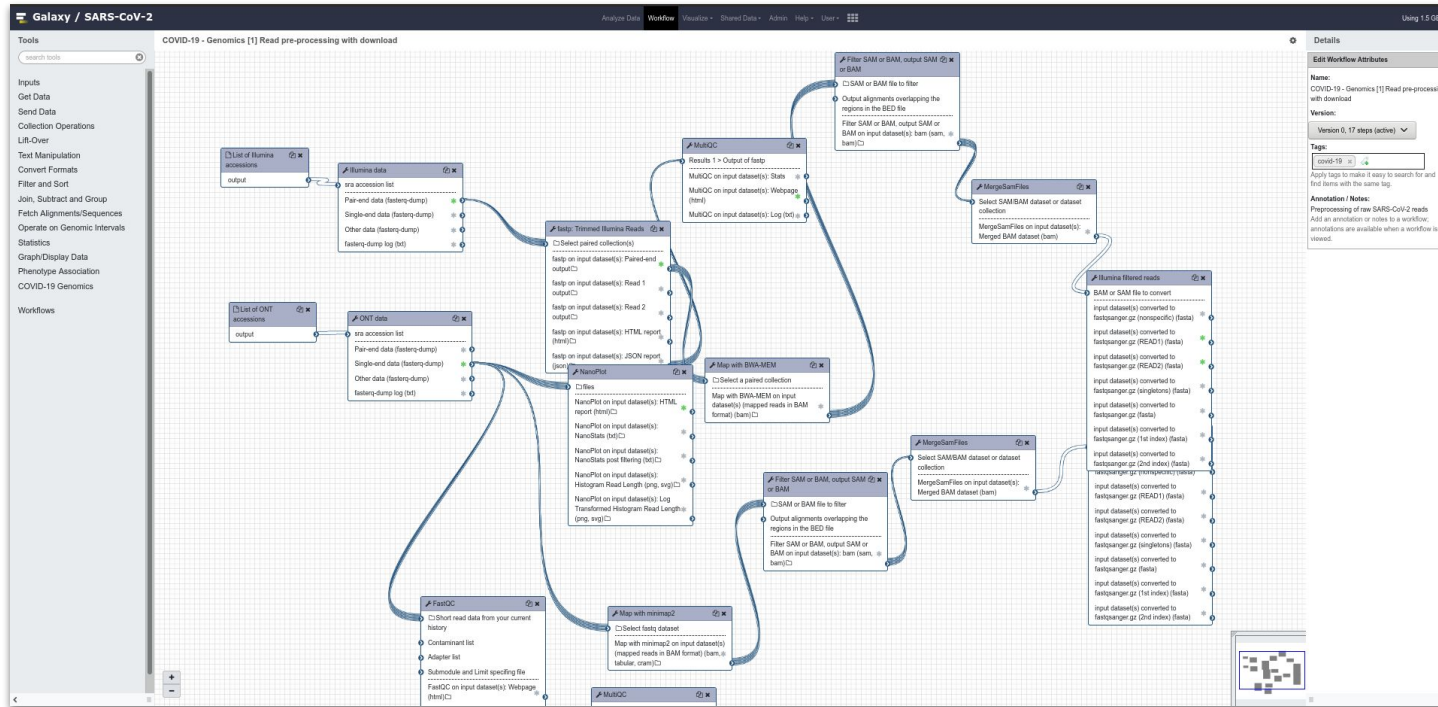
Galaxy Workflow Editor

Graphical user interface to easily add, connect and configure tools for composing workflows.

Galaxy ToolShed

Serves as an "app store" to all Galaxies worldwide.

It is a **free service** Galaxy developers to share tools.

Galaxy Administrator can install tools on their instances.

Tools dependencies automatically solved.

All Galaxy users can access to the tools available on a server.

Laniakea

LANIAKEA IS A CLOUD BASED GALAXY INSTANCE PROVIDER

https://laniakea-elixir-it.github.io/

- Laniakea relies on commonly used Life Science Open Source tools, e.g. Galaxy, RStudio, Jupyter, HashiCorp Vault, LUKS and SLURM.

- Laniakea is European Open Science Cloud service provider.

Recommended for scenarios where users need full administrative control over a private Galaxy instance.
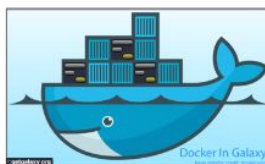
Laniakea main features

Dashboard - By hiding the technical complexity behind a user-friendly web front-end, Laniakea allows its users to configure and deploy "on-demand" Galaxy instances with a handful of clicks.

No need for the end user to know the underlying infrastructure.

No need for maintenance of the hardware and software infrastructure.
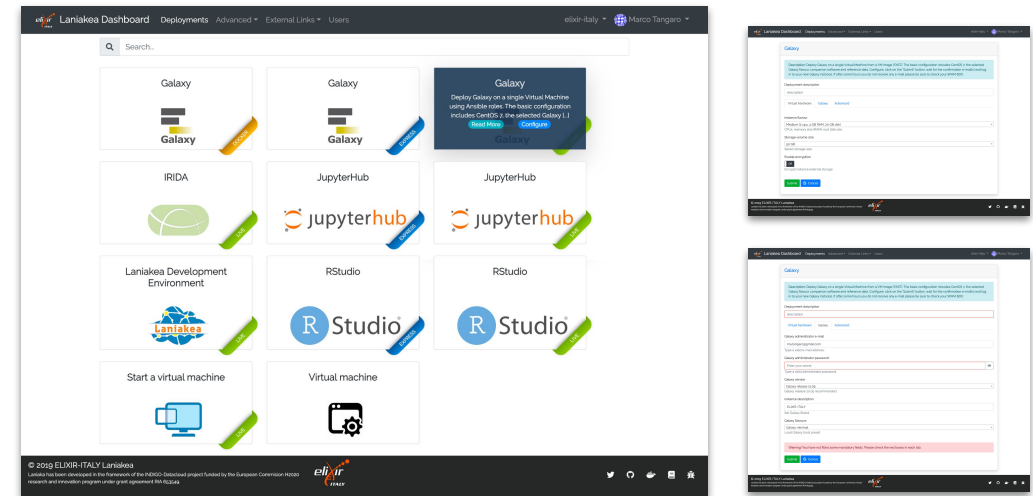
Different deployment strategies:

Live Build: build Galaxy from scratch -> always up-to-date (deployment time depending by the tools number).

Express: pre-built Galaxy images -> fast deployment, but tools not always at the last available version.

Docker: fast development of new flavours.

Laniakea main features
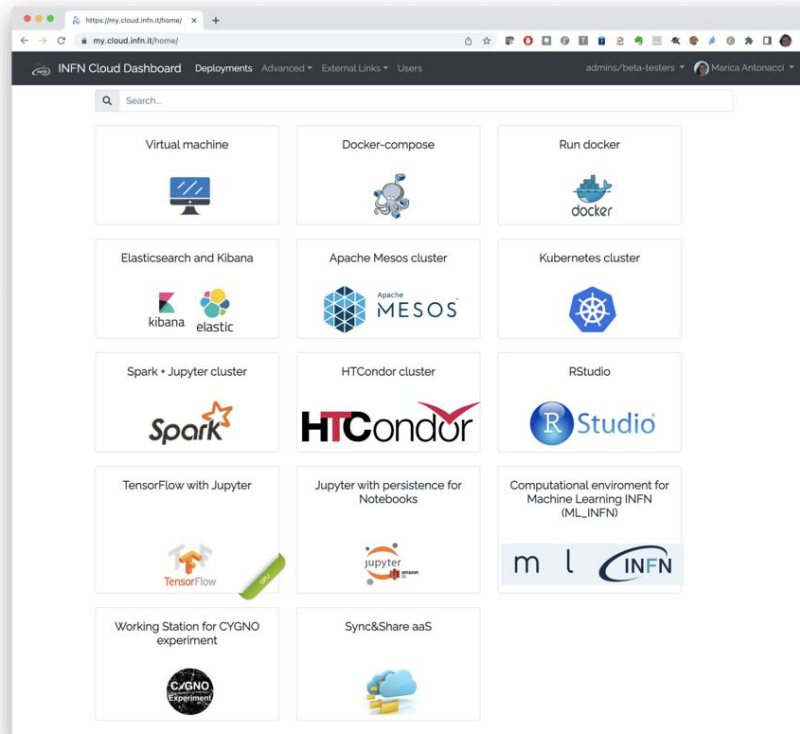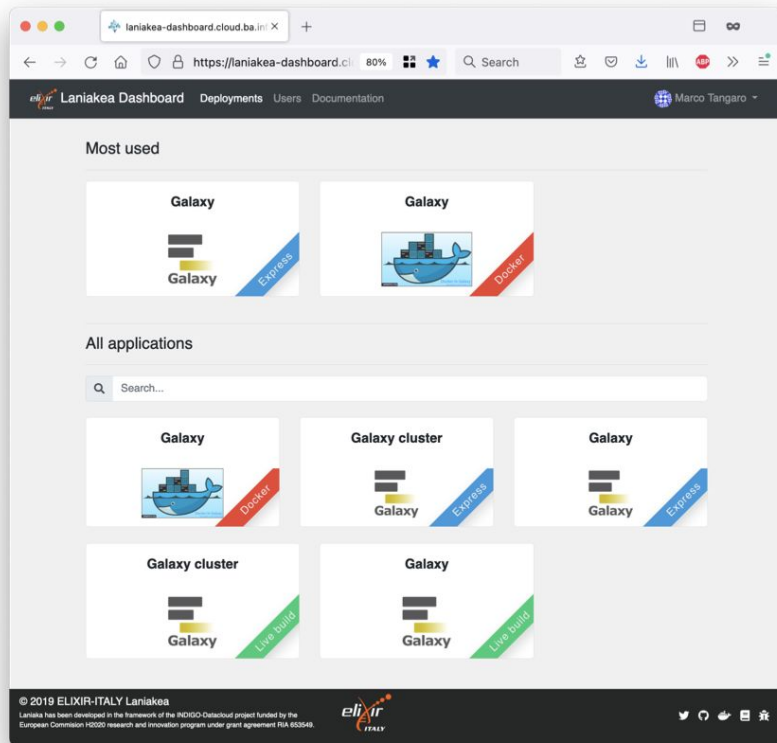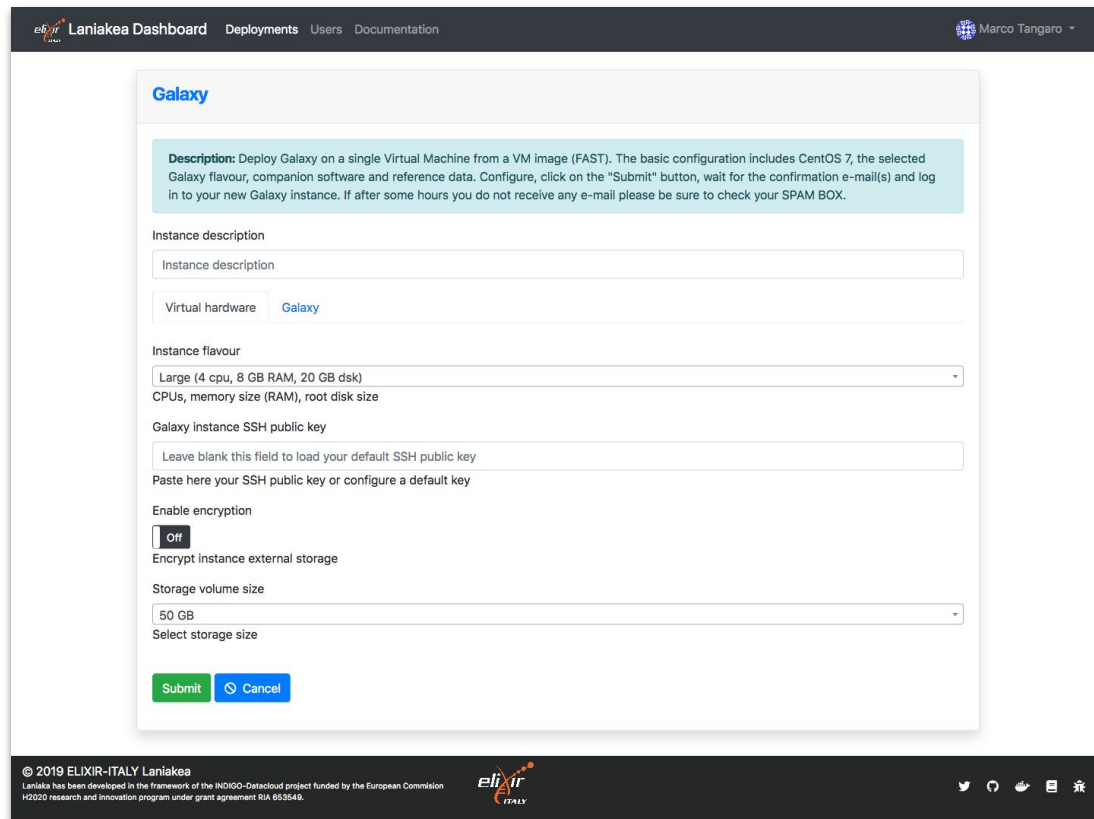
Flask web micro-framework
(flask.pocoo.org/),

Jinja2 template engine
(jinja.pocoo.org/)

Bootstrap 4 toolkit
(getbootstrap.com/).
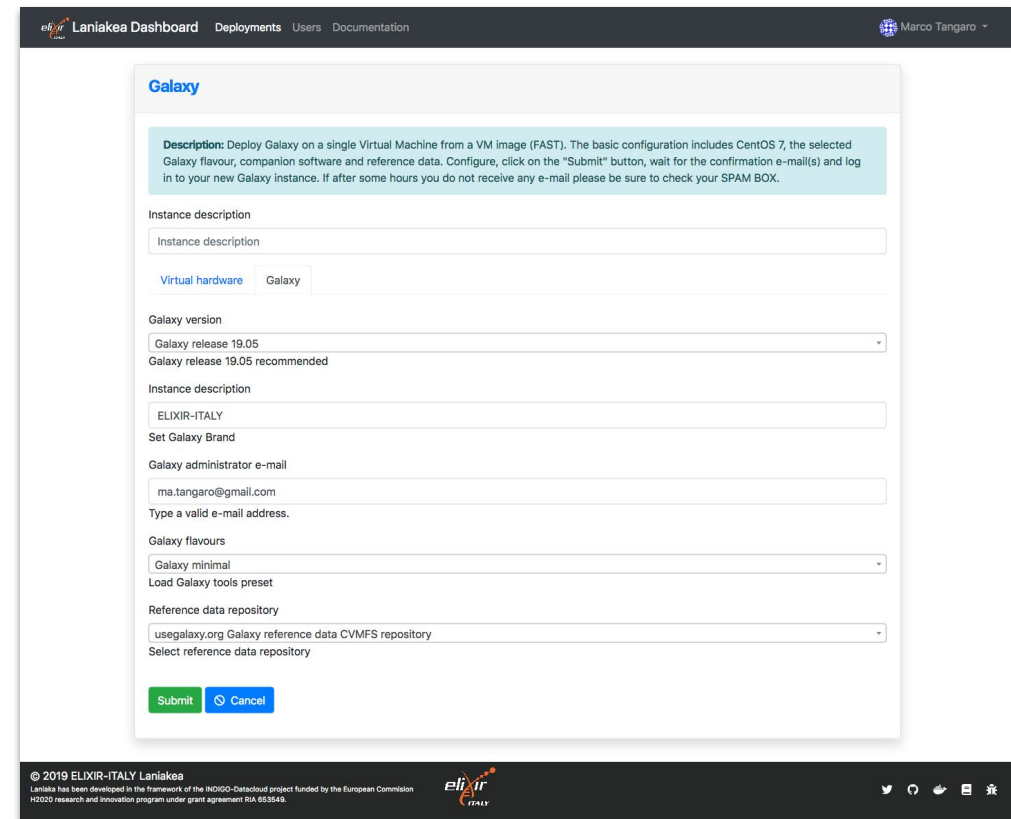
Integrated with Hashicorp
Vault for user secrets
management.

Laniakea main features

Virtual hardware: CPU,
RAM and Storage

Galaxy software: version,
credentials, flavor and
reference data.

Laniakea main features

Galaxy flavors - Deploy Galaxy with sets of tested, validated and pre installed tools, named Galaxy flavors.

Current available tools presets: Galaxy Minimal, Galaxy CoVaCS, Galaxy GDC Somatic Variant, RNA Workbench, Galaxy Epigen, Covid-19.

More Applications - No more limited to Galaxy. Jupyter Notebooks, RStudio and IRIDA available.

Environment with NextFlow, CWLtool and other development tools available.

Laniakea main features
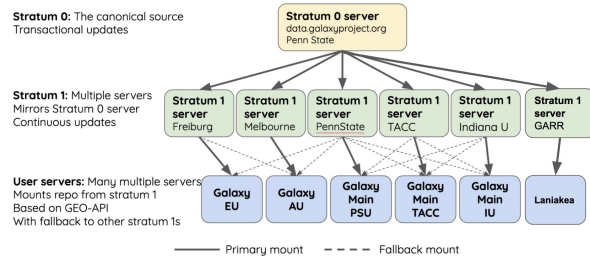
Shared reference data - Each instance comes with reference data (e.g. genomic sequences) already available for many species, shared among all the instances through the CERN-VM FileSystem .

Galaxy with cluster - allowing to instantiate Galaxy with dedicated Resource Manager, allowing to customize the number of the virtual nodes to be created and their configuration in terms of number CPU and RAM.
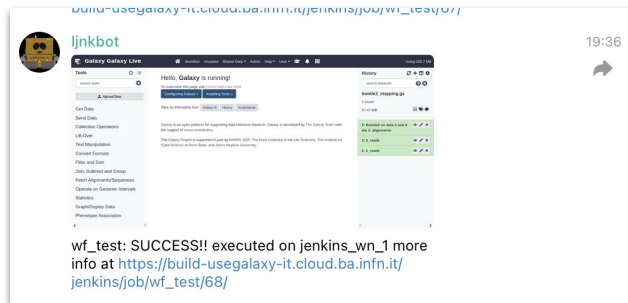
Continuous Integration - Cloud images creation and service testing are automated using Jenkins and Github.
Cloud images automatically updated when there are changes on github
Services are tested regularly every week.

Secured environment features

Storage Encryption - Data privacy is provided through encryption "on-demand".

OPENVPN

Deployments under Private Network - Automatic deployments of virtual environments on private networks.

Laniakea architecture
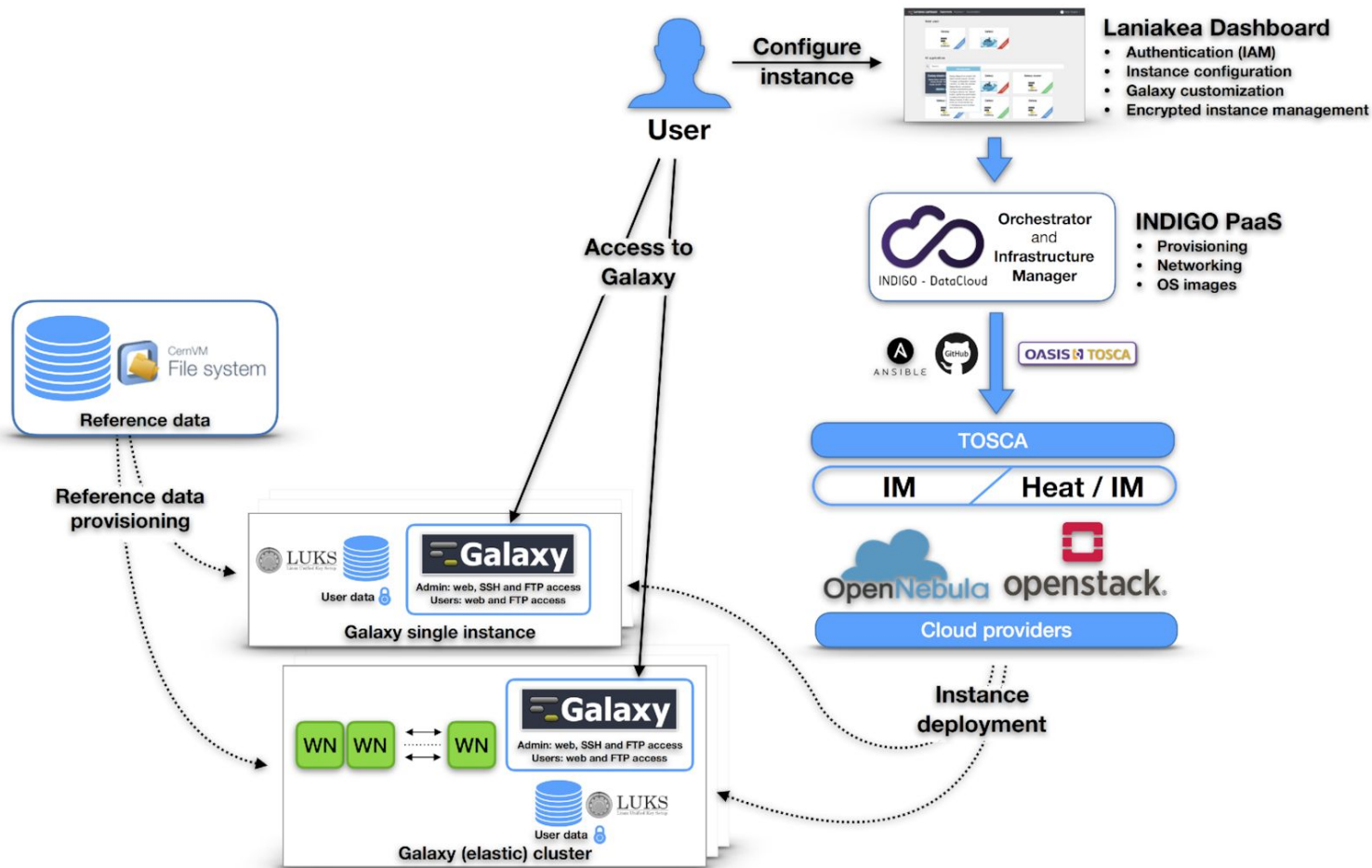
- **Dashboard** - User friendly access to configuration and and launch of a Galaxy instance
- **INDIGO-IAM** - Authentication and Authorization system
- **INDIGO-PaaS** - PaaS layer for Galaxy deployment
- Cloud Provider - ReCaS Bari
- **Persistent storage** with/without encryption
- **Hashicorp Vault** - secrets management
- **Reference data** availability with CERN-VM FS
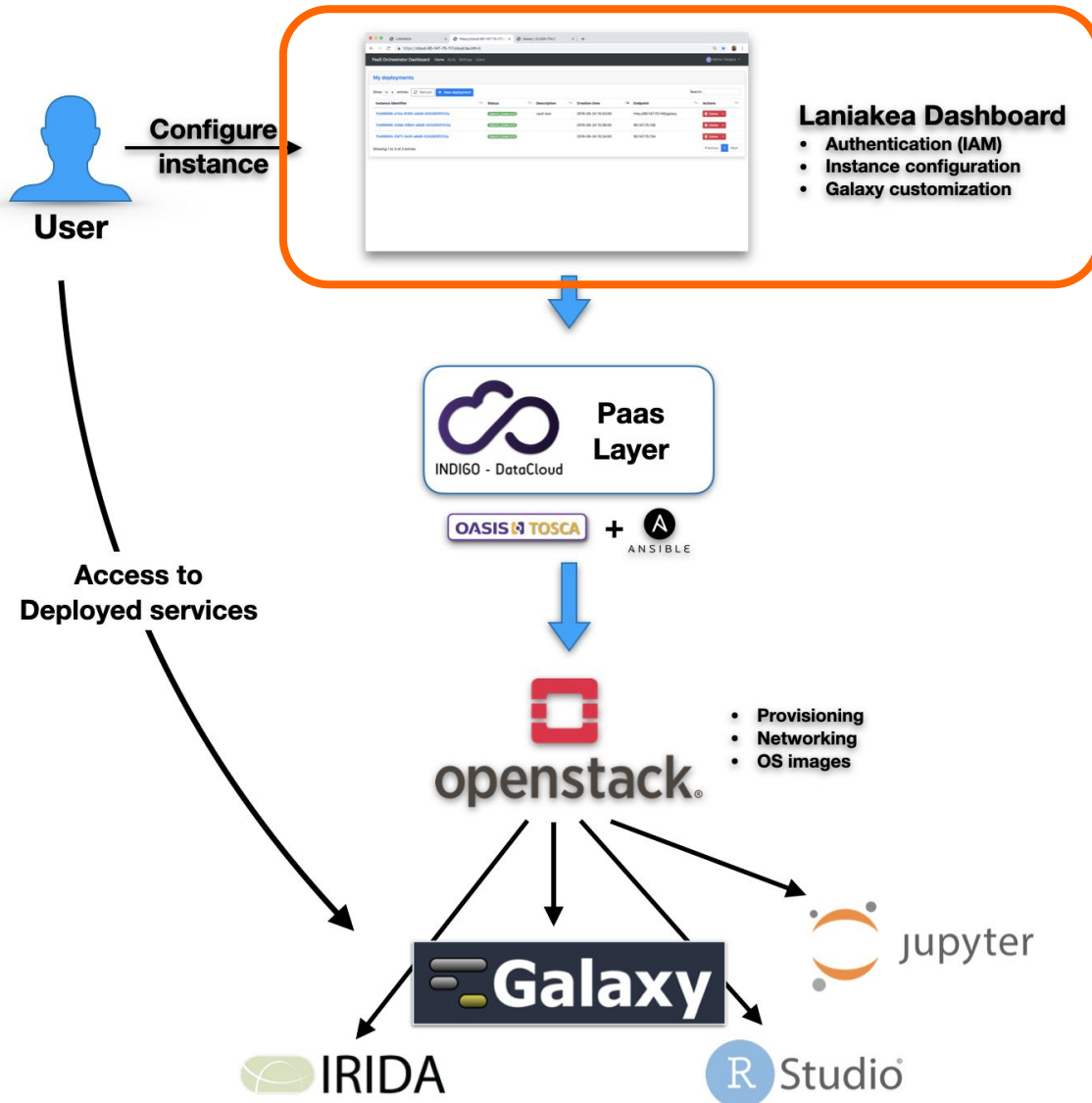
26

Laniakea architecture

- **Dashboard -** User friendly access to configure and launch a Galaxy instance

- **INDIGO PaaS -** Galaxy automatic deployment

- **Cloud Providers -** ReCaS-Bari

Laniakea architecture
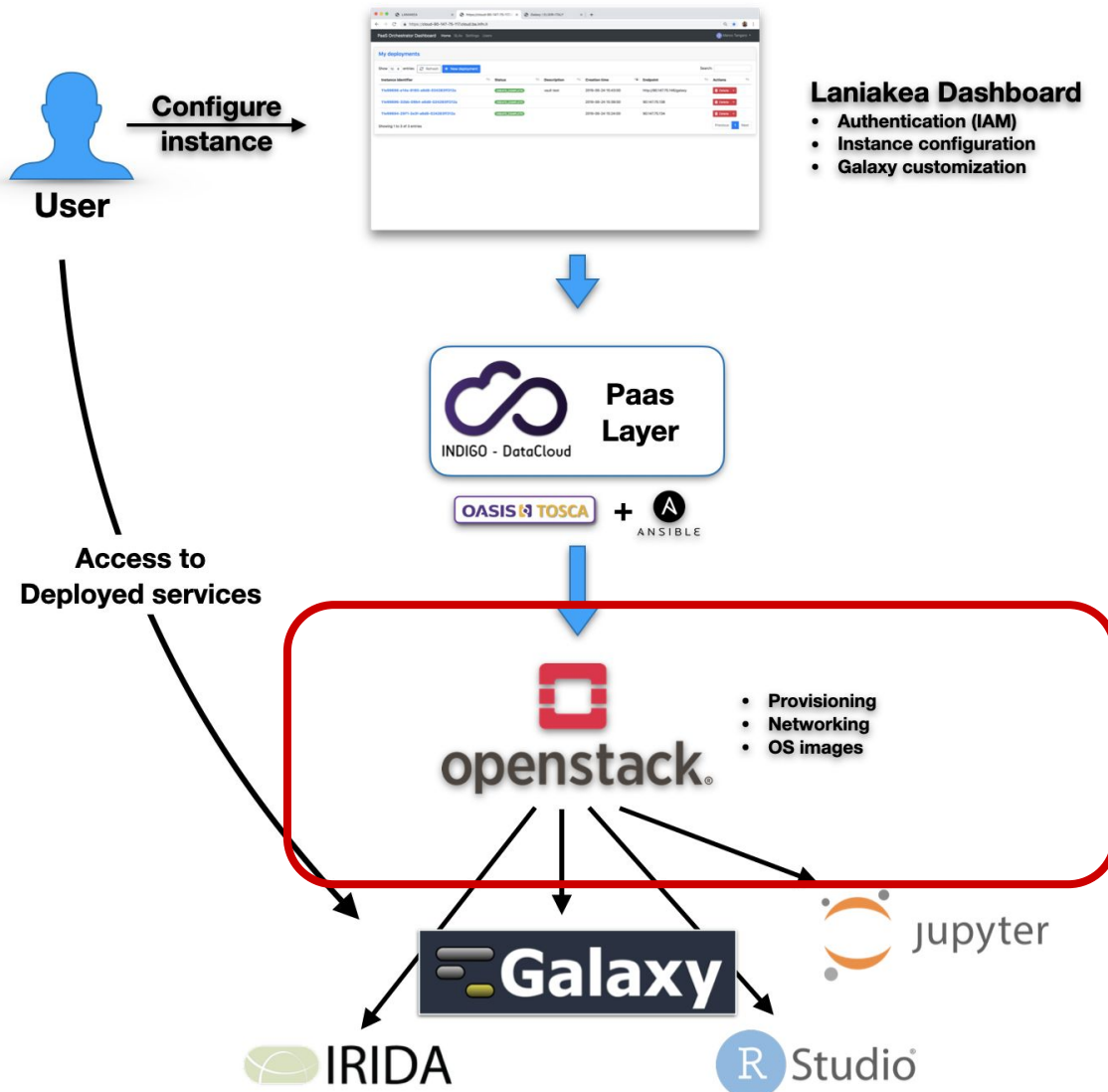
- **Dashboard -** User friendly access to configure and launch a Galaxy instance

- **INDIGO PaaS -** Galaxy automatic deployment

- **Cloud Providers -** ReCaS-Bari

28

Laniakea architecture

- **Dashboard -** User friendly access to configure and launch a Galaxy instance

- **INDIGO PaaS -** Galaxy automatic deployment

- **Cloud Providers -** ReCaS-Bari

Laniakea encryption

Laniakea encryption

The user data privacy is granted through **LUKS** storage encryption as a service: the encryption procedure is automated in order to simplify the user experience, each user can encrypt storage on-demand, using a strong random alphanumerical passphrase.

This has been achieved integrating the Dashboard and the key management system **Hashicorp Vault** (vaultproject.io) to store encryption keys, which are shown in the Laniakea Dashboard only if explicitly requested by the user.

User perspective
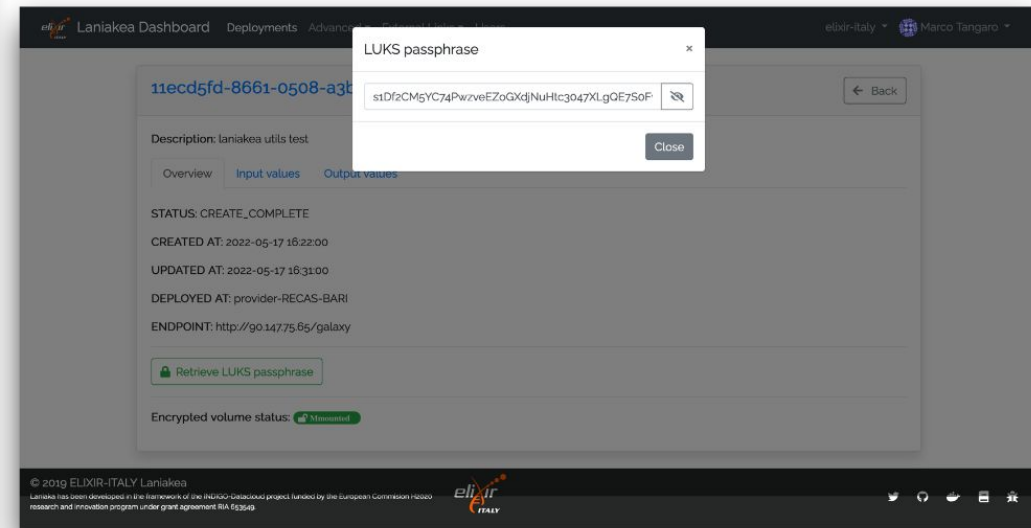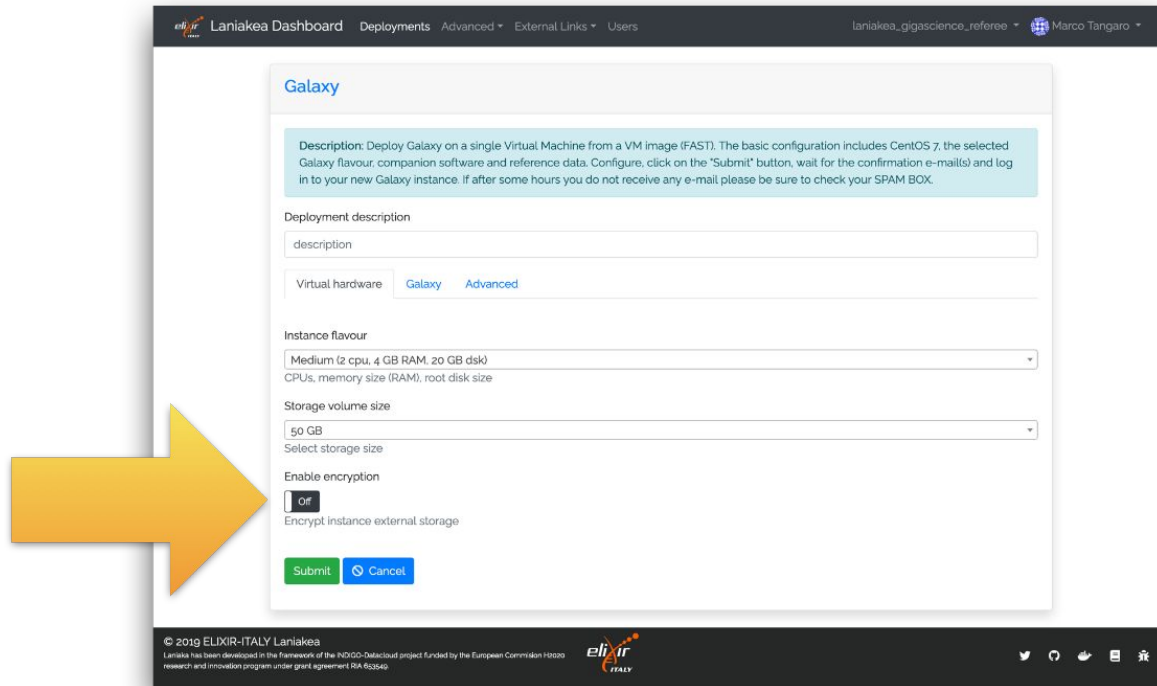
The user can enable the storage encryption using a switch toggle in the Instance "Virtual hardware" configuration tab.

The procedure is completely automated.

The storage is encrypted and the User can retrieve his random passphrase from the Instance overview page.

The underlying infrastructure

LUKS - Linux Unified Kernel Setup

A python package (pyLUKS) is used to encrypt the storage using a random passphrase and then store it on **Hashicorp Vault**.

The encryption layer sits between the physical disk and the file system.

Galaxy, or any other application, is unaware of storage encryption.

Galaxy exploits a specific mount point in order to store and retrieve files. Files are encrypted when stored to disk and decrypted when read.

Default encryption algorithm:

- aes-xts-plain64 encryption
- 256 bit key
- sha256 as hash algorithm used for key derivation.
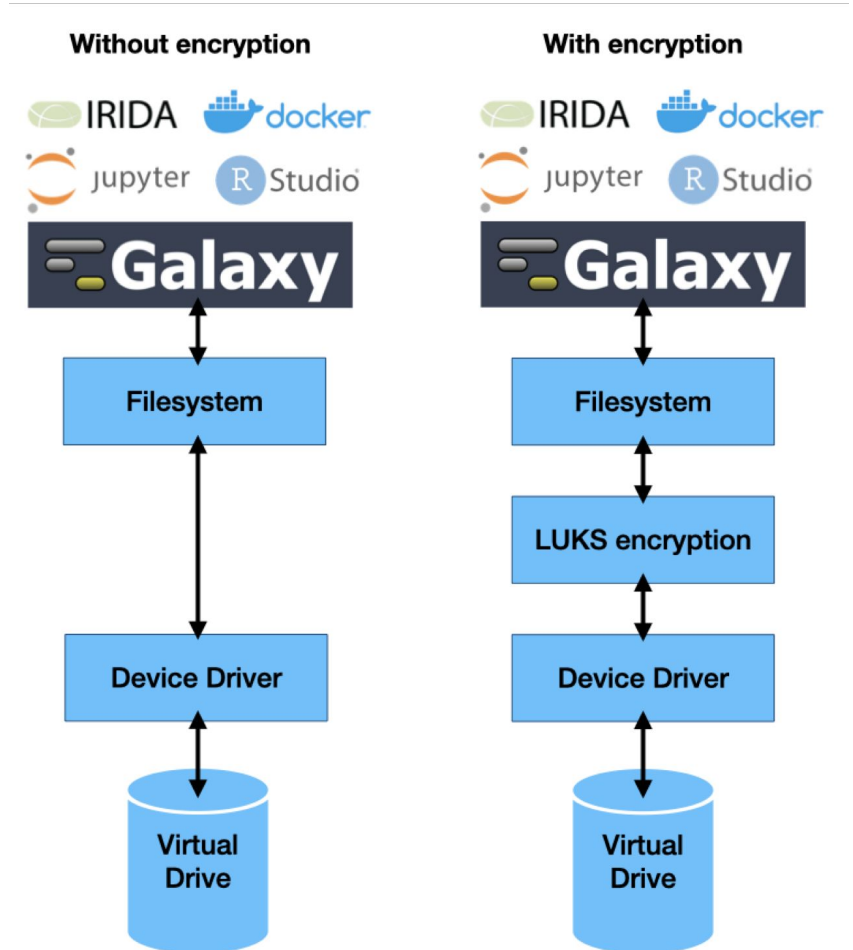
The underlying infrastructure

To evaluate the impact of the storage encryption layer on the performance of the main application supported by Laniakea, i.e., Galaxy, we measured jobs runtime and read/write speed on Virtual Machines generated by the Laniakea@ReCaS data center with and without storage encryption.

The impact on the performance of using the encryption layer, as measured in all our tests, is limited to ~5% or less across all the measured parameters and conditions.
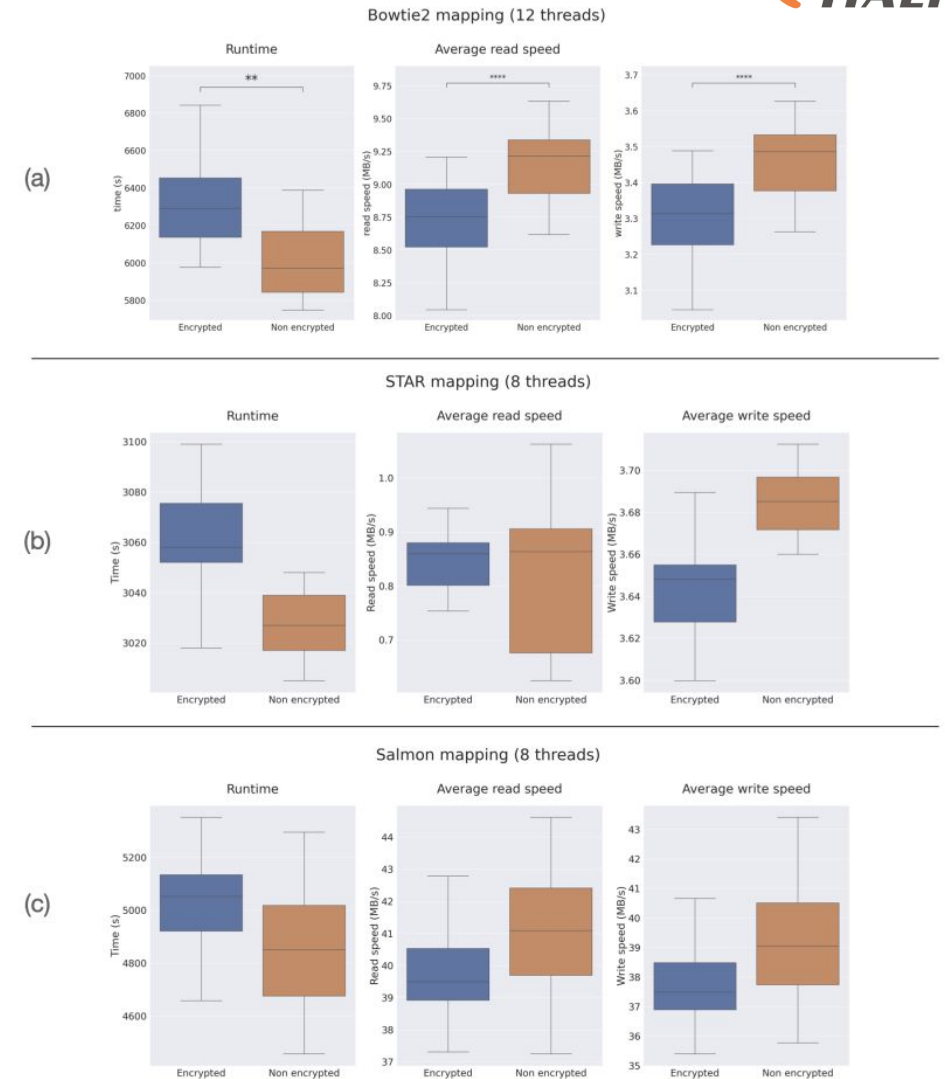
The underlying infrastructure

Key management: Vault introduction

Vault is a tool for securely accessing "secrets".
A secret is everything you want to tightly control access to, such as encryption passphrases.

Vault main concepts:
- Everything in Vault is path based: users are able to write their secrets on a specific path, **depending on their Identity**.
- Tokens are the core method for authentication within Vault. After the authentication on the Laniakea Dashboard, **tokens are dynamically generated based on user identity**.
- Policies provide a declarative way to grant or forbid access to certain path and operations, **controlling what the token holder is allowed to do within Vault**.

A token generated with a specific policy allows to write/read/update a secret in a specific path!

The underlying infrastructure

Key management: Vault authentication and authorization flow

- The "**write only**" token is exploited by LUKS script to store passphrases on Vault.
- The Laniakea Dashboard can **Read**, if required by the user, after the authentication, the passphrase from Vault.
- The Laniakea Dashboard **Deletes** the passphrase from Vault, once the deployment is deleted.
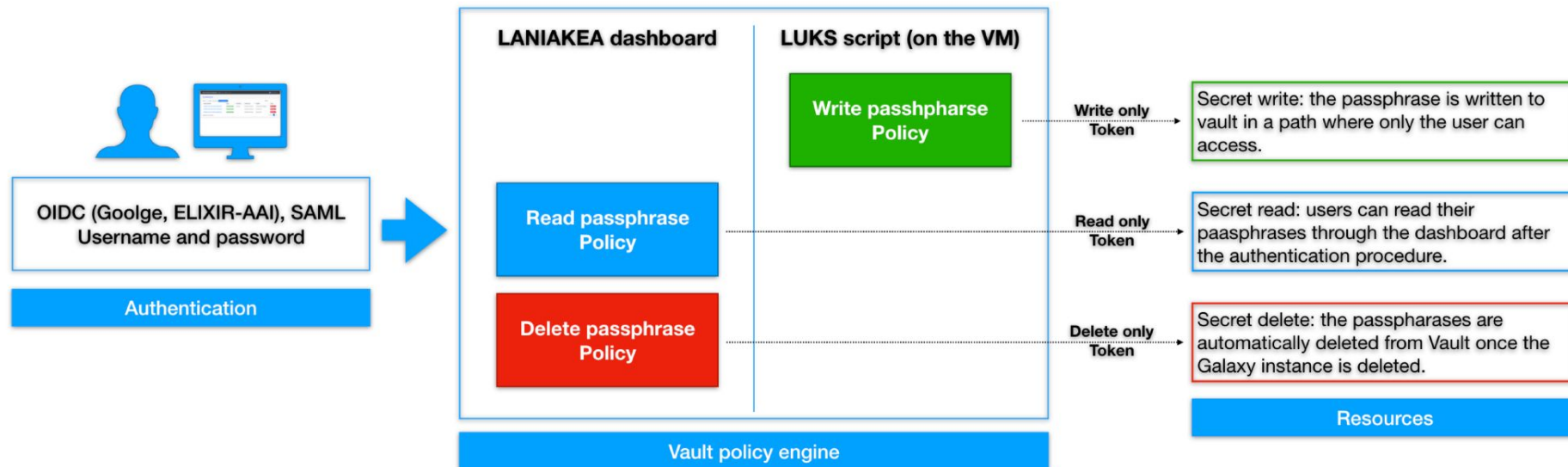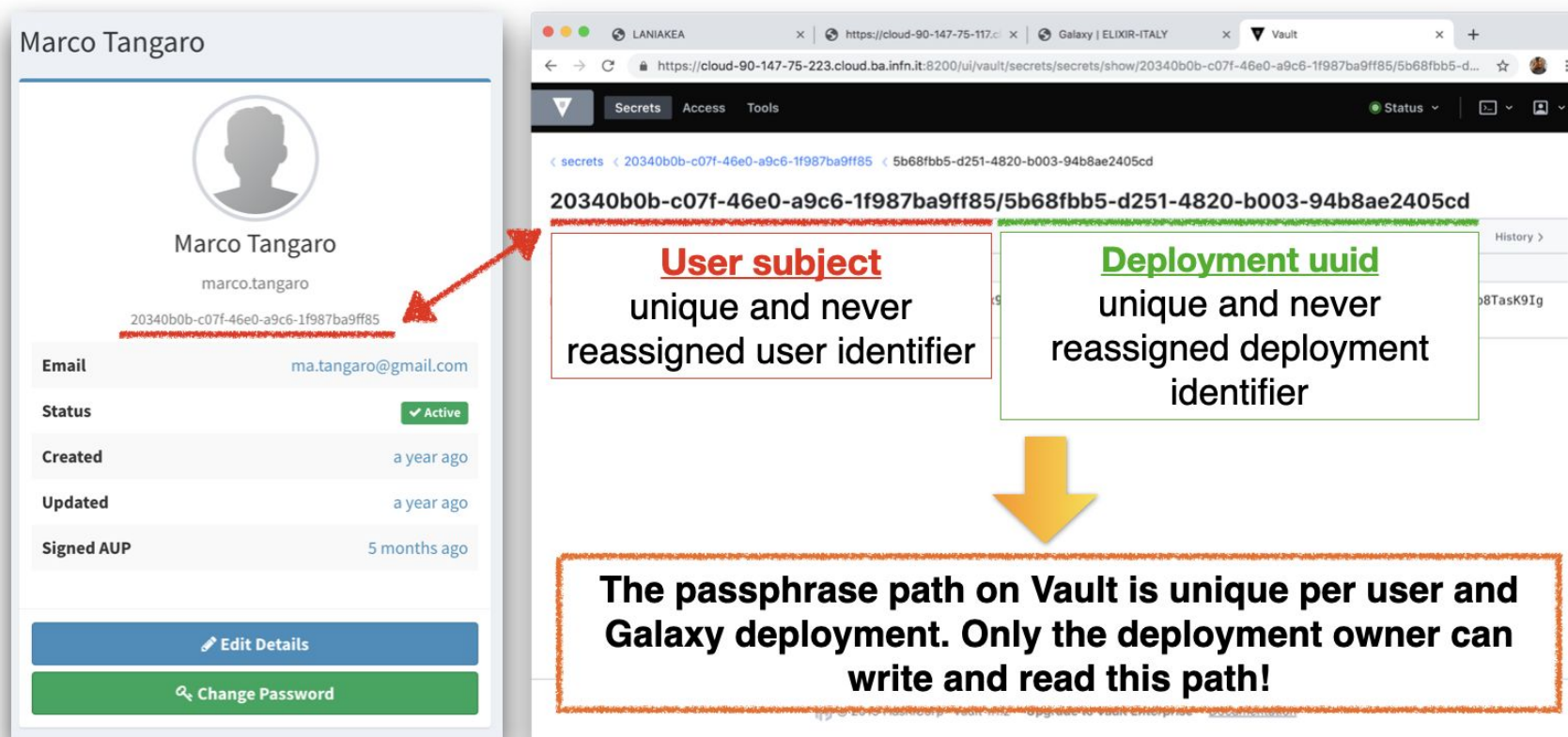
The underlying infrastructure
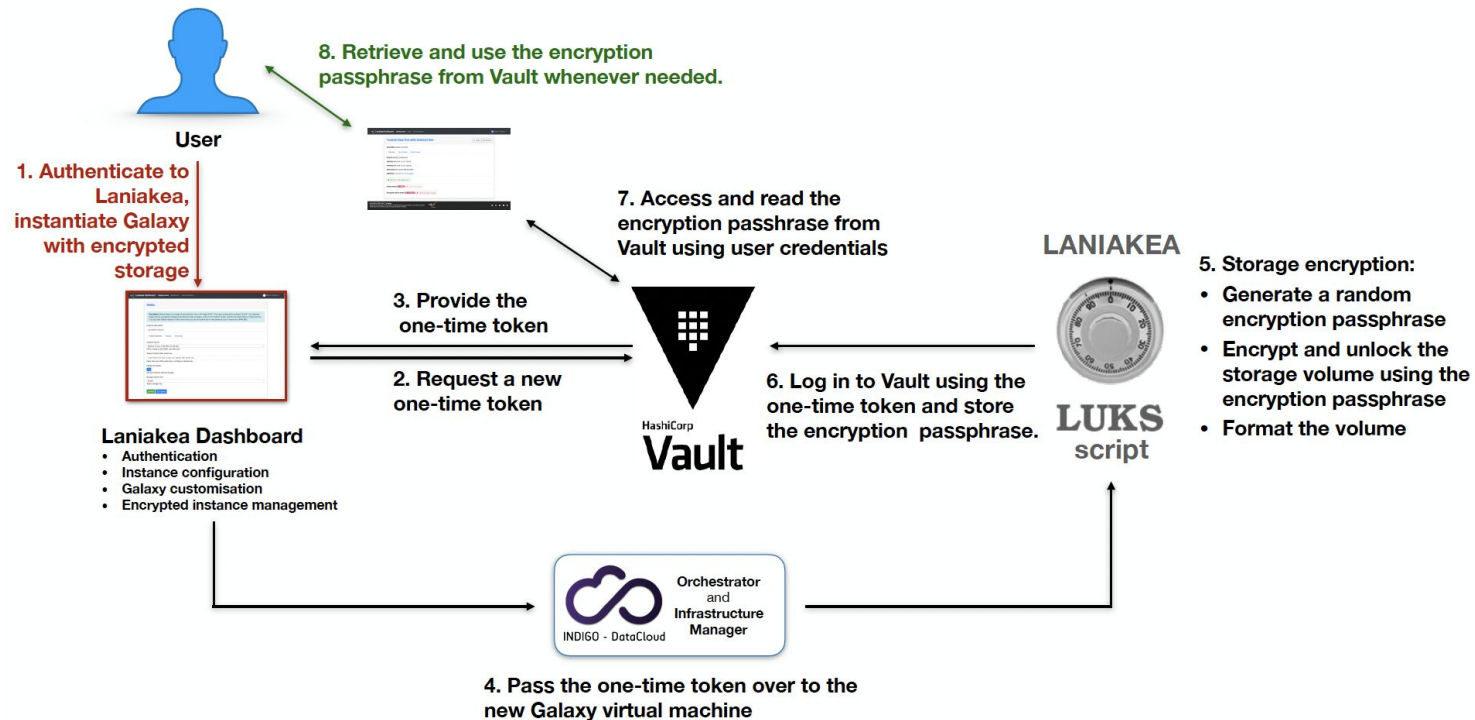
1. User Authentication.
2. **A short lived, write only token, usable only once**, is delivered to the Laniakea encryption script on the VM. There's no update policy: this token can't overwrite other passphrases for security reasons.
3. The Storage volume is encrypted by Laniakea pyLUKS package.
4. The passphrase is sent to Vault by Laniakea pyLUKS.
5. After the instance has been successfully deployed the user can retrieve his password through the Dashboard.
6. The user reads the password on the Dashboard.

The underlying infrastructure

Key management: passphrase path on Vault

User identity in IAM

Passphrase path on Vault

Applications credentials
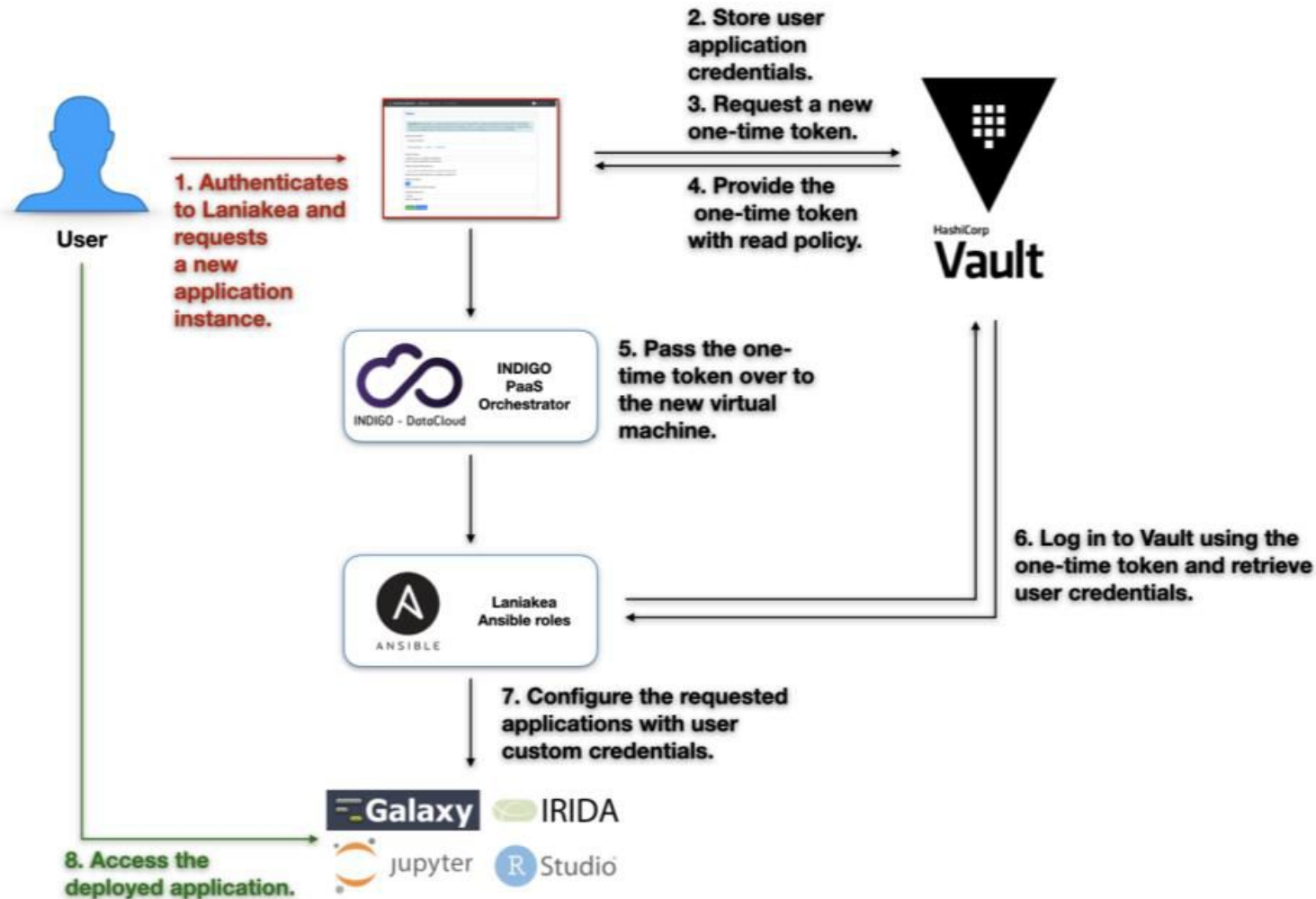
The storage encryption procedure has been extended to allow also users' credentials customisation for many applications.

In this case Ansible is responsible to retrieve credentials from Vault and correctly configure the application.

Deployments under VPN

Deployments under VPN

VPN isolated environments - Automatic deployments of virtual environments on private networks.

Isolation is reached using Tenant and security groups properties, granting the access only through VPN authentication.

User authentication to the VPN using the same Laniakea credentials.
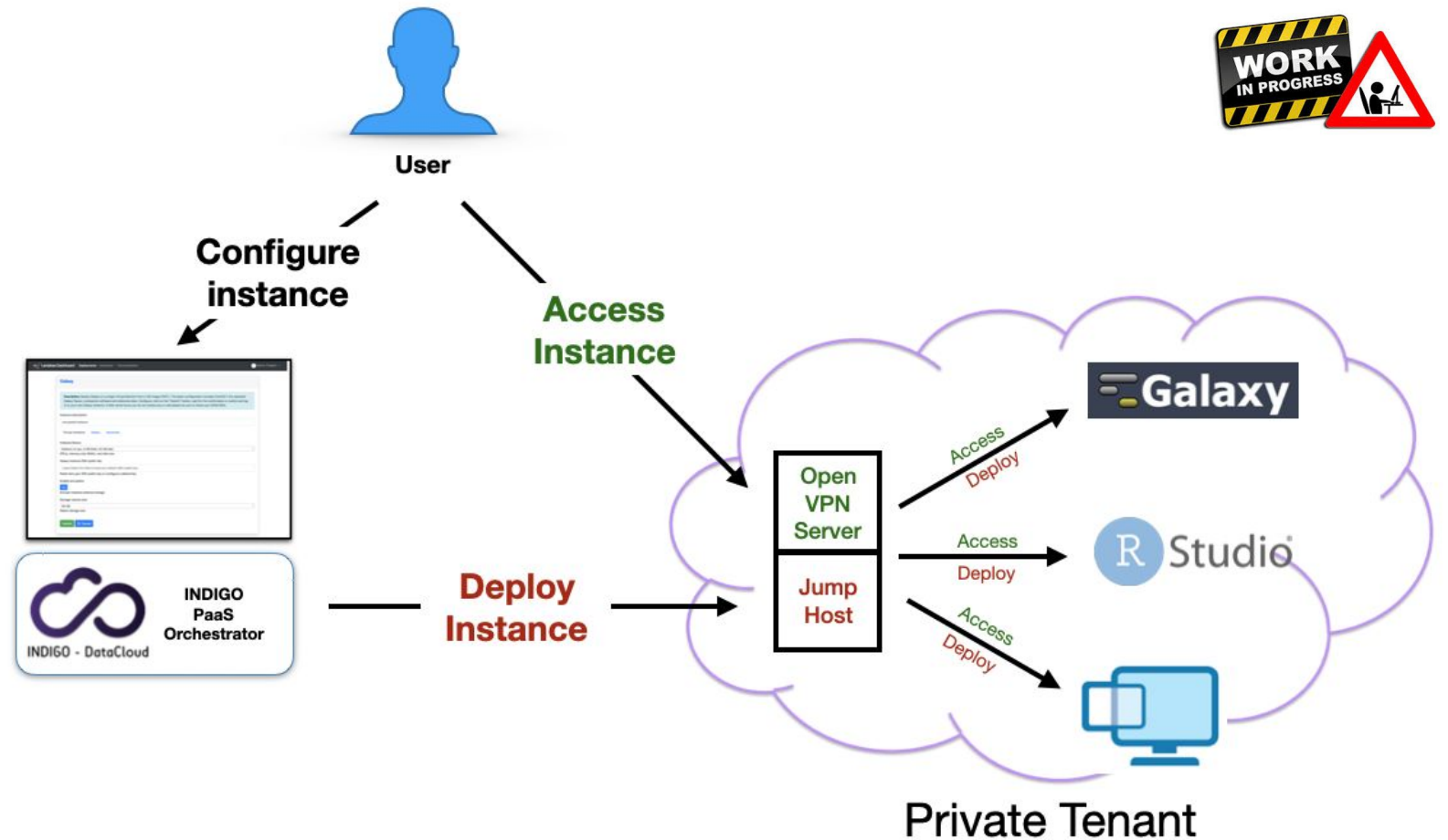
Deployments under VPN

The VPN is based on OpenVPN, with clients and server are configured to use TPC protocol.

We have developed a PAM plugin to enable authentication through OpenID Connect, exploiting Oauth2 device flow:

1. the user connects to the VPN server using an OpenVPN client
2. PAM is configured to send verification code by mail to the user.
3. the user can authenticate with its own Laniakea credentials.
4. the OIDC provider (INDIGO-IAM) sends the access token to the VPN server, that is now able to verify users identity and authorizations.
5. if the user owns the right tenant permissions, he is granted access to the private network and can finally interact with the deployed application
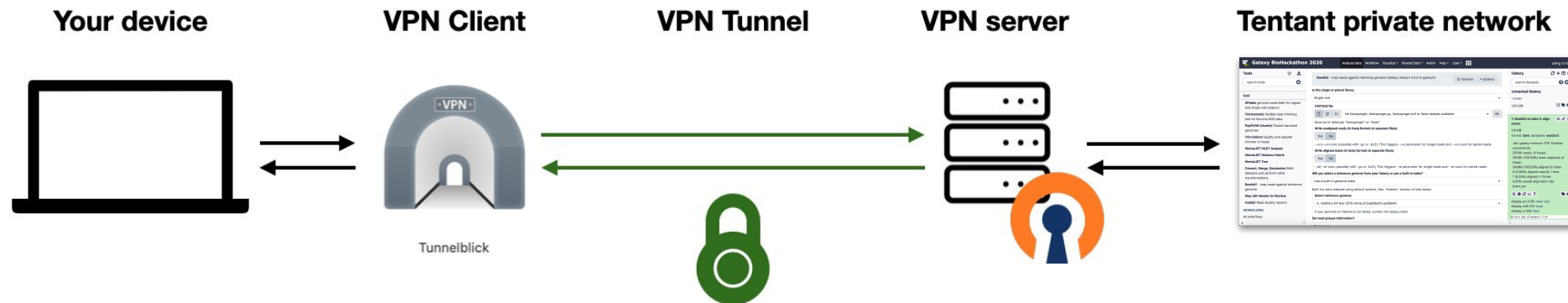
Conclusions

Storage encryption solution is already in production and exploited by several Laniakea Galaxy instances.

Data are still potentially exposed to attacks against the VM itself, where Galaxy or other applications need to consume them. We are working to provide Laniakea's users with the possibility to hide deployed applications behind a Virtual Private Network, achieving even more robust isolation of the research environment.

These approaches can help promoting the adoption of the on-demand model for Life Science and biomedical applications, making compute infrastructures more readily available to potential users even in the case of tight requirements for data protection.

Thanks for your attention

CONTACTS:

Graziano Pesole (ELIXIR-ITALY Head of Node) g.pesole@ibiom.cnr.it

Federico Zambelli (ELIXIR-ITALY technical coordinator) federico.zambelli@unimi.it

Giacinto Donvito (Compute platform ELIXIR-ITALY) giacinto.donvito@ba.infn.it

Nadina Foggetti (Legal expert) nadina.foggetti@ba.infn.it

Marica Antonacci (PaaS developer) marica.antonacci@ba.infn.it

Marco Antonio Tangaro (Laniakea chief developer) ma.tangaro@ibiom.cnr.it

Backup

- Tools graphical user interface.
- Workflow graphical user interface
- "App store" to all Galaxies worldwide
- Tools dependencies automatically solved

Allowing the community to move from command line tools to web user interfaces.
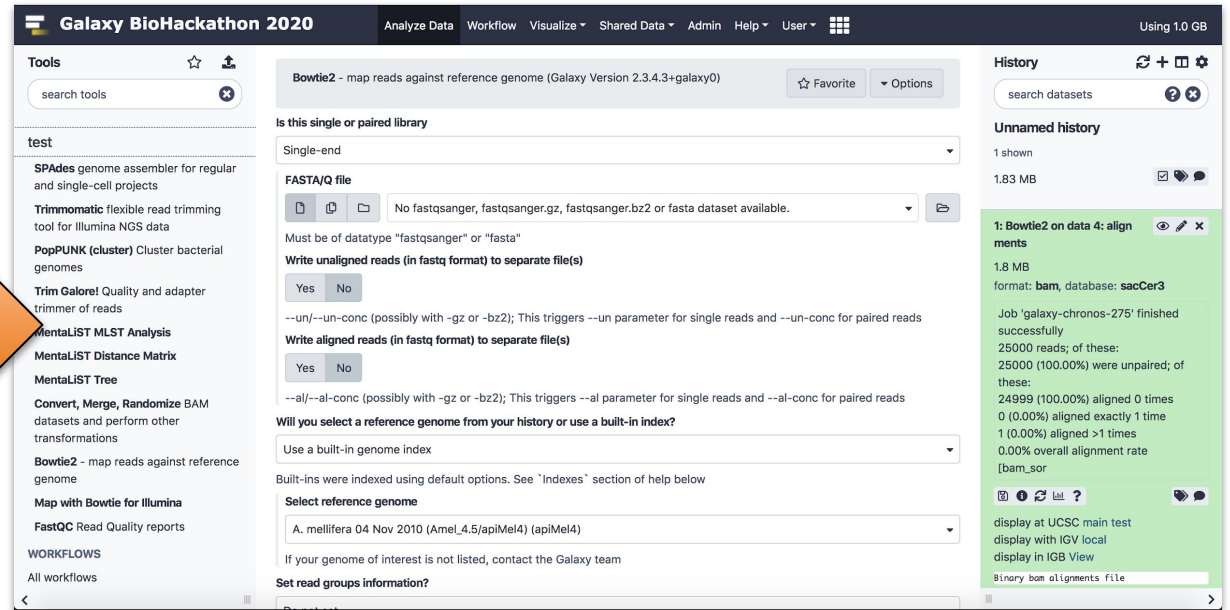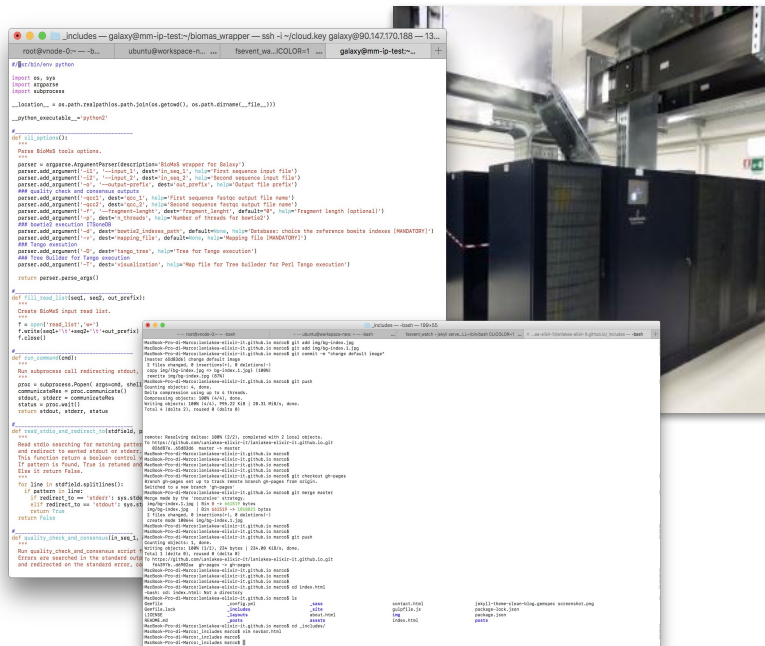
Allowing multiple users to exploit homogenous software environments, enhancing reproducibility.

Laniakea

ELIXIR-Italy partners are actively involved in the service development and/or also contribute with cloud resources.

A Laniakea service is in production for ELIXIR-ITALY partner but also for ELIXIR and external users.

The ELIXIR-ITALY **Laniakea@ReCaS** Call offers access to Cloud resources to be used for the deployment of on-demand Galaxy instances.

https://laniakea-elixir-it.github.io/laniakea_at_recas

Laniakea

The Laniakea Dashboard home page.

Each tile provides a quick explanation of the application and links to the configuration and launch section.

Also more applications available: Jupyter, RStudio, ...
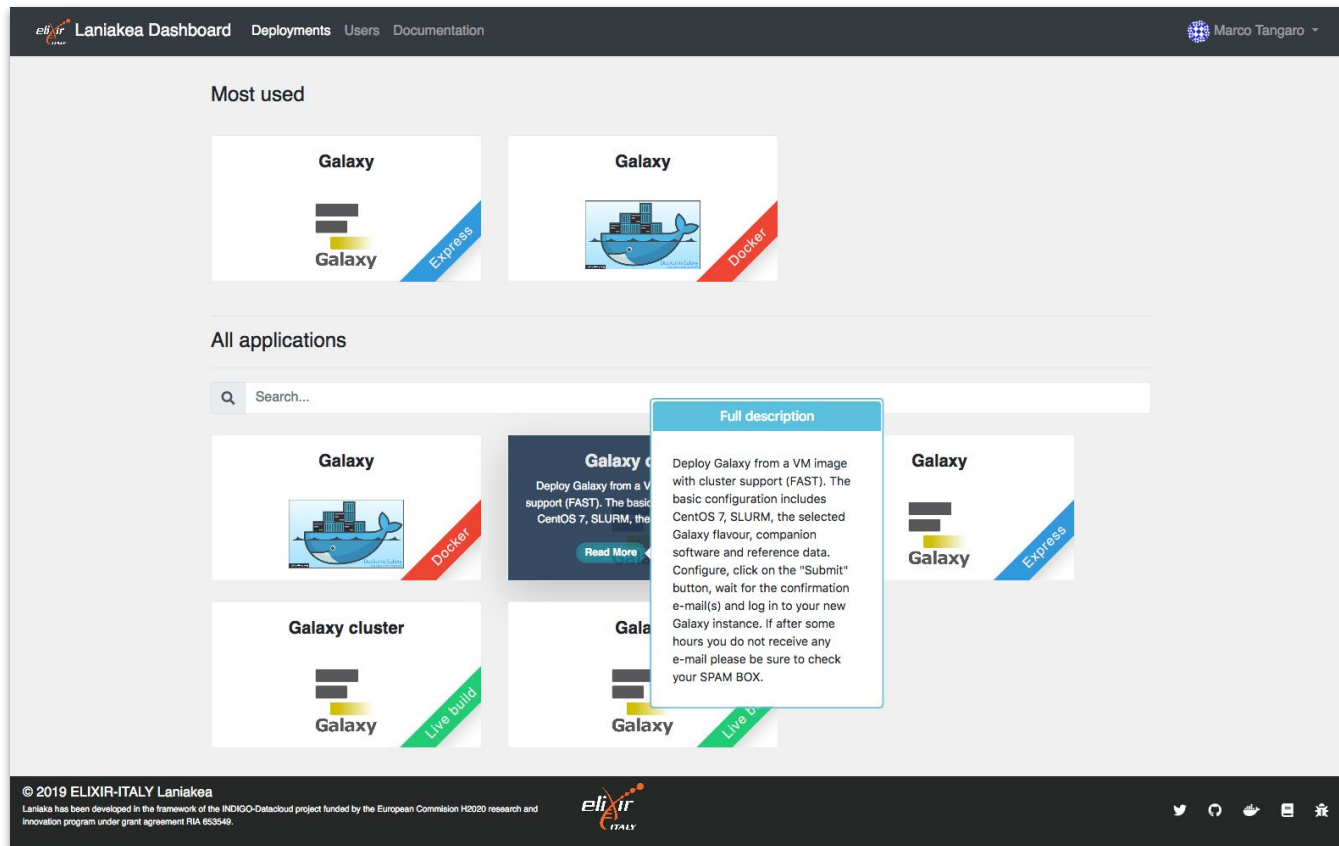
Laniakea

Different deployment strategies:

Live Build: build Galaxy from scratch -> always up-to-date (deployment time depending by the tools number).
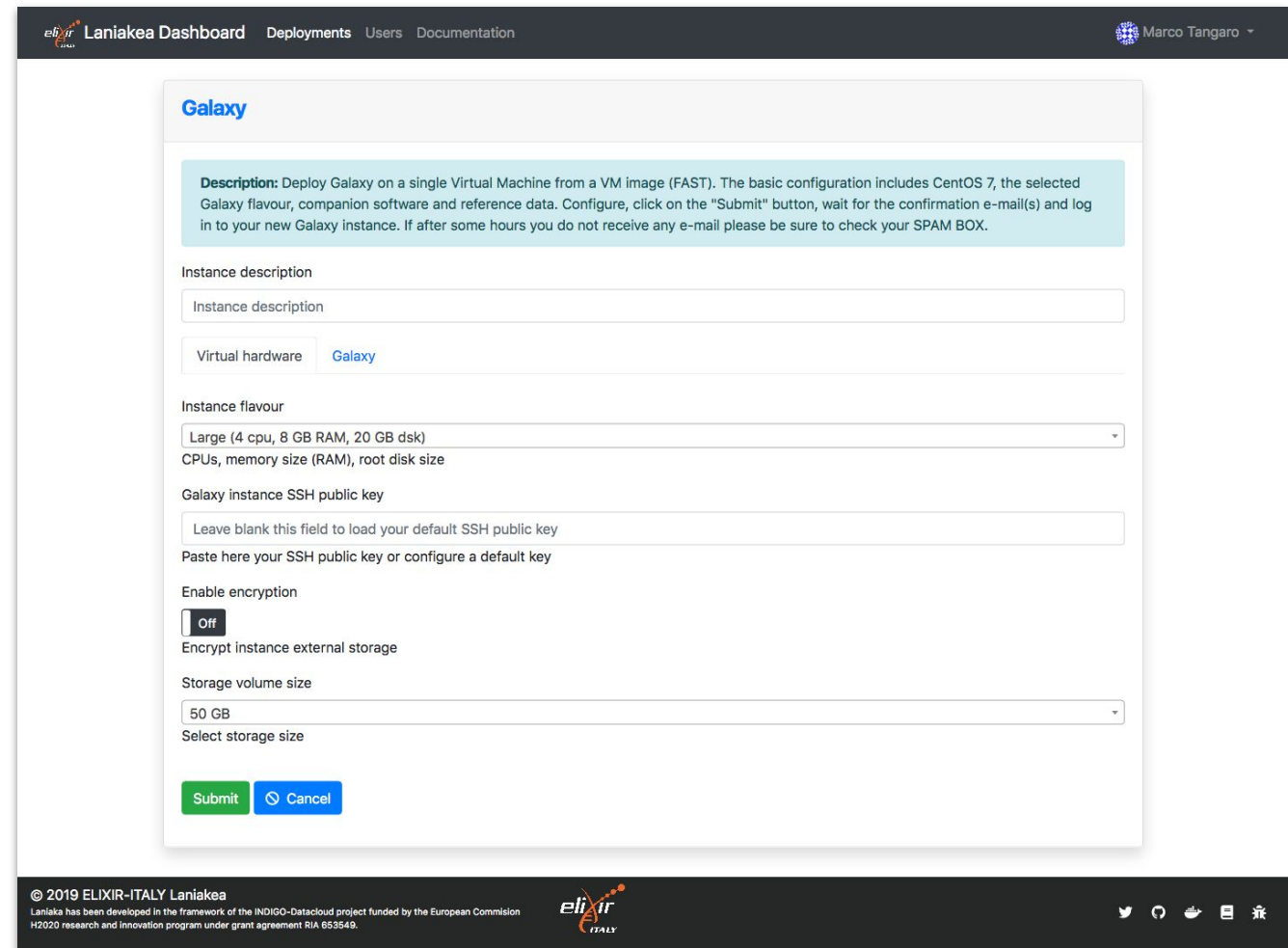
Express: pre-built Galaxy images -> fast deployment, but tools not always at the last available version.

Docker: fast deployment of new flavours.
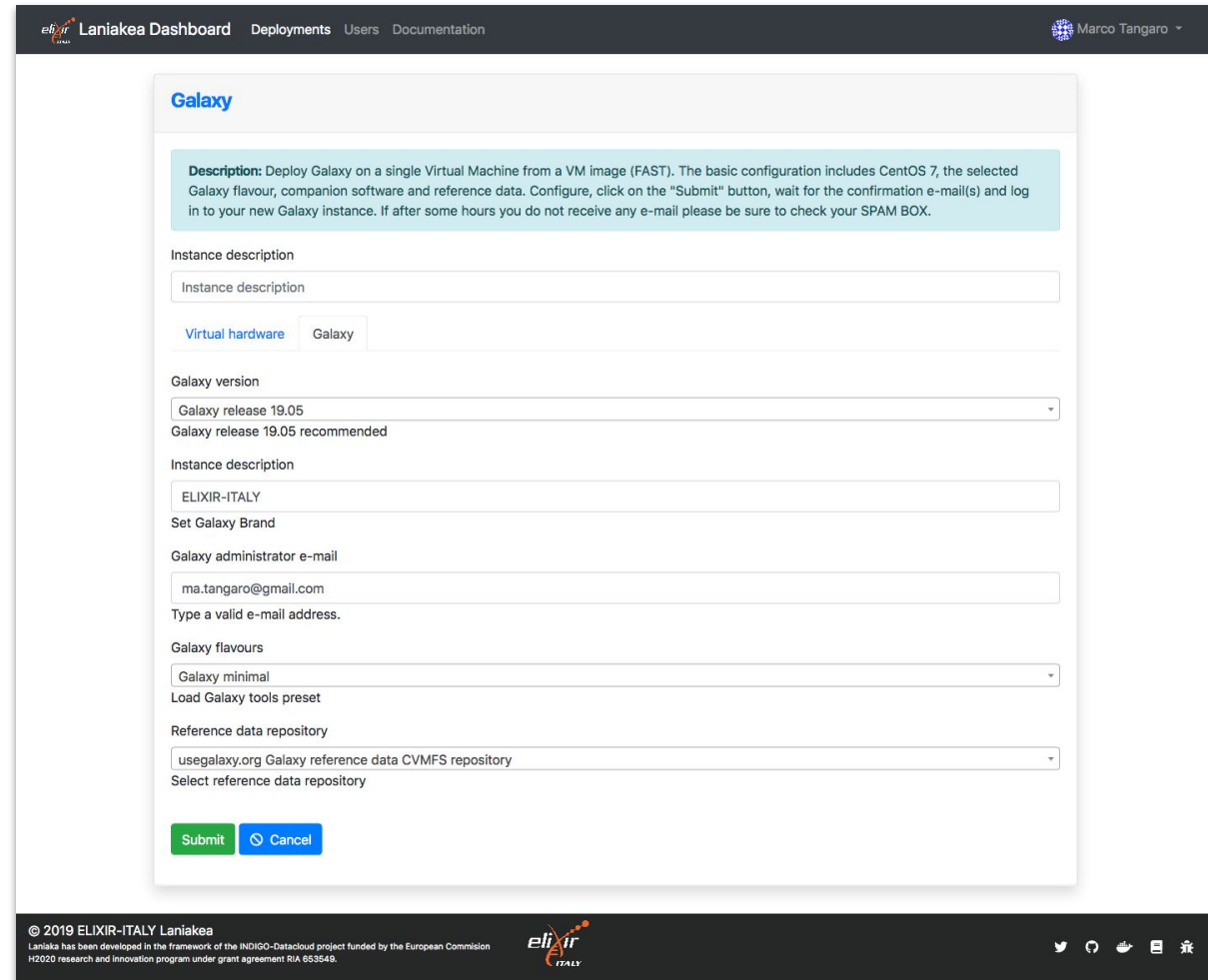
Laniakea

The web front-end provides different tabs to configure your Galaxy.

Virtual hardware: CPU, RAM and Storage

51

Laniakea

The web front-end provides different tabs to configure your Galaxy.

Galaxy software: version, credentials, flavor and reference data.

Laniakea

Laniakea

Galaxy instances management view

Laniakea architecture (simplified view)
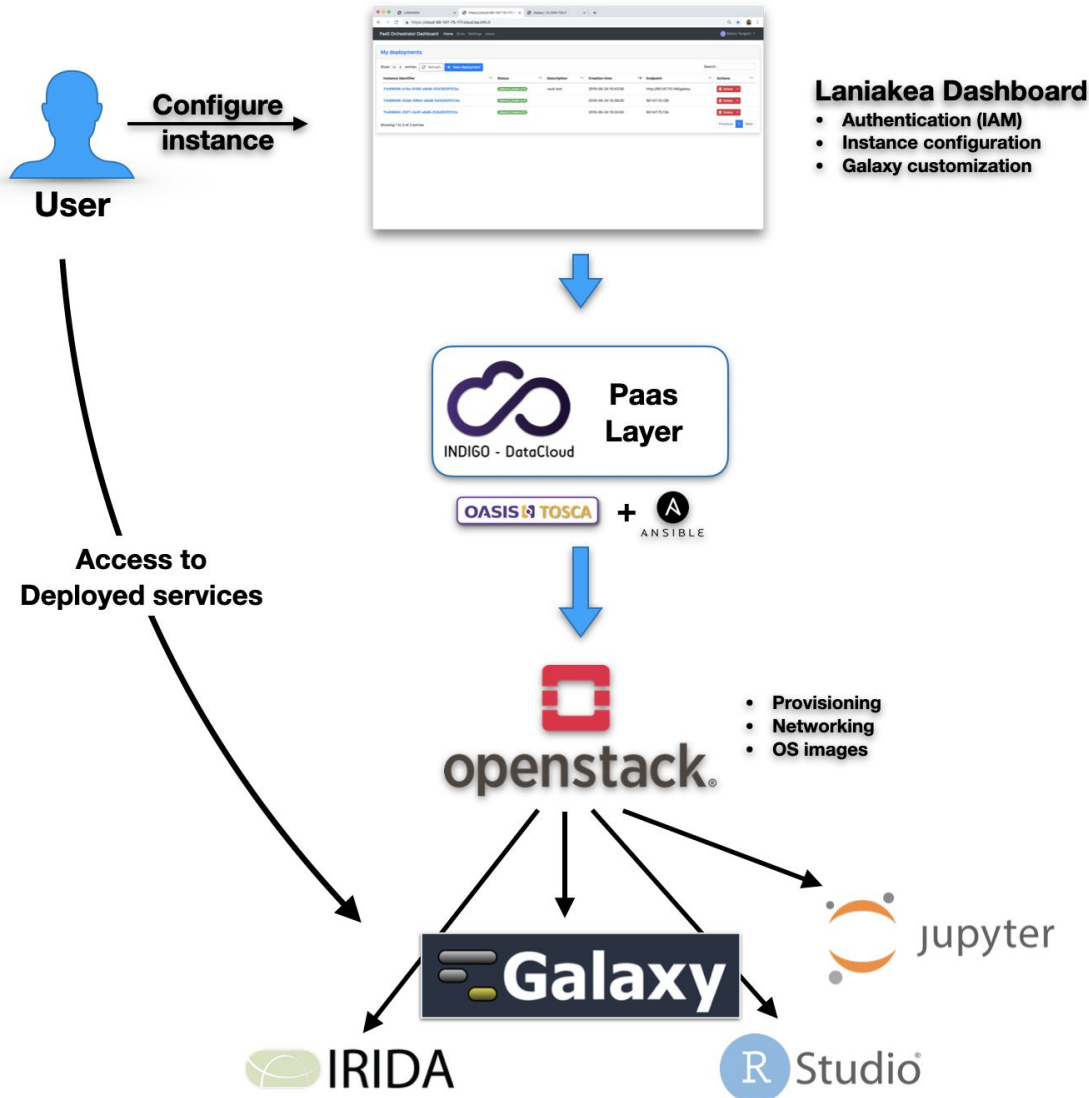
Laniakea Dashboard
- **Authentication (IAM)**
- **Instance configuration**
- **Galaxy customization**

- **Dashboard -** User friendly access to configure and launch a Galaxy instance

- **INDIGO PaaS -** Galaxy automatic deployment

- **Cloud Providers -** ReCaS-Bari

55

Laniakea@ReCaS

Currently, some important Italian Institutions are using Laniakea for their daily work:

- Istituto Ortopedico Rizzoli (2 internal Galaxy servers).
- Istituto Zooprofilattico Sperimentale della Puglia e della Basilicata (2 internal Galaxy servers and 1 IRIDA instance).
- Ospedale Pediatrico Giannina Gaslini (public server).
- University of Milan (public Galaxy server and tools development).
- IBIOM-CNR (public Galaxy server and tools development).
- University of Turin (training)

… and counting.
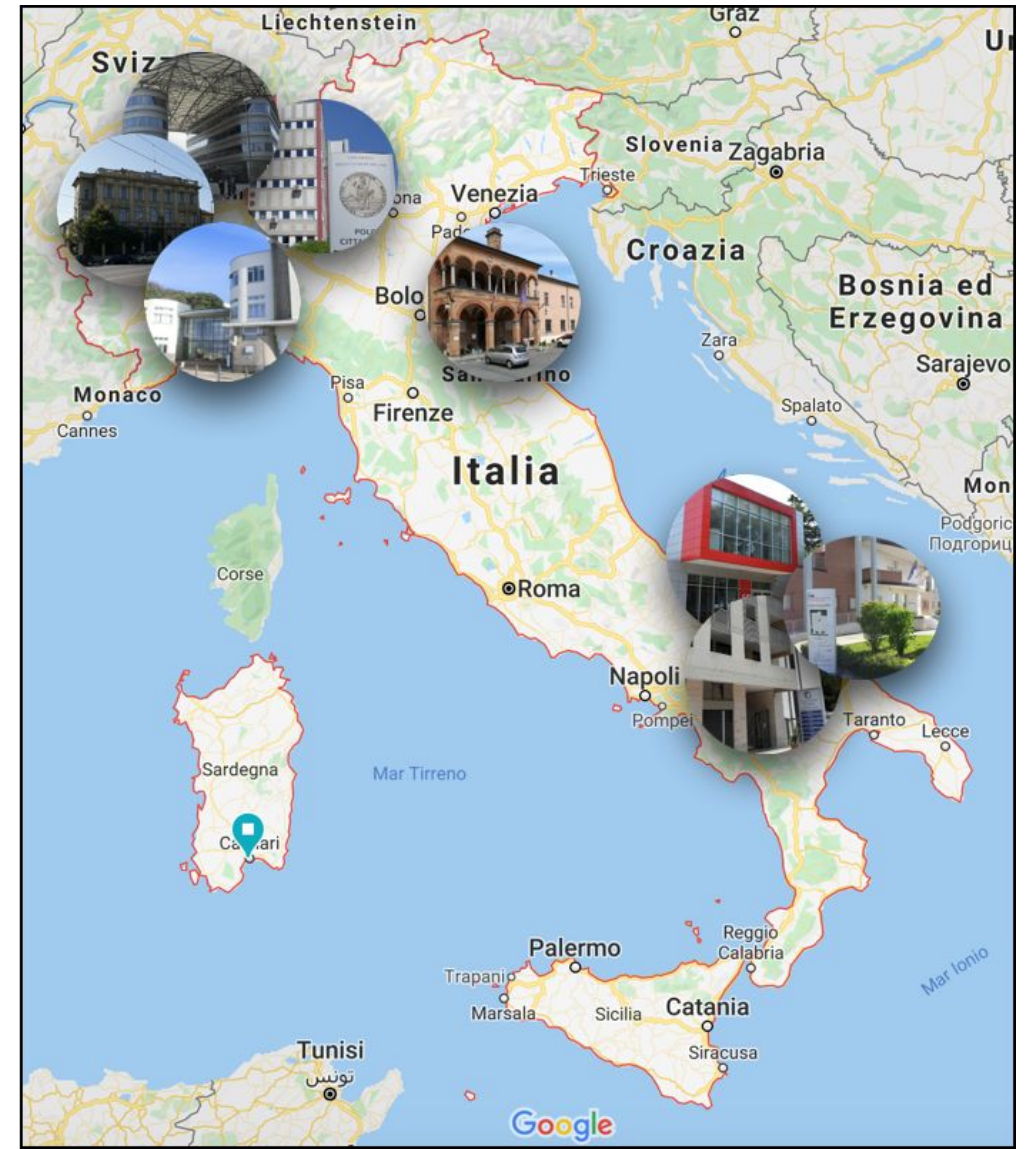
Laniakea Dashboard

The PaaS Layer accepts deployment requests in the form of TOSCA Templates: a document (YAML syntax) describing the infrastructure to deploy, e.g. the virtual hardware and the software to be installed and configured. The Dashboard parses the TOSCA document automatically and renders the user interface with user friendly forms. This allows to extend Laniakea functionalities just adding new templates without any code modification.

We extended the TOSCA templates inputs to create configurable forms. This creates a flexible web interface, allowing straightforward customisation of the user experience through human readable YAML configuration files. The dashboard template is modular and can be easily adapted adding new functionalities to the user interface (e.g. adding a dropdown menu, text fields, toggles...) based on the Laniakea administrator requirements.

Laniakea Dashboard
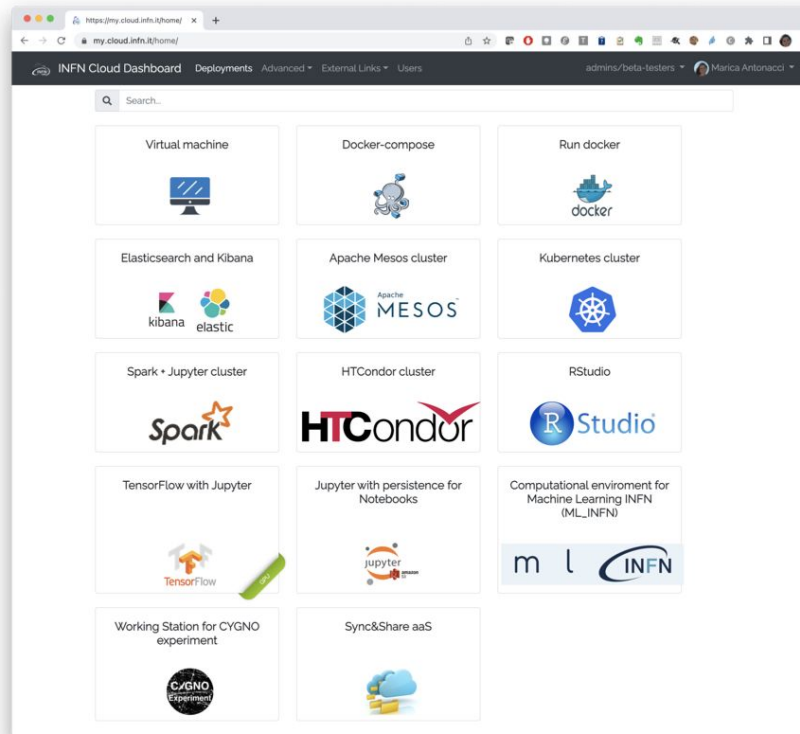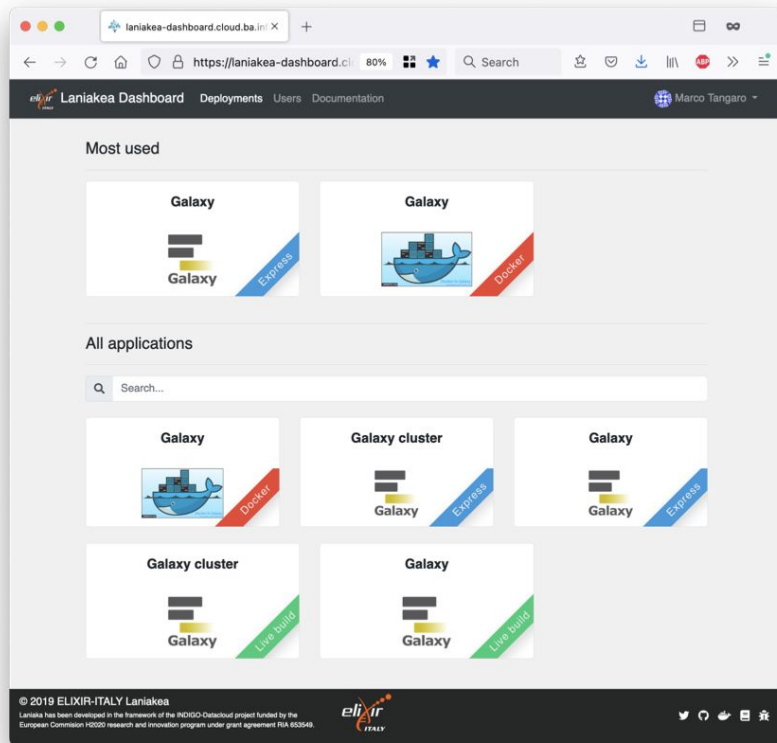
Laniakea Dashboard

Flask web micro-framework (flask.pocoo.org/),

Jinja2 template engine (jinja.pocoo.org/)

Bootstrap 4 toolkit (getbootstrap.com/).

Integrated with Hashicorp Vault for user secrets management.

Laniakea Dashboard


```
instance_flavor:                              TOSCA template
    type: string
    description: instance flavor (num_cpu, memory, disk)
    default: small
```

```
instance_flavor:                                          Parameters file
  display_name: "Instance flavour"
  tag_type: "select"
  description: "CPUs, memory size (RAM), root disk size"
  constraints:
    - { value: "small", label: "Small (1 cpu, 2 GB RAM, 20 GB dsk)" }
    - { value: "medium", label: "Medium (2 cpu, 4 GB RAM, 20 GB dsk)" }
    - { value: "large", label: "Large (4 cpu, 8 GB RAM, 20 GB dsk)" }
    - { value: "xlarge", label: "xLarge (8 cpu, 16 GB RAM, 20 GB dsk)" }
  tab: "Virtual hardware"
```

User input field in the TOSCA template to select instance flavour in terms of vCPUs, RAM and root disk storage.

Dashboard render

Virtual hardware   Galaxy   Advanced

Instance flavour

Small (1 cpu, 2 GB RAM, 20 GB dsk)

CPUs, memory size (RAM), root disk size

Dashboard render

Virtual hardware   Galaxy   Advanced

Instance flavour

Small (1 cpu, 2 GB RAM, 20 GB dsk)

Small (1 cpu, 2 GB RAM, 20 GB dsk)

Medium (2 cpu, 4 GB RAM, 20 GB dsk)

Large (4 cpu, 8 GB RAM, 20 GB dsk)

xLarge (8 cpu, 16 GB RAM, 20 GB dsk)

Laniakea Dashboard

Laniakea Dashboard

Video demo

https://youtu.be/bzJbHby89AY

Video outline:

- Authentication
- SSH key management
- Galaxy Deployment with encrypted storage
- RStudio/JupyterLab deployment
- How to build and encrypted deployment (from a simple VM to a full encrypted instance with Galaxy).