# Openstack Cloud Application Catalog

August 2017

**AUTHOR(S):**
**Victor Araujo**

**SUPERVISOR(S):**
**Belmiro Moreira**

CERN openlab

# PROJECT SPECIFICATION

The OpenStack project Murano allows users to spawn applications from a predefined catalog. This can be useful for unexperienced users that can get a preconfigured application with one click. Typical applications already available are listed at the Openstack app catalog website, but these can be further enhanced with local applications.

The project would be to investigate the use of Murano for potential future use on the CERN cloud.

# ABSTRACT

Cloud computing has transformed IT operations throughout many industries and organizations, enabling agile self-service models that allow users to get configurable resources in minutes or seconds. CERN has implemented an Openstack cloud that is used by physicists to analyze data from experiments and by other CERN staff to implement various services. The Openstack Murano project provides a cloud application catalog, which is an opportunity to give users of the CERN cloud the option to quickly deploy standardized application environments in a self-service model. Openstack Murano was successfully deployed in CERN's cloud, creating the necessary configuration management modules and adapting its code to work in CERN's Openstack production environment. The application catalog is available internally, providing several databases and application servers that can be deployed in minutes with just a few clicks. More applications can be added in the future to the application catalog, with the potential to serve different use cases for the cloud at CERN.

# TABLE OF CONTENTS

## 1. INTRODUCTION

Cloud computing enables convenient, on-demand access to configurable computing resources [1]. It is a driver for the efficient operation of Information Technology (IT) in organizations, by enabling a quick self-service model that has the potential to efficiently consolidate pools of computing resources. At CERN, cloud computing is used to provide computational resources to physicists, in order to analyze and model data related to CERN's scientific experiments. The cloud at CERN is based on Openstack [2], an open-source project for cloud Infrastructure-as-a-Service (IAAS) [1]. The Openstack deployment at CERN is one of the largest in the world, numbering more than 200 thousand computer cores in August 2017.

Openstack is made of several subprojects, which provide different cloud-related functionalities. Openstack Murano [3] is a project that provides an application catalog in Openstack clouds, allowing users to easily create reliable application environments. Murano allows users to browse an interactive catalog, with applications that can be configured and deployed effortlessly in the cloud. This can allow experienced or inexperienced users to create and configure databases, application servers and even big data environments with a few clicks. This removes the need to manually create and configure these environments, or to request them to units in IT departments and wait for their response, freeing valuable resources and increasing agility inside the organization.

The goal of this Openlab Summer Student project was to evaluate Murano inside CERN. To evaluate it, it was necessary to deploy it in CERN's Openstack cloud and adapt it to the characteristics of CERN's production environment. Some applications were targeted to be added to the catalog to let some users interact with it. Furthermore, these applications were adapted to be used with CERN's Linux images, to show the feasibility of creating packages adapted to CERN's environments and tools.

This project resulted in several outcomes. To deploy Murano, a Puppet [4] hostgroup was created to manage its configuration. The Murano Core Library was modified to work in CERN's production environment, without relying on Neutron networks, floating Ips and security groups. Some patches were contributed upstream to fix bugs in the Nova Network Driver. A set of application packages for Murano were created or extended from the community packages, with support for CERN's CC7 Linux image.

## 2. OPENSTACK MURANO

The Murano project provides an application catalog in Openstack environments. It provides a graphical web user interface and an API to compose and deploy application environments. Murano interacts with other Openstack components, like Openstack Heat, to orchestrate the deployment of the virtual machines that are part of application environments and then install the applications in them.

Applications are added to the catalog as packages. These packages contain classes, UI definitions and resources. Classes specify the logic necessary to handle the deployment of resources in Openstack, like the creation of virtual machines or the specification of security groups. The UI definitions are used in the web dashboard of the application catalog to configure environments, prompting the user for configuration options through web forms. Resources are scripts and other files that are copied to the virtual machines and executed. Tipically, resources contain the node-level logic to install and configure the applications through shell scripts.

Murano specifies three main components in its architecture, which interact among themselves and with other Openstack services to implement the functionalities of the application catalog. First Murano-API provides the point of interaction with the user. Murano-API is consumed by the CLI and the Horizon web dashboard. The API allows the users to manage packages and applications environments. When the user asks Murano to deploy applications in an environment, processing is taken over by another component, murano-engine. Murano-engine executes the logic defined in the classes of the package, and interacts with Heat to orchestrate the cloud resources that are required by the application. After the virtual machines have been spawned and configured, murano-engine sends execution plans and the resources defined in the package to Murano agents running in the Vms. These scripts and resources tipically install and configure the application inside the VM. Murano agents are installed in preconfigured images, or installed through cloud-init in the virtual machines.

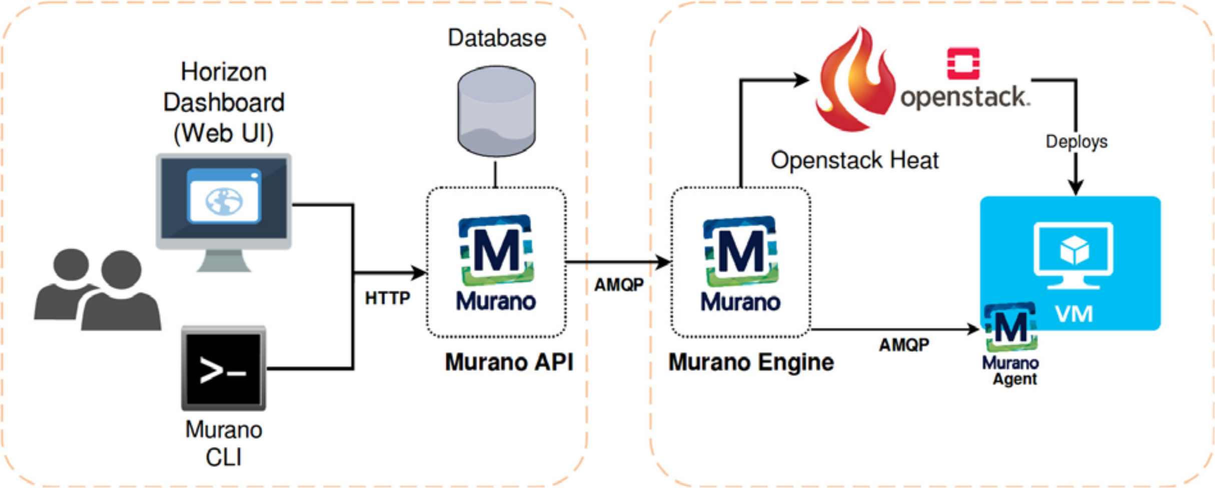


Figure 1. Murano Architecture implemented at CERN's Openstack cloud

## 3. IMPLEMENTATION

Murano was first installed in a local machine using devstack, to learn more about its functionalities and its components. Next, it was installed and configured in a development environment in CERN's cloud. Puppet is used at CERN for configuration management. Services are placed into hostgroups, whose configuration is managed using Puppet modules. Additionally, modules and hostgroups at different development levels are grouped into environments. This allows features to be tested in development environments, which are separate from the production environment. Environments can be arbitrarily defined, but at least two standard environments are used in the workflow to integrate new features: QA (for testing and development) and production.

A Puppet module was created for Murano's hostgroup inside CERN's cloud. It deploys murano-api, murano-engine and a supporting RabbitMQ. The database needed for murano-api was provisioned through the DB On Demand service from CERN's IT database group. The module for the hostgroup relies on the official Puppet module for Murano, with some modifications. The hostgroup module holds the specific configurations needed for CERN's cloud, such as the usage of Nova Network instead of Neutron.

After the Puppet module has deployed the components of Murano, the core library of Murano needs to be uploaded. The core library is available in the *meta* directory of Murano's Openstack repository. The core library required modifications to work in CERN's environment. The modifications are available in the cloud-infrastructure/murano-apps repository in CERN's Gitlab, in the *meta* directory. The modifications can be summarized as follows:

1) Use the DummySecurityGroupManager for NovaNetwork, to avoid adding security groups to the generated Heat template.
2) Disable the security_groups output in the generated Heat template, which is added by default for all instances.
3) Pass securityGroups parameter to join instances in Nova Network (patch merged upstream).
4) Fix a bug where the output for assigned IPs was not added when using Nova Network and not selecting floating IPs (patch merged upstream).

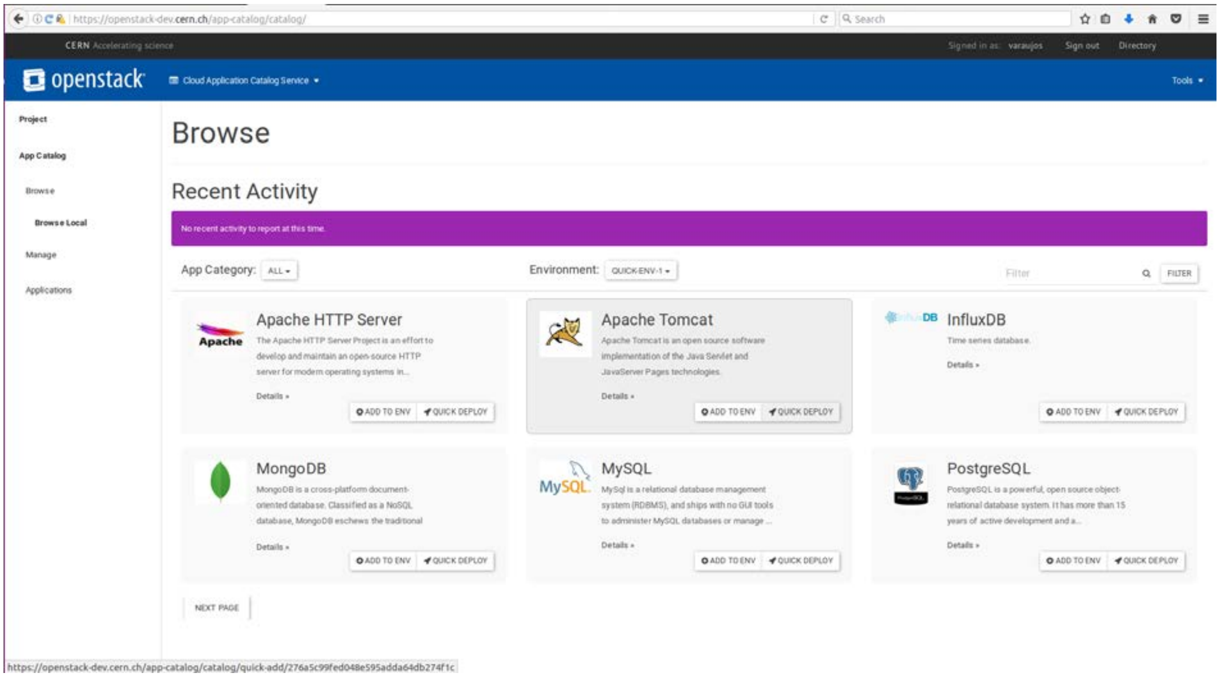
After finishing the correct configuration of the hostgroups and fixing the issues with the core library, the application catalog worked in the QA environment of CERN’s cloud. The following example packages provided for Murano were uploaded to the application catalog and tested:

1. Apache Web Server
2. MySQL
3. MongoDB
4. InfluxDB
5. PostgreSQL
6. Tomcat

These packages use Ubuntu or Debian images, so the installation scripts were modified to support CentOS 7 deployments, one of the standard Linux distributions used currently at CERN. Packages were also modified to disable floating IPs in the web configuration forms, since floating IPs are not supported in CERN’s Openstack cloud.

## 4. RESULTS

The Murano application catalog was successfully deployed and adapted to CERN’s cloud. It has packages to deploy Apache Web Server, MySQL, MongoDB, InfluxDB, PostgreSQL and Tomcat.



Browse

Recent Activity

No recent activity to report at this time.

Apache HTTP Server

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems in...

Details »

Apache Tomcat

Apache Tomcat is an open source software implementation of the Java Servlet and JavaServer Pages technologies.

Details »

InfluxDB

Time series database.

Details »

MongoDB

MongoDB is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional

Details »

MySQL

MySql is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage ...

Details »

PostgreSQL

PostgreSQL is a powerful, open source object-relational database system. It has more than 15 years of active development and a...

Details »

https://openstack-dev.cern.ch/app-catalog/catalog/quick-add/276a5c99fed048e595adda64db274f1c

*Figure 2. The Murano application catalog at CERN*

Applications can be quickly deployed by browsing the catalog (`App Catalog/Browse/Browse Local` in Horizon) and pressing the Quick Deploy button in the application's description. The forms for configuration will be shown to the user and an environment will be created automatically. The steps are the following:

1. Login to the Openstack Dashboard.

2. Navigate to `App Catalog/Browse/Browse Local`.

3. Click on the `Quick Deploy` button of the application you want to deploy.

4. Specify the desired options in the configuration dialog and click the `Create` button.

5. Deploy the application by clicking on `Deploy This Environment`.

Environments can be created manually in `App Catalog/Applications/Environments`. A name and a default network have to be specified. After the environment is created, applications can be dragged to be added to the environment. After pressing the "Deploy this Environment" button, Murano will begin the deployment and report back the completed steps or errors that come up during the deployment. The detailed steps are as follows:

1. Login to the Openstack Dashboard.

2. Navigate to `App Catalog/Applications/Environments`.

3. Click on the `Create Environment` button.

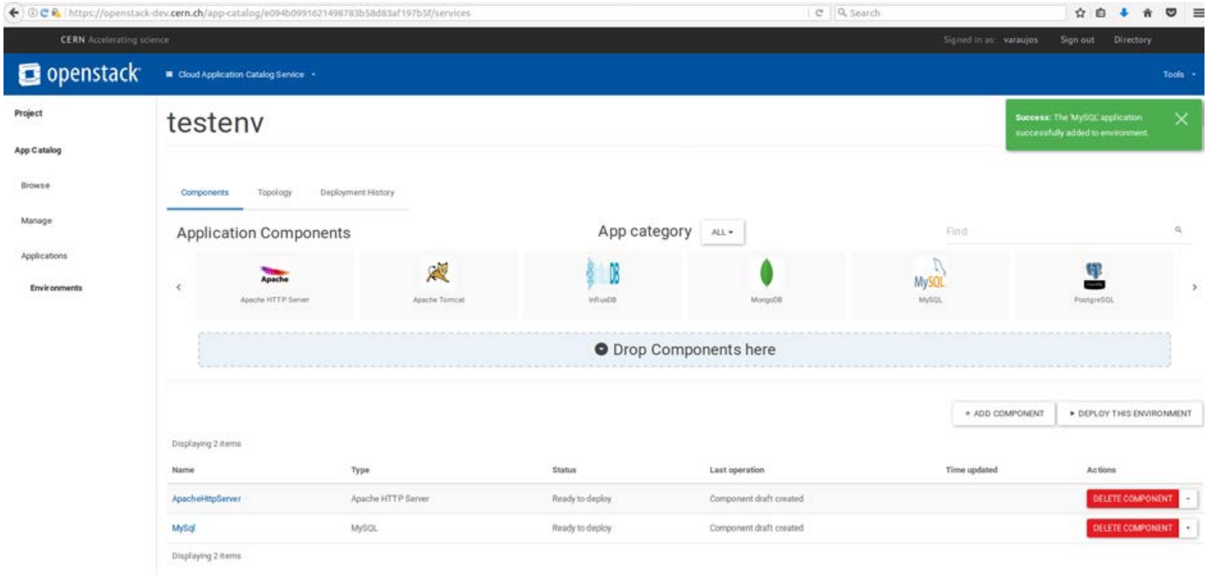
4. Fill the environment name and leave the default option for the network.

5. Click on `Create`.

6. Add applications by dragging them to the `Drop components here` section or by clicking on `Add Component`.

7. Fill the configuration form for every added application.

8. Finally, click on `Deploy This Environment`. Notifications about the status of each application will be shown in the same page. Deployment history or the logs of the latest deployment can be checked using the available tabs in the environment's page.



testenv

**Success:** The 'MySQL' application successfully added to environment.

Displaying 2 items

| Name | Type | Status | Last operation | Time updated | Actions |
|---|---|---|---|---|---|
| ApacheHttpServer | Apache HTTP Server | Ready to deploy | Component draft created | | DELETE COMPONENT |
| MySql | MySQL | Ready to deploy | Component draft created | | DELETE COMPONENT |

Displaying 2 items

Figure 3. Deploying applications in environments

## 5. CONCLUSIONS

Murano enables convenient self-service application deployment. It allows both experienced and inexperienced users to quickly provision standardized application environments. This can increase organizational agility and free valuable human resources from having to manually provision environments related to common use cases. A particular example is database provisioning, where instead of having to request a database to another IT group and wait for the fulfillment of the request, it can be provisioned in minutes through the application catalog.

Future work at CERN will need to further adapt the application catalog to the needed use cases and policies of CERN. The development of packages should be guided to serve common use cases, and to make sure that they comply with internal security standards.

## 6. REFERENCES

[1] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

[2] Sefraoui, O., Aissaoui, M., & Eleuldj, M. (2012). OpenStack: toward an open-source solution for cloud computing. *International Journal of Computer Applications*, *55*(3).

[3] Openstack. (2017). Openstack Murano. Available at: https://docs.openstack.org/murano/latest/

[4] Loope, James. (2011) *Managing Infrastructure with Puppet: Configuration Management at Scale*. " O'Reilly Media, Inc."