DEEP-Hybrid-DataCloud releases Documentation

DEEP-Hybrid-DataCloud consortium

Jan 15, 2021

Contents

You'll find here useful information regarding the DEEP-HybridDataCloud services and components **releases**, their schedules, documentation and support.

CHAPTER 1

DEEP-Hybrid-DataCloud releases

1.1 DEEP-2 Rosetta

The DEEP - HybridDataCloud project is pleased to announce the general availability of its **second public software release**, codenamed **Rosetta**

1.1.1 Included components

Alien4Cloud - DEEP plugin

Table of Contents

- *Alien4Cloud - DEEP plugin*
    - *Release Notes*
    - *Documentation*
    - *Support*

The **Alien4Cloud - DEEP plugin** is the first step towards the adoption of a visual composition approach of TOSCA templates (based on Alien4Cloud), in order to facilitate the widespread adoption of the standard, and its integration with the INDIGO-DataCloud Orchestrator, which is already part of the EOSC-HUB service catalogue.

- With this approach, the user can visually compose complex applications that involve, for example, the dynamic deployment of a container orchestration platform on an IaaS Cloud site that executes a highly-available Docker-based application to facilitate application delivery.

- The users can also deploy an Apache Mesos cluster with GPU support that contains a deep learning application for the recognition of certain plant species, offered as a service to a community of users. This introduces unprecedented flexibility, from visual composition, to the automated application delivery, using a graphical

interface that is already integrated with an Orchestrator layer that performs resource provision from multiple Clouds and application configuration.

- The integration of easy-to-use graphical interfaces builds a bridge between the users and the orchestration services.It also represents a step forward to foster the adoption of innovative computing services that are hidden from the user, as they can focus on the high-level description of the services requirements and definition, instead of working on their technical implementation.

Release Notes

v 2.0-r0

What's new

This is the second major release of the Alien4Cloud - DEEP plugin providing:

- Support for the OpenStack normative TOSCA types (implementation available here)
- Support for the IndigoDC TOSCA custom types (implementation available here)
- Support for DEEP-OC templates (available here)
- Improved support for outputs in the templates
- Outputs editor
- Outputs retrieval and display from the Indigo Orchestrator
- Improved support for TOSCA functions
- Improved support for inputs
- Inputs editor
- IAM authentication support

List of RfCs

Most important features and improvements:

- Jira issues under Alien4cloud-deep component version "Alien4cloud 2.0-r0"

Github issues for DEEP-2 (everything done in the DEEP-2 milestone):

- Alien4Cloud-deep (the plugin & Dockerfile) : https://github.com/indigo-dc/alien4cloud-deep/milestone/1?closed=1
- Alien4Cloud repo (the fork of the official A4C): https://github.com/indigo-dc/alien4cloud/milestone/1

Known Issues

Please see the list of issues at Github

Dependencies

The Alien4Cloud-DEEP plugin v. 2.0-r0 is known to work with the following:

- Alien4Cloud custom version: 2.1.0-UPV-1.0.0-SNAPSHOT
- Plugin version: 2.2.0-SNAPSHOT
- Alien4Cloud Settings Manager version: 2.1.0
- TOSCA custom-types version: 4.0.0
- Spring OIDC social plugin version: 1.4

Installation methods

- automatic deployment by using the Docker image

List of Artifacts

- **Docker Container:**
    - indigodatacloud/alien4cloud-deep:DEEP-2 (signed)
    - indigodatacloud/alien4cloud-deep:2.0-r0

Documentation

Detailed documentation is available at:

- GitHub README

Support

- GitHub issues: https://github.com/indigo-dc/alien4cloud-deep/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/
- Mian developers: Andy S Alic (asalic@i3m.upv.es), Germán Moltó (gmolto@dsic.upv.es)

CloudInfoProvider-DEEP

Table of Contents

The **Cloud Information Provider** generates a representation of cloud resources, that can be published inside a BDII (using the provided LDIF templates for a GlueSchema v2 representation) or any other component like the INDIGO Configuration Management Database (CMDB) (Using specific templates).

The generated representation is described using a Mako template having access to the cloud middleware information.

The current document describe the main features implemented for the Cloud Information Provider under the DEEP-Hybrid-Datacloud project.

Main documentation for the product is maintained upstream, including usage and contribution (for developers) guidelines. The documentation for the features pending to be merged upstream and/or specific to DEEP-Hybrid-Datacloud will be added below, associated with the release version.

Release Notes

v 0.10.5

What's new

- Support for multitenancy in CMDBv1
- Support for new Mesos providers
- Support for fetching GPU information from OpenStack and Mesos

List of RfCs

Installation methods

- yum/apt/python-pip: https://github.com/indigo-dc/cloud-info-provider-deep
- Ansible role: https://github.com/indigo-dc/ansible-role-cloud-info-provider

List of Artifacts

- **CentOS-7 RPMS**
    - cloud-info-provider-deep-0.10.5-1.el7.noarch.rpm
    - cloud-info-provider-deep-openstack-0.10.5-1.el7.noarch.rpm
- **Ubuntu 16.04 DEBS**
    - python-cloud-info-provider-deep-openstack_0.10.5-1_all.deb
    - python-cloud-info-provider-deep_0.10.5-1_all.deb
- **Ubuntu 18.04 DEBS**
    - python-cloud-info-provider-deep-openstack_0.10.5-1_all.deb
    - python-cloud-info-provider-deep_0.10.5-1_all.deb

Documentation

Detailed documentation is available at:

- https://github.com/EGI-Foundation/cloud-info-provider

Documentation specific for the DEEP-HDC version is available at:

- https://github.com/indigo-dc/cloud-info-provider-deep/blob/DEEP/README-DEEP.md

Support

- GitHub issues: https://github.com/indigo-dc/cloud-info-provider-deep/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/
- EGI GGUS Support Unit: https://wiki.egi.eu/wiki/GGUS:EGI_Cloud_Information_Discovery_FAQ
- Main developer: Pablo Orviz (orviz@ifca.unican.es)

CloudProviderRanker

Table of Contents

The **Cloud Provider Ranker** is a standalone REST WEB Service which ranks cloud provider services using rules implemented with the Drools framework. The **INDIGO PaaS Orchestrator** interacts with this service in order to obtain the rank of two or more cloud services depending on the match with specific rules. The aim of this micro component is to fully decouple the ranking logic from the Orchestrator's business logic.

Release Notes

v 0.7.0

What's new

This is the first release that sees contributions from the DEEP-HDC project it features a complete code re-engineered using Spring Boot framework, modified ranking algorithm, many improvements and bug fixes described bellow

List of RfCs

- **Added:**
  - Added new /info endpoint
- **Changed:**

- – Code re-engineered using Spring Boot framework

- – **Modified ranking algorithm:**

  * the ranking is no longer performed at the provider level; each single service is ranked, including also the potential dependencies among services (e.g. the rank for Marathon services takes into account also the metrics related to the Mesos cluster hosting the Marathon framework)

  * the Drools rules have been updated accordingly

- – Modified interface for /rank API (json request/response)

- – Updated documentation

- **Fixed:**

  - – Fixed issue with ranking services that have the same SLA priority, now they are ranked using the monitoring metrics, if available.

  - – Fixed code style issues

  - – Fixed rmp/deb creation

- **Removed:**

  - – The following endpoints have been removed:

Installation methods

- **automatic deployment by using the Docker image**

  - – See documentation at: CloudProviderRancker Guide

List of Artifacts

- **CentOS-7 RPM**

  - – CloudProviderRanker-0.7.0-1.noarch.rpm

- **Docker image:**

  - – indigodatacloud/cloudproviderranker:v0.7.0

  - – indigodatacloud/cloudproviderranker:DEEP-2

Documentation

Detailed documentation is available at:

- GitHub README
- GitBook Guides

Support

- GitHub issues: https://github.com/indigo-dc/CloudProviderRanker/issues

- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

DEEPaaS API

DEEP as a Service (DEEPaaS) is a REST API that is focused on providing access to machine learning models. By using DEEPaaS users can easily run a REST API in from of their model, thus accessing its functionality via HTTP calls.

Release Notes

v 1.0.0

What's new

- New debug option, can be enabled with the `--debug-endpoint`. This will activate the `/debug` endpoint where all the output generated by the model (i.e. using print) or logged in the deepaas.api logger will be shown.

- New V2 version of the API. This is a major change that is not backwards compatible. We also have moved away from Flask to aiohttp in order to support async requests. Please read these notes carefully so that you can upgrade your model along with this new version.

- Moved from the OpenAPI Specification version 3 and Swagger UI 3.X.

- Added new `-predict-workers` and `-train-workers` options, allowing to specify the number of workers that will be used for model operations (train, predict).

- New warm method allow to initialize the model before the API is spawned.

- Support for multiple, versioned endpoints.

List of RfCs

- Complete list is avaiable at DEEPaaS Release Notes

Known Issues

- None

Installation methods

- automatic deployment by pip - DEEPaaS guide

List of Artifacts

- Python library, released in PyPI
- **Docker Container:**

  - indigodatacloud/deepaas:DEEP-2
  - indigodatacloud/deepaas:1.0.0

Documentation

Detailed documentation can be found at:

- DEEPaaS API ReadTheDocs

Support

- GitHub issues: https://github.com/indigo-dc/DEEPaaS/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

Infrastructure Manager

Table of Contents

IM is a tool that deploys complex and customized virtual infrastructures on IaaS Cloud deployments (such as AWS, OpenStack, etc.). It eases the access and the usability of IaaS clouds by automating the VMI (Virtual Machine Image) selection, deployment, configuration, software installation, monitoring and update of the virtual infrastructure. It supports APIs from a large number of virtual platforms, making user applications cloud-agnostic. In addition it integrates a contextualization system to enable the installation and configuration of all the user required applications providing the user with a fully functional infrastructure.

This version evolved in the projects INDIGO-Datacloud and DEEP – Hybrid DataCloud. It is used by the **INDIGO Orchestrator** to contact Cloud sites to finally deploy the VMs/containers.

Release Notes

v1.8.6

What's new

- Fixed some issues deploying hybrid clusters.
- Enable to force the deletion of an infrastructure

- Enable to boot the IM in "ReadOnly" modes.

- Some minor bugfixes and enhancements.

List of RfCs

- GitHub solved issues

Known Issues

- None

Installation methods

- The recommended option to use the Infrastructure Manager service is using the available docker image. A Docker image named grycap/im has been created to make easier the deployment of an IM service using the default configuration. Information about this image can be found here: https://registry.hub.docker.com/u/indigodatacloud/im/. More details at: https://github.com/indigo-dc/im#1-docker-image

- The IM provides a script to install the IM in one single step (using pip). You only need to execute the following command:

```
$ wget qO https://raw.githubusercontent.com/indigo-dc/im/master/install.sh | bash
```

And you can also use this ansible playbook to install it: https://github.com/indigo-dc/im/blob/master/ansible_install.yaml

List of Artifacts

- **CentOS-7 RPMS**
  - IM-1.8.6-1.el7.noarch.rpm
- **Ubuntu 16.04 DEBS**
  - python-im_1.8.6-1_all.deb
- **Ubuntu 18.04 DEBS**
  - python-im_1.8.6-1_all.deb
- **Docker Container:**
  - indigodatacloud/im:DEEP-2 (signed)
  - indigodatacloud/im:1.8.6.1

Documentation

Detailed documentation can be found at:

- GitHub README
- IM ReadTheDocs

- Ansible Role for IM

Support

- GitHub issues: https://github.com/indigo-dc/im/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

Monitoring

The **Monitoring Framework** is a set of tools which allow performing several monitoring operations in the platform resulting from the INDIGO-Datacloud project (https://www.indigo-datacloud.eu/). The Monitoring Framework is based on Zabbix, as the collector of the monitoring information coming from different sources, due to its maturity, its community support and its flexibility for different environments. The Monitoring Framework is divided in several main parts: * The Zabbix server (with the corresponding configuration and some support scripts); * The Zabbix wrapper, created for enabling a REST API for Zabbix; * Several probes, with different monitoring purposes (OCCI, Heapster, etc.).

This repository contains the supporting scripts for the Zabbix server (in order to perform automatic backups of the Zabbix database and configuration), the wrapper to be deployed with Zabbix (as a way to facilitate integration) and the probes released in the first version: a probe for monitoring OCCI interfaces of Infrastructure Providers and a probe for monitoring the Kubernetes cluster where the Indigo platform is deployed (by means of the Heapster tool).

Release Notes

v DEEP-2

What's new

This is the first release of Monitoring framework through the DEEP-HDC project, providing new version of Openstack, Mesos and QCG probes:

- Openstack zabbix probe v. 1.4
- Mesos probe v. 1.4
- QCG probe v. 1.4

Some of the main new common new features are:

- Integration with the Orchestrator services and providers convention
- Integration with DEEP IAM
- Integration with DEEP CMDB

List of RfCs

Installation methods

- Mesos: https://github.com/indigo-dc/Monitoring/blob/master/doc/mesos.md#3-installation

- Openstack: https://github.com/indigo-dc/Monitoring/blob/master/doc/OPENSTACK.md#3-installation

- QCG: https://github.com/indigo-dc/Monitoring/blob/master/doc/QCG.md#3-installation

List of Artifacts

- **CentOS-7 RPMS**
    - Openstack probe openstack-zabbix-probe-1.4.2-2.noarch.rpm
    - Mesos probe MesosZabbixProbe-1.4-1.noarch.rpm
    - QCG probe QCGZabbixProbe-1.0-1.noarch.rpm
- **Ubuntu 16.04 DEBS**
    - Openstack probe openstack-zabbix-probe-1.4.2_all.deb
    - Mesos probe mesos-zabbix-probe_1.4_all.deb
    - QCG probe qcg-zabbix-probe-1.0_all.deb
- **Ubuntu 18.04 DEBS**
    - Openstack probe openstack-zabbix-probe-1.4.2_all.deb
    - Mesos probe mesos-zabbix-probe_1.4_all.deb
    - QCG probe qcg-zabbix-probe-1.0_all.deb

Documentation

Detailed documentation can be found at:

- GitHub documentation

Support

- **GitHub issues:**
    - https://github.com/indigo-dc/Monitoring/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/
- **Main develoeprs:**
    - jose.sanchezm@atos.net
    - rut.palmero@atos.net
    - damian.kaliszan@man.poznan.pl

PaaS Orchestrator

The **INDIGO PaaS Orchestrator** is a component of the PaaS layer that allows to instantiate resources on Cloud Management Frameworks (like OpenStack and OpenNebula) and Mesos clusters.

It takes the deployment requests, expressed through templates written in TOSCA YAML Simple Profile v1.0, and deploys them on the best cloud site available. In order to do that

- it gathers SLAs, monitoring info and other data from other platform services,
- it asks to the cloud provider ranker for a list of the best cloud sites.

Release Notes

v2.2.0-FINAL

What's new

Between the many changes introduced we can highligth the new support for compute GPU, flavors and for vRouter deployment

List of RfCs

- New Features
    - Added support for Compute GPU and flavors
    - Added Vault integration for supporting Marathon secrets
    - Added initial QCG support
    - Added /configuration endpoint
    - Added enableHttps flag in MarathonApp (enable_https in tempplate)
    - Added support for vRouter deployment
- Improvements
    - Support new A4C Tosca Parser library (v2.1.0-DEEP-1.0.0)
    - Support new CPR interface
    - Support new CMDB Schema
    - Support new Monitoring data structure
    - Improved hybrid deployment support for basic scenario

- Bug Fixes

    - Fix log errors in case of cloud services retry

    - Fix bug in the selection of the cloud service

Dependencies

The PaaS Orchestrator v.2.1.2-FINAL has the following services dependencies

- CMDB release v0.5

- SLAM release v2.0.0

- CPR release v0.7.0

- Monitoring - Zabbix Wrapper 1.0.2

- Monitoring probes: Openstack probe 1.4.2, Mesos probe 1.4 and QCG probe 1.0

- IM release 1.8.6.1

- (Optional) Onedata > = v18.0.2-rc[11,12]

- (Optional) Vault 1.1.2

- tosca-types v4.0.0

Installation methods

- Automatic deployment is available through Ansible roles

    - Project URL: https://github.com/indigo-dc/indigopaas-deploy

List of Artifacts

- Docker Container:

    - indigodatacloud/orchestrator:DEEP-2 (signed)

    - indigodatacloud/orchestrator:2.2.0-final

Documentation

Detailed documentation is available at:

- GitHub README

- GitBook Guides

Support

- GitHub issues: https://github.com/indigo-dc/orchestrator/issues

- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

PaaS Orchestrator Dashboard

The **INDIGO PaaS Orchestrator - Simple Graphical UI** is a Python application built with the Flask microframework; Flask-Dance is used for Openid-Connect/OAuth2 integration. The docker image uses Gunicorn as WSGI HTTP server to serve the Flask Application.

Release Notes

v.1.0.0

What's new

This is the first stable release.

List of RfCs

- **Important Functionalities:**
    - IAM authentication
    - Display user's deployments
    - Display deployment details, template and log
    - Delete deployment
    - Create new deployment

Known Issues

- None

Installation methods

- **automatic deployment by using the Docker image**
    - See documentation at: PaaS Orchestrator Dashboard README

List of Artifacts

- **Docker image:**

    - orchestrator-dashboard:v1.0.0

    - orchestrator-dashboard:DEEP-2 (signed)

Documentation

Detailed documentation can be found at:

- GitHub README

Support

- **GitHub issues:**

    - https://github.com/indigo-dc/orchestrator-dashboard/issues

- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

- Main Developer: Marica Antonacci (marica.antonacci@ba.infn.it)

Spring Social OIDC

Table of Contents

    - *Release Notes*

    - *Documentation*

    - *Support*

This project provides integration with the DEEP orchestrator and OpenID connect log-in support through Spring Social to Spring projects.

Release Notes

v1.4

What's new

- First inclusion in a DEEP release

- Integration of the orchestrator functionality into Spring projects such as Alien4Cloud through Spring Social

List of RfCs

Known Issues

- GitHub open issues

Installation methods

- Dependency managers such as Maven or Gradle can use the method described in https://github.com/indigo-dc/spring-social-oidc/blob/master/README.md#use-in-a-spring-project

List of Artifacts

- **jar to be included in Spring projects through gradle or maven**
    - Maven artefact

Documentation

Detailed documentation can be found at:

- GitHub README

Support

- GitHub issues: https://github.com/indigo-dc/spring-social-oidc/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/
- Main developer: jose.sanchezm@atos.net

INDIGO Virtual Router Configurator

Table of Contents

The **INDIGO-DC Virtual Router** does not carry its own code. It consists of open components and all automation is contained in Ansible roles.

The purpose of the INDIGO-DC Virtual Router is establishing an overlay virtual private network to interconnect nodes in a cloud platform deployment even if deployed in multiple, geographically distant sites.

Release Notes

DEEP-v2

What's new

- support for OpenStack
- added parameter for MTU size - to prevent hanging out connections in case of MTU problems
- added option for private subnet
- use-case for elastic clusters - certificate for machine with re-used hostname
- various other fixes (CentOS, internal certificates)

List of RfCs

Known Issues

- None

Installation methods

- Automatic deployment - ansible module used by Infrastructure Manager.
  - Project URL: https://github.com/indigo-dc/ansible-role-indigovr

List of Artifacts

- **Ansible Roles**
  - https://galaxy.ansible.com/indigo-dc/indigovr

Documentation

Detailed documentation is available at:

- GitHub README

Support

- GitHub issues: https://github.com/indigo-dc/ansible-role-indigovr/issues
- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

udocker

udocker is a basic user tool to execute simple docker containers in user space without requiring root privileges. Enables download and execution of docker containers by non-privileged users in Linux systems where docker is not available. It can be used to pull and execute docker containers in Linux batch systems and interactive clusters that are managed by other entities such as grid infrastructures or externally managed batch or interactive systems.

udocker does not require any type of privileges nor the deployment of services by system administrators. It can be downloaded and executed entirely by the end user.

udocker is a wrapper around several tools to mimic a subset of the docker capabilities including pulling images and running containers with minimal functionality.

Release Notes

v1.1.4

What's new

This version provides a number of new features, improvements and bug fixes

List of RfCs

New features

- Use hub.docker.com as default registry
- Listing of containers size with ps -s
- Listing of containers execution modes with ps -m
- Added support for nameat() and statx() in Pn and Fn modes
- Added Fn libraries for Ubuntu18, Ubuntu19, Fedora29, Fedora30, Fedora31, CentOS8
- Added Fn libraries for Alpine 3.8, 3.9, 3.10, 3.11
- Added support for sha512 hashes
- Added support for opaque whiteouts
- Added search –list-tags to available tags for a given repository
- Add CLI support for image names in format host/repository:tag
- Support for fake root in Sn execution modes via –user=root
- Added run –containerauth to enable direct use of the container passwd and group fixing user and group creation

- Added run –env-file= to load file with environment variables
- Added support for file mount bindings in singularity
- Added UDOCKER_USE_PROOT_EXECUTABLE env var to select proot location
- Added UDOCKER_USE_RUNC_EXECUTABLE env var to select runc location
- Added UDOCKER_USE_SINGULARITY_EXECUTABLE env var to select singularity
- Added UDOCKER_DEFAULT_EXECUTION_MODE env var to select default execution mode
- Added R2 and R3 execution modes for PRoot overlay execution in runc enabling better root emulation under these modes.
- Added setup –purge for cleanup of mountpoints and files
- Added setup –fixperms to fix container file permissions
- Add rename command for renaming of containers
- Create processes without shell context
- Added loading and handling of container images in OCI format

Improvements

- Improved search using v1 and v2 APIs and better display of results
- Improved handling of links and pathname translation in Fn modes
- Improved fix of SECCOMP accelerated mode for P1 mode
- Improved verify of loaded/pulled images
- Improved handling of mountpoints
- Improve file and directory binding support for Singularity and runc
- Safer parsing of config files and removal of directories
- Improved installation

Bugs

- Fixes for udocker in ARM aarch64

Detailed list available on udocker GitHub

Known Issues

- None

Installation methods

- Via PIP or other method as described in the Installation Manual

List of Artifacts

Tarball:

- udocker-1.1.4.tar.gz

Documentation

Detailed documentation can be found at:

- GitBook Guide
- GitHub documentation

Support

- **GitHub issues:**

  – https://github.com/indigo-dc/udocker/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

TOSCA types & templates

The **TOSCA types** repository shows a YAML description of new types added first in the INDIGO project, and afterwards in the DEEP-HybridDataCloud and eXtreme DataCloud projects, to extend TOSCA Simple Profile in YAML Version 1.0 to add high level entities. In the examples directory there are a set of TOSCA documents using these types that will be supported by the INDIGO, DEEP and XDC components.

The **TOSCA Templates** repository contain templates Supporting the Use Cases for INDIGO-DataCloud, DEEP-HybridDataCloud and eXtreme DataCloud projects.

Release Notes

v. 4.0.0

What's new

This version provides a number of new features, improvements and bug fixes

List of RfCs

- TOSCA types v. 4.0.0, tested with Ansible 2.6.20
  - Added new type for QCG job
  - Added new types and artifacts for vRouter components
  - Added new type for DEEPaaS

- – Added support for Marathon secrets

- – Added icons for each type

- – Added metadata and description in custom_types.yaml

- – Added type for BlockStorage and for AttachesTo relationship

- – Improved support for hybrid deployments

- – Improved policy definitions

- TOSCA templates v. 4.0.0

  - **– Updated templates to ensure the compliance with Simple Profile in YAML 1.0**

    - *** these changes are mandatory for working with Orchestrator version >= 2.2.0**

  - – Updated templates for DEEP use-cases

  - – Fixing typos (e.g. priviliged –> privileged)

  - – Added template for Slurm Elastic Cluster

  - – Deleted obsolete templates

  - – Deleted templates with hard-coded values for SLAs

- None

List of Artifacts

- **Tarballs:**

  - tosca-types-4.0.0.tar.gz

  - tosca-templates-v4.0.0.tar.gz

Documentation

Detailed documentation can be found at:

- TOSCA types README

- TOSCA templates README

- Brief guide to TOSCA

- How to deploy a TOSCA Template

Support

- **GitHub issues:**

  - https://github.com/indigo-dc/tosca-types/issues

  - https://github.com/indigo-dc/tosca-templates/issues

- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

1.1.2 Highlights

Key technical highlights:

- Alien4Cloud-DEEP plugin * Support for the OpenStack normative TOSCA types, for the IndigoDC TOSCA custom types, for DEEP-OC templates * Outputs retrieval and display from the Indigo Orchestrator * IAM authentication support

- DEEP-Hybrid-DataCloud Cloud Information Provider * Support for multitenancy in CMDBv1, for new Mesos providers, for fetching GPU information from OpenStack and Mesos

- CloudProviderRanker

    - new release featuring a complete code re-engineered using Spring Boot framework and modified ranking algorithm

- DEEPaaS API

    - New V2 version of the API, major change that is not backward compatible, supporting async requests, following

      Swagger UI 3.X Specifications, with new options allowing to specify the number of workers that will be used for model operations

- Infrastructure Manager (IM)

    - new version with multiple bug fixes and enhancements, improving deployment of hybrid clusters.

- INDIGO Virtual Router Installer

    - new verion with many bug fixes and improvements, supporting OpenStack

- Monitoring Framework

    - first release through the DEEP-HDC project, providing improved versions of Openstack and Mesos probes and new QCG probe

- PaaS Orchestrator

    - new version that add support for compute GPU, flavors and for vRouter deployment

- PaaS Orchestrator Dashboard

    - first stable release with INDIGO-IAM authentication, allowing easier management of infrastructures deployments

- Spring Social OIDC

    - first release in DEEP-HDC allowing for the integration of the orchestrator functionality into Spring

      projects such as Alien4Cloud through Spring Social

- TOSCA types and templates

    - new types for QCG jobs and DEEPaaS, support for Marathon secrets and hybrid deployments

- udocker

    - many new features, improvements and bug fixes, like the improved support of singularity,

      added support for sha512 hashes, safer parsing of config files and removal of directories, improved installation

1.1.3 Release Notes

The DEEP-2/Rosetta release consists in 10 Products and a number of technical guides:

- 22 packages and tarballs, for CentOS 7, Ubuntu 16.04, Ubuntu 18.04

- 6 Docker containers

You can find in the later sections the full list of products, with detailed release notes and instructions for their installation/configuration.

1.1.4 Installation Notes

All DEEP - HybridDataCloud products are distributed from standard Operating Systems (OS) repositories and DockerHub registry of the indigodatacloud and deephdcgo organization.

The packages repositories have the following structure:

- DEEP-HDC **production** (stable):

    - deep-hdc/production/2/centos7/x86_64/{base|updates}

    - deep-hdc/production/2/ubuntu/dists/xenial/main/{binary-amd64,source}

        * containing signed, well tested software components

    - third-party:

        * deep-hdc/production/2/centos7/x86_64/third-party

        * deep-hdc/production/2/ubuntu/dists/{xenial,bionic}/third-party{binary-amd64,source}

            · containing packages that are not part of DEEP, or not part of the base OS or EPEL, but used as dependencies by other DEEP components

- DEEP-HDC **testing**: deep-hdc/testing/2/{centos7,ubuntu}/

    - containing packages that will become part of the next stable distribution; in the certification and validation phase.

- DEEP-HDC **preview**: deep/preview/2/{centos7,ubuntu}/

    - containing signed packages that will become part of the next stable update, available for technical-previews

All packages in production and preview repositories are signed with the INDIGO - DataCloud gpg key. The public key can be downloaded from here, and the fingerprint from here.

Please import the key BEFORE starting!

- for CentOS7 save the key under /etc/pki/rpm-gpg/

```
# rpm --import https://repo.indigo-datacloud.eu/repository/RPM-GPG-KEY-indigodc
```

- for Ubuntu:

```
# wget -q -O - https://repo.indigo-datacloud.eu/repository/RPM-GPG-KEY-indigodc |
↪sudo apt-key add -
```

Configuring the use of DEEP-HDC repositories

YUM & APT configuration files are available at:

- CentOS7 - https://repo.indigo-datacloud.eu/deep-hdc/repos/deep-2.repo
- Ubuntu 16.04 - https://repo.indigo-datacloud.eu/deep-hdc/repos/deep-2-ubuntu16_04.list
- Ubuntu 18.04 - https://repo.indigo-datacloud.eu/deep-hdc/repos/deep-2-ubuntu18_04.list

or use the deephdc-release package to install DEEP-HDC repositories:

- CentOS7:

```
# wget https://repo.indigo-datacloud.eu/repository/deep-hdc/production/2/centos7/x86_
↪64/base/deephdc-release-2.0.0-1.el7.noarch.rpm
# yum localinstall -y deephdc-release-2.0.0-1.el7.noarch.rpm
```

- Ubuntu 16.04:

```
# wget https://repo.indigo-datacloud.eu/repository/deep-hdc/2/ubuntu/dists/xenial/
↪main/binary-amd64/deephdc-release_2.0.0-1_amd64.deb
# dpkg -i deephdc-release_2.0.0-1_amd64.deb
```

- Ubuntu 18.04:

```
# wget https://repo.indigo-datacloud.eu/repository/deep-hdc/2/ubuntu/dists/bionic/
↪main/binary-amd64/deephdc-release_2.0.0-1_amd64.deb
# dpkg -i deephdc-release_2.0.0-1_amd64.deb
```

These packages will install required dependencies, the INDIGO - DataCloud public key and ensures the precedence of DEEP-HybridDataCloud repositories over EPEL and Ubuntu.

It is strongly recommended the use of the lastest version of the **deephdc-release** packages containing the public key and the YUM and APT repository files.

Enable the DEEP-HDC Containers repositories

On the DockerHub Registry, DEEP - HybridDataCloud uses the INDIGO - DataCloud and DEEP-HDC Organizations:

- indigodatacloud, for Core Services
- deephdc, for DEEP-OC modules

Containers present in those repositories and released in DEEP-1 major release are tagged with "DEEP-2" tag and signed, leveraging the Docker's trust features so that users can pull trusted images.

Currently, content trust is disabled by default. You must enable it by setting the DOCKER_CONTENT_TRUST environment variable, like below:

```
export DOCKER_CONTENT_TRUST=1
```

For more details regarding the "Content Trust in Docker" please read Docker's Documentation

Content trust is associated with the TAG portion of an image. For DEEP-2 (Rosetta) release the signed tag is DEEP-2. See example bellow if you want to ensure the correct use of DEEP-HDC images:

```
$ export DOCKER_CONTENT_TRUST=1

$ docker pull indigodatacloud/orchestrator:2.2.0-FINAL
```

(continues on next page)

```
No trust data for 2.2.0-FINAL

$ docker pull indigodatacloud/orchestrator:DEEP-2
Pull (1 of 1): indigodatacloud/orchestrator:DEEP-
↪2@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be: Pulling from↪
↪indigodatacloud/orchestrator
93857f76ae30: Pull complete
[...]
e8c92b40b492: Pull complete
Digest: sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
Status: Downloaded newer image for indigodatacloud/
↪orchestrator@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
Tagging indigodatacloud/
↪orchestrator@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be↪
↪as indigodatacloud/orchestrator:indigo_2
$ docker images
REPOSITORY                      TAG             IMAGE ID         CREATED          ↪
↪     SIZE
indigodatacloud/orchestrator    DEEP-2          xxxxxxxxxxx      1 hours ago      ↪
↪ 843MB
```

1.1.5 Documentation

Please find DEEP-2 services and components documentation at *Included components*.

1.1.6 Support

Most complex software contains bugs, we are not an exception. One of the features of free and open source software is the ability to report bugs, helping to fix or improve the software you use.

DEEP - HybridDataCloud project uses the INDIGO Catch-All GGUS - Support Unit and the *deep-support@listas.csic.eu* for general support requests. More details regarding each product support channels are provided in the respective products release pages.

Developers, researchers and IT enthusiasts: feel free to write to *deep-info@listas.csic.es* to ask for more information on how to use DEEP solutions for your work. For automatic notifications you can register to the DEEP - HybridDataCloud release RSS feed or subscribe to the DEEP - HybridDataCloud Announce Mailing list. You can also socialize with us via Twitter or by joining our LinkedIn group.

1.2 DEEP-1 Genesis

The DEEP - HybridDataCloud project is pleased to announce the general availability of its **first public software release**, codenamed **Genesis**

1.2.1 Included components

Alien4Cloud - DEEP plugin

The **Alien4Cloud - DEEP plugin** is the first step towards the adoption of a visual composition approach of TOSCA templates (based on Alien4Cloud), in order to facilitate the widespread adoption of the standard, and its integration with the INDIGO-DataCloud Orchestrator, which is already part of the EOSC-HUB service catalogue.

- With this approach, the user can visually compose complex applications that involve, for example, the dynamic deployment of a container orchestration platform on an IaaS Cloud site that executes a highly-available Docker-based application to facilitate application delivery.

- The users can also deploy an Apache Mesos cluster with GPU support that contains a deep learning application for the recognition of certain plant species, offered as a service to a community of users. This introduces unprecedented flexibility, from visual composition, to the automated application delivery, using a graphical interface that is already integrated with an Orchestrator layer that performs resource provision from multiple Clouds and application configuration.

- The integration of easy-to-use graphical interfaces builds a bridge between the users and the orchestration services.It also represents a step forward to foster the adoption of innovative computing services that are hidden from the user, as they can focus on the high-level description of the services requirements and definition, instead of working on their technical implementation.

Release Notes

v 1.1-r0

What's new

This is the first release of the Alien4Cloud - DEEP plugin providing:

- Support for visual composition of TOSCA templates
- Support for the PaaS Orchestrator

List of RfCs

Most important features and improvements:

- [DPD-179] - Support for users in Alien4Cloud
- [DPD-180] - Support for user on the fly
- [DPD-182] - Indigo Tosca Error simple compute node
- [DPD-183] - Repository in the indigo tosca types not handled by A4C
- [DPD-184] - Add a README
- [DPD-185] - Description of IndigoDC data types
- [DPD-186] - Get status for deployed infrastructure

- [DPD-187] - Rm non-required props with null
- [DPD-188] - Generate and write Javadoc
- [DPD-189] - Expose the imports section indigo types
- [DPD-192] - Missing Properties in the Configuration of the Orchestrator Plugin
- [DPD-197] - All TOSCA methods should be strings
- [DPD-198] - Adhere to Google Java Style
- [DPD-270] - /usr/bin/env: bad interpreter: Text file busy
- [DPD-280] - Missing users after restart
- [DPD-281] - Dockerfile custom config
- [DPD-282] - Add support for orchestrator instance confi
- [DPD-283] - De-activate logging of sensitive data
- [DPD-285] - Test coverage >=70%
- [DPD-288] - Error loading certificate
- [DPD-294] - README - Obtain Certificate
- [DPD-309] - Change envs name for certificates name holders
- [DPD-311] - Change IDs used to identify deployment
- [DPD-335] - [README] Add how to use REST API
- [DPD-336] - [README] Add documentation versioning
- [DPD-341] - A4C-DEEP security dependency issue
- [DPD-343] - A4C-DEEP documentation
- [DPD-359] - Release notes for Alien4Cloud-deep v. 1.1-r0
- [DPD-365] - Alien4Cloud-deep v. 1.1-r0 - test reports

Complete list of solved issues is available at: Github

Known Issues

Please see the list of issues on Github

Dependencies

The Alien4Cloud-DEEP plugin v. 1.1-r0 is known to work with the following:

- Alien4Cloud custom verision: 2.0.0-UPV-1.0.0
- Plugin version: 1.0.4-SNAPSHOT
- Alien4Cloud Settings Manager version: 1.0.1
- TOSCA custom-types version: 1.0

List of Artifacts

- **Docker Container:**

    – [indigodatacloud/alien4cloud-deep](#):DEEP-1 (signed)

    – [indigodatacloud/alien4cloud-deep](#):1.1-r0

Documentation

Detailed documentation is available at:

- [GitHub README](#)

Support

- GitHub issues: https://github.com/indigo-dc/alien4cloud-deep/issues

- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

CloudInfoProvider-DEEP

The **Cloud Information Provider** generates a representation of cloud resources, that can be published inside a BDII (using the provided LDIF templates for a GlueSchema v2 representation) or any other component like the INDIGO Configuration Management Database (CMDB) (Using specific templates).

The generated representation is described using a [Mako](#) template having access to the cloud middleware information.

The current document describe the main features implemented for the Cloud Information Provider under the DEEP-Hybrid-Datacloud project.

Main documentation for the product is maintained [upstream](#), including usage and contribution (for developers) guidelines. The documentation for the features pending to be merged upstream and/or specific to DEEP-Hybrid-Datacloud will be added below, associated with the release version.

Release Notes

v 0.10.4

What's new

This versions adds InfiniBand support for OpenStack in LDIF format:

- cloud-info-provider shall provides now information, through GLUE2ResourceID, when a given OpenStack flavor provides InfiniBand.

- The convention is to use the extra_specs parameter from the flavor's metadata. This parameter is a list of key:value pairs, where the both key's and value's names are not fixed. The provider should allow to define the key's name and the value string to match via command-line options. The InfiniBand support will be published through the GLUE2ExecutionEnvironmentNetworkInfo attribute.

List of RfCs

- DPD-284 - Add InfiniBand support for OpenStack in LDIF format

Known Issues

- None

List of Artifacts

- **CentOS-7 RPMS**
  - cloud-info-provider-deep-0.10.4-1.el7.noarch.rpm
  - cloud-info-provider-deep-openstack-0.10.4-1.el7.noarch.rpm
- **Ubuntu 16.04 DEBS**
  - python-cloud-info-provider-deep-openstack_0.10.4-1_all.deb
  - python-cloud-info-provider-deep_0.10.4-1_all.deb

Documentation

Detailed documentation is available at:

- https://github.com/EGI-Foundation/cloud-info-provider

Documentation specific for the DEEP-HDC version is available at:

- https://github.com/indigo-dc/cloud-info-provider-deep/blob/DEEP/README-DEEP.md

Support

- GitHub issues: https://github.com/EGI-Foundation/cloud-info-provider/issues?q=is%3Aissue+is%3Aclosed

- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

- EGI GGUS Support Unit: https://wiki.egi.eu/wiki/GGUS:EGI_Cloud_Information_Discovery_FAQ

DEEPaaS API

Table of Contents

- *DEEPaaS API*

    – *Release Notes*

    – *Documentation*

    – *Support*

DEEP as a Service (DEEPaaS) is a REST API that is focused on providing access to machine learning models. By using DEEPaaS users can easily run a REST API in from of their model, thus accessing its functionality via HTTP calls.

Release Notes

v 0.1.0

What's new

This is the initial release of DEEPaaS API (deepaas) exposing basic model functionality.

List of RfCs

- Task implemented/partially implemented - Expose model functionality to end users - Expose functionality via API - Provide access to application - Provide resources to efficiently train a model - Facilitate the deployment of a trained model for inference - Provide a easy way to access a deployed application by end users - Provide a way to submit input data to model

Known Issues

- None

List of Artifacts

- **Docker Container:**

    – indigodatacloud/deepaas:DEEP-1 (signed)

    – indigodatacloud/deepaas:0.1.0

v 0.2.0

What's new

This new version allows the user to specify which arguments the underlying model accepts. In this release users can implement a "get_train_args" function in the module entry point, allowing to expose those arguments through the API.

List of RfCs

- Ability to specify parameters for "train"

Known Issues

- None

List of Artifacts

- **Docker Container:**

    – indigodatacloud/deepaas:DEEP-1 (signed)

    – indigodatacloud/deepaas:0.2.0

Documentation

Detailed documentation can be found at:

- DEEPaaS API ReadTheDocs

Support

- GitHub issues: https://github.com/indigo-dc/DEEPaaS/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

Infrastructure Manager

Table of Contents

IM is a tool that deploys complex and customized virtual infrastructures on IaaS Cloud deployments (such as AWS, OpenStack, etc.). It eases the access and the usability of IaaS clouds by automating the VMI (Virtual Machine Image) selection, deployment, configuration, software installation, monitoring and update of the virtual infrastructure. It supports APIs from a large number of virtual platforms, making user applications cloud-agnostic. In addition it integrates a contextualization system to enable the installation and configuration of all the user required applications providing the user with a fully functional infrastructure.

This version evolved in the projects INDIGO-Datacloud and DEEP – Hybrid DataCloud. It is used by the **INDIGO Orchestrator** to contact Cloud sites to finally deploy the VMs/containers.

Release Notes

v1.7.5

What's new

This new version adds a number of new features and improvements, like the support for hybrid deployments, as well as some bug fixes, detailed bellow

List of RfCs

New features and improvements

- [DPD-167] - Enable hybrid deployment

- [DPD-172] - Add option to disable XML-RPC API

- [DPD-173] - Create a connector to Apache CloudStack

- [DPD-174] - Add volume_id to TOSCA BlockStorage type

- [DPD-175] - Add support to disk.*.image_url in OpenStack connector

- [DPD-240] - Facilitate the deployment of a trained model for inference

- [DPD-263] - Enable to specify spot to EC2 instances in TOSCA

- [DPD-264] - Get EC2 instances info from ec2instances.info

Bugs fixed

- [DPD-168] - Problem installing specific branch of Ansible galaxy roles

- [DPD-169] - OCCI server 2.0.4 authentication error

- [DPD-170] - OCCI infrastructure: Volume deleting returns forbidden and takes too long

- [DPD-171] - Fix error reporting memory units in OCCI connector

- [DPD-265] - Error geting the subnets of the default VPC on EC2 conn

- [DPD-271] - Error detecting ansible process end

Known Issues

- None

List of Artifacts

- **CentOS-7 RPMS**

    – IM-1.7.5-1.el7.noarch.rpm

- **Ubuntu 16.04 DEBS**

    – python-im_1.7.5-1_all.deb

- **Docker Container:**

    – indigodatacloud/im:DEEP-1 (signed)

    – indigodatacloud/im:1.7.5

v1.8.2

What's new

This new version adds a number of new features and improvements, as well as some bug fixes, detailed bellow

List of RfCs

New features and improvements

- [DPD-534] - Enable to set ansible module version.
- [DPD-535] - Fix error attaching volumes in OCCI conn (OpenNebula site)
- [DPD-537] - Remove new line characters inserted into cloud-init shell scripts
- [DPD-538] - Enable to create Azure VM from user disk
- [DPD-539] - Enable to set cidr in Fogbow federated networks
- [DPD-540] - Add node creation error message in CloudStack conn
- [DPD-541] - Add sleep among ctxt agent task retries

Bugs fixed

- [DPD-531] - Error in dockerfile not installing libcloud library
- [DPD-532] - Fix error port 22 is not opened in private nets
- [DPD-533] - Fix error in directory creation in case of using external ansible host
- [DPD-536] - Fix error in OCCI conn contacting with KeyStorm

Known Issues

- None

List of Artifacts

- **CentOS-7 RPMS**
    - IM-1.8.2-1.el7.noarch.rpm
- **Ubuntu 16.04 DEBS**
    - python-im_1.8.2-1_all.deb
- **Docker Container:**
    - indigodatacloud/im:1.8.2

v1.8.4

What's new

This new version adds a number of new features and improvements, as well as some bug fixes, detailed bellow

List of RfCs

New features and improvements

- [DPD-589] - Enable to add routes in GCE conn
- [DPD-590] - Get last version of AppDB image in OCCI conn
- [DPD-591] - Change FogBow connector to new APIs (Buzios)
- [DPD-597] - Handling of special characters in error messages from OpenStack
- [DPD-598] - Keypairs left behind on OpenStack
- [DPD-600] - Add missing volume_url parameter to OpenStack conn
- [DPD-601] - Remove keypair creation in EC2 conn
- [DPD-602] - Add error messages in case of error creating volumes
- [DPD-603] - Remove ec2 classic supoort
- [DPD-606] - Enable network creation in connectors

Bugs fixed

- [DPD-587] - Error creating FW in GCE conn if outports is empty
- [DPD-588] - Issues with Ansible 2.8
- [DPD-592] - Error adding a floating IP when there is a floating ip free
- [DPD-593] - Some floating IPs does not appear in the IM ctxt config files
- [DPD-594] - On infrastructure creation the VM status is unknown instead of pending
- [DPD-595] - Error 500 is returned in case of TOSCA parser errors
- [DPD-596] - Error adding two volumes to a VM in OCCI conn
- [DPD-599] - Errors with PyYAML version 5.1
- [DPD-604] - Error reconfiguring with a new RADL
- [DPD-605] - Error getting keystone token in OCCI conn
- [DPD-607] - Error in openstack conn in case of setting auth data with a host without port
- [DPD-608] - Error in OpenNebula conn in case of error in function call

Known Issues

- None

List of Artifacts

- **CentOS-7 RPMS**
    - IM-1.8.4-1.el7.noarch.rpm
- **Ubuntu 16.04 DEBS**
    - python-im_1.8.4-1_all.deb
- **Docker Container:**

- indigodatacloud/im:DEEP-1 (signed)

- indigodatacloud/im:1.8.4

Documentation

Detailed documentation can be found at:

- GitHub README
- IM ReadTheDocs
- Ansible Role for IM

Support

- GitHub issues: https://github.com/indigo-dc/im/issues

- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

Orchent

Orchent is a command line application to manage deployments and their resources through the PaaS Orchestrator in a fast and easy way.

Release Notes

v1.2.2

What's new

The new version provides a number of new features, improvements and bug fixes

List of RfCs

- Bug fuxes:

  - Fixed communication with oidc-agent >= 2.0.0

- New features:

  - Added support for two new flags in depcreate/depupdate operations:

* maxProvidersRetry: Maximum number of cloud providers to be used in case of failure (Default: UNBOUNDED).

* keepLastAttempt: In case of failure, keep the resources allocated in the last try (Default: true).

- These flags are supported starting from Orchestrator version 2.1.0

Known Issues

- None

List of Artifacts

- **CentOS-7 RPMS**

    – orchent-1.2.2.el7.centos.x86_64.rpm

- **Ubuntu 16.04 DEBS**

    – orchent-1.2.2-amd64.deb

- **Docker Container:**

    – indigodatacloud/orchent:DEEP-1 (signed)

    – indigodatacloud/orchent:1.2.2

Documentation

Detailed documentation is available at:

- GitHub README
- GitBook Guides

Support

- GitHub issues: https://github.com/indigo-dc/orchent/issues
- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

PaaS Orchestrator

Table of Contents

The **INDIGO PaaS Orchestrator** is a component of the PaaS layer that allows to instantiate resources on Cloud Management Frameworks (like OpenStack and OpenNebula) and Mesos clusters.

It takes the deployment requests, expressed through templates written in TOSCA YAML Simple Profile v1.0, and deploys them on the best cloud site available. In order to do that

- it gathers SLAs, monitoring info and other data from other platform services,
- it asks to the cloud provider ranker for a list of the best cloud sites.

Release Notes

v2.1.1-FINAL

What's new

This version provides a number of new features, improvements and bug fixes, like:

- Hybrid deployments on multiple sites
- Support to specifying specialized computing hardware
- Improved support for deployment failures

List of RfCs

- New features:
    - Implement Cloud Providers Retry Logic (#288 DPD-121)
    - Retrieve of information from CMDB (fixes #290 DPD-124)
    - AuthN through OIDC (fixes #291 DPD-125)
    - Add support for mesos tasks with GPUs (fixes #292 DPD-126)
    - Support IM async infrastructure creation #289
    - Allow to import custom CAs in the java truststore #238
- improvements
    - Improve OneData Spaces definition through templates (fixes #293 XSD-28)
- Bug fixed:
    - Return 409 status code for SQLTransientExceptions #285
    - (MySql) Timestamp columns have implicit default value assigned #283

Known Issues

- None

Dependencies

The PaaS Orchestrator v.2.1.1-FINAL has the following services dependencies

- CMDB release v0.4 (docker image indigodatacloud/cmdb:indigo_2)
- SLAM release v2.0.0 (docker image indigodatacloud/slam:v2.0.0)
- CPR release v0.6.0 (docker image indigodatacloud/cloudproviderranker:indigo_2)
- Monitoring - Zabbix Wrapper 1.0.2 (docker image indigodatacloud/zabbix-wrapper:indigo_2)
- IM release >= 1.7.4 * (Optional) Onedata v18.0.2-rc[11,12]
- tosca-types v3.0.0

List of Artifacts

- **Docker Container:**
    - indigodatacloud/orchestrator:DEEP-1 (signed)
    - indigodatacloud/orchestrator:1.7.5

v2.1.2-FINAL

What's new

This version only updates the im-java-api client to 0.4.13

List of RfCs

- N/A

Known Issues

- None

Dependencies

The PaaS Orchestrator v.2.1.2-FINAL has the following services dependencies

- CMDB release v0.4 (docker image indigodatacloud/cmdb:indigo_2)
- SLAM release v2.0.0 (docker image indigodatacloud/slam:v2.0.0)
- CPR release v0.6.0 (docker image indigodatacloud/cloudproviderranker:indigo_2)
- Monitoring - Zabbix Wrapper 1.0.2 (docker image indigodatacloud/zabbix-wrapper:indigo_2)
- IM release >= 1.7.4 * (Optional) Onedata v18.0.2-rc[11,12]
- tosca-types v3.0.0

List of Artifacts

- **Docker Container:**

    - indigodatacloud/orchestrator:DEEP-1 (signed)

    - indigodatacloud/orchestrator:2.1.2-final

Documentation

Detailed documentation is available at:

- GitHub README

- GitBook Guides

Support

- GitHub issues: https://github.com/indigo-dc/orchestrator/issues

- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

TOSCA types & templates

Table of Contents

The **TOSCA types** repository shows a YAML description of new types added first in the INDIGO project, and afterwards in the DEEP-HybridDataCloud and eXtreme DataCloud projects, to extend TOSCA Simple Profile in YAML Version 1.0 to add high level entities. In the examples directory there are a set of TOSCA documents using these types that will be supported by the INDIGO, DEEP and XDC components.

The **TOSCA Templates** repository contain templates Supporting the Use Cases for INDIGO-DataCloud, DEEP-HybridDataCloud and eXtreme DataCloud projects.

Release Notes

v.3.0.0

What's new

This version provides a number of new features, improvements and bug fixes

List of RfCs

- Updated example template for deploying a Mesos cluster

- Added example template for deploying a Mesos cluster with GPU support and a tensorflow container on top of it that uses GPU(s)

- Added GPU support for compute nodes and dockerized jobs (chronos) and apps (marathon)

- Added preemtible_instance property to support "spot" instances

- Added example template for launching the generic_deepaas mesos job

- Added new types for describing a Kubernetes cluster; new Ansible roles implemented.

- Added new type for describing a JupyterHub node; new Ansible role implemented.

- Fixing for Galaxy on cloud:

    - Update support to Galaxy release_18.05 in indigo-dc.galaxycloud

    - Fix proftpd in indigo-dc.galaxycloud

    - New ansible role for galaxy tools installation, named indigo-dc.galaxycloud-tools

    - Update tosca.nodes.indigo.GalaxyShedTool with new ansible role

    - Fix CERN-VM FS reference data mount on cluster worker nodes on galaxy artifacts

    - Reworked ansible role (indigo-dc.galaxycloud-fastconfig) to reconfigure an image with Galaxy already installed

    - Reworked storage encryption script on indigo-dc.galaxycloud-os

    - Fix tasks order in Galaxy elastic cluster tosca template: now galaxy user is created before slurm configuration.

Known Issues

- None

List of Artifacts

- **Tarballs:**

        - tosca-types-3.0.0.tar.gz

        - tosca-templates-3.0.0.tar.gz

Documentation

Detailed documentation can be found at:

- TOSCA types README

- TOSCA templates README

- Brief guide to TOSCA

- How to deploy a TOSCA Template

Support

- **GitHub issues:**

    - https://github.com/indigo-dc/tosca-types/issues

    - https://github.com/indigo-dc/tosca-templates/issues

- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

udocker

Table of Contents

udocker is a basic user tool to execute simple docker containers in user space without requiring root privileges. Enables download and execution of docker containers by non-privileged users in Linux systems where docker is not available. It can be used to pull and execute docker containers in Linux batch systems and interactive clusters that are managed by other entities such as grid infrastructures or externally managed batch or interactive systems.

udocker does not require any type of privileges nor the deployment of services by system administrators. It can be downloaded and executed entirely by the end user.

udocker is a wrapper around several tools to mimic a subset of the docker capabilities including pulling images and running containers with minimal functionality.

Release Notes

v1.1.3

What's new

This version provides a number of new features, improvements and bug fixes

List of RfCs

Tasks

- [DPD-296] - Add support for nvidia-docker like functionality

- [DPD-301] - Add support for runC to v1.0.0-rc5

- [DPD-303] - Command line interface improvements

- [DPD-304] - Improve metadata handling

- [DPD-306] - Add kill-on-exit to proot on Pn modes

- [DPD-307] - Force python2 as interpreter

- [DPD-310] - Directory absolute symlinks in Fn modes not being converted

Bugs

- [DPD-297] - Improve parsing of quotes in the command line
- [DPD-298] - Issues in file downloads
- [DPD-299] - Mitigation for upstream limitation in runC without tty
- [DPD-300] - Fix detection of executable with symlinks in container
- [DPD-305] - Improve pathname translation in Fn modes for mounted dirs

Known Issues

- None

List of Artifacts

Tarball:

- udocker-1.1.3.tar.gz

Documentation

Detailed documentation can be found at:

- GitBook Guide
- GitHub documentation

Support

- **GitHub issues:**

    - https://github.com/indigo-dc/udocker/issues
- DEEP project's internal support ticketing system: https://jira.deep-hybrid-datacloud.eu/

INDIGO Virtual Router Configurator

Table of Contents

    - *Release Notes*
    - *Documentation*
    - *Support*

The **INDIGO-DC Virtual Router** does not carry its own code. It consists of open components and all automation is contained in Ansible roles.

The purpose of the INDIGO-DC Virtual Router is establishing an overlay virtual private network to interconnect nodes in a cloud platform deployment even if deployed in multiple, geographically distant sites.

Release Notes

DEEP-v1

What's new

The Ansible roles are intended for automated deployment of the INDIGO-DC Virtual Router through the Infrastructure Manager [2]. They can be deployed manually with Ansible if required.

List of RfCs

- N/A

Known Issues

- None

List of Artifacts

- **Ansible Roles**
    - https://github.com/indigo-dc/ansible-role-indigovr

Documentation

Detailed documentation is available at:

- GitHub README

Support

- GitHub issues: https://github.com/indigo-dc/ansible-role-indigovr/issues
- DEEP project's internal support ticketing system: http://jira.deep-hybrid-datacloud.eu/

1.2.2 Highlights

Key technical highlights:

- Alien4Cloud-DEEP plugin
    - Support for visual composition of TOSCA templates
    - PaaS orchestrator support

- cloud-info-provider
    - Support for GPU and Infiniband resources
- DEEPaaS API
    - Support for training a machine learning application
    - Support for performing inferences/analisys/predictions
    - Support only for synchronous requests
    - OpenID Connect support
    - Support for standalone service & OpenWhisk action
- Infrastructure Manager (IM)
    - Improved support for hybrid deployments
    - Support for additional TOSCA types
- PaaS Orchestrator
    - Hybrid deployments on multiple sites
    - Support to specifying specialized computing hardware Improved support for deployment failures
- uDocker
    - Improved support for GPUs and Infiniband
- Visual application topology composition and deployment
    - Graphical composition of complex application topologies
    - Deployment through PaaS orchestrator
- DEEP as a Service
    - Deployment of DEEP Open Catalog components as server-less functions
- DEEP Open Catalog
    - Ready-to-use machine learning and deep learning applications, including:
        * Machine learning frameworks + JupyterLab
        * Machine learning ready to use models
        * Deep learning ready to use models
        * BigData analytic tools

1.2.3 Release Notes

The DEEP-1/Genesis release consists in 9 Products and a number of technical guides:

- 17 OS packages, for CentOS 7 and Ubuntu 16.04
    - 8 RPMS & SRPMS
    - 6 binary & source DEBS
    - 3 tarballs
- 5 Docker containers

You can find in the later sections the full list of products, with detailed release notes and instructions for their installation/configuration.

1.2.4 Installation Notes

All DEEP - HybridDataCloud products are distributed from standard Operating Systems (OS) repositories and DockerHub registry of the indigodatacloud organization.

The packages repositories have the following structure:

- DEEP-HDC **production** (stable):

    - deep-hdc/production/{1,2}/centos7/x86_64/{base|updates}

    - deep-hdc/production/{1,2}/ubuntu/dists/xenial/main/{binary-amd64,source}

        * containing signed, well tested software components

    - third-party:

        * deep-hdc/production/{1,2}/centos7/x86_64/third-party

        * deep-hdc/production/{1,2}/ubuntu/dists/xenial/third-party{binary-amd64,source}

            · containing packages that are not part of DEEP, or not part of the base OS or EPEL, but used as dependencies by other DEEP components

- DEEP-HDC **testing**: deep/testing/{1,2}/{centos7,ubuntu}/

    - containing packages that will become part of the next stable distribution; in the certification and validation phase.

- DEEP-HDC **preview**: deep/preview/{1,2}/{centos7,ubuntu}/

    - containing signed packages that will become part of the next stable update, available for technical-previews

All packages in production and preview repositories are signed with the INDIGO - DataCloud gpg key. The public key can be downloaded from here, and the fingerprint from here.

Please import the key BEFORE starting!

- for CentOS7 save the key under /etc/pki/rpm-gpg/

```
# rpm --import https://repo.indigo-datacloud.eu/repository/RPM-GPG-KEY-indigodc
```

- for Ubuntu:

```
# wget -q -O - https://repo.indigo-datacloud.eu/repository/RPM-GPG-KEY-indigodc |
→sudo apt-key add -
```

Configuring the use of DEEP-HDC repositories

YUM & APT configuration files are available at:

- CentOS7 - https://repo.indigo-datacloud.eu/deep-hdc/repos/deep-1.repo

- Ubuntu 16.04 - https://repo.indigo-datacloud.eu/deep-hdc/repos/deep-1-ubuntu16_04.list

or use the deephdc-release package to install DEEP-HDC repositories:

- CentOS7:

```
# wget https://repo.indigo-datacloud.eu/repository/deep-hdc/production/1/centos7/x86_
↪64/base/deephdc-release-1.0.0-1.el7.noarch.rpm
# yum localinstall -y deephdc-release-1.0.0-1.el7.noarch.rpm
```

- Ubuntu 16.04:

```
# wget https://repo.indigo-datacloud.eu/repository/deep-hdc/1/ubuntu/dists/xenial/
↪main/binary-amd64/deephdc-release_1.0.0-1_amd64.deb
# dpkg -i deephdc-release_1.0.0-1_amd64.deb
```

These packages will install required dependencies, the INDIGO - DataCloud public key and ensures the precedence of DEEP-HybridDataCloud repositories over EPEL and Ubuntu.

It is strongly recommended the use of the lastest version of the **deephdc-release** packages containing the public key and the YUM and APT repository files.

Enable the DEEP-HDC Containers repositories

On the DockerHub Registry, DEEP - HybridDataCloud uses the INDIGO - DataCloud and DEEP-HDC Organizations:

- indigodatacloud, for Core Services
- deephdc, for DEEP-OC modules

Containers present in those repositories and released in DEEP-1 major release are tagged with "DEEP-1" tag and signed, leveraging the Docker's trust features so that users can pull trusted images.

Currently, content trust is disabled by default. You must enable it by setting the DOCKER_CONTENT_TRUST environment variable, like below:

```
export DOCKER_CONTENT_TRUST=1
```

For more details regarding the "Content Trust in Docker" please read Docker's Documentation

Content trust is associated with the TAG portion of an image. For DEEP-1 (Genesis) release the signed tag is DEEP-1. See example bellow if you want to ensure the correct use of DEEP-HDC images:

```
$ export DOCKER_CONTENT_TRUST=1

$ docker pull indigodatacloud/orchestrator:2.1.1-FINAL
No trust data for 2.1.1-FINAL

$ docker pull indigodatacloud/orchestrator:DEEP-1
Pull (1 of 1): indigodatacloud/orchestrator:DEEP-
↪1@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be: Pulling from
↪indigodatacloud/orchestrator
93857f76ae30: Pull complete
[...]
e8c92b40b492: Pull complete
Digest: sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
Status: Downloaded newer image for indigodatacloud/
↪orchestrator@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
Tagging indigodatacloud/
↪orchestrator@sha256:441c8b037684422ccdf2fdec322c9c09904ed3ce74d9fcc7d2862a9f53ad36be
↪as indigodatacloud/orchestrator:indigo_2
$ docker images
```

(continues on next page)

```
REPOSITORY                      TAG             IMAGE ID         CREATED         ␣
↪    SIZE
indigodatacloud/orchestrator    DEEP-1          bdbe758d9f32     37 hours ago    ␣
↪    843MB
```

1.2.5 Documentation

Please find DEEP-1 services and components documentation at *Included components*.

1.2.6 Support

Most complex software contains bugs, we are not an exception. One of the features of free and open source software is the ability to report bugs, helping to fix or improve the software you use.

DEEP - HybridDataCloud project uses the INDIGO Catch-All GGUS - Support Unit and the *deep-support@listas.csic.eu* for general support requests. More details regarding each product support channels are provided in the respective products release pages.

Developers, researchers and IT enthusiasts: feel free to write to *deep-info@listas.csic.es* to ask for more information on how to use DEEP solutions for your work. For automatic notifications you can register to the DEEP - HybridDataCloud release RSS feed or subscribe to the DEEP - HybridDataCloud Announce Mailing list. You can also socialize with us via Twitter or by joining our LinkedIn group.

CHAPTER 2

Release repositories

2.1 Source Code repositories

Source code repositories are available on GitHub under the **indigo-dc** and **deephdc** organizations - http://bit.ly/deep-hdc

2.2 Artefacts repositories

DEEP-HDC **production** (stable) repositories:

- deep-hdc/production/{1,2}/centos7/x86_64/{base|updates}
- deep-hdc/production/{1,2}/ubuntu/dists/xenial/main/{binary-amd64,source}
    - containing signed, well tested software components
- third-party:
    - deep-hdc/production/{1,2}/centos7/x86_64/third-party
    - deep-hdc/production/{1,2}/ubuntu/dists/xenial/third-party{binary-amd64,source}
        * containing packages that are not part of DEEP-HDC project, or not part of the base OS or EPEL, but used as dependencies by other DEEP-HDC components

YUM & APT **configuration files** are available here or use the **deephdc-release** package to install DEEP-HDC repositories

Release schedule

- **Time-based releases**

    - **projects' Major releases** - the DEEP-HybridDataCloud project foresees two major releases, distributions, during its lifetime, at around 10 months since the start of the project.

- **As-soon-as-available**

    - **components' Minor/Revision releases** - in a project Major release, Development teams (aka Product Teams) can release updated versions of their components as soon as the DEEP-HDC software quality criteria are met. Thought the project Continuous Integration and Delivery System tests are continuously run giving feedback on the status of the components.

3.1 Support Model

- in a Major Release for each component or service only the latest revision released is supported.

- for each component or service a (major) release is supported at least for the lifetime of the projects' major release in which this version was released the first time.

CHAPTER 4

Supported platforms

- DEEP-HybridDataCloud releases are supported for the following platforms:

    - **CentOS7 & Ubuntu 16.04**

        * for the products distributed through rpms and deb packages

    - **all platforms supporting Docker containers**

        * for the products distributed as docker images

4.1 Supported artifacts & packaging formats

- Packages:

    - Binaries: executable packages

    - Sources: when available, package files that contain all of the necessary files to compile/build the respective piece of software

    - Tarballs: clients are distributed as tarballs for all the platforms

- Containers: Docker images are available for almost all project software