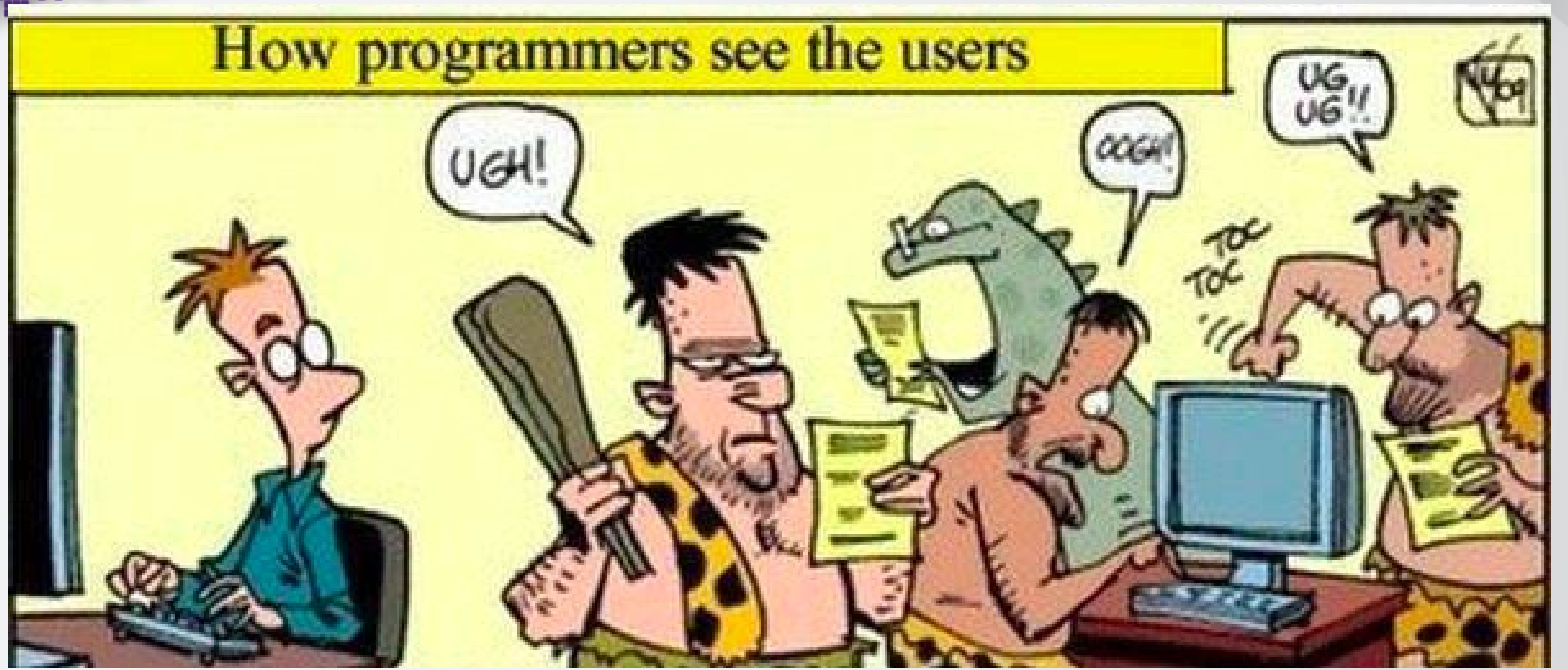
RIA-653549

The First INDIGO-DataCloud Software Release

Presenter : Paul Millar for
Patrick Fuhrmann for
Davide Salomoni
INDIGO-DataCloud Project Coordinator
HGF “Physics at the Terascale”, November 2016, DESY
davide.salomoni@cnaf.infn.it

INDIGO-DataCloud is co-founded by the Horizon 2020Framework Programme

Something ~~is~~ was still missing in the Cloud world...



Source: http://goo.gl/wT8XEq

From the Paper “Advances in Cloud”

EC Expert Group Report on Cloud Computing,

- http://cordis.europa.eu/fp7/ict/ssai/docs/future-cc-2may-finalreport-experts.pdf

The Expert Group identified the following unsolved issues in the currently deployed ecosystem :

- Open Interoperation across (proprietary) Cloud solutions at IaaS, PaaS and SaaS level has not yet been developed.
- No solutions are available to manage multitenancy at large scale and heterogeneous environments.
- No dynamic and seamless elasticity from in- house Cloud to public Clouds ...
- Datamanagement: Problems with bandwidth, security and privacy between public and private clouds.

From the Paper “Advances in Cloud”

Suggestions :

[...] A major opportunity for Europe involves finding a SaaS interoperable solution across multiple CLOUD platforms.

Another lies in migrating legacy applications without losing the benefits of the CLOUD, i.e. exploiting the main characteristics, such as elasticity etc.

Gap analysis

- What is missing:
  - Open **interoperation** / federation across (proprietary) CLOUD solutions at
    - IaaS,
    - PaaS,
    - and SaaS levels
  - Managing **multitenancy**
    - At large scale…
    - … and in heterogeneous environments
  - Dynamic and seamless **elasticity**
    - For both private and public cloud…
    - … and for complex or infrequent requirements
  - **Data management** in a Cloud environment
    - Due to technical…
    - … as well as to legal problems

Filling these gaps should lead to:

- Interoperable PaaS/SaaS solutions addressing both public and private Cloud infrastructures
- Migration of legacy applications to the Cloud

INDIGO-DataCloud

- **An H2020 project** approved in January 2015 in the EINFRA-1-2014 call
  - 11.1M€, 30 months (**from April 2015 to September 2017**)
- **26 European partners** in 11 European countries
  - Coordination by the Italian National Institute for Nuclear Physics (INFN)
  - Including developers of distributed software, industrial partners, research institutes, universities, e-infrastructures
- **Develop an open source Cloud platform** for computing and data ("DataCloud") tailored to science.
- **Targeting Multi-disciplinary scientific communities**
  - E.g. structural biology, earth science, physics, bioinformatics, cultural heritage, astrophysics, life science, climatology
- Deployable on **hybrid (public or private) Cloud infrastructures**
  - INDIGO = **IN**tegrating **D**istributed data **I**nfrastructures for **G**lobal Expl**O**itation
- **In response** to the technological **needs of scientists** seeking to easily exploit distributed Cloud/Grid compute and data resources.

User (Scientist) first

Users first: from here...

Use-Cases from
LifeWatch
EuroBioImaging
INSTRUCT
LBT
CTA
WeNMR
ENES
eCulture
ELIXIR
EMSO
Dariah
WLCG

100 distinct requirements

Converted to concrete activities in the Project DoW

Computational

- Software as a Service
- Execution of Workflows
- Cloud Bursting
- X-Site Execution
- Improved Scheduling
- Access to GP-GPU's

Storage

- Distributed Storage, accessible via POSIX
- Persistent Data Storage

Infrastructure

- Global Level AAI
- Software Defined Networks

Using “Champion” approach :

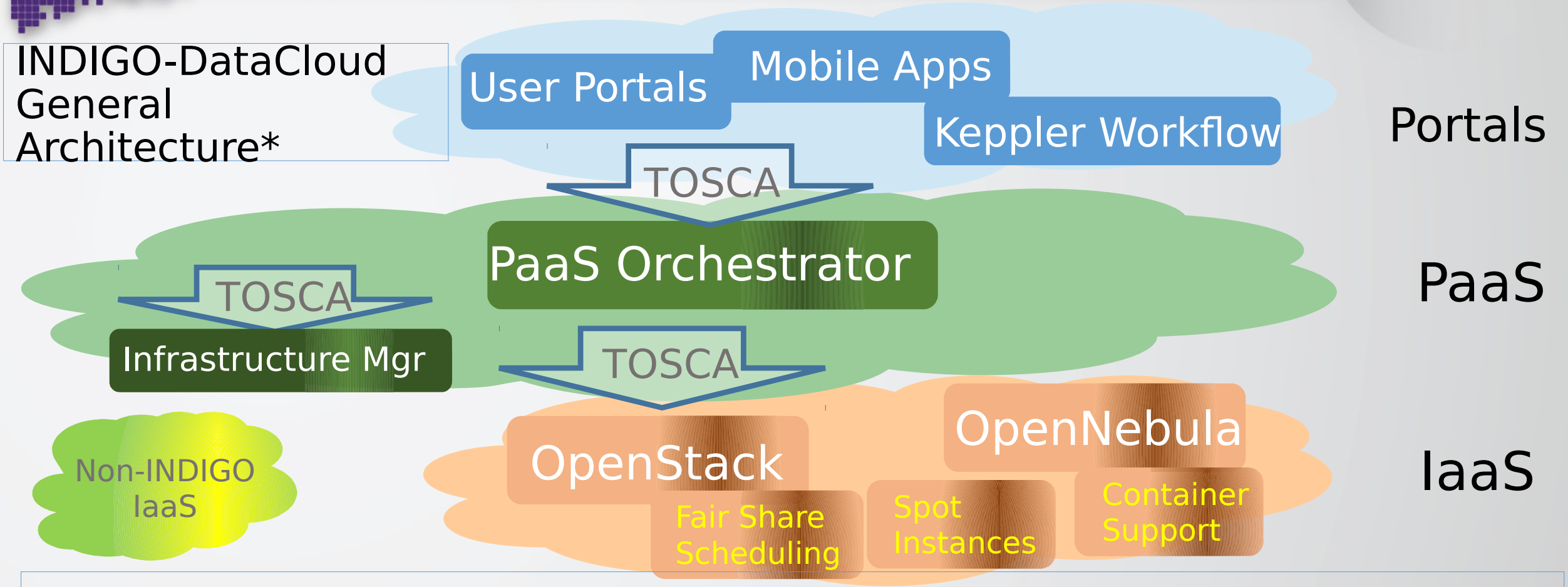
Communities have to provide a scientist, becoming an expert in computing and INDIGO terminology.

- Data Repository platform for DARIAH

https://www.indigo-datacloud.eu/documents-deliverables

The long road to the release, from the architecture…

INDIGO-DataCloud General Architecture*



*: see details in http://arxiv.org/abs/1603.09536 or in https://www.indigo-datacloud.eu/documents-deliverables

Software Release and Maintenance

1. Release management
   1. Publish release schedules
   2. Manage project public repositories
   3. Continuous integration
   4. Continuous delivery
2. Software maintenance & support
   1. Change Management processes
   2. Support to released software
   3. Problem Management process

The first INDIGO-DataCloud Software Release

INDIGO - DataCloud

August 8, 2016

INDIGO-DATACLOUD FIRST PUBLIC RELEASE IS OUT!

INDIGO MIDNIGHTBLUE

On August 8, 2016 INDIGO-DataCloud project announced the general availability of its first public software release, codenamed MidnightBlue. The release comes after an initial phase of requirement gatherings which involved several European scientific collaborations in areas as diverse as structural biology, earth sciences, physics, bioinformatics, cultural heritage, astrophysics, life sciences, climatology, etc. This resulted in the development of many software components addressing existing technical gaps linked to easy and optimal usage of distributed data and compute resources.

INDIGO - DataCloud
Better Software for Better Science

INDIGO - DataCloud

INDIGO MidnightBlue Service Catalogue

Updates and new releases of the INDIGO services are expected to come in the forthcoming months.
The first scientific applications and use cases adopting this first INDIGO release are expected starting from September 2016.

https://www.indigo-datacloud.eu/communication-kit

INDIGO-DATACLOUD FIRST PUBLIC RELEASE IS OUT!

INDIGO MIDNIGHTBLUE

INDIGO - DataCloud

Common Solutions	
Identity and Access Management	14
Data Center Solutions	
Fairshare Scheduler for OpenStack	15
Partition Director Service for Batch and Cloud resources	16
Cloud Provider Ranker	17
Infrastructure Manager	23
OCCI support for OpenStack and OpenNebula	24
Extended OpenStack and OpenNebula Functionalities	25
Data Solutions	
Global Data Access	18
Storage Quality of Service and Data Lifecycle support	27
Automated Solutions	
PaaS Orchestrator	20
Core PaaS	26
QoS/SLA Management Service	30

User-oriented Solutions	
Userspace Container Support	21
Data Mining and Analytics for eScience Server	22
Future Gateway (Programmable Scientific Portal)	28
INDIGO Plug-ins for scientific workflow systems	29
INDIGO Mobile Toolkit	31

The AAI common glue, implemented across the entire INDIGO architecture.

1. Identity and Access Management

Short Service Name

IAM

Solution Type

Common Solution

Installation Area

The IAM service provides a layer where identities, enrolment, group membership, attributes and policies to access distributed resources and services can be managed in a homogeneous and interoperable way. It supports the federated authentication mechanisms behind the INDIGO AAI.

The IAM service provides user identity and policy information to services so that consistent authorization decisions can be enforced across distributed services.

Identity and Access Management is provided through multiple methods (SAML, OpenID Connect and X.509) by leveraging on the credentials provided by the existing Identity Federations (i.e. IDEM, EDUGAIN, etc). Distributed authorization policies and Token Translation Service will guarantee selected access to the resources as well as data protection and privacy.

2. Fairshare Scheduler for OpenStack

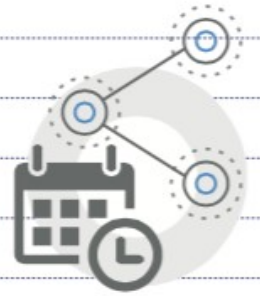
Short Service Name

Synergy

Solution Type

Data Center Solution

Installation Area



Synergy, the NDIGO Fairshare Sceduler for Openstack, is an extensible all-purpose management service for integration with OpenStack Infrastructures. It is implemented by a collection of independent pluggable tasks and executed periodically (e.g. cron jobs) or interactively (e.g. RESTful API). Synergy can be used to allocate a set of dynamic OpenStack resources to be shared among different projects. Moreover, Synergy offers a queuing mechanisms for requests until relevant resources are available. It can oversee the instantation of both virtual machines and containers managed via the nova-docker service.

3. Partition Director Service for Batch and Cloud resources

Short Service Name

Dynpart

Solution Type

Data Center Solution

Installation area

Dynpart, the Partition Directore Service for Batch and cloud resources, facilitates the management of a hybrid data center that provides both batch-system based services and cloud-based services. Physical computing resources, in fact, can act as member of batch system cluster or as compute node in a cloud environment.

Dynpart can easily manage such mutual exclusive approach of physical resources making the data center dynamic and flexible.

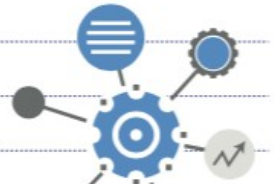
6. PaaS Orchestrator

Short Service Name

Orchestrator

Solution Type

Automated Solution



«**OneData allows us** to implement a first prototype of distributed archive for the Cherenkov Telescope Array (CTA) project. The distributed architecture of the CTA Archive will allow to lower costs with respect to a single huge data centre including easy manageability and maintenance».

INAF

Eva Sciacca,
Researcher at INAF – Astrophysics Observatory of Catania, Italy

collects high-level deployment requests from the software layer, and coordinates the resource or service deployment over dynamic Mesos clusters or directly over IaaS platforms.

5. Global Data Access

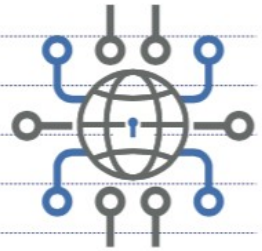
Short Service Name

OneData

Solution Type

Data Solution

Installation Area



Global Data Access is the global data management system providing easy access to distributed storage resources and supports a wide range of use case, from data management to data-intensive scientific computations

INDIGO communities

Physics & Astrophysics

«**We tried Onedata** for the Large Binocular Telescope (LBT) use case and configured this service for a simulated distributed archive. We tried to install and configure all Onedata components (Onezone, Oneprovider and Oneclient) on dedicated virtual machines, deploying docker images. The main goal was to use Onedata to store and distribute data to different sites according to defined data policies».

«Onedata provides an intuitive configuration interface and a very flexible framework to store data using global distributed storage providers».

Andrea Bignamini,
Researcher at the Astronomical Observatory of Trieste, Italy

9. Infrastructure Manager

Short Service Name

INDIGO communities

Biological & Medical science

«The Infrastructure Manager gives you full flexibility to write your own recipes to deploy and configure your cluster. The user can also use already available RADLs to deploy a cluster. The interface is user friendly, and tutorials are helpful. The user can also store his/her credentials for not only one but multiple resource providers (Amazon, Google, EGI etc), which is a big plus"

Zeynep Kurkcuoglu,

Bijvoet Center for Biomolecular Research, the Netherlands

INDIGO Champion for the "Virtualization of the HADDOCK portal" use case

virtual infrastructure.

12. Core Pa

Short Service N

Kubernetes, Mor

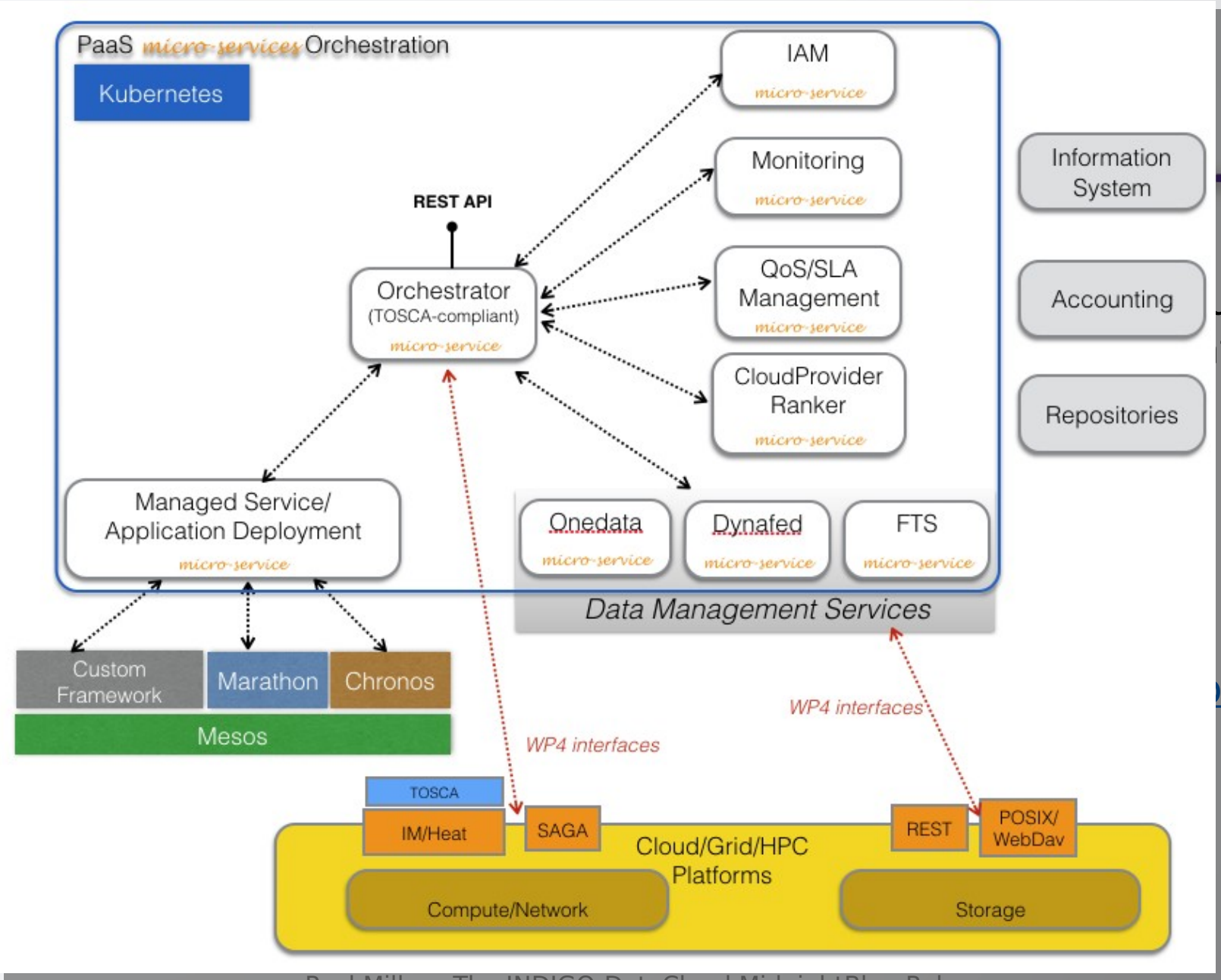
Solution Type

Automated Solu

Installation Are

Core PaaS provi
performance of a
particular:

- availability an
- monitoring of PaaS µService
- accounting of



INDIGO - DataCloud

uilt upon a set of
terfaces) that are:

://kubernetes.io),
managing
across multiple

13. Storage Quality of Service and Data Lifecycle support

Short Service Name

CDMI

Solution Type

Data Solution

Installation Area

This solution implements the INDIGO-DataCloud CDMI Server, a set of functionalities aimed at Improving QoS capabilities of storage resources for better support of high-level storage requirements, such as flexible allocation of disk or tape storage space and support for data life cycle.

- This is an enhancement also with respect to what is currently available in public clouds, such as Amazon Glacier and Google Cloud Storage.

CDMI provides the official reference implementation of the SNIA Cloud Data Management Interface (CDMI), an ISO standard, and also a Spring Boot application port of the SNIA CDMI-Server. The CDMI server has been extended to support Quality-of-Service (QoS) and Data Life-cycle (DLC) operations for multiple storage back-ends like dCache, Ceph, GPFS, Gemss+TSM, StoRM and HPSS.

Technical Support

Most complex software contains bugs, and 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. The INDIGO-DataCloud project uses the GGUS (Global Grid User Support) tool as its user support system. It provides sophisticated search functionality, report generation, interfaces to bug tracking systems used by different middleware components, and automatic ticket reminder including escalation indication. Please use the INDIGO-DataCloud Catch-All GGUS Support Unit or directly contact us through the indigo-su@lists.indigo-datacloud.eu mailing-list.

Share the INDIGO Experience

Developers, researchers and IT enthusiasts: feel free to write to info@indigo-datacloud.eu to ask for more information on how to deploy your PaaS-based solution for your work. For automatic notifications, you can register to the INDIGO-DataCloud RSS release feed or subscribe to the INDIGO-DataCloud Announce Mailing list. You can also socialize with us via Twitter, Facebook and LinkedIn. Finally, you can also subscribe to **INDIGO Newsletters** and receive communications about the project, such as new releases, community events and other events where to meet the INDIGO team, tutorials, workshops, webinars, guides, and more.

- INDIGO-Datacloud Website: https://www.indigo-datacloud.eu
- Technical Support: https://www.indigo-datacloud.eu/indigo-support-and-technical-services
- Twitter: https://twitter.com/indigodatacloud
- Facebook: https://www.facebook.com/indigodatacloud
- Linkedin: https://it.linkedin.com/in/indigodatacloud

What this means for dCache

- INDIGO-DataCloud providing some €0.5M of project money to dCache team:

  Funding 3 FTEs working on dCache.

- Work follows the DoW:

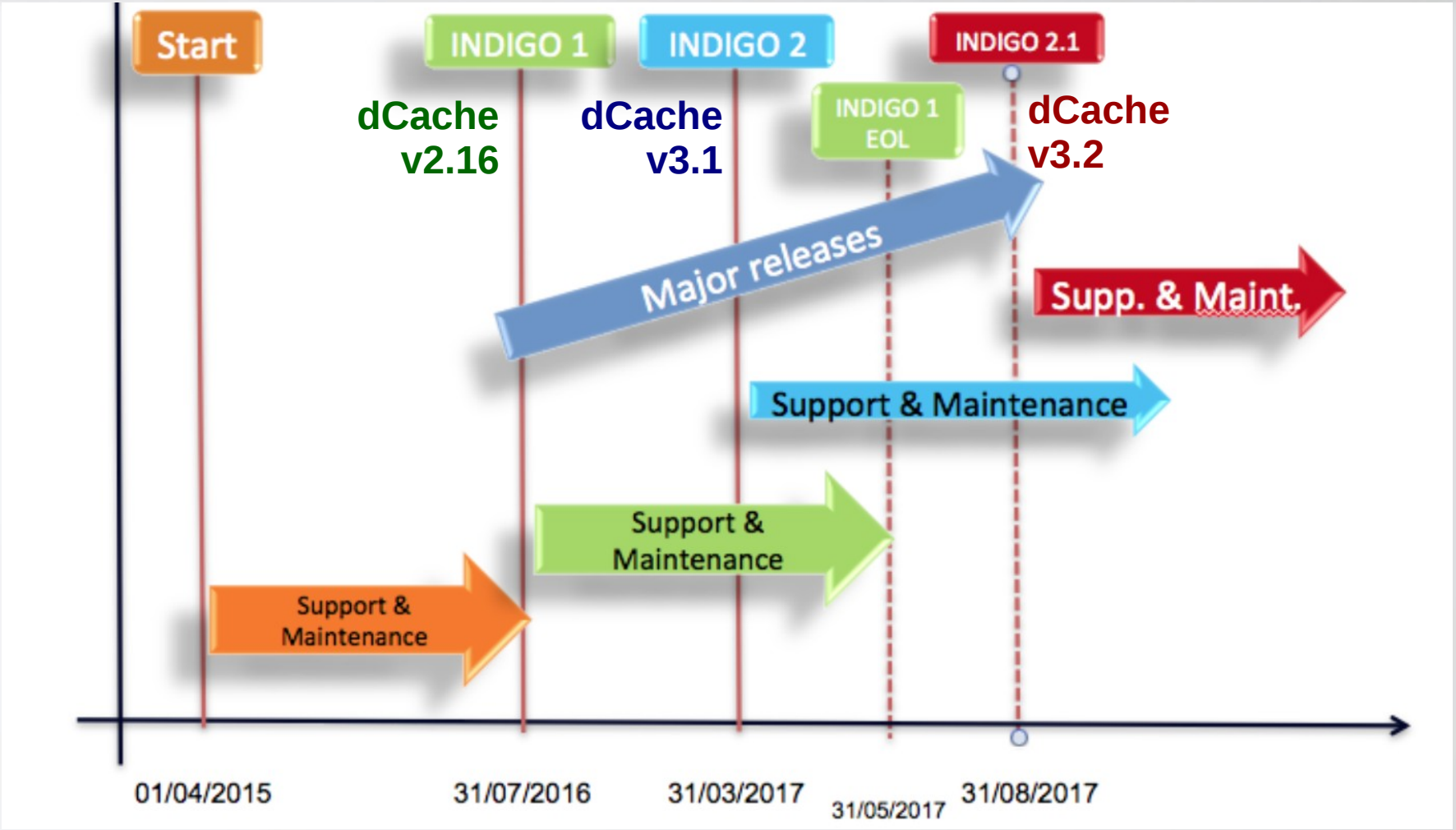
  HEP (through WLCG) has supplied use-cases

- Improvements within dCache include:
  - Adding OpenID Connect support to dCache
  - Improving Quality of Service management options,

    Media, number of copies, ...
  - Improving Quality of Service management,

    Focus is on QoS management via CDMI.

Release Timeline



Conclusions

- The **first public INDIGO release** was issued at the beginning of August 2016.
- Its services are already available in several testbeds.
- Concrete use cases are currently being implemented by many scientific communities.
- A lot of important developments are being carried on in coordination with upstream developers, so that code maintenance is not only upon us.
- **Now looking** for early adopters / people willing to test and run INDIGO components with their applications or requirements. **If interested, please contact us**.
- We look forward to providing these components in a future **European Open Science Cloud through INFRADEV-4-2016 and EINFRA-12-2017 projects**.
  - And extending them through **EINFRA-21-2017** projects.

Thank you

INDIGO - DataCloud

https://www.indigo-datacloud.eu

Better Software for Better Science.

Sample use cases

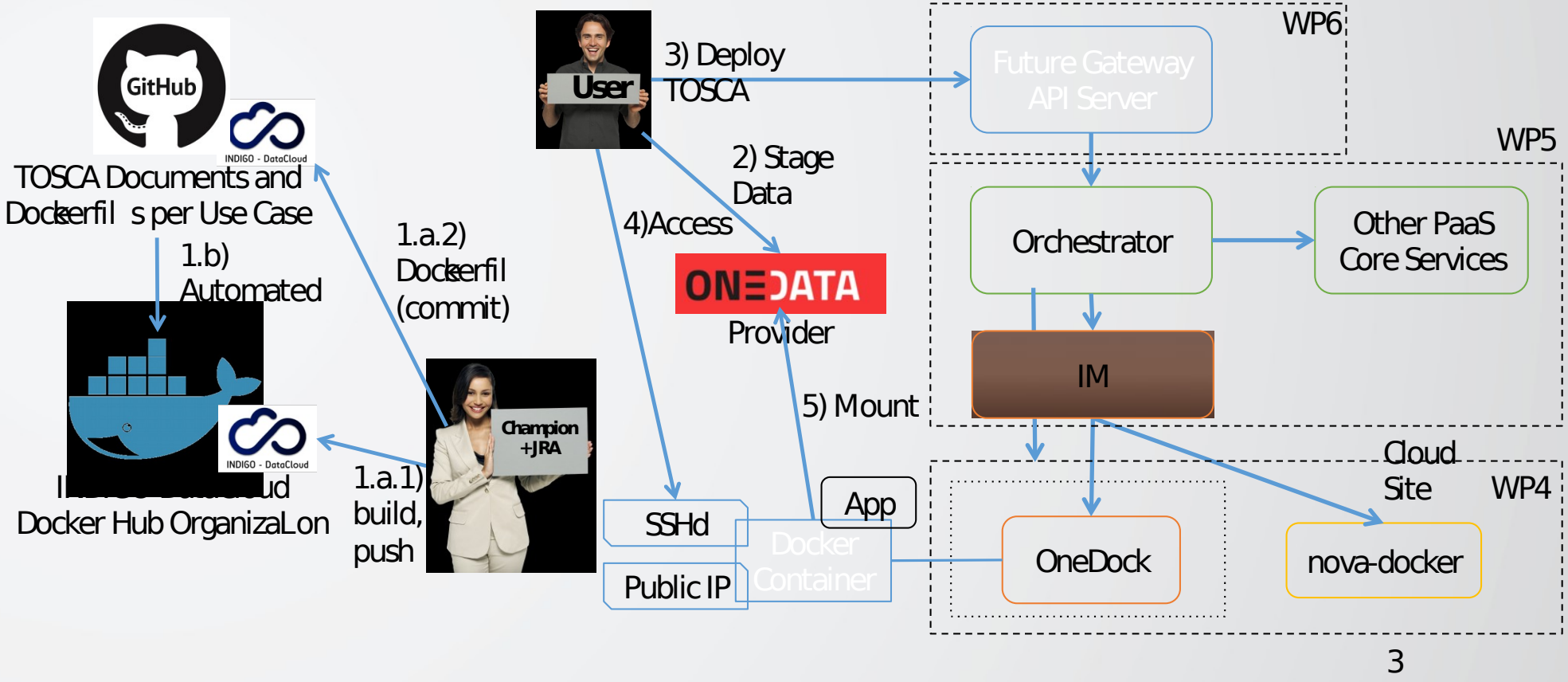
Sample use cases:

1. Interactive usage of a Docker container with *ssh*
2. A web portal that uses a batch system to run applications
3. Virtual infrastructures for Medical Imaging Biobanks
4. An application to CMS
5. Running Docker containers without Docker

UC #1: Interactive usage of a Docker container with *ssh*

- A Docker container is instantiated automatically after a simple request on the web portal from an end-user.
  - This will exploit a **TOSCA Template** through the INDIGO orchestrator.
- The container has a public IP address and the user (or the portal) can directly get access to it.
- Users can mount a local **or remote** posix filesystem through INDIGO **Onedata**.
- The application in the Docker container is able to simply read the files provided via web browser by the end user and to write posix files that are available to users via web browsers.
- The same Docker container could be used to execute a large list of applications in a batch-like behaviour.

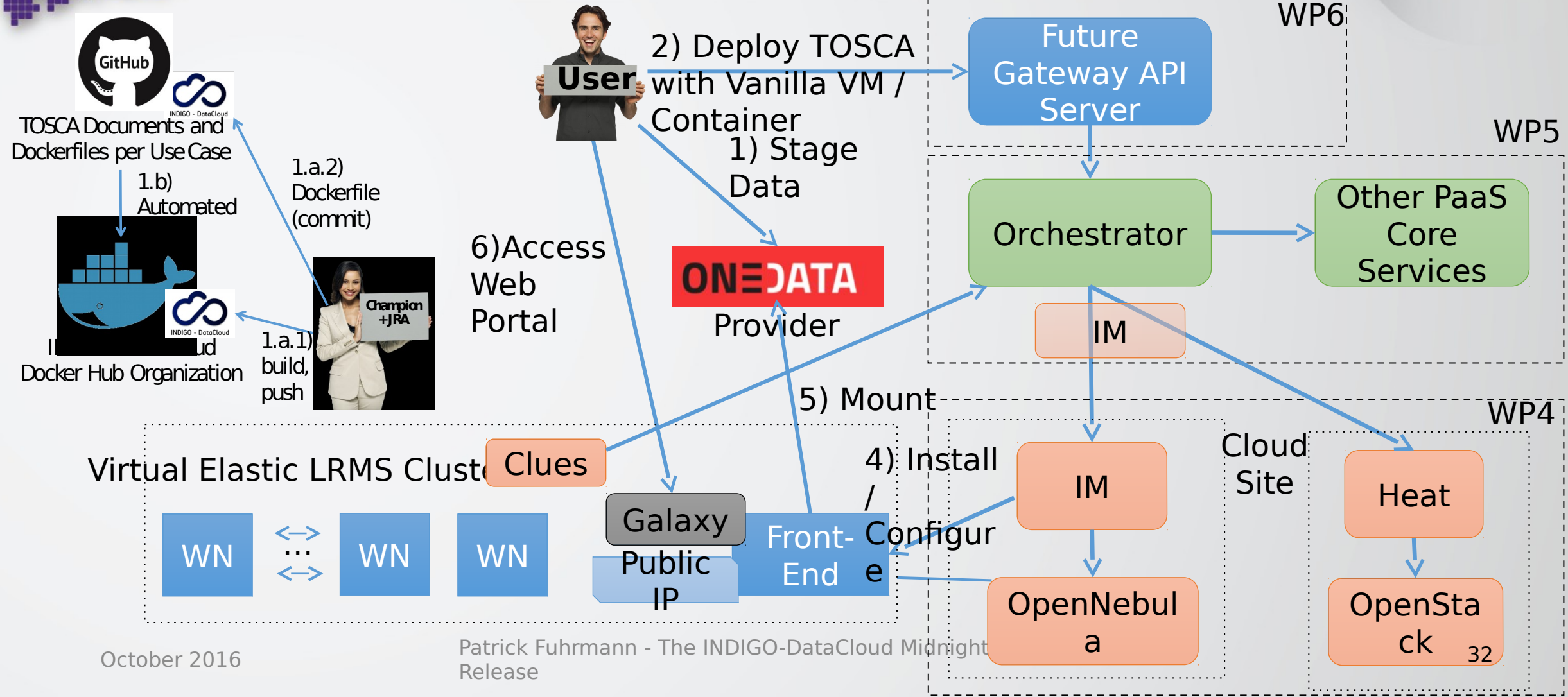
UC #1: Interactive usage of a Docker container with *ssh* - Overview



UC #2: A web portal that uses a batch system to run applications

- A user community maintains a “vanilla” version of a portal using Galaxy, a computing image, plus some specific recipes to customize software tools and data
  - Portal and computing are part of the same image that can take different roles.
  - Customization may include creating special users, copying (and registering in the portal) reference data, installing (and again registering) processing tools.
  - Typically web portal image also has a batch queue server installed.
- All the running instances share a common directory.

UC #2: A web portal that uses a batch system to run applications - Overview



UC#3: Virtual infrastructures for Medical Imaging Biobanks

The goal: provisioning of scalable and fully customized virtual infrastructures with access to data subsets.

Four steps:

1. Data fetching and anonymization using dcm2nii and copy of the data into a volume.
2. Defining the execution environment, involving a batch queue (condor/torque) and the tools on the nodes (Caffe, octave and user-specific image processing components).
3. Deployment of the environment with the volume mounted (Read only), and additional (Read & Write) volumes for results.
4. Data processing using the interface.

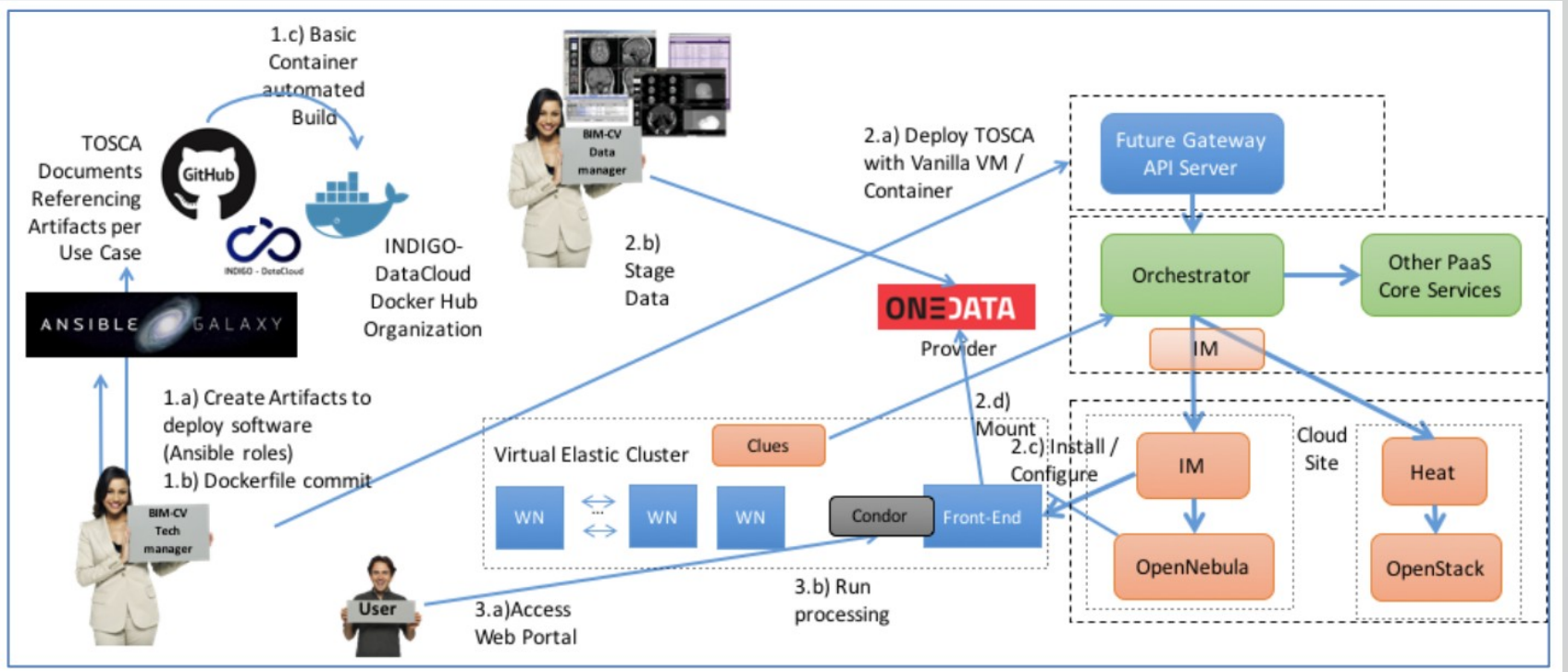


Figure 4. 4: Medical Imaging Databanks architecture with INDIGO

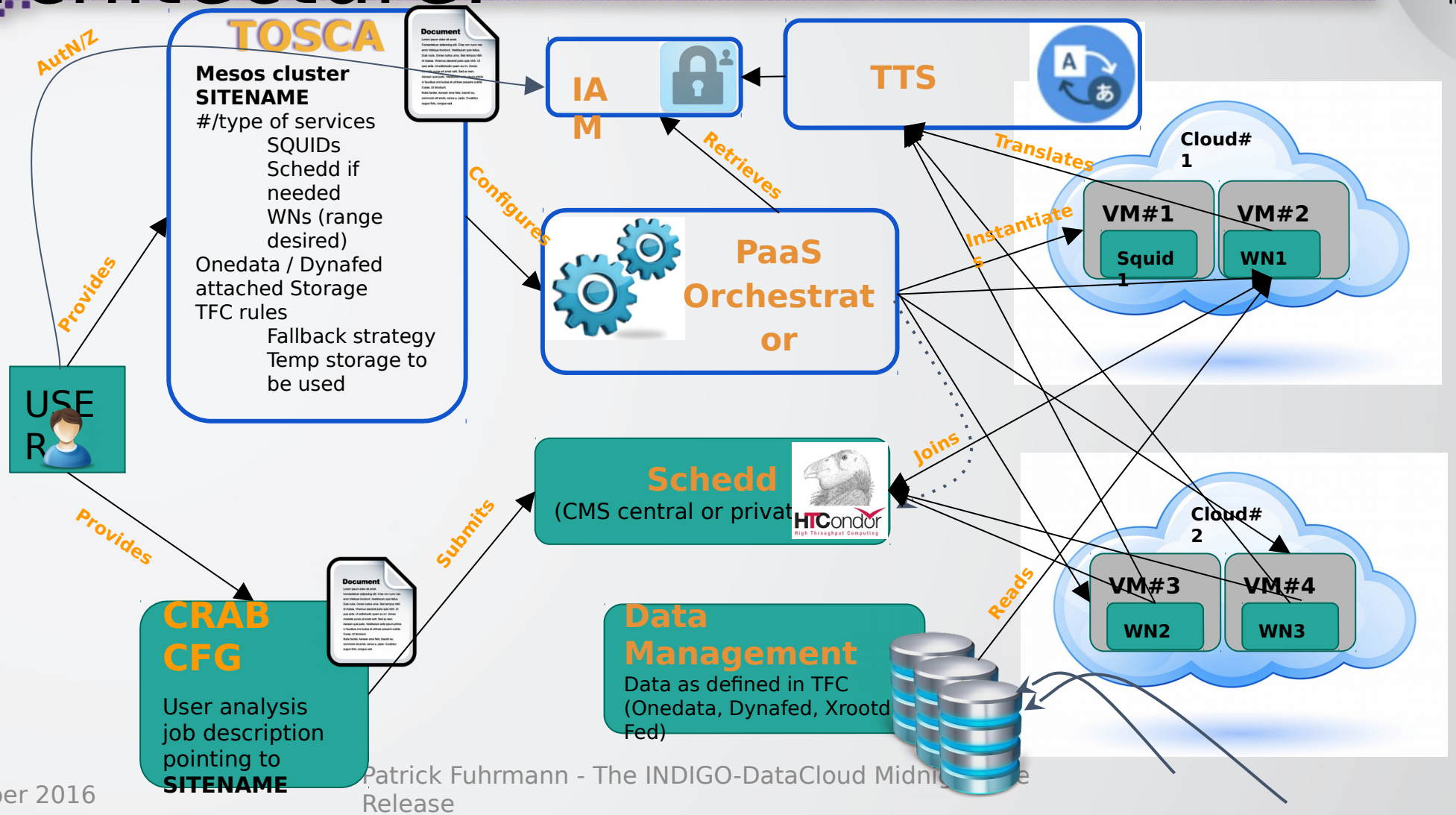
UC#4: An application to LHC/CMS

- The **goal** is to develop a solution for generating automatically an on-demand, container-based cluster for CMS in order to allow:
  - The effective use of **opportunistic resources**, such as general purposes campus facilities.
  - The **dynamic extension** of an already existing dedicated facility.
- By simplifying and automating the process of creating, managing and accessing a pool of computing resources the project aims to **improve**:
  - **Sites management:**
    - A simple solution for dynamic/elastic T2 extensions on “opportunistic”/stable resources
    - A friendly procedure to dynamically instantiate a spot “Tier3-like resource center”
  - **Users experience:**
    - Generation of an ephemeral on-demand T3 seen by the Experiment computing infrastructure as a personal WLCG-type facility, in order to serve a group of collaborators. The system must allow the use of standard/regular CMS Tools such as CRAB.
  - **Experiment-Collaboration resources:**
    - A comprehensive approach to opportunistic computing. A solution to access and orchestrate e.g. multiple campus centers, harvesting all the free CPU cycles without major deployment efforts.

UC#4: Application to CMS, four pillars:

- **Cluster Management:**
  - Mesos clusters as a solution in order to execute docker for all the services required by a regular CMS site (Worker Nodes, HTCondor Schedd and squids).
  - Marathon guarantees us the dynamic scaling up and down of resources, a key point.
- **AuthN/Z & Credential Management:**
  - The INDIGO Identity Access Management (IAM) service is responsible for AuthN/Z to the cluster generation.
  - The Token Translation Service (TTS) enables the conversion of IAM tokens into an X.509 certificate
    - NOTE: This allows Mesos slaves (running HTCondor_startd daemon) to join the CMS central queue (HTCondor_schedd) as a regular Grid WN
- **Data Management:**
  - Dynafed is the approach currently followed by the project. A further possibility we will investigate is Oneclient (from Onedata) as a tool allowing to mount remote Posix filesystems.
- **Automation:**
  - TOSCA templates, meant to be managed by INDIGO PaaS Orchestrator, allow the automation of the overall setup.
    - The aim is to produce a single YAML file describing the setup of all required services and deps.

UC#4: Application to CMS, architecture:



UC#5: Running Docker containers without Docker

- Adoption of Docker is very slow in HPC centers
- Thus the typical situation is that Docker is not installed and one cannot run containers without some support from system software.
- In general, Docker adoption will be slow in any computing farm or interactive linux system shared by many users.
  - It will take time for sysadmins to overcome the concerns of their security teams.
  - It is yet another service to maintain…
  - …. you name it.

UC#5: INDIGO udocker

- **A tool to execute content of docker containers in user space** when docker is not available
  - enables download of docker containers from dockerhub
  - enables execution of docker containers by non-privileged users
- **It can be used to execute the content of docker containers in Linux batch systems and interactive clusters managed by others**
- **A wrapper around other tools to mimic docker capabilities**
  - current version uses **proot** to provide a chroot like environment without privileges (it runs on CentOS 6, CentOS 7, Fedora, Ubuntu)
- **More info and downloads at:**
  - https://www.gitbook.com/book/indigo-dc/udocker/details
  - https://indigo-dc.gitbooks.io/udocker/content/doc/user_manual.html

UC#5: INDIGO udocker

- Examples:

```
# download, but could also import or load a container exported/save by docker
$ udocker.py  pull   ubuntu:latest
$ udocker.py  create  --name=myubuntu   ubuntu:latest

# make the host homedir visible inside the container and execute something
$ udocker.py  run   -v $HOME    myubuntu   /bin/bash   <<EOF
cat   /etc/lsb-release
ls  -l  $HOME
EOF
```

udocker is NOT an alternative to docker: we need the container image built by docker.

It is a tool to handle and run containers with regular user privileges and/or when docker is not
available for some reason: **it is very convenient to access clusters and Grid resources**

UC#5: INDIGO udocker

- Everything is stored in the user home dir or some other location
- Container layers are download to the user home
- Directory trees can be created/extracted from these container layers
- proot uses the debugger ptrace mechanism to change pathnames and execute transparently inside a directory tree
- No impact on read/write or execution, only impact on system calls using pathnames (ex. open, chdir, etc)

- Does not require installation of software in the host system:
  - udocker is a python script
  - proot is statically compiled