DEEP Hybrid-DataCloud

Intensive computing techniques for applications needing specialised hardware

Workshop: Creating Platform-Driven E-Infrastructure Innovation On EOSC
10 July 2019

Mario David, Cristina Duma, Valentin Kozlov and Alessandro Costantini
On behalf of all the partners of DEEP
DEEP HybridDataCloud

- **Designing and Enabling E-Infrastructures for intensive data Processing in a Hybrid DataCloud**

- Started as a spin-off project (together with eXtreme DataCloud - XDC) from INDIGO-DataCloud technologies

- H2020 project, EINFRA-21 call

- Runs November 1st 2017 – April 2020

- 9 academic partners + 1 industrial partner:
  - CSIC, LIP, INFN, PSNC, KIT, UPV, CESNET, IISAS, HMGU, Atos
DEEP-HybridDataCloud

- **Goal**: prepare a new generation of e-Infrastructures that harness latest generation technologies, supporting deep learning and other intensive computing techniques to exploit very large data sources.

- **Global objective**: promote the use of intensive computing services by different research communities and areas, and the support by the corresponding e-Infrastructure providers and open source projects.

- **Ease and lower** the entry barrier for non-skilled scientists:
  - Transparent execution on e-Infrastructures
  - Build ready to use modules and offer them through a catalog or marketplace
  - Implement common software development techniques also for scientist’s applications (DevOps)

---

WP1 – Project Management & Exploitation
Lead by CSIC

WP2 – Intensive Computing PILOT Applications
Lead by HMGU

WP3 – Testbed and Integration with EOSC services
Lead by LIP

WP4 – Accelerated and HPC in the Cloud
Lead by IISAS

WP5 – High Level Hybrid Cloud
Lead by INFN

WP6 – DEEP as a Service
Lead by CSIC
DEEP architecture

Data management services (XDC, dCache, onedata)

CMDB
SLAM
Zabbix

CPR
Orchestrator

INDIGO PaaS

DEEP as a Service

Virtual router

WP2
Federated AAI DEEP IAM

WP6

WP5

WP4

https://a4c.ncq.ingrid.pt
https://github.com/deephdc/
https://cloud.docker.com/u/deephdc/
https://iam.deep-hybrid-datacloud.eu/
https://paas.cloud.cnaf.infn.it/orchestrator/
https://github.com/deephdc/
https://cloud.docker.com/u/deephdc/
https://im.deep-hybrid-datacloud.eu:8800
https://cereus.man.poznan.pl/mesos
https://mesos.ui.sav.sk/mesos-web/
https://cereus.man.poznan.pl
https://portal.cloud.ifca.es
https://dcache-xdc.desy.de:3880/
https://onezone.cloud.cnaf.infn.it
## DEEP core components

<table>
<thead>
<tr>
<th>Component</th>
<th>Workpackage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes</td>
<td>WP4</td>
</tr>
<tr>
<td>Mesos/Marathon/Chronos</td>
<td>WP4</td>
</tr>
<tr>
<td>OpenStack nova-lxd</td>
<td>WP4</td>
</tr>
<tr>
<td>udocker</td>
<td>WP4</td>
</tr>
<tr>
<td>PaaS Orchestrator</td>
<td>WP5</td>
</tr>
<tr>
<td>Orchent</td>
<td>WP5</td>
</tr>
<tr>
<td>Infrastructure Manager</td>
<td>WP5</td>
</tr>
<tr>
<td>CLUES-indigo</td>
<td>WP5</td>
</tr>
<tr>
<td>TOSCA types and templates</td>
<td>WP5, WP6</td>
</tr>
<tr>
<td>Monitoring System</td>
<td>WP5</td>
</tr>
<tr>
<td>CloudProviderRanker</td>
<td>WP5</td>
</tr>
<tr>
<td>Cloud Information Provider</td>
<td>WP5</td>
</tr>
<tr>
<td>INDIGO Virtual Router</td>
<td>WP5</td>
</tr>
<tr>
<td>Alien4Cloud - DEEP</td>
<td>WP6</td>
</tr>
<tr>
<td>DEEPaaS API</td>
<td>WP6</td>
</tr>
</tbody>
</table>

### Documentation

- Configuration
- Deployment
- Development
- Upstream

### Development & Upstream Contribution

- Requirements
- User communities
- Delivery
- WP2
- WP3
- WP4, WP5, WP6
IaaS components

udocker: improve support of GPUs, improve support of low latency interconnects (Infiniband)

Openstack: nova-lxd - add/improve support of GPUs

Mesos/Marathon/Chronos: authn/authz with oidc/oauth, documentation/configuration support for GPUs and Infiniband, recipes for on-demand deployments.

Kubernetes: same as mesos

HPC Integration Tools: Provides the way to access HPC resources from the PaaS Orchestrator - containers in HPC through udocker
PaaS components

- Develop/Improve support of deployments requiring GPUs and Infiniband
- Better support for the hybrid multi-site deployments
- Provide Tosca recipes for all required deployments
DEEPaaS components

Alien4Cloud: TOSCA template composition & deployment

DEEP Open Catalog/Marketplace: provides the universal point of entry to all services offered by DEEP.
DEEPaaS components

Deep Learning Applications

This webpage gathers all the applications developed at the Instituto de Física de Cantabria (IFCA) using deep learning techniques.

Image Recognition

Plants

Author: Ignacio Heredia

Description: This application classifies a plant image among 60 plant species mainly from Western Europe.

More Info
- Paper
- Github

REST API that is focused on providing unskilled users with access to machine learning models.

DEEP as a Service API endpoint

[Image of the API endpoint]

DEEP as a Service (DEEPaaS) API endpoint.

models  Model information, inference and training operations

GET  /models/  Return loaded models and its information

GET  /models/models  Return model's metadata

POST  /models/models/predict  Make a prediction given the input data

PUT  /models/models/train  Retrain model with available data

Models

- Models
- ModelResponse
- ModelMetadata
Plan and design

Implementation
DEEP core components: CI/SQA phase

1. Code fetching
2. Code style check
3. Unit testing coverage
4. SLOC metrics gathering
5. Functional and integration testing
6. Code Review
7. Documentation
8. Automated Deployment
9. Security linter/scanner
10. Vulnerability check on dependencies
11. Delivery
12. Notifications

Open access, open for contributions: https://github.com/indigo-dc/sqa-baseline
SQA controls

GitHub PR

Bandit (Python): security check

hardcoded_bind_all_interfaces: Possible binding to all interfaces.
Test ID: B104
Severity: MEDIUM
Confidence: MEDIUM
File: lib/config.py
More info: https://github.com/deep-hybrid-datacloud/DEEP-HDC-products/issues/5

Technical documentation

These pages contain technical notes software documentation, guides, tutorials, logbooks and similar documents produced with DEEP Hybrid DataCloud project

Mesos
- Introduction
- Toolshed Setup
- Prepare the agent (slave) node
- Testing Chrome patch for GPU support
- Testing GPU support in Marathon
- Running tensorflow/job container
- References
- Enabling open-id connect authentication

Kubernetes
- DEEP: Installing and testing GPU Node in Kubernetes - CentOS7
- Installing GPU nodes and adding it to Kubernetes cluster

OpenStack nova-lxd
- OpenStack nova-lxd installation via Ansible
- Deploying OpenStack environment with nova-lxd via Ansible
- Installing nova-lxd with Juju
- OpenStack nova-lxd testing configuration

uDocker
- uDocker new GPU Implementation

Miscellaneous
- DEEP hardware test
- DEEP software test
**Delivery:** Automatic build
- RPMs/DEBs.
- Docker images.
- Python packages.

**DEEPaaS component → PyPI automatically.**

**Notification:** Immediate availability of artefacts:
- **Automatic creation** of JIRA issue.
- Component version and artefact location.
- **Notification** sent to WP3 for testing.
DEEP core components: Software releases

Release, Maintenance & support schedule

Time-based releases:

2 major releases planned during the project lifetime.

Define periods for full and standard maintenance, security updates and end-of-life.

Updates of the components and services are provided as soon as new versions are announced and the SQA pipelines and controls finish successfully.
# DEEP-1 Genesis QC reports

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DEEP-1/Genesis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Upstream contributions**

<table>
<thead>
<tr>
<th>Upstream Code</th>
<th>Link</th>
<th>Partner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tosca-parser</td>
<td><a href="https://github.com/openstack/tosca-parser/commit/3af63cb0a8863e8ef8f899ca379f16b23f3e3">https://github.com/openstack/tosca-parser/commit/3af63cb0a8863e8ef8f899ca379f16b23f3e3</a></td>
<td>UPV</td>
</tr>
<tr>
<td></td>
<td><a href="https://github.com/openstack/tosca-parser/commit/23ed9914eb4b67481178a69ee30fd38b5eb7606e8">https://github.com/openstack/tosca-parser/commit/23ed9914eb4b67481178a69ee30fd38b5eb7606e8</a></td>
<td></td>
</tr>
<tr>
<td>PaaS Orchestrator</td>
<td><a href="https://github.com/openstack/tosca-parser/commit/3af63cb0a8863e8ef8f899ca379f16b23f3e3">https://github.com/openstack/tosca-parser/commit/3af63cb0a8863e8ef8f899ca379f16b23f3e3</a></td>
<td>docker image:</td>
</tr>
<tr>
<td>Orchestor</td>
<td>1.2.2</td>
<td>rpm &amp; deb:</td>
</tr>
<tr>
<td>TOSCA Types &amp; Templates</td>
<td>3.0.0</td>
<td>N/A</td>
</tr>
<tr>
<td>udocker</td>
<td>1.1.3</td>
<td>0.92 (lowest 02)</td>
</tr>
<tr>
<td>vRouter</td>
<td>DEEPv1</td>
<td>-</td>
</tr>
</tbody>
</table>
User communities and applications

**Citizen Science: Plant classification** ▶ **Image Classification**

For training and testing image classifiers (CNNs; TensorFlow). From this model the following services are derived:

- Plants (dataset: up to 1 TB)
- Conus marine snails
- Seeds
- Phytoplankton

**Earth Observation: Satellite Imagery**

Explore application of Machine Learning for satellite imagery (e.g. remote object detection, terrain segmentation, meteorological prediction).

Currently being developed is **super-resolution service** to upscale low resolution bands to high resolution with deep learning (e.g. DSen2; TensorFlow. Dataset: ca. 1 TB)
User communities and applications

**Biological and Medical Science: Retinopathy**

Diabetic retinopathy is a fast-growing cause of blindness worldwide. The use-case focuses on a deep learning approach (CNNs; TensorFlow) to automated classification of retinopathy based on color fundus retinal photography images (DR=0 (=healthy) .. 4 (blind)).

Dataset: ca. 100 GB

**Computing Security:**

**Massive Online Data Streams: Online analysis of data streams**

Intrusion detection systems: provide an architecture able to analyze massive on-line data streams, also with historical records, in order to generate alerts in real-time. Based on proactive time-series prediction adopting artificial neural networks (e.g. LSTM, GRU; TensorFlow). Dataset: 100 GB currently, then up to 2 TB /day

**Physics: Post-processing**

Of HPC simulations (Lattice QCD): analysis of a large number of configurations for Lattice QCD simulation. Move the configurations to long-term storage, perform checks and metadata operations. Requirements: Infiniband, data of 1 TB
DEEP serves different users’ profiles

**Unskilled* Users:** want to use a trained deep neural network for prediction/classification of their own data → no expert knowledge of ML → no access to high-level computing resources

**Intermediate* Users:** want to use a trained deep neural network and adapt it for solving their problem (transfer learning) → some knowledge of ML → access to limited computing resources

**Advanced* Users:** want to develop their own deep neural network (with special requirements as e.g. data privacy) → expert knowledge of ML → need access to high-level computing resources

(*) - knowledge level
DEEP from a user point of view


main entry point

DEEP Marketplace

Basic Users

Intermediate Users

Advanced Users

DEEPaaS

DEEP HybridDataCloud Pilot e-Infrastructure

Authentication · Authorization · Storage · Computing · Orchestration

Unified token-based authentication
DEEP CI/CD for user applications

Development and integration in cloud resources.

Jenkins pipeline for user applications:

CI/SQA: Code style, security scan.

CD:

- **Immediate availability** of application.
- **Automatic building** (Docker images).
- **Automatic publishing** (Docker Hub).
- **Notification** (email to developers).
DEEP user applications

WP3
Code version control
- flake8 / PEP8
- Bandit security scanner
- Unit tests
- Coverage report
- OWASP Dependency-check

WP2
Code testing & Quality Control
- Jenkins
- dockerhub

Docker build & push to test registry
- Jenkins

Deploy to test environment

Post-deployment testing

Under discussion
Possible extension to be included in Jenkins pipeline

Blue-Green deployment on production
Deployment for long training

Engagement
Achievements: users perspective

- **Encapsulation and isolation** of different environments (using container)
  - Enable operativity in different infrastructure
- **DEEPaaS** as an entry points with **flexible design**
  - Allow different training arguments for each use-case
- **DEEP DS template** and **DEEPaaS API**
  - ready-made template that facilitates standardisation and (semi)automatic creation of necessary files and codes
Achievements: users perspective

- **DEEP OC software automation** DevOps: CI/CD pipelines for user applications
- **DEEP Leading Interaction**
  - A good way to provide technical support for users as well as accompanying "HowTo" documentation with details
- **Udocker extended support**
  - Older Linux kernels are also supported and does not require root privileges
  - Functionalities were critically valuable in test running docker images at an old local cluster.
- **Orchestrator Dashboard:** WebUI where users can submit TOSCA templates to the orchestrator
## Use cases status

<table>
<thead>
<tr>
<th>Plant Classification</th>
<th>Problem</th>
<th>Goal</th>
<th>DEEP services</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plant Classification</td>
<td>Automatically identify plant species from images using Deep Learning</td>
<td>Perform image classification on different datasets by performing the so called transfer learning</td>
<td>Tensorflow, Keras via DEEP PaaS Orchestrator, OIDC-agent</td>
<td>Deployed through orchestrator and oidc-agent in the DEEP testbed</td>
</tr>
</tbody>
</table>

| Satellite Imagery | Explore possible applications of machine learning techniques to satellite imagery from different sources | Support a super-resolution service to upscale low resolution bands to high resolution with Deep Neural Networks | Tensorflow, Keras, Pytorch via DEEP Paas Orchestrator, OIDC-agent, Alien4Cloud | Only DEEPaas API available Will be implemented in the DEEP testbed via A4C |

| Retinopathy | Automated classification of retinopathy based on color fundus retinal photography images | Perform deep learning approach for image classification | Tensorflow via DEEP PaaS Orchestrator, OIDC-agent | Deployed through orchestrator and OIDC-agent in the DEEP testbed |

| MODS | Intrusion detection systems: provide an architecture able to analyze massive on-line data streams, also with historical records, in order | Generate alerts in real-time using ML and DEEP learning approaches. | Tensorflow, Keras via DEEP PaaS Orchestrator, OIDC-agent, Alien4Cloud | Deployed through orchestrator and OIDC-agent in the DEEP testbed. To be implemented via A4C |
DEEP-HybridDataCloud highlights
DEEP vision & work on Software Quality Assurance

● Vision:
  ○ We support the HLEG vision on delivering quality software for the EOSC
  ○ We produced “A set of common software quality assurance baseline criteria for research projects”
    ■ http://hdl.handle.net/10261/160086
    ■ Done together with the eXtreme DataCloud and INDIGO projects
  ○ Objective: align baseline criteria within different projects

● Work:
  ○ Current EOSC synergies (eXtreme-DataCloud)
    ■ SQA baseline
    ■ Automation: continuous integration and delivery for core products
      ● Common library for CI/CD pipeline functionalities
    ■ Agile software development: jump-started from WP2 requirements
  ○ DEEP goes beyond: automation techniques supporting user communities
    ■ Continuous integration and delivery pipelines in place: Docker Hub images re-creation triggered by changes in i) DEEPaaS and ii) application itself
    ■ Initial continuous deployment prototype: readiness/provision of training and inference as a service
    ■ Rendering and generation of the marketplace portal: leveraging (JSON) schema- validated metadata descriptions
DEEP-Genesis: 1st platform and release

- First software release and prototype platform released Jan. 2018
- More than 12 software components, 4 different services, several upstream contributions, more than 10 models in marketplace
- DEEP-Genesis: initial service catalog
  - All services are OIDC-ready and follow the AARC and AARC2 blueprint recommendations

<table>
<thead>
<tr>
<th>Service</th>
<th>Functionalities</th>
<th>Preview endpoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual application topology composition and</td>
<td>• Graphical composition of complex application topologies</td>
<td><a href="https://a4c.ncg.ingrid.pt">https://a4c.ncg.ingrid.pt</a></td>
</tr>
<tr>
<td>deployment</td>
<td>• Deployment through PaaS orchestrator</td>
<td></td>
</tr>
<tr>
<td>ML/DL training facility as a service</td>
<td>• Provide continuous training and retraining of developed models</td>
<td>On demand</td>
</tr>
<tr>
<td>DEEP as a Service</td>
<td>• Deployment of DEEP Open Catalog components as server-less functions</td>
<td>(beta, internal preview only)</td>
</tr>
<tr>
<td></td>
<td>• Provide inference/prediction endpoints</td>
<td><a href="https://deep.cloud.ifca.es/">https://deep.cloud.ifca.es/</a></td>
</tr>
<tr>
<td>DEEP Open Catalog</td>
<td>• Ready-to-use machine learning and deep learning applications, including:</td>
<td><a href="https://marketplace.deep-hybrid-datacloud.eu">https://marketplace.deep-hybrid-datacloud.eu</a></td>
</tr>
<tr>
<td></td>
<td>▶ Machine learning frameworks + JupyterLab</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ ML/DL ready to use models</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▶ BigData analytic tools</td>
<td></td>
</tr>
</tbody>
</table>
Collaboration with ongoing initiatives

- **Collaboration with EINFRA-21 projects:**
  - eXtreme-DataCloud: Integration of data management solutions (XDC) and computing solutions (DEEP), exploiting event driven executions. Work on software quality.
  - DARE: Provide ML/DL services to integrate into workflows

- **Collaboration with other initiatives:**
  - EOSC-Hub: integration of developments into production tools (cloud-info-provider, TOSCA-Parser).
  - EGI.eu: Improved support for accelerators in Cloud resources
  - Developments merged upstream

- **Collaborations with external user communities:**
  - Royal Botanical Garden of Madrid, LifeWatch ERIC, Mouse Motor Lab (Rowland Institute Harvard), Centre for Automatic and Robotics (CSIC)
Selected DEEP early results


- User communities publications:

- Published articles by user communities not in the project, exploiting DEEP-HybridDataCloud software components:
Thank You
Next steps: users perspective

- Serverless framework planned to be developed
  - based on OpenWisk platform
- Further development of FLAAT:
  - FLA sk support for handling OIDC Access Tokens
- Extend the CI/CD pipeline:
  - Include deployment, testing of produced application Docker images from its DockerHub repository.
- General improvements:
  - Improve documentation based on feedback from users.
  - Perform training actions: through dedicated video conferences and webinars
  - Individual support through 1-to-1 TeleConferences
- Development Docker Image (DDI)