



INDIGO - DataCloud

INDIGO-DataCloud

STATUS REPORT AND UPDATED PLAN OF WP3 ACTIVITIES

EU DELIVERABLE: D3.2

Document identifier:	INDIGO-WP3-D3.2-v1.0.docx
Date:	21/12/2016
Activity:	SA1
Lead Partner:	INFN
Document Status:	FINAL
Dissemination Level:	PUBLIC
Document Link:	

Abstract

This document reports on the software QA, software maintenance and support, pilot infrastructure services and exploitation activities and achievements during the first year of the project. Delivers the status and updated plans for these activities, software quality control results and feedback from the 1st public software release.□



INDIGO - DataCloud

I. COPYRIGHT NOTICE

Copyright © Members of the INDIGO-DataCloud Collaboration, 2015-2018.

II. DELIVERY SLIP

	Name	Partner/Activity	Date
From	Cristina Aiftimiei	INFN (WP3)	December 14 th 2016
Reviewed by	Moderator: Isabel Campos Reviewers: I.Campos, M.Plocienniek	CSIC (WP1) PSNC (WP6)	December 19 th 2016
Approved by	PMB		December 22 th 2016

III. DOCUMENT LOG

Issue	Date	STATUS	Author/Partner
1	July 20 th 2016	TOC	Cristina Aiftimiei/INFN
2	August 10 th 2016	Adjustments to the TOC and layout Draft - v0.1	Cristina Aiftimiei/INFN Jorge Gomes/LIP
3	September 23 th 2016	Exploitation	Matthew Viljoen/EGI
4	November 16 th 2016	Release management	Cristina Aiftimiei/INFN
5	November 25 th 2016	Pilot services, SQA and metrics	Mario David/LIP Pablo Orviz/CSIC Michal Orzechowski/AGH
6	November 28 th 2016	Pilot services and related metrics	Mario David/LIP
7	December 7 th 2016	Executive summary, introduction, exploitation and pilot services	Cristina Aiftimiei/INFN Mario David/LIP Matthew Viljoen/EGI Baptiste Grenier/EGI
8	December 11 th 2016	Executive summary, introduction, conclusions, risks, SQA, metrics, pilot services, release management and exploitation.	Cristina Aiftimiei/INFN Mario David/LIP Matthew Viljoen/EGI Pablo Orviz/CSIC Michal Orzechowski/AGH Jorge Gomes/LIP Baptiste Grenier/EGI
9	December 12 th 2016	Draft - v0.2	Cristina Aiftimiei/INFN Mario David/LIP Matthew Viljoen/EGI



INDIGO - DataCloud



			Pablo Orviz/CSIC Michal Orzechowski/AGH Jorge Gomes/LIP
10	December 21 th 2016	Final - v1.0, addressing comments from Isabel Campos and Marcin Plociennik	Cristina Aiftimiei/INFN Mario David/LIP Matthew Viljoen/EGI Pablo Orviz/CSIC Michal Orzechowski/AGH Jorge Gomes/LIP Ian Neilson/STFC



INDIGO - DataCloud



TABLE OF CONTENTS

[1. INTRODUCTION](#)

[1.1 DOCUMENT PURPOSE](#)

[1.2 DOCUMENT ORGANIZATION](#)

[1.3 Notational conventions](#)

[1.4 Acronyms](#)

[2. EXECUTIVE SUMMARY](#)

[3. SOFTWARE QUALITY ASSURANCE](#)

[3.1 Introduction](#)

[3.2 Quality Criteria](#)

[3.2.1 Code style](#)

[3.2.2 Unit testing](#)

[3.2.3 Functional and integration testing](#)

[3.2.4 Code review](#)

[3.2.5 Documentation](#)

[3.2.6 Automated deployment](#)

[3.3 Services for continuous integration and SQA](#)

[3.4 Metrics](#)

[3.5 SQA and security](#)

[4. SOFTWARE RELEASE AND MAINTENANCE](#)

[4.1 Introduction](#)

[4.2 Software Maintenance and Support](#)

[4.2.1 – Maintenance plan & process, support organization](#)

[4.2.2 – Maintenance & Support – Implementation Status](#)

[4.2.2.1 INDIGO-DataCloud software](#)

[4.2.2.1.1 Source Code](#)

[4.2.2.1.2 Technical & User Documentation](#)

[4.2.2.2 Maintenance process - status and operation](#)

[4.2.2.3 Support Activities](#)

[4.3 Release Management](#)

[4.3.1 Release Management process & plan](#)

[4.3.2 Status of Implementation](#)



INDIGO - DataCloud



[4.3.3 INDIGO-1 \(MidnightBlue\)](#)

[4.3.3.1 Contributions to Upstream Distribution](#)

[4.4 Release & Maintenance and security vulnerabilities](#)

[5. PILOT SERVICES](#)

[5.1 Introduction](#)

[5.2 Services for Continuous Integration and Software Release](#)

[5.2.1 Project management service](#)

[5.2.2 Artefacts Repository](#)

[5.2.3 Artefacts Repository - Docker containers](#)

[5.2.4 Continuous Integration service](#)

[5.2.5 Version Control System](#)

[5.2.6 Issue/Bug tracker services](#)

[5.2.7 Code metrics services](#)

[5.2.8 User support](#)

[5.2.9 Documentation](#)

[5.2.10 Automatic deployment and configuration](#)

[5.2.11 Code review service](#)

[5.3 Development Infrastructure](#)

[5.4 Integration Infrastructure](#)

[5.5 Pilot Preview Infrastructure](#)

[6. EXPLOITATION](#)

[6.1 Planned activities and outcomes](#)

[6.2 High level products](#)

[6.2.1 High-level view of the software](#)

[6.2.2 Technical view of the software](#)

[6.2.3 Deployers' View of the Software](#)

[6.3 Service Providers' Board](#)

[6.3.1 SPB Terms of Reference](#)

[6.3.2 SPB Wiki](#)

[6.3.3 SPB Mailing List](#)

[6.3.4 Membership and Future Direction of the SPB](#)

[6.4 Staged Rollout](#)

[6.4.1 Staged Rollout Process Definition](#)

[6.4.2 Invitation for Early Adopters](#)



INDIGO - DataCloud



[6.5 Integration into the EGI Cloud Middleware Distribution \(CMD\)](#)

[7. KEY PERFORMANCE INDICATORS](#)

[7.1 Metrics](#)

[7.2 SQA Process KPIs](#)

[7.2.1 SQA Adherence](#)

[7.2.2 Automation testing index](#)

[7.3 SQA Schedule KPIs](#)

[7.4 SQA Quality KPIs](#)

[8. RISK ASSESSMENT](#)

[9. PLANS FOR THE SECOND PERIOD](#)

[9.1 Software quality assurance](#)

[9.2 Software release and maintenance](#)

[9.3 Pilot services](#)

[9.4 Exploitation](#)

[10. CONCLUSIONS](#)

[11. REFERENCES](#)

[A1. Appendix 1](#)

[A2. Appendix 2](#)

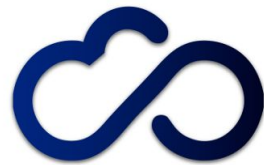
[A2.1 Integration Infrastructure](#)

[A2.2 Pilot Preview Infrastructure](#)

[A3. Appendix 3](#)

[A3.1 Metrics values](#)

[A3.2 KPIs values - SQA process](#)



INDIGO - DataCloud

1. INTRODUCTION

INDIGO-DataCloud (INtegrating Distributed data Infrastructures for Global ExpLOitation) is developing and delivering software to simplify the execution of applications on Cloud and Grid based infrastructures, as well as on HTC and HPC clusters. The project is extending existing PaaS (Platform as a Service) solutions, allowing public and private e-infrastructures, including those provided by EGI, EUDAT, PRACE and Helix Nebula, to integrate their existing services.

In order to achieve this objective the WP3 role is to support service activities, that in INDIGO-DataCloud project are mainly oriented to guarantee the high quality of the released software, and a successful deployment of production-like pilot services. These pilot services enable the validation of the developed software with real use-cases provided by the user communities in WP2 (definition of Support to Research Communities). The pilot services are targeting a direct integration of project outcomes into state-of-the-art infrastructures like the federated cloud services and large computing federated and storage systems, offered by EGI.eu and by national partners, or HPC resources, also including commercial systems. This is guaranteed through the participation of relevant partners operating those resources (INFN, KIT, DESY, CSIC, LIP, ATOS, T-System). WP3 is also addressing the exploitation of the released software.

WP3 is providing the software lifecycle services and related support to project developers, user communities, and infrastructure providers. In particular WP3 is providing a continuous software improvement process (Figure 1.1) that includes software quality assurance, software release management, maintenance, support services, and the pilot infrastructures needed for software integration and testing.

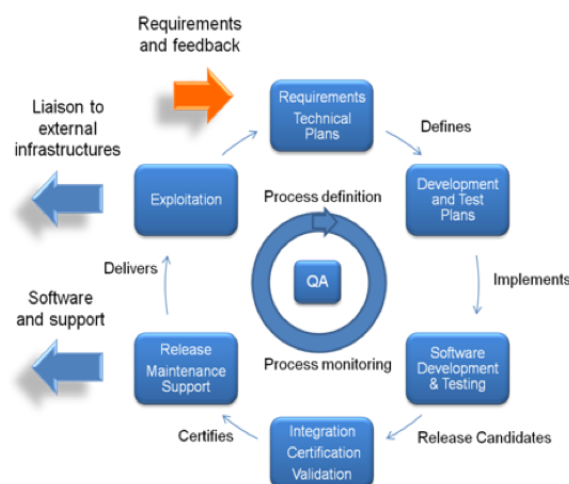


Figure 1.1: Software improvement process

The software requirements and lifecycle processes are designed to ensure compatibility with a wide range of infrastructures and user communities, through a requirements gathering process conducted in collaboration with partner infrastructures, individual resource providers, and complemented by



INDIGO - DataCloud

feedback collected from selected applications through which the final software assessment is performed.

Finally, WP3 provides bridges with external entities by developing and implementing an exploitation strategy directed at scientific and commercial infrastructure providers, and by operating the INDIGO-DataCloud helpdesk and first line of support.

During the first project period WP3 started by establishing its initial plan, which was defined in deliverable D3.1 [R1]. The processes and services in the initial plan were successfully deployed to support the development, testing, integration and building of the first INDIGO-DataCloud software release codenamed MidnightBlue [R2].

1.1 DOCUMENT PURPOSE

This document is a formal EC deliverable which records the status of the action towards the implementation of processes and services for Software Quality Assurance, Software Release and Maintenance and Pilot services for all software developed in INDIGO-DataCloud, as well as the Exploitation plan to foster the adoption of this software by e-Infrastructures and user communities.

1.2 DOCUMENT ORGANIZATION

The document is organized in four main sections, one for each of the operative tasks of the WP3 Work Package: Software Maintenance and Release Management, Pilot Services and Exploitation. Each section is further divided in subsequent parts providing summaries of the main prescriptions foreseen in the initial plan, deliverable D3.1 [R1]; status of the implementation; quality assurance checks; issues and suggestions for improvement. Metrics and KPIs are also provided in a separated section.

1.3 Notational conventions

The information within this document is mandatory for WP3 and for software developers in WP4, WP5 and WP6 (unless otherwise specified). The keywords “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 [R3].



INDIGO - DataCloud

1.4 Acronyms

AAI	Authentication and Authorization Infrastructure
API	Application Programming Interface
CD	Continuous Delivery
CDMI	Cloud Data Management Interface
CI	Continuous Integration
CLI	Command Line Interface
CLUES	CLUster Energy Saving
CMD	Cloud Middleware Distribution
CMDB	Configuration Management Database
CMF	Cloud Management Framework
DB	Database
DC	DataCloud
DLC	Data Life-cycle
EA	Early Adopter
EGI	European Grid Infrastructure
EMT	Engineering Management Team
EOL	End Of Life
FG	FutureGateway
GA	General Availability
GGUS	Global Grid User Support
GUI	Graphical User Interface
HPC	High Performance Computing
HTC	High Throughput Computing
IaaS	Infrastructure as a Service
IAM	Identity and Access Management Service
IDH	Identity Harmonization
IM	Infrastructure Manager
JRA	Joint Research Activity
KPI	Key Performance Indicator
OCCI	Open Cloud Computing Interface
OGF	Open Grid Forum
OMT	Open Mobile Toolkit
OOI	OCCI implementation for OpenStack
OPIE	OpenStack Preemptible Instances Extension
OS	Operating System
OSC	OpenStack Client



INDIGO - DataCloud

PaaS	Platform as a Service
PMB	Project Management Board
PR	Pull Request
PT	Product Team
QA	Quality Assurance
RC	Release Candidate
RfC	Request for Change
RFC	Request For Comment
RM	Release Manager
SaaS	Software as a Service
SLAM	SLA Manager
SLOC	Source Lines of Code
SME	Small and Medium-sized Enterprise
SP	Service Provider
SPB	Service Providers Board
SQA	Software Quality Assurance
SQAP	Software Quality Assurance Plan
SR	Staged Rollout
SVG	Software Vulnerability Group
TB	Technical Board
ToR	Terms of Reference
TTS	Token Translation Service
UMD	Unified Middleware Distribution
URL	Uniform Resource Locator
VCS	Version Control System
WP	Work Package
VCS	Version Control System
VM	Virtual Machine



INDIGO - DataCloud



2. EXECUTIVE SUMMARY

INDIGO-DataCloud (INtegrating Distributed data Infrastructures for Global ExpLOitation) is a project approved in January 2015 within the E-INFRA-1-2014 call of the Horizon 2020 framework program of the European Community. It aims at developing a data/computing platform targeting scientific communities, deployable on multiple hardware and provisioned over hybrid (private or public) e-infrastructures.

This document provides a report on the software quality assurance, software maintenance and support, pilot infrastructure services and exploitation activities carried within the Software management and Pilot Services Work Package. Delivers the status and updated plans for these activities together with quality control results and feedback from the 1st public software release.

The Software Quality Assurance (section 3), has defined in “D3.1 - Initial Plan” [R1] the set of quality criteria for the INDIGO-DataCloud software together with the processes and means of verifying the software compliance. This deliverable (D3.2) provides a report of the software quality assurance implementation (section 3.3), quality criteria (section 3.2), together with the implemented metrics and KPIs (section 3.4).

The development and release processes are backed by a state of the art software quality assurance (SQA), composed of public and private services, with a continuous integration system that exploits containers technology. This level of sophistication and flexibility is necessary to cope with the project’s inherent diversity that includes new products, contributions to existing products and contributions to external projects. These developments include contributions to cloud middleware frameworks (OpenStack, OpenNebula, EGI Cloud Middleware Distribution) and address different environments and applications (data intensive, HTC, HPC, cloud, grid).

The SQA areas of intervention include code style, unit testing, functional testing, documentation, code reviews. A new automated deployment area has been introduced to facilitate the software deployment. Ansible Galaxy service is being used to provide ansible roles to automatically install and configure INDIGO-DataCloud components and applications. About 37 ansible roles have already been produced. Reports regarding the software quality assurance compliance are produced and summarized weekly. The Appendix A3 provides an overview of the SQA metrics and KPIs.

With respect to the Software release and maintenance activities (section 4), this deliverable presents the work done in the first reporting period of the project in terms of:

- Defining the processes to guide the Change and Problem Management, and Software Maintenance, and the continuous execution of the related procedures.



INDIGO - DataCloud

- Coordinating the release of the INDIGO-1 (MidnightBlue) [R2] providing features as requested by project users,
- Defining updates schedules and coordinating the release of INDIGO-1 (MidnightBlue) updates providing bug fixes, improvements and new features
- Supporting the users, either system administrators of Pilot services testbeds or members of scientific communities part of the project.

All the activities followed plans described in the “D3.1 - Initial Plan” [R1]. The many differences in experience and background of the people participating in the project made the preparation of the plans a complex task and it took considerably more time than originally planned in the Technical Annex [R4]. However, the contents of the resulting plans were sufficiently clear on how to maintain, support and release INDIGO-DataCloud products. These were complemented with details regarding the Software Quality Assurance criteria, and by a number of documents covering aspects like Release Management, Testing (unit and functional), and Documentation.

The Release Management task (section 4.5) focused almost exclusively on the preparation of the first INDIGO-DataCloud major release, planned for the end of July 2016. The release, codenamed MidnightBlue, was officially announced on 8th of August 2016 [R2], completed with repositories of digitally signed packages (binary and source) and Docker [R5] containers, documentation and release notes. The first INDIGO-DataCloud major release contains 167 digitally signed binary packages and 40 Docker containers that allow the installation of 32 products. In addition 13 Docker containers for applications have been produced. The integration and release effort required strong coordination performed via the Engineering Management Team (EMT) coordinated by WP3.2. More than 500 tickets (OpenProject tasks) have been used to track all activities coordinated by WP3.

The Software Maintenance task (section 4.2) is based on the clear mandate to keep the stability of the software deployed in production, introducing only those changes that are necessary to keep the software running in a way that is adequate for its users. Since no proper INDIGO-DataCloud release existed until beginning of August, the maintenance task concentrated on understanding the different components and services, and on how to integrate their development workflows with the INDIGO-DataCloud. Change management started to be applied after the first release, with monthly scheduled updates. At the time of writing five updates were already released.

The User Support activities (section 4.4) were organized taking advantage of the well-known support chain between infrastructure providers, like EGI, and software providers, based on the use of the GGUS tool [R6]. An INDIGO-DataCloud Generic Support Unit [R7] was defined, and a support mailing-list was also created, in order to ease the communication, especially for new communities. Support is provided to users both internal to the project and external.

The pilot services (section 5) were organized around three types of services: dedicated to developers (section 5.2), software integration (section 5.3) and software preview (section 5.4). The activities were initially focused in implementing the services needed to support the software development and release



INDIGO - DataCloud



management and included among others the source code repository, and continuous integration system. The INDIGO-DataCloud source code repository and Version Control System on GitHub has 161 repositories, 28 teams and 83 persons registered. A total of 298 requests have been opening on the GitHub tracker of which 219 have been closed. Details of these services are given in section 5.1. The integration infrastructure played a crucial role in the integration tests and preparation of the release (section 5.3). After the first release the focus shifted to the deployment of the preview infrastructure (section 5.4) where the software is validated using real use cases provided by WP2.

The exploitation activities (section 6), were focused in bridging with infrastructure providers which are the targets for the INDIGO-DataCloud software together with the user communities (WP2). The exploitation task created a Service Providers Board (section 6.3), and established communication channels with providers with a particular relevance for EGI, with which an agreement was established for integration of INDIGO-DataCloud products in the EGI Cloud Middleware Distribution (section 6.5). The exploitation task also established the Staged Rollout process to gather feedback from software deployment (section 6.4).

The methodology and processes implemented by WP3 have been designed to enable and foster contributions to upstream projects and distributions. This approach has already produced significant results with 11 products contributed and merged upstream, while others are in progress.



3. SOFTWARE QUALITY ASSURANCE

3.1 Introduction

The INDIGO-DataCloud Software Quality Assurance (SQA) defines and monitors the set of quality standards and procedures that the software components developed under the INDIGO-DataCloud project must adhere throughout each phase of the software life cycle. The objectives of the SQA activity in INDIGO-DataCloud are:

- strengthen the software quality during the development phase aiming at releasing high quality software;
- enable smoother maintenance and increase long term sustainability;
- facilitate the release process by ensuring that the software is always ready for being released;
- enable future exploitation by adhering and enforcing the target platforms SQA requirements.

Instead of providing a unique and monolithic solution, the INDIGO-DataCloud software is a coherent set of components that enables integration and interoperability between existing frameworks. Therefore the nature of the software being developed is quite diversified including:

- code contributions to external projects;
- forking of existing products;
- enhancements to existing products;
- new products developed from scratch.

The scope of the software being developed is also quite diversified including:

- contributions to several middleware stacks (OpenStack, OpenNebula, EGI CMD);
- support for several types of environments (data intensive, HTC, HPC, cloud, grid).

Due to this heterogeneity in the project's definition, the INDIGO-DataCloud SQA is based on a well-defined spectrum of design principles and testing methodologies, flexible enough to cope with the project's inherent diversity. The SQA baseline of each INDIGO-DataCloud product was adjusted to its targeted platform (e.g. OpenStack, OpenNebula, dCache, etc) thus facilitating contribution to external projects and therefore the long term sustainability of the INDIGO-DataCloud software. For the new products being developed from scratch, that are not related to any external project, a set of commonly used baseline requirements was established.

3.2 Quality Criteria

As defined in the WP3 initial plan, the INDIGO-DataCloud SQA has five main quality criteria areas: code style, unit testing, functional testing, documentation and code reviews. These five areas have been complemented with the addition of a sixth area: automated deployment. Table 3.1 provides a summary of the six quality criteria areas.



INDIGO - DataCloud

Table 3.1: SQA areas of intervention

SQA area	Characteristics
Code Style	Performed on Continuous Integration (CI), tests on these areas run for each newly submitted source code change
Unit testing	
Functional testing	
Documentation	Made publicly available in project's GitBook organization
Code reviews	Performed via GitHub's Pull Requests, visible to external collaborators
Automated deployment	Post D3.1 requirement, Ansible roles and Puppet modules.

3.2.1 Code style

Code style enforcement promotes good practices that seek the maintainability of the code, emphasizing its readability. The code style is enforced to be consistent with any of the common/well-known coding conventions. This is essential for 3rd-party upstream contributions and leads to a high diversity of code styles in INDIGO-DataCloud than can be observed in the Figure 3.1. The code style verification is performed by the continuous integration system (section 5.1.4) and is triggered by changes in the products' source code repositories in GitHub (section 5.1.5).

Code style standards used by INDIGO products

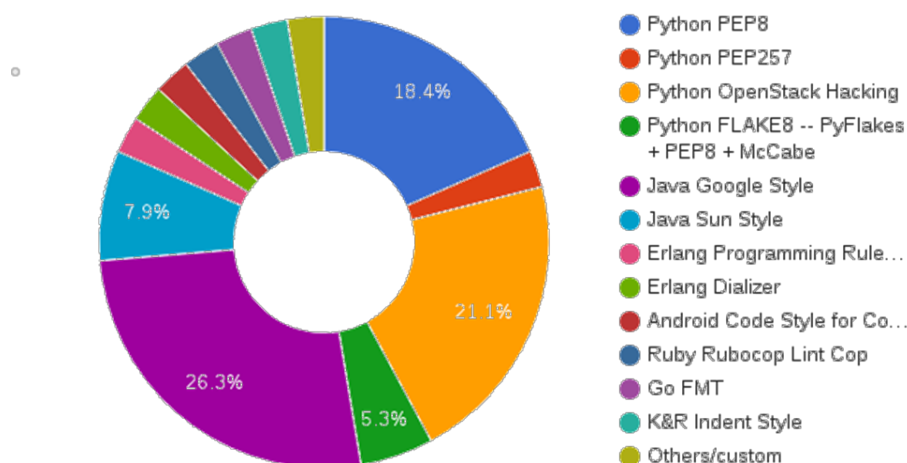




Figure 3.1: Code styles

3.2.2 Unit testing

Unit testing aims to early detect failures in functions and areas of the code or incoming changes that could potentially break any working functionality. Unit testing is a “white-box” testing since it emphasizes in the evaluation of all the possible flows in the internal design of the code. The recommended target coverage for unit testing that INDIGO-DataCloud aims to achieve in the second release is 70%. This value will increase over time. For the first release the coverage is depicted in the Figure 3.2. Several products are already inline with the target coverage. The unit tests are executed by the continuous integration system and are triggered by changes in the product's' source code repositories in GitHub.

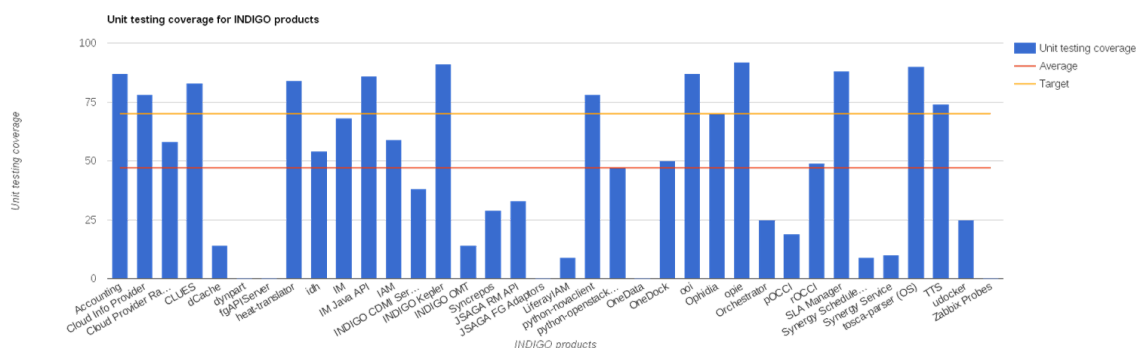


Figure 3.2: Unit testing coverage per product

3.2.3 Functional and integration testing

The functional testing consists in the verification of the software components functionalities, based on requested requirements and agreed design specifications. Unlike unit testing, functional testing is classified as “black-box” testing since it evaluates the functionalities that the product exposes. Currently about 30% of the products have the functional testing automatized in the continuous integration system. For the products whose functional testing is performed externally evidence of the test reports is collected and archived for reference.

Integration testing focuses on interactions with external components, making sure that these components work together. It is being performed in the integration infrastructure outside of the continuous integration system. An extensive list of tests has been identified (see also section 5.2). Integration testing is currently performed as the last step before the product goes into the release process.



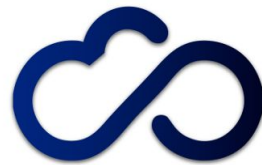
INDIGO - DataCloud

3.2.4 Code review

The code review is the last step prior to the source code integration into the main component's codebase. The code review is an informal non-automated (peer) human-based review of the candidate code, and takes place after the software passes all the continuous integration SQA verifications: code style, unit tests and functional tests. The Figure 3.3 shows an example of the tests that must succeed in the continuous integration for a product code review to take place.

The code review has been implemented taking advantage of the GitHub Pull Requests capability [R8] as explained in section 3.3, and is open to external feedback fostering collaboration. Once it is accomplished the code can be merged.

In the second project period the possibility of using GerritHub [R9] to complement the GitHub based reviews is envisaged.



INDIGO - DataCloud

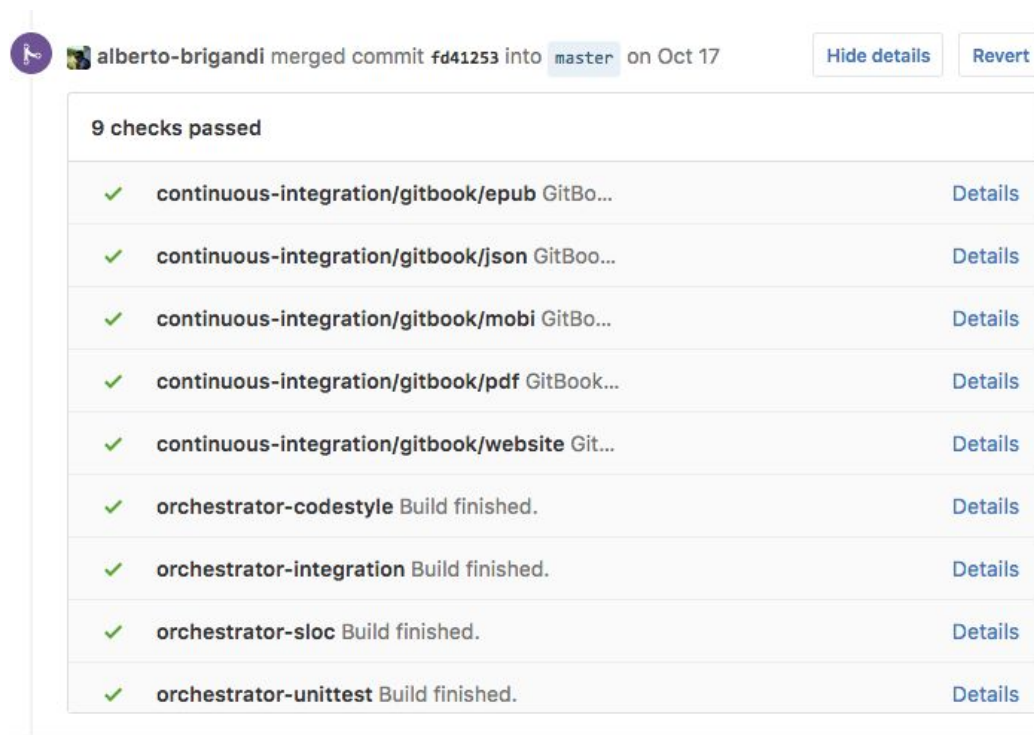


Figure 3.3: Example of continuous integration tests after which code review can take place

3.2.5 Documentation

Documentation is provided for all products being released under the INDIGO-DataCloud project scope. The following documentation policies have been implemented:

- Documentation is treated like code, and is included in the product's codebase repository. This allows documentation evolution to be tracked and easily matched against the source code.
- All INDIGO-DataCloud products documentation is centrally available (online) in the project's GitBook organization. This is accomplished through tight integration between GitHub and GitBook. Once new documentation is uploaded to GitHub it is made automatically visible in GitBook.
- The documentation completeness is evaluated in a revision for each INDIGO-DataCloud major release. A revision took place for the software documentation provided with the first release.
- WP3 works with the developers to improve the documentation quality and completeness.

The deliverable D3.1 refers to three different types of documentation:

- developer documentation;
- deployment and administration documentation;
- command line interface (CLI) and application programming interface (API) documentation.



INDIGO - DataCloud

In addition to these types of documentation, the products must also provide user guides describing the interaction with a given piece of software from the user perspective. The need of a user guide may not apply to all products. Only those products where interaction with end-users is expected are required to provide a user guide. This requirement was added after the initial plan definition in D3.1.

3.2.6 Automated deployment

The addition of quality guidelines for tackling product's automated deployment occur after the initial plan definition in D3.1. It imposes a mandatory requirement on the usage of open-source software configuration management tools to ease the deployment of the products. The product teams (PTs) must provide such configuration capabilities via adequate configuration modules. Examples of recommended software configuration management tools are: Puppet, Ansible and Chef. With Ansible being the preferred solution.

The product documentation must also refer the software configuration modules, indicating the location from which the modules can be obtained (e.g. Puppet Forge, Ansible Galaxy, etc), together with a brief description of the module, its usage, examples, and a reference of the available module parameters.

3.3 Services for continuous integration and SQA

The INDIGO-DataCloud SQA process is depicted in Figure 3.4. The INDIGO-DataCloud SQA process is depicted in Figure 5. the process, in its different steps, reflects some of the main and important achievements of the WP3.2 during the first reporting period, like:

- The pull request of a feature branch in GitHub triggers the execution of SQA jobs in the INDIGO-DataCloud Jenkins continuous integration system.
- The SQA jobs perform the code style verification and calculate unit and functional test coverage.
 - The tools necessary for tackling these tests are packaged in Docker images (available in DockerHub).
 - Each test then initiates a new container that provides a clean environment for its execution.
 - This is an **innovative approach** that provides the flexibility needed to cope with the INDIGO-DataCloud software diversity.
- The results of the several SQA jobs are made available in the Jenkins service which notifies back to GitHub their exit status.
 - Only if the tests have succeeded, the source code is validated and is ready to be merged into the production branch.
- The last step in the workflow is the code review, where a human review of the change is performed. After code review the source code can be merged and becomes ready for integration and later release.



INDIGO - DataCloud

INDIGO-DataCloud provides a full set of development services for continuous integration and software quality assurance (version control system, code review, issue tracker, containers repository, continuous integration, artefacts repository, code metrics services, documentation repository). As a general rule, these services must be used by the all software products. However there are exceptions to this rule that fall into two main categories:

- mature products that have well established development services already in place;
- new software products that aim to contribute to existing projects/frameworks (such as OpenStack, OpenNebula and others) and that make use of dedicated development services made available by the corresponding projects.

For these cases where products are using external development services, the product teams must account that:

- continuous integration testing must satisfy the INDIGO-DataCloud SQA requirements;
- the product teams must provide metrics to WP3 whenever requested.

Evidence of successful execution of the relevant tests (unit tests, functional tests), must be provided to WP3 for metrics collection and SQA management purposes. The INDIGO-DataCloud SQA team keeps track of the exceptions and continuous integration systems being used by the several products. This is a dynamic process. As products evolve and become accepted in their target projects they may move from the INDIGO-DataCloud development system to external development systems. This is indeed one of the objectives of the SQA in INDIGO-DataCloud, improve the quality of the software, so that it can be more easily accepted in target projects ensuring a path for sustainability and exploitation.

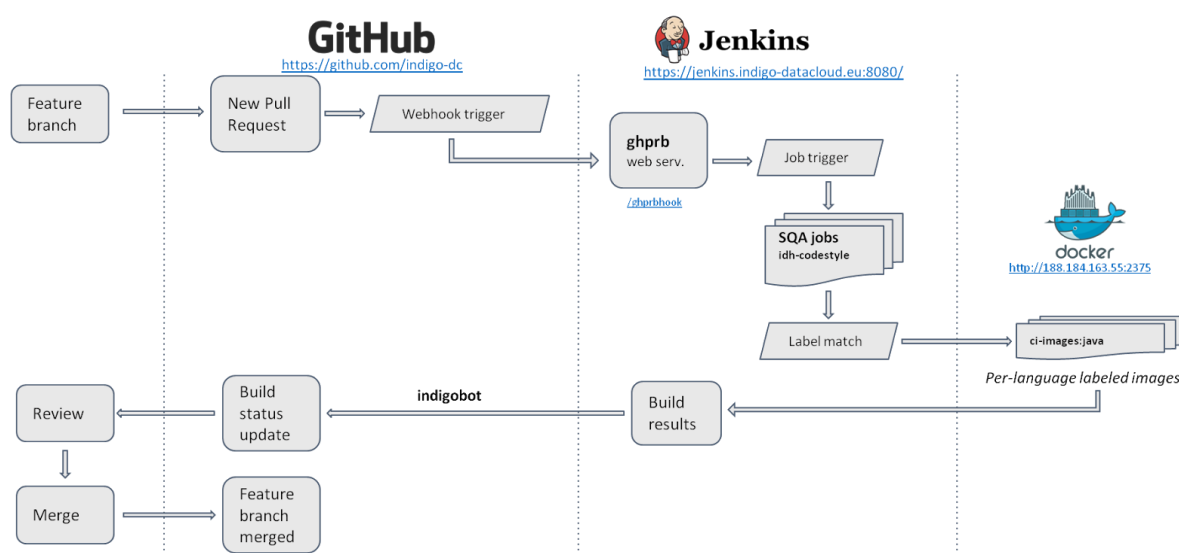




Figure 3.4: The SQA automated process

3.4 Metrics

The INDIGO-DataCloud SQA provides the means to measure and monitor the software development and the release life cycle, to positively track and improve the processes and products being developed within the project.

For each major software release the SQA main areas of quality criteria described in section 3.3 are monitored using information gathered by several sources such as GitHub, Jenkins continuous integration and OpenProject (see section 5.1.1). OpenProject tickets are used to track the evolution of the products. Weekly reports are produced for each product detailing the product SQA conformance. The Figure 3.5 shows several extracts of a specific product report, where the product general progress and status is analysed. A weekly summary is also created providing a global view of all products evolution, using a metric based on the mean value of the different areas of quality criteria expressed in percentage. The Figure 3.6 shows an extract of the summary table for some of the products in the INDIGO-DataCloud first release. The complete table is available in Appendix A3.2.

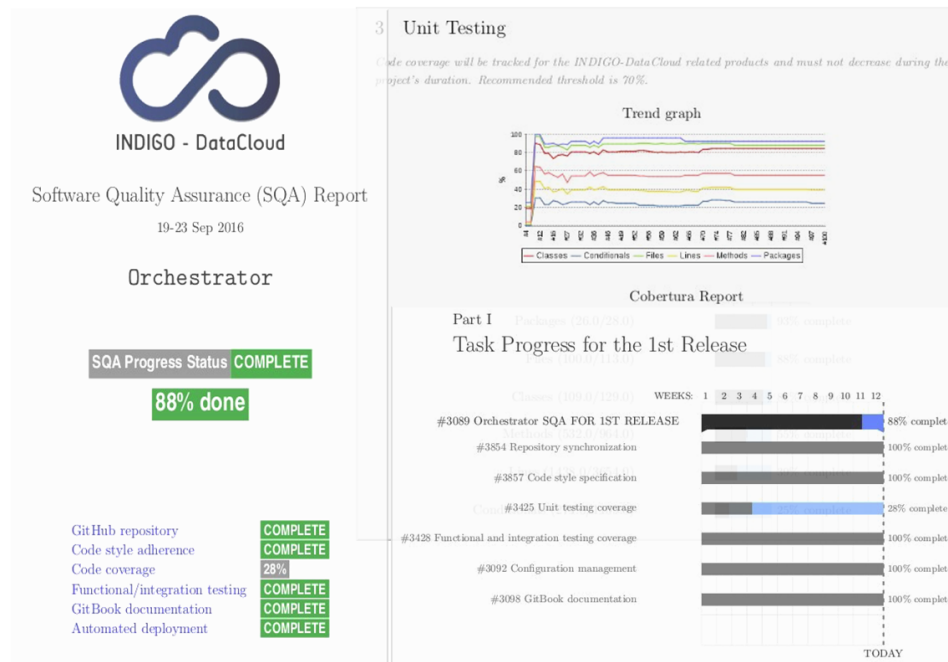


Figure 3.5: Extracts of a product SQA report

The percentage range is based on the maximum possible value of SQA fulfillment. This range goes significantly beyond the baseline requirements. For instance the units testing coverage for the second



INDIGO - DataCloud

release is expected to be about 70%, however, only with 100% coverage together with 100% in all other areas may a product achieve a score of 100% in the summary. Therefore values below 100% can still satisfy the baseline requirements set for a release. An asterisk near the percentage indicates whether the product satisfies the baseline criteria. The summary main purpose is to provide a simple method for verifying the product's evolution. The compliance details for each product are available in the individual reports.

The products full reports and the global summary are made available to the developers, WP leaders and Project Management Board (PMB). This information is used to assess the progress of the products and is discussed at the EMT meetings and PMB. The summary table is made available in the INDIGO-DataCloud OpenProject service, with links to the full reports that are made available in the INDIGO-DataCloud owncloud service

Infrastructure Manager (IM)	72%	87%	95% (**)	95% (**)	95% (**)	95% (**)	95% (**)	95% (**)	87% (**)	87% (**)	95% (**)	95% (**)	95% (**)	95% (**)	95% (**)	95% (**)
Infrastructure Manager Java API (IM-Java-API)	97% (**)	98% (**)	98% (**)	98% (**)	98% (**)	98% (**)	98% (**)	98% (**)	90% (**)	90% (**)	90% (**)	90% (**)	90% (**)	90% (**)	90% (**)	90% (**)
Synergy Service	52%	68%	68%	68%	68%	83% (**)	83% (**)	83% (**)	75% (**)	83% (**)	83% (**)	83% (**)	83% (**)	83% (**)	85% (**)	85% (**)
Synergy Scheduler Manager	67%	77%	77%	77%	77%	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)	85% (**)
OpenStack OCCL Interface (ool)	48%	48%	82%	97% (**)	97% (**)	97% (**)	98% (**)	98% (**)	90% (**)	90% (**)	90% (**)	98% (**)	98% (**)	98% (**)	98% (**)	98% (**)
OpenStack Preemptible Instances Extension (opie)	48%	48%	65%	65%	65%	65%	65%	97% (**)	88% (**)	88% (**)	88% (**)	97% (**)	97% (**)	97% (**)	97% (**)	97% (**)
ONEDock	58%	73%	90%	90%	92%	92% (**)	92% (**)	92% (**)	83% (**)	83% (**)	83% (**)	92% (**)	92% (**)	92% (**)	92% (**)	92% (**)

Figure 3.6: Extract of the SQA global summary report

SQA metrics dashboards for each INDIGO-DataCloud product are available on GitHub exploiting MetricsGrimoire [R10] and vizGrimoire [R11] as specified in D3.1. These dashboards use information automatically collected from: Jenkins reports, GitHub history, and GitBook. The dashboards provide numeric data and charts that help developers and WP3 to assess the product status and related development activity. Figure 3.7 is a snapshot of a product metrics dashboard.



INDIGO - DataCloud

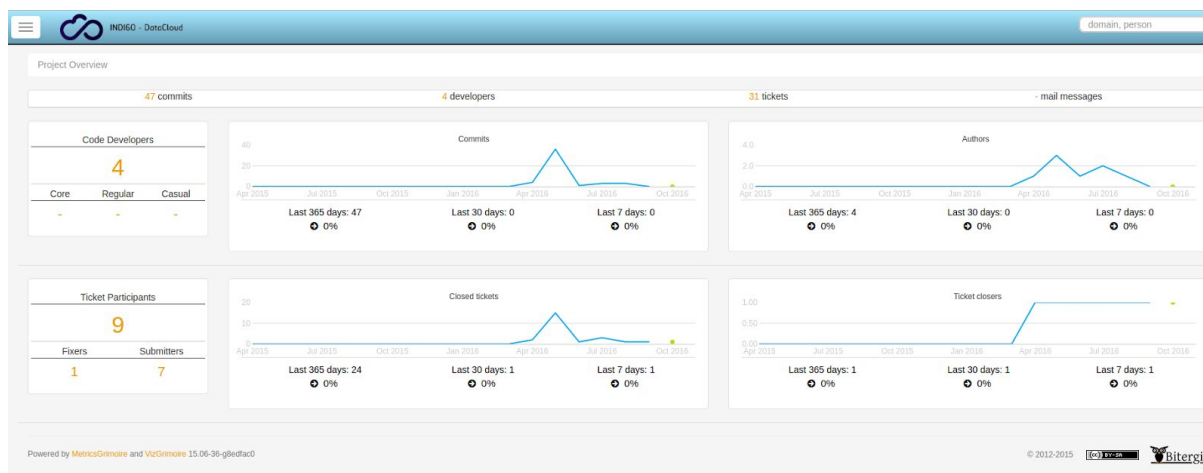


Figure 3.7: Metrics dashboard for the “udocker” component <http://indigo-dc.GitHub.io/udocker/>

Finally a full set of metrics and KPIs is provided in Appendix A3. The information provided in Appendix A3 is collected both automatically and manually from different sources and stored on Kibana for reference and analysis. These metrics and KPIs are produced in accordance with the adopted quality model defined in D3.1.

3.5 SQA and security

During the first project period, security has been mainly focused on the project architecture. WP3 has members with expertise in security that are participating in the project Technical Board (TB) that follow the development and release processes in order to provide advices. Code review recommendations also include an assessment of the inherent security risk of changes and a validation that the security model has not been downgraded by the changes.



4. SOFTWARE RELEASE AND MAINTENANCE

4.1 Introduction

The main objective of this task is to make the certified software components available as a set of coherent high quality releases, supported by an efficient maintenance process. Figure 4.1 shows the position of the release and maintenance activities in the continuous process of handover of software from the developers, Product Teams, passing through the quality checks, testing on the integration and testing infrastructures, acceptance testing from the user communities, until the deployment on external production infrastructures.

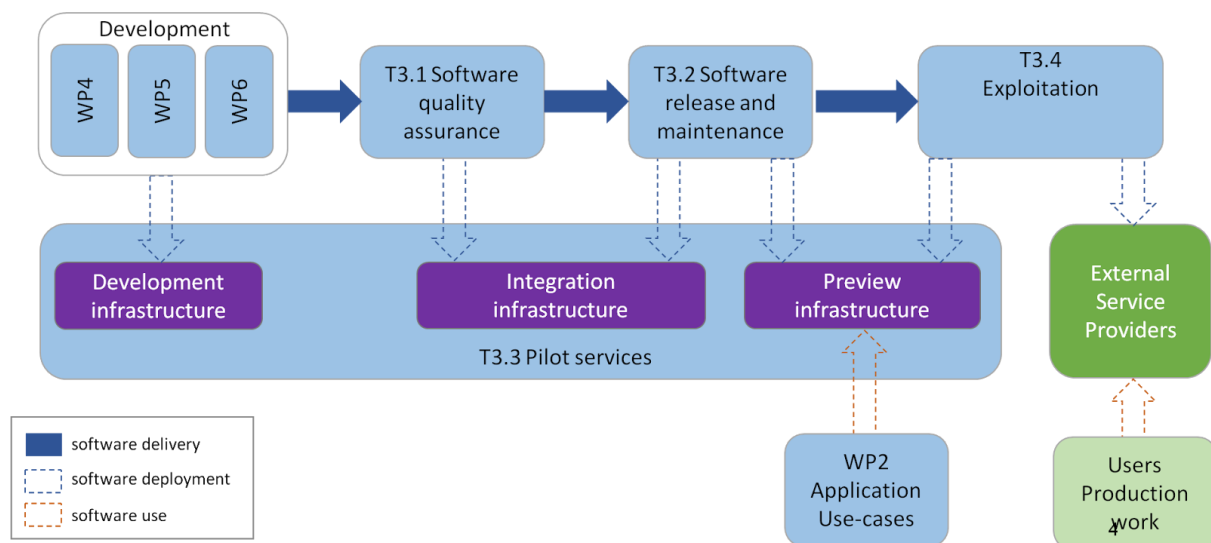


Figure 4.1: Software handover

4.2 Software Maintenance and Support

The Software Maintenance task deals with the analysis and correction of software defects found in released components by users and infrastructure technicians. This task includes the definition of the Software Maintenance, Problem Management and Change Management processes and the continuous execution of the related procedures.

The first reporting period was dedicated to the preparation of the foundations of the maintenance activities, as there was no INDIGO-DataCloud release available yet. The plan about how the goal of the task is achieved is described in D3.1 and is summarized in Section 4.2.1. The status of its



INDIGO - DataCloud

implementation is described in Section 4.2.2. They include also references to the support organization and status of its implementation.

4.2.1 – Maintenance plan & process, support organization

The Software Maintenance Plan & Process are described in Section 4.2 and 4.3 of D3.1 [R1]. The main items are:

- INDIGO-DataCloud software
 - The definition of different types of components developed through the project, service, client, library, virtual image
 - Source Code, Technical & User Documentation
- Maintenance process - following the the guidelines of the ISO/IEC 14764:2006 (Software Engineering — Software Life Cycle Processes — Maintenance) standard [R15]
 - Organization and roles – needed in order to handle maintenance implementation, change management and validation, software release, migration and retirement, support and helpdesk activities
 - Product Team & Product Team Leader
 - Development teams implementing the necessary processes and procedures, respecting established requirements and priorities as globally defined and endorsed by the INDIGO-DataCloud project executive and technical leadership
 - Technical Director, Release Manager, JRA WP Leaders
 - Decision Making Bodies: Engineering Management Team (EMT, Technical Board (TB) and Project Management Board (PMB)
 - Types of maintenance, corrective, preventive, adaptive & perfective that are foreseen to be addressed, and the tracking of the respective changes together with their prioritization according to factor like severity, impact, urgency and cost.
 - Maintenance & support schedule
 - Time-based releases with two major releases planned during the project lifetime, and different periods for full and standard maintenance, security updates and end-of-life periods

The organization of support activities is described in Section 4.4 of D3.1, and the main items are:

- The INDIGO-DataCloud support model – integrated with the overall support structure adopted in EGI, based on the use of the GGUS [R6] tool, where INDIGO-DataCloud teams cover the third-level (expert) support
- The Generic Support Unit - “**INDIGO-DataCloud Catch-All**” [R7]
- INDIGO-DataCloud support units and incident resolution workflow.

4.2.2 – Maintenance & Support – Implementation Status



INDIGO - DataCloud

The first and very important and fundamental activity from the maintenance and support point of view, was the collection of the information regarding the development teams, and their software components part of the INDIGO-DataCloud architectural design.

The information requested, described in WP3 wiki page "[INDIGO - DataCloud Product Teams, their Components and Services](#)" [R16] was:

- **Component Name** - name of the software component or service as specified in D4.1, D5.2, D6.1
- **Short Description** - very short, one-line, description, or full name in case acronyms are used for the Component Name
- **Leader(s)/contacts** - Product Team Leader and/or Deputy name(s) and contact e-mail(s)
- **Language(s)** - programming languages
- **VCS** - Version Control System, examples.: CVS, SVN, GIT, with link or reference to the code-repository, ex.: <https://GitHub.com/indigo-dc> , <https://baltig.infn.it/explore>
- **Issue Tracker(s)** - link to Issue Tracking Systems - if any, both development and support, if they are separate. Ex: GitHub issue tracker, GGUS
- **Build Tools** - build tools or systems, if any. Ex.: Make-based tools, Non-Make-based tools, Build script generation tools or other tools for the creation of images, containers
- **Build Units** - how many specfiles, pom.xml
- **No. of artefacts** - number of packages or images
- **C. I.** - Continuous Integration tools/systems, if any. Ex.: Jenkins, Hudson, etc.
- **C. M.** - Configuration Management tools, if any:
 - Tool: for ex: Ansible, Puppet, Chef, etc.
 - Link to the respective code repository.
- **Dependencies**
 - **Internal:**
 - Int. Deps - internal (to the project) dependencies, if any
 - **External:**
 - Deps Type: O.S. - depending on operating system (OS) - depending on cloud management framework (OpenStack, etc)
 - No of Ext. Deps: - number of external dependencies if any, ex.: "~35 build-time + 6 runtime", "~40"
 - Location of External Deps: example, YUM, APT or Maven repos
 - Ext. Vers.: - versions of the external dependencies, ex.: "OS defaults", or "selected versions"
- **C. S.:** Code style specification - Name and URL to the specification. The PT will have to provide a file or reference to the specification following the template table
- **C. R.** - Code Review indicate if (yes/no) and (if yes) what type of code review is already in use. ex.: GitHub Pull Requests, GerritHub
- **Documentation**



INDIGO - DataCloud

- Type of Docs - type of documentation if already existing and eventual links. Ex: developers documentation, deployment/admin documentation, API/CLI documentation, other (which one)
- **Test Infra.** - "friendly" testing infrastructure, if any, available to contribute to the testing of the specific component(s) or service(s) as part of the Pilot Testbed (T3.3), with link

The results are collected in the *indigo_pts.xls* [R17] file that is a live document, where new components and services are added, related information updated, and recorded in a stable table, once the information is confirmed, present in the wiki “*INDIGO - DataCloud Product Teams, their Components and Services*” [R16]. This information played a crucial role in the decisions concerning definition of the maintenance process and release management, as described in the following sections.

4.2.2.1 INDIGO-DataCloud software

Taking in consideration the WP4, 5 and 6 deliverables describing planned work a number of services and components were identified, together with their development teams. They are presented in Annex 1:

- Table A1.1 containing 21 WP4 developed components and services and documentation on how to integrate the use of projects solutions in already existing services
- Table A1.2 containing 14 WP5 developed components and services
- Table A1.3 containing 6 components and services developed by WP6

4.2.2.1.1 Source Code

Following the plan defined, (D3.2, Sections 4.2.1.2 and 5.1.2) all the INDIGO-DataCloud developed software was made available through a publicly available repository hosted on GitHub (reference), under the project organization, *indigo-dc*, with individual repositories per component either for new development as well as forks for already existing software prior to the project for which important contributions are foreseen during INDIGO-DataCloud lifetime. More details on the content of this repository are available in Section 5.5 below.

4.2.2.1.2 Technical & User Documentation

As mentioned in the SQA Section 3.2.5 above, a specific policy was defined in order to guarantee the availability of Technical & User Documentation, and their structure.

Detailed information on how to provide the required documents on the project space on GitBook (reference), was provided to the PTs on the WP3 wiki [R34]

Releases and Maintenance Updates have a dedicated book, collecting the release notes of all the products, as well as the contents of each update:

- <https://indigo-dc.GitBooks.io/indigo-datacloud-releases/content/>
- <https://indigo-dc.GitBooks.io/indigo-datacloud-releases/content/indigo1/updates.html>



INDIGO - DataCloud

4.2.2.2 Maintenance process - status and operation

After setting the ground in D3.1 [R1], Section 4.2.2 and 4.3 by defining the process, the roles, the bodies, the support model and structure, the maintenance schedule, the activities concentrated on the implementation and daily operations:

- Product Teams (PT) - were defined:
 - Not only at the level of project management - with around 50 groups, constantly updated to reflect new components that are added - to be used to track the release and updates tasks
 - But also in the indigo-dc GitHub organization - with around 28 teams - in order to allow more fine-grained permissions definition for individual components
- Request for Changes (RfCs) - were implemented:
 - As *tasks* for the Release and Updates, category “INDIGO - DC Releases” - to track releases and updates requirements
 - As *issues* on GitHub - mandatory for all the changes (bugs, features, improvements) to be introduced by the PTs in new versions of their components, using the priorities schema defined in D3.1, section 4.3.4 “Problem and Modification Analysis”
- Engineering Management Team (EMT):
 - Was established to coordinate the change management and release processes
 - <https://owncloud.indigo-datacloud.eu/index.php/s/8XsS2MHKQfE3s7i>
 - The EMT Terms of Reference [R35] - was prepared and approved by the PMB
 - Regular meetings were established where discussions took place and decisions were taken around the following items:
 - SQA status - issues found and planning their solution
 - Status of RfCs, issues found by users or new developments - planning new updates to address them
 - Releases preparations
 - Schedules
 - Mandatory Operating Systems and Cloud Management Frameworks to be supported
 - Pilot services status
 - Deployment of services in the integration and preview testbeds - issues encountered and
 - Status of request for resources from the PTs
 - Exploitation
 - Staged Rollout status for the components and services released
 - Service Providers Board news
 - During the reporting period around 40 meeting took place, and minutes were prepared so that a reference to the decision taken is always available
 - <https://agenda.indigo-datacloud.eu/categoryDisplay.py?categId=12>
 - The mailing list, indigo-emt@lists.indigo-datacloud.eu was setup, and archives are available at [mailing-list Archive](#)



INDIGO - DataCloud

- Maintenance Schedules - were defined
 - Following what was described in the D3.1, section 4.3.3 “Maintenance and Support Schedule”:
 - The major releases support timelines was defined (see Fig. 4.2)
 - Updates were planned for the INDIGO-1 (MidnightBlue) release
 - <https://owncloud.indigo-datacloud.eu/index.php/s/v4EwmGqDgmw4UrB>
 - Foreseen monthly updates, with PTs planning their components updates announcing and discussing them during the EMT meetings

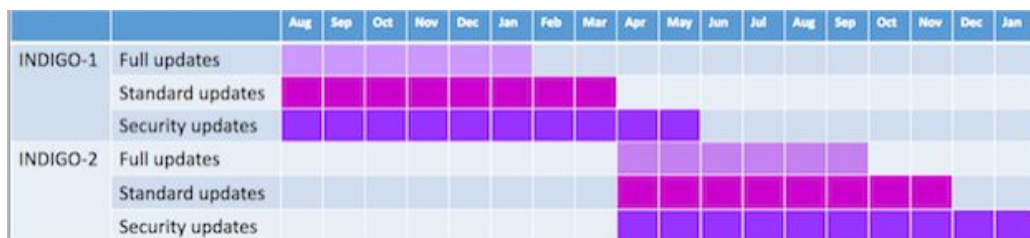


Figure 4.2: INDIGO-DataCloud maintenance Schedule

- Continuous Integration
 - Together with WP3.1 and WP3.3 the project Continuous Integration system was setup, and different type of jobs were defined in order to check in an automatic manner the SQA criterias, the packaging for the different mandatory Operating Systems, for automatic deployment from INDIGO-1 repositories, in order to check their integrity.
 - As some of the PTs had their own CI systems, and the integration with the projects one is a long and complex process - results of the individual CI systems, especially for the ones already integrated in upstream distributions, are accepted in order to certify the SQA criterias defined by the project. Details regarding those systems are present in the “INDIGO - DataCloud Product Teams, their Components and Services” wiki page [R16].
 - More details regarding the Jenkins CI system are present in the Section 5.1.4 below.

4.2.2.3 Support Activities

During the reporting period the support activities took place as planned and described in the D3.1, section 4.4 “Organization of Support Activities”

- Generic Support Unit
 - An INDIGO-DataCloud Catch-all [R7] was set up in GGUS in order to serve EGI community users encountering issues with projects’ components or willing to ask for improvement or new features



INDIGO - DataCloud



- A generic support mailing-list, indigo-su@indigo-datacloud.eu, was set up in order to ease communication with user communities.
- The support model (Figure 4.3) was implemented (reference wiki User Support), and during the reporting period only two issues were notified - http://bit.ly/indigo_ggus_all
 - One was notifying a missing dependency in INDIGO-DataCloud repositories, solved by updating the projects' third-party repositories
 - The second was a request for a new feature in a component, java-reposync, that was implemented in an updated version of the respective component
- Working together WP3.3, Exploitation, and EGI.eu, a “Technology Provider Corporate level Underpinning Agreement” (reference doc in wiki) was prepared and approved by the PMB. It specifies the support levels for the software provided to EGI for inclusion into the Unified Middleware Distribution (UMD) [R36]/Cloud Middleware Distribution /CMD) [R37].

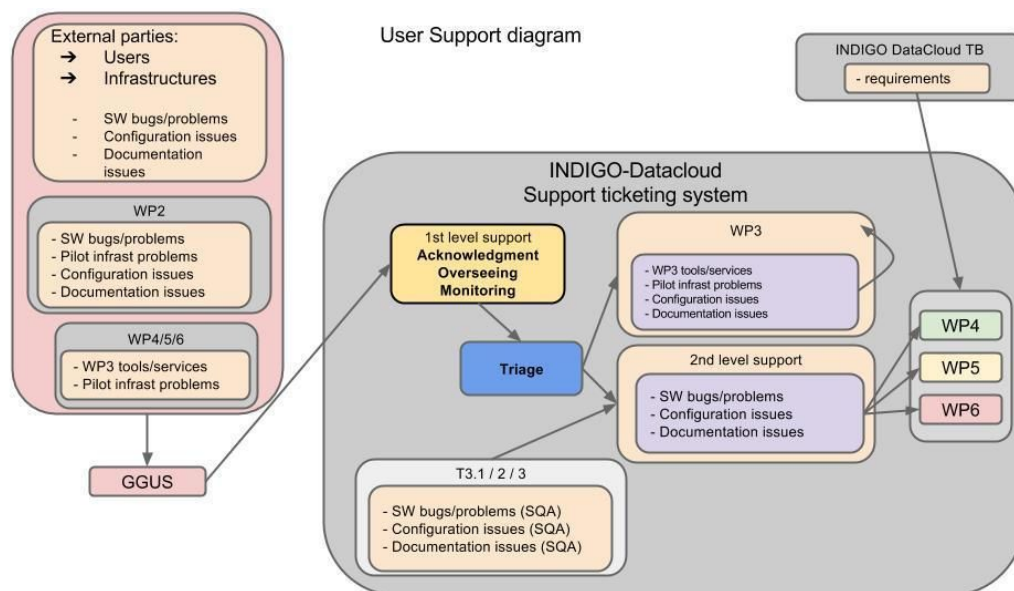


Figure 4.3: INDIGO-DataCloud Support Diagram

4.3 Release Management

The main scope of the Release Management covers:

- Release management and coordination
- Maintenance of the package repositories
- Definition of policies and release cycle



INDIGO - DataCloud

The Release Management plans and processes are described in D3.1, section 4.5 “INDIGO-DataCloud Release Process” and are summarized in Section 4.3.1. The status of its implementation is described in Section 4.3.2. It follows a section, 4.3.3 dedicated to the INDIGO-1 release as one of the major achievements of the reporting period activity.

4.3.1 Release Management process & plan

The Release Management is described in D3.1, section 4.5. The main items are:

- A 10 months release cycle: the release process has a 10 month period, with five well-scheduled phases: requirements analysis; definition of the development and test plans; development, testing and certification; release certification and validation; release and maintenance. Each INDIGO-DataCloud Work Package (2, 3, 4, 5, 6) is differently involved in each phase, with WP3 mainly concerned with the last two phases.
- Major releases: approximately at the end of a 10 month project period the delivery of an INDIGO-DataCloud major release is foreseen, containing all the significant changes, notably new features, planned and developed during the previous cycle. An INDIGO-DataCloud major release offers also the opportunity to include non backwards-compatible changes in the distribution.
- Updates to major releases: within an INDIGO-DataCloud major release, updates to component releases are possible, but cannot break their interface. The changes are managed according to the Change Management Policy, which is described in the D3.1
- Release criteria: only software that satisfies well-defined release criteria can be included in the distribution. They cover SQA criteria, packaging, testing and documentation.
- Build and test environment: one common integration, build and test infrastructure is used for all the software components. Authoritative Quality Assurance checks are integrated into this infrastructure. The chosen infrastructure is based on the Jenkins system.
- Supported platforms: the INDIGO-DataCloud distribution is made available on multiple platforms, some of them mandatory for all the software, whereas others can be optional or addressed only for some selected components. The distribution mechanism of choice, in particular for what concerns packaging and hence deployment, is the one used natively on the specific platform (e.g. YUM and RPM for RedHat-derived systems, APT and Deb for Debian-derived systems)
- Tracking releases: releases and updates are tracked with an appropriate tool. The choice is to use the WP3 OpenProject Work Packages. Component releases are scheduled as much as possible, but emergency releases are possible.
- Release Manager role: the Release Manager is responsible for governing the whole process and is assisted by the Engineering Management Team (EMT). The EMT is composed by the PT leaders, a QA representative, a Security representative, representatives of the operations teams of the major infrastructures.

4.3.2 Status of Implementation

For each of the items included in the plan presented in the previous section, the status of the implementation at the time of writing is as follows:



INDIGO - DataCloud

- A 10 months release cycle: the five stages of the first release were followed and after the publication of D2.1 [R38] - “Initial requirements from research communities”, D1.8 [R39] - “INDIGO-DataCloud General architecture”, D4.1 [R40] - “Software design document for WP4”, D4.2 [R41] - “Detailed Work Plan of the WP4 activities for the whole duration of the project”, D5.2 [R42] - “Design Document and Work Plan for the PaaS architecture”, D6.1 [R43] - “Software architecture and Work Plan for the WP6”, work concentrated on the production of one unique distribution of all the INDIGO-DataCloud software components, complying with the project policies.
- Major releases: on 8th of August 2016 the first major release of the INDIGO-DataCloud distribution, codenamed MidnightBlue, was officially announced providing support for CentOS7 and Ubuntu 14.04 platforms, and OpenStack Liberty and OpenNebula 4.14 [R44]. The MidnightBlue release is the first fundamental milestone along the sustainability roadmap identified by the project. The distribution repository contains 167 digitally signed binary packages and 40 Docker containers that allow the installation of 32 products. The distribution comes with proper documentation and release notes.

The release of INDIGO-1 was the final step of a long process that started with software components new or that have been developed in very different contexts. Many iterations were needed to achieve this very ambitious goal and the plans needed hard work, patience, collaboration and many adjustments to address the problems as they appeared
- Updates to major releases: change management to INDIGO-DataCloud releases started to be applied on a period based, monthly, after the first major release. At the time of writing this document five updates were already released
 - <https://indigo-dc.GitBooks.io/indigo-datacloud-releases/content/indigo1/updates.html>
- Release criteria: the release criteria were met by the majority of the products, and reports regarding the SQA compliance are weekly collected in the [INDIGO OwnCloud area](#), with a summary available in Section A3.2 and at
 - <https://owncloud.indigo-datacloud.eu/index.php/s/sEyA1US5zNhpWTJ>
- Build and test environment: the Jenkins Continuous integration (CI) infrastructure, complemented by tools like Grimoire, together with Integration and Preview testbeds were made available by the WP3.3. Detailed description is presented in the section “Pilot Services” below.
- Supported platforms: for INDIGO-1 a survey was done through the PTs and the resource providers of the Integration and Preview testbeds (reference http://bit.ly/cmfv_versions) regarding the supported Cloud Management Frameworks, and eventual plans to update their infrastructures (for the testbeds resource providers) - the decision was to support the OpenStack Liberty and OpenNebula 4.14 versions, and as a consequence the mandatory supported operating systems were CentOS7 and Ubuntu 14.04
- Tracking releases: all the component releases that compose the overall INDIGO-1 major release and subsequent update are tracked in the WP3 OpenProject task tracker (reference) using the “INDIGO - DC Releases” category. Each item followed a well-defined workflow that expresses the procedure for scheduled releases, which includes of course a validation step by the SQA team.



INDIGO - DataCloud

- Release Manager role: the EMT, chaired by the Release Manager, has met regularly, typically on Friday mornings, since early in the project. Meetings were complemented by a heavy use of the dedicated mailing list. The activity of the EMT (meetings, corresponding minutes, archives) is fully tracked at
 - <https://owncloud.indigo-datacloud.eu/index.php/s/8XsS2MHKQfE3s7i>

4.3.3 INDIGO-1 (MidnightBlue)

The first reporting period contains the preparation and release of the first INDIGO-DataCloud major release - INDIGO-1 (codename MidnightBlue). The activities that lead to this major achievement are summarized below:

- According to the Release Schedule methodology described in the D3.1, section 4.5.3.1 “Schedule Methodology”, the Release Manager prepared and presented for approval to the EMT the INDIGO-1 release schedule:

Task/Milestone	Start	Length	Date
Planning and Development	Day after the end of the Planning phase for the previous GA Release	More than 5 months	-
Feature Submission Deadline	3-4 months before the Feature Freeze	n/a	
Feature Freeze	2 month before the RC	Until GA	30.05.2016
Final Change Deadline (code freeze)	End of Testing and Certification	Until GA	20.06.2016
Release Candidate	End of Testing and Certification Start Acceptance Testing 2 months before GA-release day 3 days after the Final Change Deadline	Until GA	23.06.2016
Compose Final RC	Same as Release Candidate	1 week	01.07.2016
Technical-Previews	1 month before the GA	Should last 2 weeks	04.07.2016 (01.07.2016 is Friday)
Test Final RC	2 weeks before GA	6 days	18.07.2016
GA Release	Release Date	n/a	01.08.2016 (31.07.2016 is Saturday)
Maintenance	GA release day	~10 months	01.08.2016
End of Life	10 months after the GA of current release	n/a	31.05.2017

- With respect of the schedule the release was delayed with just one week, needed in order to collect all the SQA reports in order to understand the readiness of all the components, the preparation of the production repositories and the final release notes.
- As mentioned in the previous section



INDIGO - DataCloud

- the mandatory platforms for this major release were decided
- All the infrastructural and operational resources were made available:
 - Jenkins CI
 - Repositories - testing, preview and production; DockerHub organizations
 - EMT agendas and mailing-list
 - Release and SQA tracking tool
- The first demos/technical previews of selected components were made available well in advance and presented at the “INDIGO Champions meet Developers” event in Amsterdam, on 04-05 April 2016 [R57]
- The release preparation was followed through an INDIGO-DC Release task, [#3947](#) - as parent task for 43 individual components task, having dependencies on 47 SQA (http://bit.ly/sqa_indigo1). The differences in the number are because some of the SQA tasks were referring to components of the same service, some of the components were in fact 3rd party contributions (like jOCCI (reference) or documentation contributions regarding the use of already existing software solutions as part of the INDIGO-DataCloud architecture (like Kubernetes (ref), Mesos Cluster (ref)) or guides on how to configure already existing services to enable support for INDIGO-DataCloud solutions (like OpenStack Keystone AAI Support (ref))
- During the testing phase artefacts made available from the testing repository (ref) were used, while for the technical previews from the preview repositories, where packages were signed with INDIGO-DC official signing key.

The public availability of a new INDIGO-DataCloud release was announced through:

- general announcements
 - On the project News section
 - <https://www.indigo-datacloud.eu/news/first-indigo-datacloud-software-release-out>
- INDIGO-DataCloud RSS Feed
 - <http://repo.indigo-datacloud.eu/INDIGODataCloudNews.rss.xml>
- indigo-all mailing-list
 - <https://lists.indigo-datacloud.eu/sympa/arc/indigo-all/2016-08/msg00000.html>

4.3.3.1 Contributions to Upstream Distribution

Many of the INDIGO-DataCloud solutions released through INDIGO-1 were also contributed to the respective upstream distributions like:

- OpenStack
 - Changes/contribution done **already merged** upstream
 - Nova Docker
 - Heat Translator (*INDIGO-DataCloud is 3rd overall contributor and core developer*)



INDIGO - DataCloud

- TOSCA parser (*INDIGO-DataCloud is 2nd overall contributor and core developer*)
- OpenID Connect CLI support
- OOI: OCCI implementation for OpenStack
- Changes/contribution **under discussion** to be merged upstream
 - OpenStack Preemptible Instances support (extensions)
- OpenNebula
 - Changes/contribution done **already merged** upstream
 - ONEDock
- Changes/contribution done already merged upstream for:
 - Infrastructure Manager
 - CLUES
 - Onedata
 - jSAGA Adaptors
 - MitreID connect library

4.4 Release & Maintenance and security vulnerabilities

The D3.1 - Initial Plan contained the description of the Emergency Release procedure, in section 4.5.2.2, describing the procedure to be followed when a major incident, a defect considered critical or a security vulnerability, is discovered during usage of a software component or service in a production-infrastructure.

After the agreement on the [Technology Provider Corporate level Underpinning Agreement](#) a dedicated section on the Software Maintenance wiki was added on how to address security vulnerability [R45], based on the documentation available from the EGI Security Vulnerability Group [R46]:

- for Security Vulnerabilities assessed as such by the EGI-SVG:
 - for LOW (target date = 1 year) & MODERATE (target date = 4 months) risk assessment - issues will be treated as a normal RfC to be addressed in one of future updates of affected products, to be made available through a normal Update, by the "due date" mentioned in the assessment
 - for HIGH (target date = 6 weeks) risk assessment - issues will be further assessed, together with EGI-Operations taking in consideration the impact on the production-sites/users. In case it is evaluated as CRITICAL risk (target date = 3 days) the EMERGENCY procedure for an Immediate priority RfC must be followed. Otherwise: if by the "due date" it is available a normal Update Cycle, the issues should be treated as a High priority RfC to be fixed by an update of the affected product through the first normal update cycle; if by the "due date" there is no normal Update Cycle, the issues should be treated as an Immediate priority RfC to be released following the EMERGENCY procedure.
 - for CRITICAL (target date = 3 days) risk assessment - the EMERGENCY procedure should be followed.
- RfC for security vulnerabilities:



INDIGO - DataCloud



- in case of security vulnerabilities assessed as such by the EGI-SVG or by the PT itself
 - the RfC must reference the EGI-SVG internal, private, RT tkt. No description or details regarding the vulnerability should be present in publicly available places.
- Release Task Tracking:
 - Component Release Notes:
 - in case the updated product/component contains the fix for a security vulnerability the Release Notes/What's New section must contain the reference to the EGI SVG: Advisory, no other details regarding the vulnerability should be present.



5. PILOT SERVICES

5.1 Introduction

The task WP3.3 is providing the set of services needed for the integration and testing of software components developed by the product teams in WP4, WP5 and WP6. In addition is also providing testbed infrastructures: for software development, integration and preview.

A set of tools and services have been deployed or provided to support:

- Software developed by JRAs (in WP4, WP5 and WP6).
- The project management (for all WPs).
- The SQA, CI and delivery processes (WP3.1).
- The software release, management activities and support for end-users and infrastructure managers (WP3.2).

The Development infrastructure provides:

- Resources for developers to test software under development.

The Integration infrastructure provides:

- Resources to support the continuous integration process and SQA, these resources are kept separate and are fully dedicated to the continuous integration system.
- Test integration/interaction between software components.

The Pilot Preview infrastructure is target to:

- Deploy software components with INDIGO-DataCloud released versions in a production-like conditions.
- User communities (WP2), providing an environment to deploy and validate use-cases under realistic conditions using released software.
- Support application and workflow porting to DCI environments.

These infrastructures are using diversified computing and storage resources operated by the project partners with a wide geographic coverage in Europe, thus demonstrating the ability of the INDIGO-DataCloud software to operate properly over heterogeneous distributed resources.



INDIGO - DataCloud



5.2 Services for Continuous Integration and Software Release

WP3.3 operates a set of tools and services to support the product teams, the software quality assurance (SQA), the continuous integration (CI), the software release, software maintenance and the user support. These services have been outlined in the WP3 initial plan [R1] and later were extensively tested, integrated and made available. Aiming at minimizing the services deployment and maintenance effort, WP3 opted to use publicly available services whenever adequate. Thus minimizing the set of services that needed to be deployed and maintained by WP3 itself. The main reasons supporting this decision were:

- The use of publicly available services allows:
 - Higher public visibility and discoverability.
 - Better fit to open source software development.
 - Easier path for external contributions.
 - Provides a path to further development, support and exploitation beyond the end of the project.
- Smaller effort needed inside the project to operate and manage those services. Makes service delivering more cost effective, simultaneously enabling WP3 to focus on the services operation and project objectives.


The list of services is shown in the Table 5.1 containing the name of the service, the URL and type; either public or deployed/operated by WP3. As it can be seen only 3 of the 11 services are deployed and operated by WP3.

Table 5.1: Services table

Name		URL	Location/Type
	OpenProject	https://project.indigo-datacloud.eu/	WP3 - INFN-CNAF
	Artefacts repository - packages RPMs and DEBs	http://repo.indigo-datacloud.eu/	WP3 - INFN-CNAF and hosted CERN
	Artefacts repository Dockerhub - Docker containers	https://hub.docker.com/u/indigodatacloud/ https://hub.docker.com/u/indigodatacloudapps/	Public, under projects' organizations
	Jenkins CI	https://jenkins.indigo-datacloud.eu:8080	WP3 - INFN-CNAF and hosted CERN
	GitHub - Source code repository and version control	https://GitHub.com/indigo-dc	Public, under projects' organization
	GitHub - issue tracker	https://GitHub.com/issues?user=indigo-dc	Public, under projects' organization



INDIGO - DataCloud

	Grimoire (code metrics) plus GitHubPages - gh pages	<a href="https://indigo-dc.GitHub.io/<GitHub_REPO>">https://indigo-dc.GitHub.io/<GitHub_REPO> Examples: https://indigo-dc.GitHub.io/tts/browser/ https://indigo-dc.GitHub.io/im/browser/	Public, under projects' organization
	Global Grid User Support - GGUS	https://ggus.eu/	Public, project support team with mailing list
	GitBook - Documentation	https://www.GitBook.com/@indigo-dc/ https://www.GitBook.com/book/indigo-dc/indigo-datacloud-releases/details	Public, under projects' organization
	Ansible Galaxy - Automatic deployment and configuration	https://galaxy.ansible.com/indigo-dc/	Public, under projects' organizations
	GitHub Pull Requests (PR) Gerrithub - source code review	<a href="https://GitHub.com/indigo-dc/<GitHub_REPO>/pulls">https://GitHub.com/indigo-dc/<GitHub_REPO>/pulls https://review.gerrithub.io/	Public, under projects' organizations

More complete service descriptions along with some statistics are detailed in the subsequent subsections.

5.2.1 Project management service

OpenProject: <https://project.indigo-datacloud.eu/>

The Project management service is providing tools such as a task tracker, wiki, a placeholder for documents and project management. The service is being hosted and supported by the INDIGO-DataCloud partner INFN-CNAF.

Each of the INDIGO-DataCloud WPs has a “project” in OpenProject service. There is also one “project” for each Quarterly Report and several cross-WP activities: AAI task force, Outreach and Exploitation, and for the Service Provider Board (SPB). WP3 uses OpenProject extensively.

Figure 5.1 is a snapshot of the WP3 “project” summary with the “Type”, “Priority”, “Category”, “Assignee” of the track tickets.



INDIGO - DataCloud

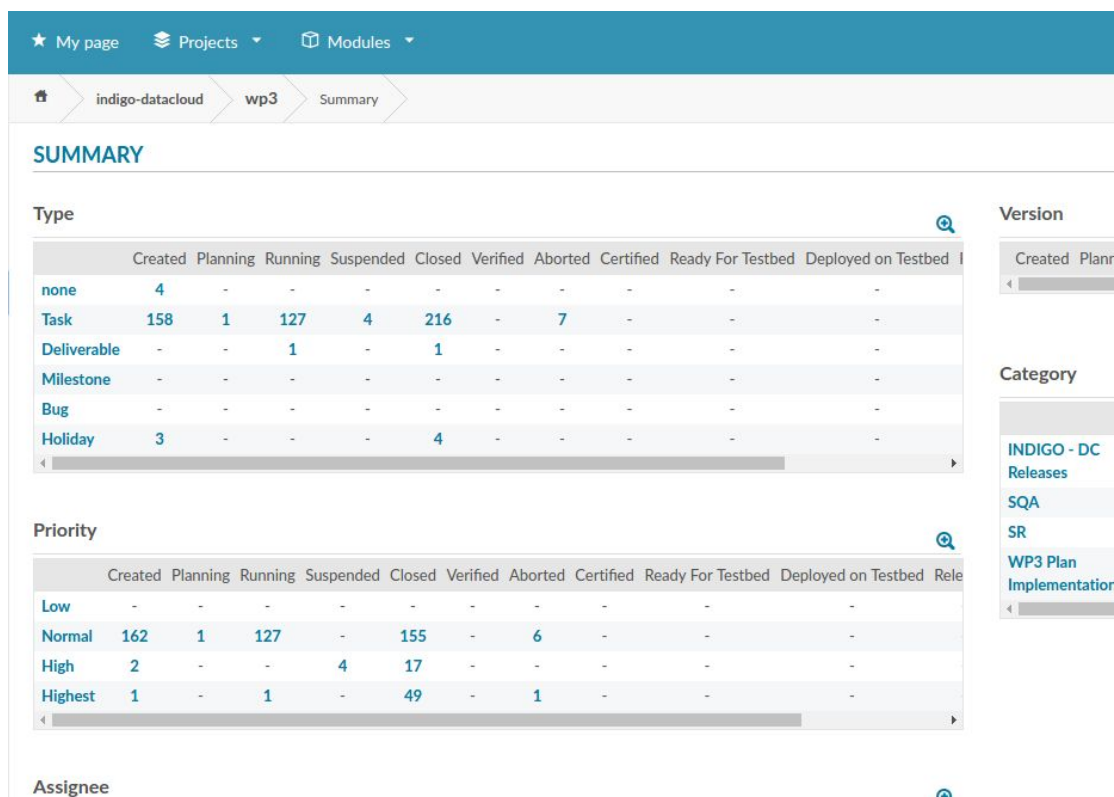


Figure 5.1: WP3 Summary page. Shows the “Type”, “Priority”, “Category”, “Assignee” of the track tickets.

Table 5.2 shows the statistics regarding the “Tasks” (track tickets) of WP3 in OpenProject.

Table 5.2: Track tickets statistics

Created	Planning	Running	Suspended	Closed	Aborted	Total
158	1	127	4	216	7	513

The following Task “Categories” have been defined:

- INDIGO-DataCloud Releases: tasks used to track each release and updates (WP3.2);
- SQA: tasks used to track the SQA activities (WP3.1);
- SR: tasks used to track the Staged Rollout test of software components (WP3.4);
- WP3 Plan Implementation: tasks used to track the implementation of the services and workflows of the WP3 activities.

Although not defined as a “Category”, there are tasks used to track the request from JRA developers (WP4, WP5, and WP6) when requesting resources from the development/integration infrastructure for their development work.



INDIGO - DataCloud

5.2.2 Artefacts Repository

Repository: <http://repo.indigo-datacloud.eu/repository/>

INDIGO-DataCloud has deployed and operates a software artefacts repository, publicly exposed through the web. The repository stores the following artefacts for the supported Operating Systems - Centos7 and Ubuntu 14.04, and Cloud Management Frameworks (CMF) OpenStack Liberty version and OpenNebula 4.14

- packaged software:
 - YUM repositories for RPMs:
http://repo.indigo-datacloud.eu/repository/indigo/1/centos7/x86_64/
 - APT repositories for DEBs:
<http://repo.indigo-datacloud.eu/repository/indigo/1/ubuntu/dists/>
- tarballs:
http://repo.indigo-datacloud.eu/repository/indigo/1/centos7/x86_64/tgz/.

The service is hosted at CERN, was deployed and is operated by CNAF. Figure 5.2 show the artefacts repository. It contains links and information about the INDIGO-DataCloud software releases and updates, documentation, configuration, information contact, and support contact.



INDIGO - DataCloud

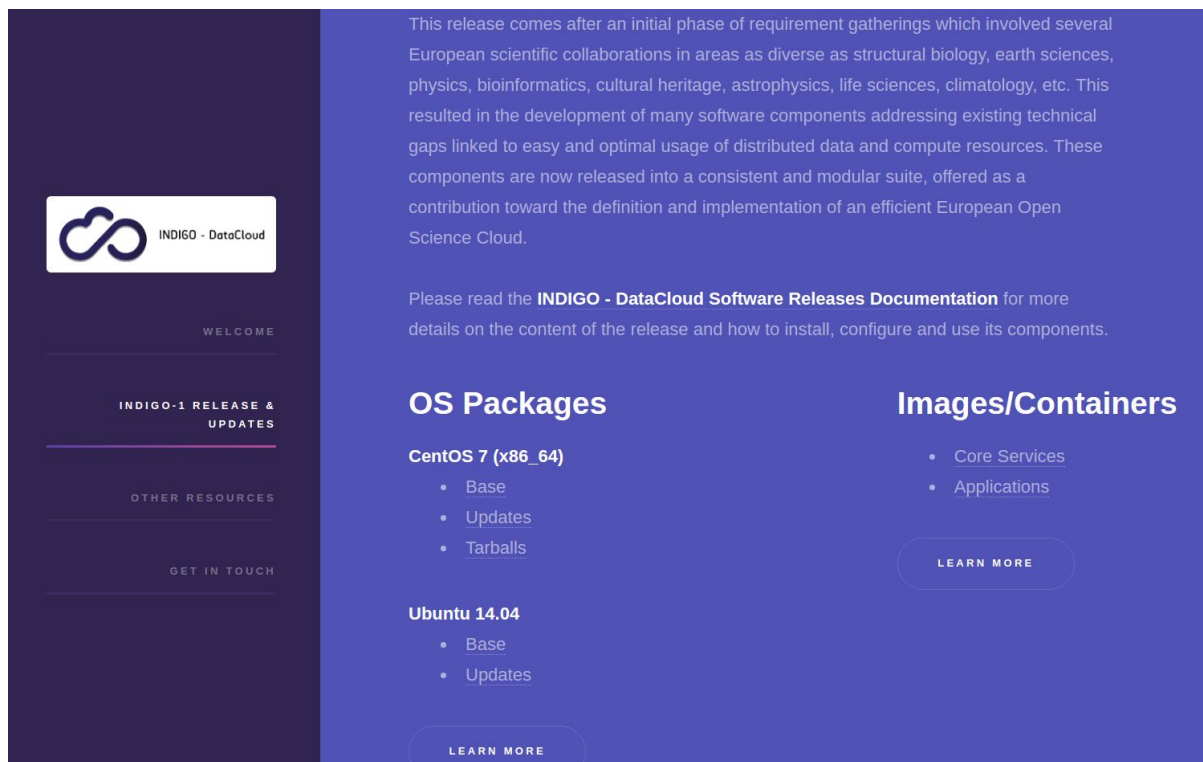


Figure 5.2: Artefacts repository frontend

The repository has the following structure:

- **Production** repository: containing signed, tested and validated packages
 - **base**: packages released in the INDIGO-1 - MidnightBlue release;
 - **updates**: packages released in the software updates - at the time of writing there have been 3 updates;
 - **third-party**: external packages needed by INDIGO-DataCloud components to satisfy dependencies, not available in other repositories
 - **source** RPMs and tarballs
- **Preview** repository: containing production-ready packages, signed with the INDIGO-DataCloud key aimed to be deployed into the Preview testbed, before the official release, in order to allow projects' user communities to validate the developed software against their use cases
- **Testing** repository: containing packages provided for testing mainly by developers and for integration purposes, before the transition to "Preview" repository.

Table 5.3 shows the number of artefacts/packages in each repository at the moment of writing the document, after the INDIGO-1 release and three updates:



INDIGO - DataCloud

Table 5.3: artefacts per repository

Centos7			Ubuntu 14.04 (trusty)			Tarballs
Base	Updates	Third-party	Base	Updates	Third-party	Source & Binaries
34	21	39	55	26	38	43

5.2.3 Artefacts Repository - Docker containers

Dockerhub:

<https://hub.docker.com/u/indigodatacloud/>

<https://hub.docker.com/u/indigodatacloudapps/>

Dockerhub container images are automatically built from GitHub repositories where the respective Dockerfiles reside. The built docker images are thus treated as software with version control, tagging, integrated in the SQA CI process, to be included in the INDIGO-DataCloud releases.


Table 5.4 shows the deployed Dockerhub organizations, its description and the number of repositories (docker images). Figure 5.3 is a snapshot of the Dockerhub dashboard of the “indigodatacloud” organization that host the docker images for INDIGO-DataCloud services, next to each image repository is also shown the number of downloads/pulls of the image.

Table 5.4 show a snapshot of the dockerhub “indigodatacloud” organization with some of the repositories and the number of downloads of images next to each one.

Organization	Description	Number of repositories
Indigodatacloud	docker images with services and components	40
Indigodatacloudapps	docker images with applications used by the user communities of WP2	13



INDIGO - DataCloud

 Dashboard Explore Organizations

indigodatacloud

Repositories Teams Billing Settings

Repositories







 <div>indigodatacloud/im public automated build</div>	2 STARS	10K+ PULLS	DETAILS
 <div>indigodatacloud/orchestrator public automated build</div>	1 STARS	3.9K PULLS	DETAILS
 <div>indigodatacloud/ubuntu-sshd public automated build</div>	0 STARS	859 PULLS	DETAILS
 <div>indigodatacloud/zookeeper public automated build</div>	0 STARS	800 PULLS	DETAILS
 <div>indigodatacloud/mesos-slave public automated build</div>	0 STARS	786 PULLS	DETAILS
 <div>indigodatacloud/mesos-master public automated build</div>	0 STARS	705 PULLS	DETAILS

Figure 5.3: Dockerhub “indigodatacloud” organization dashboard.

5.2.4 Continuous Integration service

Jenkins: <https://jenkins.indigo-datacloud.eu:8080>

The Jenkins Continuous Integration server is the orchestrator for the automated builds, other automated tasks needed to accomplish the SQA, and to produce packages for the Artefacts repository. It schedules jobs to slave nodes that execute the necessary tasks.



INDIGO - DataCloud

The Jenkins-CI server and slave nodes are hosted at CERN and are operated by CNAF. The Definition and implementation of the Jenkins jobs are performed by the developers (PT) and WP3.1. The execution is in general automatic but can also be triggered on demand. Report checking, visualization and inspection is performed by CNAF as release manager.

Figure 5.4 show the Jenkins-CI web frontend, where the status of jobs can be visualized. It also allows the management of the Jenkins-CI server such as definition/inclusion of jobs, users and roles, the configuration of slave nodes, among others..

The job status panel has several tabs where one can select the visualization of specific types of jobs. A few examples are: “Apps” for application components, “OpenNebula” and OpenStack” components, “SLOC” for the number of lines of code and comments, “Test - Code style” for code style, Unit and Functional tests.

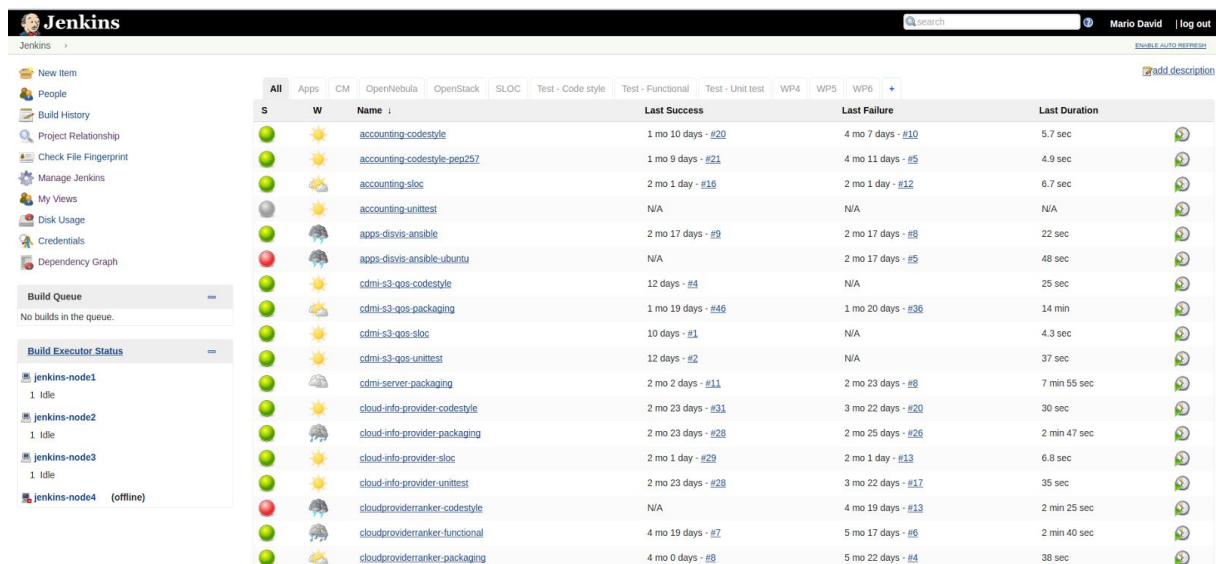


Figure 5.4: Jenkins-CI web frontend

Table 5.5 shows the number of jobs for each category. The total number of jobs (164) is not the sum in each category since a fraction of the jobs is not yet categorized, while some of the jobs in the WP4-6 category are also in some of the other categories.

Table 5.5: Jenkins jobs per category



INDIGO - DataCloud

Apps	CM	Open Nebula	Open Stack	SLOC	Code style	Functional	Unit	WP4	WP5	WP6	All
2	1	7	10	27	30	4	31	46	27	19	168

5.2.5 Version Control System

GitHub: <https://GitHub.com/indigo-dc>

GitHub is the source code repository and Version Control System. An organization has been created “indigo-dc” to host all repositories with source code developed by the project. There are also repositories which are forks from external projects and others that are contributions to external projects

Besides the source code, GitHub is also hosting configuration modules (ansible configuration roles or puppet templates), Dockerfiles that are later used to automatically build the docker images in the Dockerhub repositories, documentation and release notes for the components.

Table 5.6 contains statistics regarding the GitHub “indigo-dc” organization.

Table 5.6: GitHub “indigo-dc” number of repositories, persons and teams.

Number of repositories	Number of persons	Number of teams
161	83	28

Figure 5.5 is a snapshot of the GitHub “indigo-dc” organization. It show some of the repositories along side with a commit statistics graphic, the number of people, teams, and the main programming languages.



INDIGO - DataCloud



The screenshot shows the GitHub organization page for 'indigo-dc'. The header includes the organization's name, a search bar, and navigation links for Pull requests, Issues, and Gist. Below the header, the organization's profile is displayed with the logo and name. The main section lists repositories with their names, descriptions, languages, and update times. On the right, there are sidebars for 'Top languages' and 'People'.

Repositories:

- tts**: the Token Translation Service (TTS). Languages: Erlang. Updated 4 hours ago.
- fgAPIServer**: Forked from FutureGateway/fgAPIServer. Language: Python. Updated 6 hours ago.
- indigo-datacloud-releases**: Repository for the INDIGO-DC Releases gitbook. Language: CSS. Updated 16 hours ago.
- puppet-synergy**: Language: Puppet. Updated 22 hours ago.
- dcache**: Forked from dCache/dcache. Description: dCache - a system for storing and retrieving huge amounts of data.

Top languages: Python, Java, Shell, Erlang, Ruby.

People: A grid of 16 user avatars.

Figure 5.5: GitHub “indigo-dc” organization source code repositories



INDIGO - DataCloud

5.2.6 Issue/Bug tracker services

GitHub issue dashboard: <https://GitHub.com/issues?user=indigo-dc>

OpenProject: <https://www.openproject.org/>

One of the requirements for the management of the software maintenance activity was that the issues and improvements found in the software developed through the INDIGO-DataCloud project to be tracked in the GitHub “indigo-dc” organization dashboard. At the time of writing this deliverable the vast majority of teams followed the request, with just few of them using different issue-tracker systems, mainly because the respective software already existed before the project, and those systems were already in use.

Table 5.7 show the statistics for all INDIGO-DataCloud components in GitHub. Figure 5.6 is a snapshot of the GitHub issues dashboard showing the issues currently opened.

Table 5.7: issues in GitHub dashboard

# open issues	# closed issues	# total issues
79	219	298



INDIGO - DataCloud

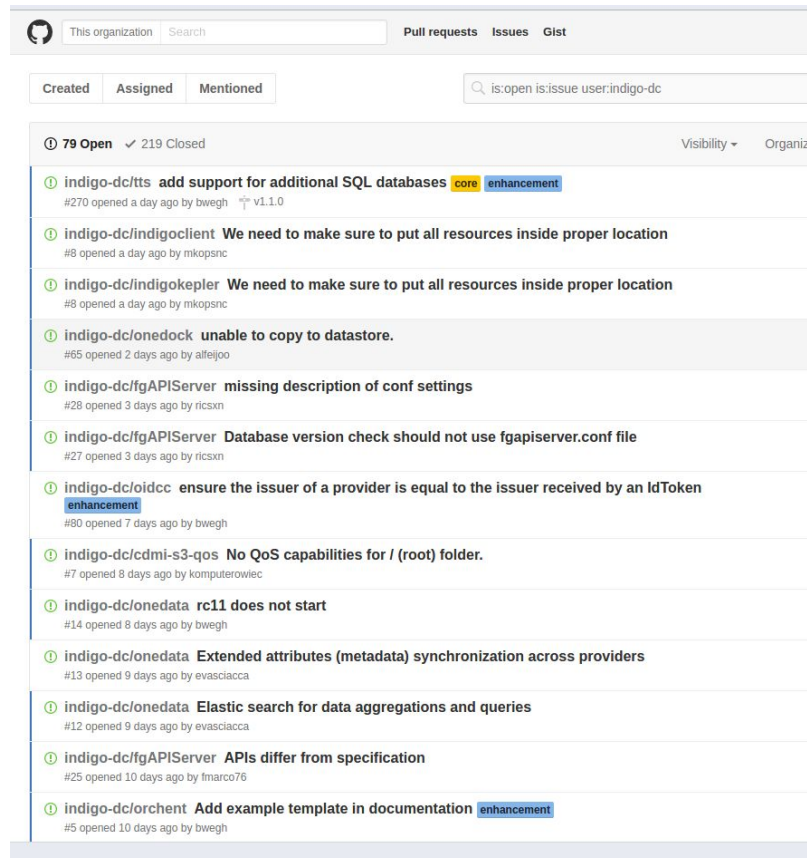


Figure 5.6: GitHub issues dashboard for indigo-dc organization.

5.2.7 Code metrics services

GitHub pages powered by MetricsGrimoire, VizGrimoire:
https://indigo-dc.GitHub.io/<GitHub_REPO>

Metrics and some Key Performance Indicators (KPIs) are gathered in GitHub pages. The backend is implemented with Metrics and Viz Grimoire. There is a dashboard for each component/repository in GitHub in the location shown above.

At the time of writing not all repositories have implemented such dashboards. Nonetheless, metrics and KPIs are being gathered and summarize regarding the SQA for all components.

As examples, the dashboard locations for some components are:

- udocker: <http://indigo-dc.GitHub.io/udocker/browser/>. Figure 3.7 in section 3.4 shows a snapshot of the dashboard for this component.



INDIGO - DataCloud

- IAM: <https://indigo-dc.GitHub.io/iam/browser/>
- OOI: <https://indigo-dc.GitHub.io/ooi/browser/>

5.2.8 User support

Global Grid User Support (GGUS): <https://ggus.eu>

The GGUS is the service that external users and service providers administrators should use for support of INDIGO-DataCloud components. This service is provided by the EGI European Infrastructure.

An INDIGO-DataCloud support team has been set-up in GGUS with a mailing list. The details can be found in section 4.2.2.3. Figure 5.7 shows a snapshot of the GGUS Helpdesk for the INDIGO-DataCloud support unit.

The screenshot shows the GGUS Helpdesk interface. On the left is a navigation menu with links like 'Did you know...', 'Documentation', 'Registration', 'My dashboard', 'Search ticket', 'Submit ticket', and 'Support staff'. The main area is titled 'GGUS - the Helpdesk' and features a 'Ticket search engine' section with various filters for Ticket-ID, Submitter, Type, VO, Site, Priority, Resp. Unit, Status, Last Update, and more. Below the filters, there are input fields for 'Ticket ID?', 'Support Unit' (set to 'INDIGO DataCloud'), 'Concerned VO?', 'User?', 'Keyword?', 'Involved support?', 'Assigned to person?', 'Notified site', and 'Special attributes'. A 'GO!' button is present. Below the search filters, there is a table showing 4 tickets found. The table has columns: Ticket-ID, Type, VO, Site, Priority, Resp. Unit, Status, Last Update, and Subject.

Ticket-ID	Type	VO	Site	Priority	Resp. Unit	Status	Last Update	Subject
123995		vo.indigo-datacloud.eu		urgent	INDIGO DataCloud Catch-all ▶ involved	solved	2016-12-05	java-syncrepos should support adding ten...
123769		vo.indigo-datacloud.eu		less urgent	INDIGO DataCloud Catch-all ▶ involved	verified	2016-11-16	Unable to install python-novacient from...
123266		dteam		less urgent	INDIGO DataCloud Catch-all	closed	2016-08-24	test INDIGO Catch-All SU
116517		none		less urgent	INDIGO DataCloud Catch-all	verified	2015-10-01	TEST ticket

Tot pages: 1
[1]

Figure 5.7: Global Grid User Support dashboard for support unit “INDIGO-DataCloud catchall”.

5.2.9 Documentation

Documentation: <https://www.GitBook.com/@indigo-dc>

The INDIGO-DataCloud is using the GitBook to host the documentation for all components as well as for the releases. GitBook allows to automatically produce “Books” from the documentation in the GitHub repositories through a link to the respective repository. This allows treating the



INDIGO - DataCloud

INDIGO-DataCloud documentation as software with version control. Figure 5.8 shows the dashboard of the GitBook “indigo-dc” organization.

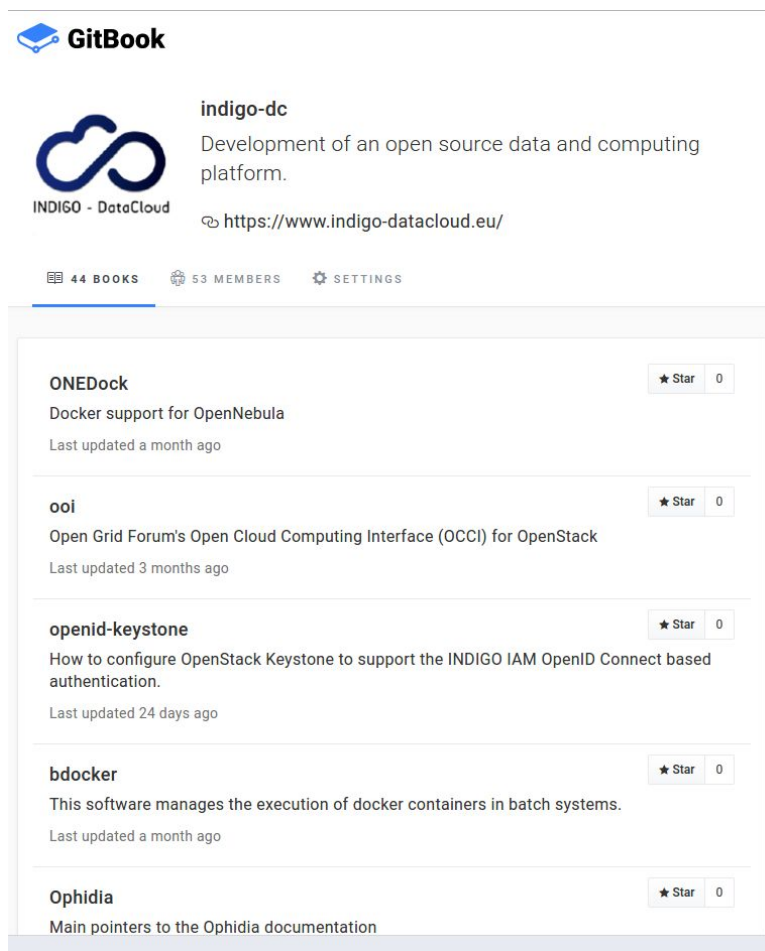


Fig. 5.8: GitBook dashboard for the “indigo-dc” organization is the placeholder for all documentation. It includes the documentation about the INDIGO-DataCloud releases.

There are cases where GitHub repositories only host documentation (no source code) such as the information about the releases and updates. At the time of writing there are 44 books published.



5.2.10 Automatic deployment and configuration

Ansible Galaxy: <https://galaxy.ansible.com/indigo-dc/>

Ansible Galaxy service is being used to provide ansible roles to automatically install and configure INDIGO-DataCloud components and applications.

Role	Description
ambertools	Ambertools applications
ansible-role-idh-master	Install the Identity Harmonization (IDH) service
ansible-role-liferay-iam	Deployment of IAM authentication modules on a running Liferay Instance version 7.0.x
calico	Configure and run calico
cdmi-master	Install CDMI
cdmi-s3-qos	Install cdmi-s3-qos module together with embedded CDMI server
chronos	Install Chronos
cloud-info-provider	Install and configure cloud-info-provider
clues	Install Clues
ansible-role-cdmiserver	Configure and install CDMIServer

Fig. 5.9: Ansible Galaxy dashboard for the “indigo-dc” organization. Placeholder for ansible roles for automatic deployment and configuration.

These roles are managed by GitHub repositories and automatically published in Ansible Galaxy by linking it to the respective GitHub repository. As is the case of the Docker images in Dockerhub and the Documentation in GitBook, the management of those roles is accomplished in the respective GitHub repositories. Figure 5.9 shows the dashboard of the Ansible Galaxy “indigo-dc” organization. At the time of writing there are 37 ansible roles available in the “indigo-dc” organization.



INDIGO - DataCloud

5.2.11 Code review service

GitHub Pull Requests: https://GitHub.com/indigo-dc/<GitHub_REPO>/pulls

GerritHub: <http://gerrithub.io/>

Code review is currently being performed in the git pull requests (PRs). When a PR is triggered by a commit in a GitHub repository, the automatic tests (Jenkins-CI) take place, upon the successful pass of those tests follows the code review that finalizes the the SQA process.

Figure 5.10 shows the dashboard of the GitHub open PRs for “indigo-dc” organization, for each PR the corresponding commit message is also shown. Table 5.8 shows the statistics of PRs for all components.

Table 5.8: Number PRs in GitHub of Indigo-DataCloud components.

# PRs open	# PRs closed	# PRs total
28	1219	1247



INDIGO - DataCloud

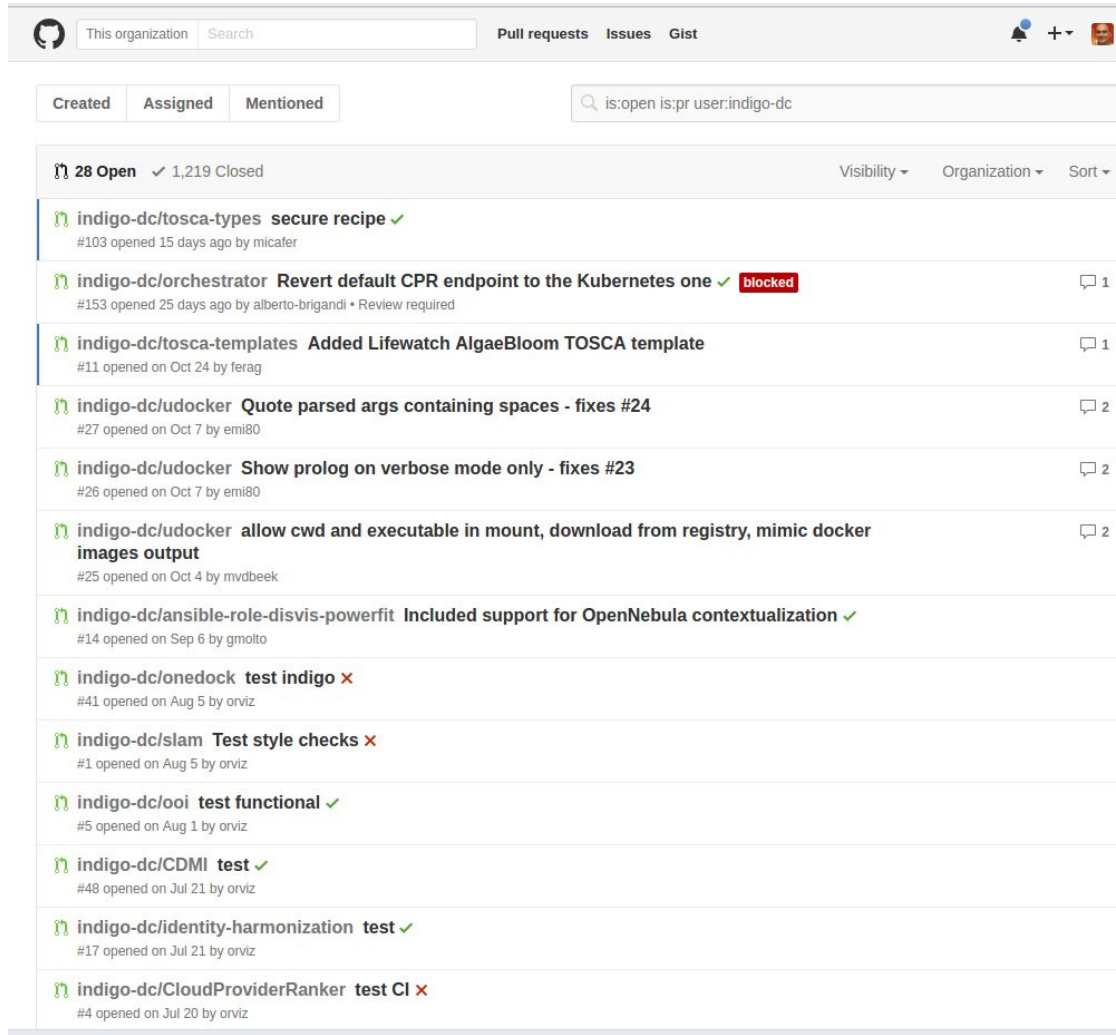


Figure 5.10: Dashboard showing the PRs in GitHub. The commit message is shown in the dashboard.

5.3 Development Infrastructure

The “Development Infrastructure” provide resources to the Product Teams for software development purposes.

A procedure was put in place by WP3.3 to manage resource requests performed by the PTs. These resources are aimed at supporting the development work and are provided mainly by the project partners in WP3.3 using private IaaS cloud resources. In very specific cases resources can be provided by partners from other WPs. In both cases the process is coordinated by WP3.3.



INDIGO - DataCloud

The complete procedure is detailed in OpenProject in the WP3 wiki [R56], and consists in the following steps:

1. Creation of new “Task” in WP3 “Work-Packages” section.
2. Defining a resource request via the field "Subject" by specifying:
 - a. “Request resources for component <ComponentName> Work Package <Indigo Work Package/Task/Subtask>”
3. Defining a description as detailed as possible of the needed resources:
 - a. Operating System, external SW, if there is already a resource provider or site that will support the PT.
4. Setting the responsible as the WP3.3 task manager.

The work is tracked in the OpenProject ticket that is closed once the PTs successfully get the required resources. Table 5.9 show the resource request tasks that have been created and closed.

Table 5.9: tasks open, closed and total.

# Tasks Open	# Tasks closed	# Tasks total
1	12	13

The number of resource request is reasonably small since a large majority of the PTs have access to their own resources.

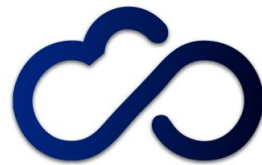
5.4 Integration Infrastructure

The “Integration Infrastructure” provides resources to the PTs to test the integration and interaction between components.

The procedure for PTs to request resources is the same as detailed previously for the “Development Infrastructure”. These resources are aimed at supporting the integration and interoperability between different components and services.

The integration testing is coordinated by WP3, a map with the partners providing resources for the Integration Infrastructure as well as the corresponding services and components is shown in Figure 5.11. An extensive list of tests has been identified and is kept continuously updated. For each test there is a responsible partner and the corresponding status is kept updated. The integration tests list contains:

- the test description, responsible partner, status;
- Work Package specific tests as well as inter Work Package tests.



INDIGO - DataCloud

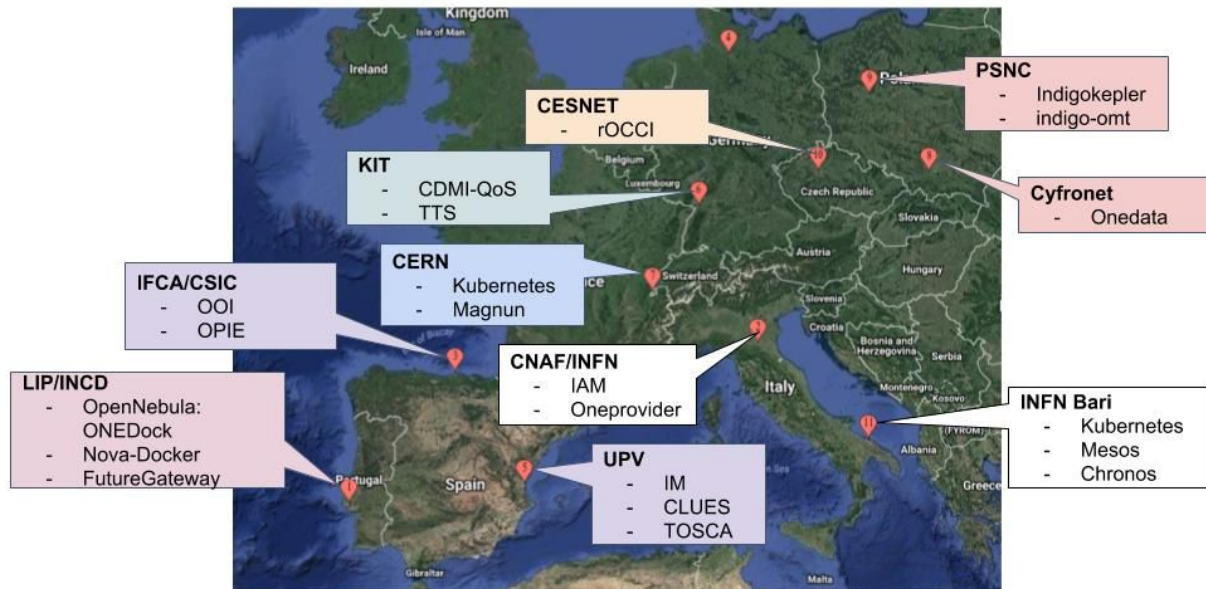


Figure 5.11: Map with partners hosting resources for the Development and Integration Infrastructure. The list is not complete for visualization purposes.

The tables shown in Appendix A2.1 contain all information about some of the integration tests, a complete list is not shown due to its extensive nature:

- Table A2.1.1: definition of tests between WP4 (IaaS layer) components.
- Table A2.1.2: definition of tests between WP4 (IaaS layer) and WP5 (PaaS layer) components.
- Table A2.1.3: definition of tests between WP5 (PaaS layer) components.
- Table A2.1.4: definition of tests between WP6 (SaaS layer) components.

5.5 Pilot Preview Infrastructure

The “Pilot Preview Infrastructure” is an environment for users (WP2) to test and validate the INDIGO-DataCloud products, with their use cases. It is operated and maintained as a production like environment but with limited capacity.

The deployment, operation and support is performed by system administrators that are in general partners contributing to WP3. The services are deployed with components that have been officially released by INDIGO-DataCloud and are kept synchronized with the released updates.

WP3.3 via the preview infrastructure also participates in the Staged Rollout process managed by WP3.4. The Staged Rollout is further detailed in the exploitation section 6. This systematic process



INDIGO - DataCloud

aims to gather feedback from system administrators that deploy and manage the software components and constitutes a final step to assess the software quality from the deployers point of view in production or near-production environments.



Figure 5.12: Map with resource providers part of the WP3.3, operating and supporting the Pilot Preview Infrastructure.

The preview infrastructure aims to demonstrate the software technology readiness for production as well as the software completeness and fulfillment of the user requirements gathered by WP2. Figure 5.12 is a map showing the partners that participate in the Pilot Preview Infrastructure. Also shown the components and services deployed by each partner.

The tables shown in Appendix A2.2 detail the full set of services and endpoints that are part of the “Pilot Preview Infrastructure”. Each table contain the components or services per category.

Figure 5.13 shows the architecture of a INDIGO-DataCloud use case. A technical description of the flow through the several components and services is described next.

The first step that users should perform is their registration in the INDIGO-DataCloud Access Management service (IAM). The IAM is used throughout all services for authentication and authorization between the user and the services. The services should also be registered in the IAM, Table A2.2.1.



INDIGO - DataCloud

On the left side of the diagram (light red) is the Future Gateway and associated components as shown in Table A2.2.7 (this is the SaaS WP6 components). It's a Scientific Gateway portal to be used directly by the users. It has an API server that is used by the FG portal itself, the LiferayIAM component to interact with the IAM service, Kepler is a Workflow engine and Ophidia is a Big-Data framework to support some of the use cases.

Tosca templates are resource descriptions that the users submit from the FG portal to the PaaS (WP5) layer (right side box in teal).

The PaaS services and endpoints are shown in Tables A2.2.5 and A2.2.6. The main service is the Orchestrator that receive Tosca templates from the Future Gateway portal and “decide” to which IaaS infrastructure (bottom right boxes - grey or blue) to send for execution or deployment.

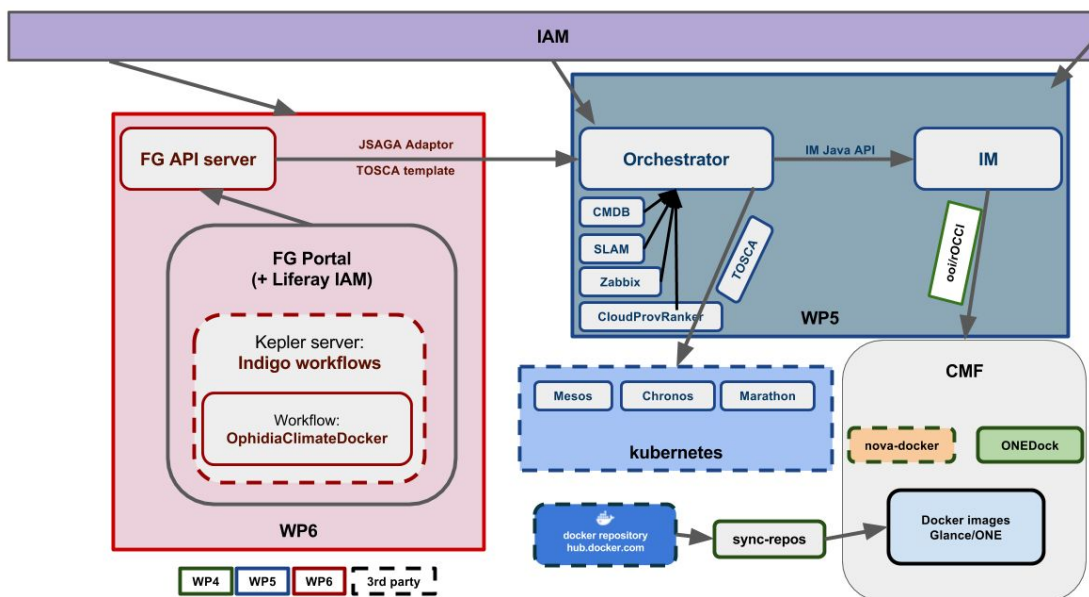


Figure 5.13: A use case of Indigo-DataCloud depicting several services and components that are deployed in the Pilot Preview Infrastructure.

Figure 5.14 is a detailed view of the PaaS (WP5) layer architecture including the Pilot Preview Infrastructure endpoints. The Orchestrator bases its “decision” with the help of several associated services:

- Zabbix: is a monitoring services of IaaS resources that include specific INDIGO-DataCloud probes.
- CMDB: is a services where IaaS resources are registered and hold information about those resources.
- SLAM: is a SLA Manager.



INDIGO - DataCloud

- CloudProviderRanker: ranks IaaS resources.

The Tosca template can be one of two main categories:

- **Application:** an application submitted for execution for a limited period of time (hours or days);
- **Service:** a user service or platform that runs for a long period of time (months or years).

The IaaS (WP4) computing and storage resources shown in Tables A2.2.2 and A2.2.3 correspond to sites and/or services where applications are actually executed or long running services are deployed by way of Tosca templates. These sites and services can be of several types:

- Cloud Management Frameworks (CMFs) such as OpenStack and OpenNebula (under study and test the possibility to use commercial/public providers such as AWS or Azure);
 - the CMFs can also support instantiation of docker containers: with nova-docker for OpenStack or ONEDock for OpenNebula;
- Docker management Frameworks such as Kubernetes, Mesos/Marathon, Chronos.

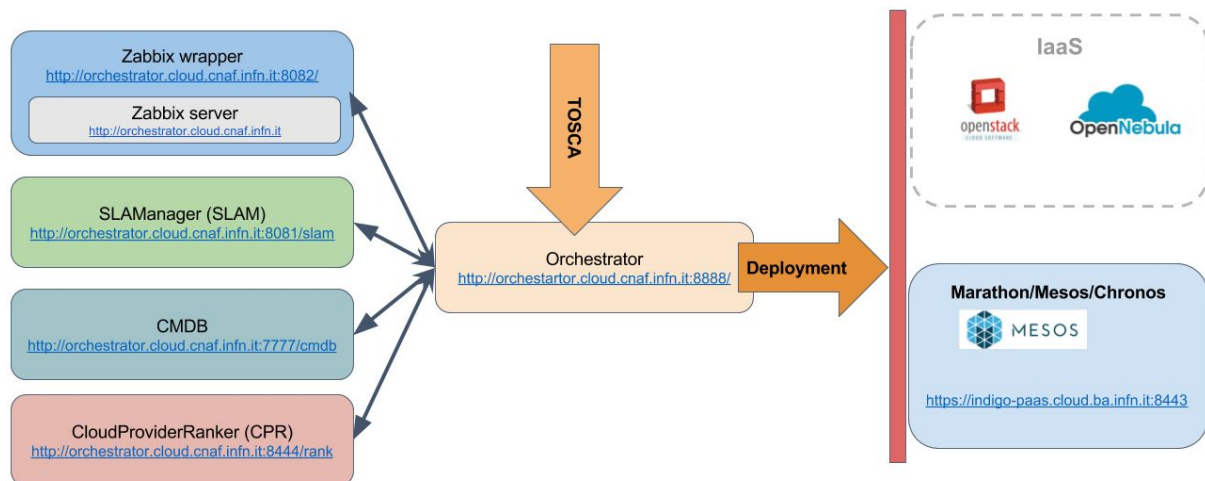


Figure 5.14: Architecture of the PaaS (WP5) layer of services including the endpoints of the Pilot Preview Infrastructure.

In these use cases, the user interacts only through the Web GUI frontend of the Future Gateway, and controls and manages the execution or deployments of docker containers (or VMs) through this GUI frontend. Access to the PaaS layer via a CLI is also possible.



6. EXPLOITATION

6.1 Planned activities and outcomes

The INDIGO-DataCloud exploitation activities aim to maximize the impact of the INDIGO-DataCloud software products by making them available, accessible and usable to users. Since the start of the project, the INDIGO-DataCloud project has been producing software, integrated solutions, appliances and services, which are aimed at being deployed in external production infrastructures used by the user communities. It is critical for the success of the project that the outputs of INDIGO-DataCloud are available in the infrastructures that are relevant to the users. The activities within this task aim to maximize (from a technical point of view) the uptake of the solutions provided by the project by the e-infrastructures. This is done by removing constraints that would prevent the users from exploiting the INDIGO-DataCloud software at the infrastructure level. For example, the output products need to be easily documented and deployable on the target e-Infrastructure and recognised and provided by service providers. Support channels need to be defined and agreed in terms of response time and timescales for resolving problems.

This section covers the work done by Task 3.4 (Exploitation) which includes the high level classification of INDIGO-DataCloud outputs, the setting up and activities of the Service Providers' Board (SPB) as well as work done in making the project outputs available to the EGI, currently the largest academic infrastructure, and the processes required to do this - notably Staged Rollout - the controlled deployment of new releases in a production environment.

6.2 High level products

A prerequisite for successful exploitation is easily understanding the different software components of INDIGO-DataCloud. There are over 32 software products in INDIGO-DataCloud, which is clearly too many for users and deployers to understand the main purposes of the software.

A key output of this activity has been to classify the diverse software outputs of INDIGO-DataCloud into different categories that can be easily understood by end users and deployers of the software. There are multiple valid ways that such a classification can be done; three different ways have been considered as part of this activity: high-level view, technical view and deployers' view of the software. The summaries of these classifications are included below:



INDIGO - DataCloud

6.2.1 High-level view of the software

Table 24: High level view of the software

Title	Description
User-level solutions	Portal and mobile toolkits to deliver scientific cloud computing and data functionality to users.
Automated solutions	Functionality enabling the the user to automate the selection of suitable resources and the running of workflows.
Data-solutions	The means to integrate data and computing while at the same time ensuring the ease of access, management and sharing of data.
Site-level solutions	Tools for site administrators to integrate with external infrastructures and services, maximizing the benefit to users in meeting their needs.

6.2.2 Technical view of the software

Table 25: Technical view of the software

Title	Description
Multi-tenant cloud infrastructure components	This allows for the deployment of a multi-tenant cloud infrastructure. Common services are shared by multiple users and communities and can be used to build new services/applications. They allow developers to orchestrate INDIGO-DataCloud Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) functionalities to create their domain-specific applications for the benefit of end users.
Gateway+UI components	This delivers user friendly Graphical User Interfaces for applications or for complex custom infrastructures. Solutions for both the Web and mobile applications are included.
Computing virtualization components	This delivers a complete, dynamic and cloud agnostic computing environment for application to run on.
Storage virtualization components	This delivers a POSIX-compliant file access to heterogeneous and distributed storage facilities to be used by applications running on the cloud. The scalable solution supports multiple data access protocols and like other INDIGO-DataCloud components, is designed to be cloud agnostic.



6.2.3 Deployers' View of the Software

Table 26: Deployers' view of the software

Type	Category	Services
Infrastructure as a Service	Authentication	TTS, IDH
	OpenStack	Synergy, OOI, OPIE, Nova-docker, TOSCA in Heat, Keystone AAI support
	OpenNebula	IM, IM Java API, ONEDock, Docker-hub sync
	Batch Farm	Partition Director, udocker
	Storage	Onedata, CDMI, dcache
Platform as a Service	Platform	Kubernetes, Zabbix Server Orchestrator, QoS/SLA, Onedata, Zabbix probes, IAM, Accounting, CloudProviderRanker, IM, Cloud Information Provider

This last category (Deployers' View) was primarily driven so that Service Providers can be presented with the software with a useful level of granularity suitable to deploy and satisfy different use cases for their different communities. As such, the deployer classification has been needed for the preparation of the Staged Rollout and organising the Early Adopters. These are explained more fully in Section 6.4.

6.3 Service Providers' Board

A key component in successful exploitation of the software outputs of INDIGO-DataCloud is successful engagement with Service Providers (SPs). The Service Providers' Board (SPB) is one of the advisory bodies of INDIGO-DataCloud. The primary purpose of this board is to have a direct communication channel with the Service Providers, who provide the resources to the user communities, to make sure that their technical requirements are addressed and that the software and platforms developed in INDIGO-DataCloud can run on their resources. The SPB however has also proven itself to be a useful forum to disseminate relevant status updates regarding the project, software development and testing to the SPs. The SPB also gives SPs a chance to ask questions relating to any of these topics.

During the first year of the project, the SPB was created and a number of subtasks (described below) were completed, with the SPB itself convening four times. An output of the SPB which has been particularly useful to the project has been to gather information on SP's infrastructure - specifically what cloud and storage solutions (and versions) are they using, and their future upgrade plans [R48]. This information, which is continuously being updated over the course of the project, enables the



INDIGO - DataCloud

project to align software support plans with the SPs' infrastructure upgrade plans - an important prerequisite for success. This information also means that the project is able to make informed decisions about how long to continue supporting older infrastructure versions.

A number of activities have been done as part of this work:

6.3.1 SPB Terms of Reference

The SPB Terms of Reference (ToR) [R49] define the purpose of the SPB, how it fits into the INDIGO-DataCloud project and logistics such as the method of appointment of membership and how the SPB meets. The ToR has been finalized and agreed with SPB members and PMB.

6.3.2 SPB Wiki

The SPB Wiki [R50] has been set up as a project within OpenProject and provides a top-level overview of the SPB and current membership. It is also the location where files such as the ToR are hosted.

6.3.3 SPB Mailing List

A mailing list, indigo-datacloud-spb@lists.indigo-datacloud.eu, has been established, consisting of all SPB members, for all communication relevant to SPs such as meeting planning, minutes, information about release status etc.

6.3.4 Membership and Future Direction of the SPB

Initially the SPB was designed to be a closed group for invited only participation. It was then decided that it is more beneficial to the project for the SPB and its documents to be completely open to anyone interested in participating. The concept of organisation membership of the SPB remains, but organisations can attend meetings prior to requesting formal membership.

6.4 Staged Rollout

The INDIGO-DataCloud Staged Rollout is a procedure based on EGI Staged Rollout [R51] whereby releases of the supported middleware are first deployed and tested by Service Providers who have agreed to be Early Adopter (EA) before being widely disseminated to all Service Providers through the production repositories. This procedure permits testing an update in a production environment and exposing the software to real end users. This environment is also more heterogeneous than what is possible during the certification and verification phases. The procedure allows for potential issues with an update to be discovered at a small scale and potential workarounds to be swiftly added to the release notes.



INDIGO - DataCloud

The following activities have been done for the preparation of Staged Rollout:

6.4.1 Staged Rollout Process Definition

The INDIGO-DataCloud Staged Rollout (SR) process has been defined on the WP3 wiki [R52] and agreed in preparation for the first release.

INDIGO-DataCloud SR is coordinated and led by the appointed SR Manager who is responsible for the adherence to the process for each release. A key part of this process is the completion and compilation of reports where the Early Adopters (EAs) respond to a series of standard questions covering the deployment process and any problems that they may have encountered.

6.4.2 Invitation for Early Adopters

Having defined the SR process and finalized the categorization of the INDIGO-DataCloud software from the point of view of the Deployer (see section 6.2) an invitation for SPs to become EAs was made. The list of EAs for the different categories of the software has been added to the WP3 wiki [R53]. The current list of EAs is constituted by sites that are members of the INDIGO-DataCloud preview infrastructure and/or members of the EGI federated cloud. Therefore there is a close collaboration with WP3.3.

6.5 Integration into the EGI Cloud Middleware Distribution (CMD)

To allow for easy deployment of INDIGO-DataCloud software onto the EGI Infrastructure, it should be made available also from the EGI Software Repository. It consists of the Unified Middleware Distribution (UMD) aimed mainly at grid-related software and the newer Cloud Middleware Distribution (CMD) specifically for components related to/integrated with/compatible with the OpenStack and OpenNebula cloud middleware frameworks. Since INDIGO-DataCloud produces cloud-based software, the natural target repository in EGI is the CMD.

Once software has successfully passed INDIGO-DataCloud Staged Rollout, it is eligible for direct inclusion into the CMD. This is due to the fact that the INDIGO-DataCloud SR is based on the EGI SR and provides the same level of assurance in terms of successful controlled deployment and testing in a production environment.

To allow for inclusion into the CMD, a Corporate-Level Technology Provider Underpinning Agreement [R54] has been defined and agreed between the INDIGO-DataCloud Technology Providers and EGI Foundation to specify information such as support hours and levels, responsibilities and contact information for the provider and customer. Much of the information in this document is provided on the EGI wiki, to facilitate it being updated in the future without needing to create and agree new versions of the Underpinning Agreement.



INDIGO - DataCloud



At the time of writing this deliverable, SR is well under way for the first release, with six components been successfully validated hence been ready for inclusion into the CMD. The first version of CMD for OpenStack (CMD-OS-1) is almost ready to be released and all components that have passed SR at the time of its release will be included in it.



INDIGO - DataCloud

7. KEY PERFORMANCE INDICATORS

Tables A.3.1.1 to A3.1.3 in Appendix A3, list the Key Performance Indicators (KPI) that measure the progress made toward the achievement of the introduced targets. Each KPI measures how close each PT is to the various quality criteria. As defined in deliverable D3.1, metrics are divided into 3 categories of applicability: mandatory (M), recommended (R) and optional (O). For more detailed information about each of the metrics please refer to deliverable D3.1 [R1].

7.1 Metrics

SQA analysis and compliance of the products that took part in the project's first release has been tracked by means of the following thirteen metrics, that correspond to the *mandatory* and *recommended* applicability, as defined in the deliverable D3.1 [R1]:

- Number of documented code style guidelines
- Adherence to documented code style guidelines
- Number of security vulnerabilities
- Number of functionalities tested
- Functional testing coverage
- Trend analysis
- Defects by severity
- Lines of code
- Comments frequency
- Developer documentation completeness
- Deployment and administration documentation completeness
- CLI/API documentation completeness
- Unit testing coverage

Concrete values for each of the metrics collected can be found in Tables A.3.1.1 to A3.1.3 in Appendix 3.

7.2 SQA Process KPIs

Table 29 summarizes the scope, definition and collecting frequency of the indicators that refer to SQA process management, namely:

- SQA adherence
- Automation testing index

Note that the statement *per reporting period* in the following definitions covers the period until the theoretical release of the present document (month 18). *Per release* matches the different releases (major and updates) occurred in the reporting period.



INDIGO - DataCloud

Table 29: Software Quality Assurance Process KPIs

KPI	Definition	Frequency
SQA Adherence	Percentage metric tracking adherence of each product to the SQA conventions and recommendations.	Per reporting period
Automation Testing Index	Percentage of automatic testing over the total of tests being performed in the pre-release or development phase (applies to code style, unit, functional and integration testing).	Per release

7.2.1 SQA Adherence

Tables A3.2.1 and A3.2.2 show the evolution of the products towards compliance with the INDIGO-DataCloud software quality criteria during the weeks prior and after the first release.

The darker (green) cells correspond to fulfillment of the criteria. As a result of the post-release documentation evaluation, the SQA adherence for some of the products lowered (week 9 to week 12). Products that constitute contributions to OpenStack project do not have a numeric metric and appear as “NA” (not applicable) as their quality criteria is different. This table was compiled from WP3 SQA Products Status wiki [R55].

7.2.2 Automation testing index

Table A3.2.3 showcases the maturity of automatic validation for each product within this period. Automated testing is performed exclusively by the continuous integration service for each change to be merged in the production source code of the individual product repositories.

Integration testing has been entirely performed manually during this period due to the high complexity of the architecture. However it is reflected in the metric calculation as it was defined as part of the automatic testing in the deliverable D3.1 [R1].



7.3 SQA Schedule KPIs

Table 33: Software Quality Assurance Schedule KPI

KPI	Definition	Frequency	Value
Release Frequency	Number of releases	Per reporting period	4
Release Efficiency	Number of release rollbacks	Per reporting period	0

7.4 SQA Quality KPIs

Table 34: Software Quality Assurance Quality KPIs

KPI	Definition	Frequency	Value
Defect Removal Efficiency	Percentage of total defects that have been found while testing and are fixed when a new software release is going live	Per release	Major release: 90.91% 1st update: 94.74% 2nd update: 100% 3rd update: 100%
Defect Removal Resolution Time	Average time of defect fixes in development after notification	Per reporting period	12.47 days
Rejected Defect Ratio	Defects rejected by developers team over the total of defects found	Per release	Major release: 2:33 1st update: 1:19 2nd update: 1:7 3rd update: 0:1
Escaped Defect Ratio	Defects found by customers post a release over the total of defects found	Per release	Major release: 0:33 1st update: 0:19 2nd update: 0:7 3rd update: 1:1
Feature Testing Time	How long a feature is tested	Per release	N/A



INDIGO - DataCloud



Incident Trend Analysis	Number of incidents - number, categories and submission time of the incidents escalated to the INDIGO-DataCloud Support Units, including the generic Support Unit, are extracted from GGUS	Per reporting period	2 1 incident 1 change request
Incident Resolution Time	Average time for resolving an incident by the 3 rd -level support	Per reporting period	48.93wd *

(*) - as extracted from GGUS system.



INDIGO - DataCloud

8. RISK ASSESSMENT

Deliverable D3.1 contains an assessment of risks associated to the WP3 initial plan and related mitigation measures. Table 8.1 shows the occurrences during the first period and the measures taken to address them.

Table 8.1: Risk occurrence

Description of risk	Risk occurrence
Name: Withdrawal of key partners or experts. Probability: Low Impact: Key competence in WP3 can be lost. Lowering the support and maintenance of software and/or services and pilot infrastructures. Lowering manual integration testing, thus quality of software.	Didn't occur.
Name: Effort needed is underestimated Probability: Medium/Low Impact: Delays in performing SQA, setting up services, in their operation, setting up and support the testing infrastructure, any manual integration testing, response to issues and problems.	Didn't occur
Name: Insufficient production quality of the deployed infrastructures of our partners. Although this operational aspect is not part of the project itself, it significantly contributes to its success. Probability: Low Impact: Problems with the deployments and validation.	Didn't occur
Name: Software developer teams not complying fully with the SQA procedures Probability: Medium Impact: Lower quality of INDIGO-DataCloud software with potential for increased cost and reputation damage of fixing bugs and responding to security vulnerabilities after release	A small number of PTs took longer to comply with the SQA procedures. Monitoring and follow-up of specific issues through the EMT addressed the issue.
Name: Delay in software release with respect to previously agreed schedule Probability: Medium Impact: Delay in the adoption of INDIGO-DataCloud software by the Resource Infrastructures and user communities. Delay in deployment of important features or bug fixes in production infrastructures.	The first software release was issued with a very small delay of one week. The delay was too short to have any impact.
Name: Change of policy or usage of one or more public free (cloud) services used in task WP3.3 to a more restricted usage or paid fee. Probability: Low Impact: Drop of the service. Delays due to migration to INDIGO-DataCloud provisioning those services. More human resources to deploy and operate in-house services.	No change of policies or restrictions. One public service had a limit that didn't apply to open source projects such as INDIGO-DataCloud and removed the limit upon request of WP3.



INDIGO - DataCloud

<p>Name: Technical problems using one or more public free (cloud) services used in task WP3.3. Slow response to issues raised by INDIGO-DataCloud.</p> <p>Probability: Medium/Low</p> <p>Impact: Delays in using the service, failures in the WP3 processes. Delays due to migration to INDIGO-DataCloud provisioning those services. More human resources to deploy and operate in-house services.</p>	<p>The above mentioned service took several days to provide an answer.</p> <p>Caused a small delay in creating documentation for a product.</p>
<p>Name: Lack of engagement of service providers and user communities.</p> <p>Probability: Medium/Low</p> <p>Impact: Lower adoption of INDIGO-DataCloud software products, difficulties in the WP3 exploitation task.</p>	<p>Didn't occur. Although the software was recently released, there is already strong interest by the user communities in WP2 and by infrastructures.</p>
<p>Name: Difficulties to implement/apply parts of the plan due to the INDIGO-DataCloud heterogeneous environment and rich architecture.</p> <p>Probability: Medium/Low</p> <p>Impact: Delays in applying the process to the involved software components. Higher consumption of resources than foreseen to operate the WP3 processes.</p>	<p>The deployment of the preview infrastructure took more time than initially foreseen as it was the first time the software was being fully deployed in a production-like environment.</p>

In addition to the risks identified in the Deliverable 3.1 a set of new security related risks and mitigation measures has been identified and is shown in the Table 8.2.

Table 8.2: Additional risks

Description of risk	Proposed risk mitigation
<p>Name: Software fails to deliver Authentication methods and Authorization granularity or levels of assurance that new user communities require to implement their security policy.</p> <p>Probability: Low</p> <p>Impact: Medium - weaker privacy or security controls or lack of future community adoption.</p>	<p>Consulting with a variety of user communities and infrastructures to ensure that the software will fulfill the requirements.</p> <p>INDIGO-DataCloud already includes a wide scope of use cases and communities.</p>
<p>Name: Software fails to implement appropriate logging due to insufficient requirements or configuration, is thereby unable to implement policy on traceability and accountability,</p> <p>Probability: Medium</p> <p>Impact: Medium - failure to be able to co-operate in security incident response.</p>	<p>Traceability testing as part of preview infrastructure..</p>
<p>Name: Software fails to provide adequate controls for the protection of personal data provided by users to enable their usage of the system.</p> <p>Probability: Low</p> <p>Impact: Low - For the research community such personal data is usually limited to identifiers such as email address and institutional affiliation which are generally made public in other contexts.</p>	<p>Consulting with a variety of user communities and infrastructures to ensure that the software will fulfill the requirements.</p> <p>INDIGO-DataCloud already includes a wide scope of use cases and communities.</p> <p>Assess the software security.</p>



9. PLANS FOR THE SECOND PERIOD

This section introduces the WP3 plans for the second project period that aim at activities and results beyond or complementing the WP3 initial plan defined in deliverable D3.1 [R1].

9.1 Software quality assurance

Automation deployment

The automation of software deployment will be pursued. The mandatory requirement on the usage of open-source software configuration management tools will be enforced. The objective will be easing and simplifying the deployment and configuration of the products. Whenever possible the processes within the SQA itself will be further automatized.

SQA and Security

Focus will be put in investigating the use of additional static and dynamic security testing tools initially based on the SWAMP [R12] static analysis packaging. SWAMP provides a wide range of open-source and commercial software security testing tools, and comes in two forms:

- MIR-SWAMP [R13]: a cloud platform that allows security assessments to be easily performed;
- SWAMP-in-a-box [R14]: an open-source distribution that can be deployed and customized.

SWAMP will be used to conduct security assessments of the INDIGO-DataCloud software. Its integration in the automated SQA testing within the Jenkins continuous integration will be evaluated.

SQA and the second release

The software quality assurance of the second INDIGO-DataCloud major release will constitute a major activity during the second project period. The objective will be the improvement of the overall quality of the products beyond the results already achieved in the first release.

9.2 Software release and maintenance

The software maintenance activity will focus on timely release of *corrective* (bug fixes) and *adaptive* (new/versions of operating systems and/or cloud management frameworks) changes as well as



INDIGO - DataCloud

perfective and *preventive* ones, by defining the roadmap and coordinating the activities heading to the INDIGO - DataCloud second release, following the defined procedures.

- The support for a new version of the OpenStack CMF, the **Mitaka** one, is already planned, and it will be provided through one of the INDIGO-1 updates to come in the next period. Software package repositories are prepared to be ready to host the new versions of the components.
- Updates of INDIGO-1 release will continue to be provided on a monthly manner taking in consideration that:
 - The full support period will end at the end of January 2017, so that no new features will be added to INDIGO-1, but moved, eventually to INDIGO-2 release
 - The standard support period will end at the end of March 2017, in concomitance with the release of the projects' second major release.
 - No more changes, bug fixes, will be introduced after March. Just support for fixing security vulnerabilities will be provided for 2 more months, until the end of May 2017.
 - After May 2017 - INDIGO-1 repositories will be closed, and the only supported components will be the one present in the INDIGO-2 release.
 - Update mechanisms will be provided for all components and services

The software support activities will concentrate on following closely the different user communities and e-infrastructure sites that will use and deploy projects solutions, ensuring that all their requests for support, for changes/improvements are addressed and correctly redirected to the right development teams, in case proven to be bugs in the released software.

Roadmap to INDIGO-2 (ElectricIndigo)

The activity of preparation of the second projects' release started with the definition of the schedule and the mandatory new operating systems and CMF versions to be supported.

Regarding the operating systems and CMFs it was decided, through discussions in the EMT meetings, that:

- The supported Operating Systems platforms will be:
 - Mandatory
 - CentOS 7
 - Ubuntu 16.04
 - Optionally PTs can support other OS platforms
- The supported CMF (Cloud Management Framework) versions will be:
 - Mandatory
 - OpenStack v. Newton,
 - OpenNebula v. 4.14 /5.2

After discussions with the Technical Director and the JRAs WP leaders, the schedule, present in the table below, was defined for the INDIGO-2 release. It was presented for approval to the Technical



INDIGO - DataCloud

Board and EMT. No comments were raised so that the schedule was published in the WP3 wiki pages, and the activity of following it up started. In the next period:

- the complete list of the components and services and their new features will be made available
- Tasks will be opened in order to track the
 - Development of the new features
 - Requests for development resources, in case needed, for the new OS and CMFs versions
 - SQA criteria compliance fulfilment
 - Release of the various components, containing references to the availability of artefacts (packages, containers) and also to the issues to be addressed and improvements to be made for the second release

Task/Milestone	Start	Length	Date
Planning and Development	Day after the end of the Planning phase for the previous GA Release	More than 5 months	09.08.2016
Feature Submission Deadline	3-4 months before the Feature Freeze	n/a	02.01.2017
Feature Freeze	2 month before the RC	Until GA	31.01.2017
Final Change Deadline (code freeze)	End of Testing and Certification	Until GA	22.02.2017
Release Candidate	End of Testing and Certification Start Acceptance Testing 2 months before GA-release day 3 days after the Final Change Deadline	Until GA	27.02.2017
Compose Final RC	Same as Release Candidate	1 week	01.03.2017
Technical-Previews	1 month before the GA	Should last 2 weeks	06.03.2017 (Monday)
Test Final RC	2 weeks before GA	6 days	20.03.2016 (Monday)
GA Release	Release Date	n/a	31.03.2017 (Friday)
Maintenance	GA release day	~10 months	31.03.2017
End of Life	10 months after the GA of current release	n/a	31.01.2018

Status and eventual issues will be discussed during periodic EMT meetings, and actions will be taken in a timely manner in order to avoid and prevent unexpected delays.

Addressing security vulnerabilities

After defining the needed policies and procedures on how to address security vulnerabilities, as mentioned in section 4.4, during the next period through the INDIGO-DataCloud Security Task Force



INDIGO - DataCloud

a direct communication channel will be opened with EGI-SVG group in order to update them, if needed, to match the requirements coming from the latest version of the EGI Strategy and Vulnerability Handling Procedure [R47]. In case of security incidents detected and assessed by the EGI-SVG group as affecting INDIGO - DataCloud software, all the necessary actions will be taken in order to ensure the availability, release, of a fixed version of the affected component or service, according to their assessed risk-level.

9.3 Pilot services

Infrastructure and software security

The deployment of the released products in the preview infrastructure in production-like conditions, using resources provided by production infrastructures, is imposing an additional focus on security. However, the deployment of the INDIGO-DataCloud in the preview infrastructure also opens new opportunities to assess and validate the software security. The WP3.3 security related plans include:

- work with infrastructure providers such as EGI to adjust procedures and understand the security aspects of introducing novel PaaS services into production;
- use scanning and penetration testing tools to look at live instances, in the preview infrastructure, in order to evaluate the security of the underlying software and related deployment configurations;
- perform traceability testing to ensure that there is a clear and complete path back up through the infrastructure to an authenticated user for his actions.

Consolidation of the Pilot Preview Infrastructure

Keep the preview infrastructure operational and updated with the released components. Complete the setup of the use cases using the components from the SaaS, PaaS and IaaS layers. The architecture depicted in Figures 5.13 and 5.14 of section 5.5, shows that INDIGO-DataCloud services and components offer a very good coverage spanning the three “aaS” layers, together with the “Data Management” services. The architecture and the components will be further tested and validated with the use cases. The integration of the use cases will be followed closely, WP3 will provide assistance and bridge with the developers.

Inclusion of public cloud providers in the Pilot Preview Infrastructure

T-Systems pledged IaaS and Kubernetes resources to WP3 and tests have already started in order to demonstrate how resources from public cloud resources can be exploited using the INDIGO-DataCloud software. In a first phase the focus will be the integration of IaaS resources. In a second phase, the plan is to use these resources to host other services of the PaaS or even SaaS layer. There are already contacts and discussions with other European Public Cloud providers to establish similar cooperations.



INDIGO - DataCloud

Plan and implement elasticity tests and stress tests

Use the deployed use cases to test elasticity and stress test the released software in the preview infrastructure. Although the Pilot Preview infrastructure has a limited set of resources, it can nonetheless and to its limit be used for elasticity tests such as multiple use cases being executed in parallel. This will allow to measure how the IaaS layer behaves to the demands from applications. Also will allow to check the behaviour of the SaaS and PaaS response to multiple request from different users and patterns. In addition service specific tests such as the ones used for integration testing or others made available by the developers will be explored to complement the use cases.

9.4 Exploitation

Expand exploitation activities

For the remainder of the project, WP3.4 is actively looking to increase participation to the SPB by including more commercial companies and small and medium-sized enterprises (SMEs). It is planned for the SPB to be instrumental in assisting new adopters of INDIGO-DataCloud outputs while trying out the software. This can be done by technical experts within the project assisting new adopters to set up testbeds. These activities can be reported back at SPB meetings for the benefit of other attendees.

Collaboration and agreements with service providers both within the project and beyond will be pursued. The objective for the second period is to promote the adoption of the INDIGO-DataCloud software by the widest possible range of providers.

Staged Rollout and Integration into the EGI Cloud Middleware Distribution

The Staged Rollout of INDIGO-DataCloud products will continue aiming at software validation for inclusion in the EGI Cloud Middleware Distribution. New versions of the products will be made available in the second release, and again these versions will be deployed in the preview infrastructure and in other locations under the SR.

Common exploitation activities

Continue collaborating with WP2 research communities and WP1 project management, exploring synergies on common exploitation activities and establish a common exploitation plan.



10. CONCLUSIONS

The WP3 plan defined in deliverable D3.1 has been successfully implemented. The processes and services operated by WP3 have enabled the development and release of the INDIGO-DataCloud software and its subsequent validation using real use-cases in a production like environment provided by the WP3 preview infrastructure, these activities will continue aiming at further validate the software its robustness and scalability. The first release has already succeeded in enabling several user-cases across scientific domains, with products are already being exploited by users. In addition several components have been accepted and merged as contributions in relevant projects and frameworks such as OpenStack, OpenNebula and others.

The first INDIGO-DataCloud software release is now under maintenance and will be followed in 2017 by the second release codenamed ElectricIndigo. Besides the software enhancements the second release will include support for newer versions of the supported operating systems and cloud management frameworks. Meanwhile the software development process will be further improved with increased automation both internally in WP3, and in the software products through the SQA automated deployment requirements.

A project wide security group has been created to manage security incidents, software vulnerabilities, establish policies, procedures and work towards improving the security of INDIGO-DataCloud software and services. The security group is being established under WP3 having a majority of members from WP3 including the SQA manager, release manager, infrastructure and services manager, and WP3 manager, among other key project members.

WP3 has successfully engaged with the user communities via WP2, and with infrastructure providers through the exploitation activities. In this context a Service Providers Board has been established including providers external to the project. The service providers board is now being enlarged to other providers both academic/scientific and commercial. Collaboration with external commercial providers has also been started aiming at deployment of the software in these infrastructures for testing and future adoption.

An agreement has been established with EGI for the exploitation of the INDIGO-DataCloud software in the EGI infrastructure, which will surely contribute to the adoption of the project developments. During the second period, continued integration in the EGI Cloud Middleware Distribution and in other infrastructures will be pursued.



INDIGO - DataCloud



11. REFERENCES

R1	D3.1 - Initial plan for WP3 https://owncloud.indigo-datacloud.eu/index.php/s/yDklCrWjKnjutVA
R2	MidnightBlue INDIGO-DataCloud first release https://www.indigo-datacloud.eu/news/first-indigo-datacloud-software-release-out
R3	Bradner, S. "RFC 2119." 1997: https://www.ietf.org/rfc/rfc2119.txt
R4	Technical Annex of the INDIGO-DataCloud project proposal
R5	Docker containerization platform https://www.docker.com/
R6	GGUS https://ggus.eu/
R7	INDIGO-DataCloud support unit https://wiki.egi.eu/wiki/GGUS:INDIGO_DataCloud_Catch-all_FAQ
R8	GitHub Pull Requests https://help.github.com/articles/creating-a-pull-request/
R9	Gerrithub https://review.gerrithub.io/
R10	MetricsGrimoire https://metricsgrimoire.github.io/
R11	vizGrimoire http://vizgrimoire.bitergia.org/
R12	SWAMP Software Assurance Marketplace https://continuousassurance.org/
R13	MIR-SWAMP https://continuousassurance.org/mir-swamp/
R14	SWAMP-in-a-box https://continuousassurance.org/swamp-in-a-box/
R15	ISO/IEC 14764:2006 (Software Engineering — Software Life Cycle Processes — Maintenance)
R16	WP3-INDIGO-PTs https://owncloud.indigo-datacloud.eu/index.php/s/6jZlhjvaXg4JNxM
R17	Indigo_pts.xls https://docs.google.com/spreadsheets/d/1ut1ZnMu0u_1dH9_TucRw2W8NcgymrI9xcuTQ_-0txAiA/edit#gid=0
R18	Open Cloud Computing Interface (OCCI) - http://www.occi-wg.org/
R19	OpenStack http://www.openstack.org/
R20	OpenGridForum (OGF) https://www.ogf.org/ogf/doku.php
R21	OpenNebula https://opennebula.org/
R22	OpenStack Preemptible Instances https://blueprints.launchpad.net/nova/+spec/preemptible-instances



INDIGO - DataCloud

R23	Heat Orchestration Template (HOT) http://docs.openstack.org/developer/heat/template_guide/hot_spec.html#hot-spec
R24	Nova-Docker https://wiki.openstack.org/wiki/Docker
R25	OpenStackClient http://docs.openstack.org/developer/python-openstackclient/
R26	OpenStack Nova API http://developer.openstack.org/api-guide/quick-start/index.html
R27	CLUES https://github.com/grycap/clues
R28	HTCondor https://research.cs.wisc.edu/htcondor/
R29	Apache Mesos http://mesos.apache.org/
R30	Drools framework http://drools.org/
R31	Kubernetes Cluster http://kubernetes.io/
R32	TOSCA Simple Profile http://docs.oasis-open.org/tosca/TOSCA-Simple-Profile-YAML/v1.0/csprd01/TOSCA-Simple-Profile-YAML-v1.0-csprd01.html
R33	Simple API for Grid Applications https://en.wikipedia.org/wiki/Simple_API_for_Grid_Applications
R34	WP3 - GitBook - HowTo https://owncloud.indigo-datacloud.eu/index.php/s/9c5JxCkdGvSNss2
R35	INDIGO-DataCloud EMT ToR https://owncloud.indigo-datacloud.eu/index.php/s/65mU1WmzduUZG8b
R36	EGI Unified Middleware Distribution http://repository.egi.eu/
R37	EGi Cloud Middleware Distribution https://wiki.egi.eu/wiki/EGI_Cloud_Middleware_Distribution
R38	D2.1 - Initial requirements from research communities https://owncloud.indigo-datacloud.eu/index.php/s/ILNAczJNBnLmLLG
R39	D1.8 - INDIGO-DataCloud General architecture http://owncloud.indigo-datacloud.eu/index.php/s/xKna8rbHq3VL5j
R40	D4.1 – Software Design Document for WP4 https://owncloud.indigo-datacloud.eu/index.php/s/4g9rCo9Cr20xHDW
R41	D4.2 – Detailed Work Plan for the duration of the project https://owncloud.indigo-datacloud.eu/index.php/s/a1MAm5x7oZ2k2Q9
R42	D5.2 - Design Document and Work Plan for the PaaS architecture https://owncloud.indigo-datacloud.eu/index.php/s/li6wUvGX4QRGd7U
R43	D6.1 – Software architecture and work plan for WP6 https://owncloud.indigo-datacloud.eu/index.php/s/Dy9kOxibpezkUik
R44	INDIGO-DataCloud - First release announcement https://www.indigo-datacloud.eu/news/first-indigo-datacloud-software-release-out



INDIGO - DataCloud



R45	INDIGO-DataCloud - Addressing Security Vulnerabilities https://owncloud.indigo-datacloud.eu/index.php/s/U7CA186wyUee5oA
R46	EGI Security Vulnerability Group https://wiki.egi.eu/wiki/SVG:SVG
R47	EGI Strategy and Vulnerability Issue Handling Procedure https://documents.egi.eu/public/ShowDocument?docid=2538
R48	Service Providers Infrastructure and versions https://owncloud.indigo-datacloud.eu/index.php/s/00OkN2e3zmoierf
R49	SPB Terms of Reference (ToR) - add owncloud link
R50	SPB wiki - https://owncloud.indigo-datacloud.eu/index.php/s/VmoE1C5O51GHKNK
R51	EGI Staged Rollout procedure https://wiki.egi.eu/wiki/Staged_Rollout
R52	SR wiki https://owncloud.indigo-datacloud.eu/index.php/s/TEmOXEuYOTD4DaH
R53	Early Adopters wiki https://owncloud.indigo-datacloud.eu/index.php/s/bfc0nnQRQkwceuj
R54	Underpinning Agreement
R55	SQA Products status https://owncloud.indigo-datacloud.eu/index.php/s/sEyA1US5zNhpWTJ
R56	Procedure to request resources for SW development from PTs https://owncloud.indigo-datacloud.eu/index.php/s/vTAf2WEEyiS2Ys5
R57	INDIGO Champions meet Developers https://agenda.indigo-datacloud.eu/conferenceDisplay.py?confId=93



INDIGO - DataCloud

A1. APPENDIX 1

Table A1.1: WP4 - IaaS components

Component	Description
dCache	Large-scale storage system with support for multiple QoS levels.
Token Translation Service (TTS)	Service that offers an easy way to self service credentials by the users.
Identity Harmonization (IDH)	Service providing a RESTful web service to link and unlink multiple local user accounts.
CDMI server with QoS	Reference implementation of the SNIA Cloud Data Management Interface (CDMI), extended to support Quality-of-Service (QoS) and Data Life-cycle (DLC).
OpenStack OCCI Interface	Implementation of the Open Cloud Computing Interface (OCCI) [R18] for OpenStack [R19].
pOCCI	OCCI standard compliance testing tool.
rOCCI	Framework that allows management of resources in arbitrary cloud management frameworks through the OCCI standard.
jOCCI	Suite of Java libraries to facilitate use of the OCCI (Open Cloud Computing Interface) protocol as standardized by OGF (the Open Grid Forum) [R20].
Synergy	FairShare Scheduler for OpenStack - new extensible general purpose management OpenStack service.
ONE_fs_sched	FairShare Scheduler for OpenNebula [R21].
OpenStack Preemptible Instances Extension (OPIE)	The materialization of the preemptible instances [R22] extension serving as a reference implementation.
OpenStack Identity Authentication Library	Keystoneauth - provides a standard way to do authentication and service requests within the OpenStack ecosystem.
Keystone AAI support	A guide on how to configure OpenStack Keystone to



INDIGO - DataCloud

	support the INDIGO-DataCloud Identity and Access Management (IAM) OpenID Connect based authentication.
Heat-Translator	An OpenStack project, with contributions through the INDIGO-DataCloud project, a command line tool which takes non-Heat templates as an input and produces a Heat Orchestration Template (HOT) [R23] which can be deployed by OpenStack Heat.
Nova-Docker	Provides detailed documentation on how to enable the use of nova-docker [R24] component in OpenStack based infrastructures. Nova-Docker is supported by INDIGO-DataCloud.
ONEDock	Project intending to provide support for OpenNebula to create Docker containers, and deliver them to the end user as if they were Virtual Machines.
OpenStack Client (OSC)	New version of the OSC [R25] adding OpenStack support for INDIGO-DataCloud developments (AAI, preemptible instances, etc).
OpenStack NOVA Client	New version of the client for the OpenStack Nova API [R26] adding support for INDIGO-DataCloud developments (AAI, preemptible instances, etc).
java-RepoSync	Component used to synchronize Docker images from DockerHub to instances of either OpenStack or OpenNebula with Docker support already configured.
Partition Director (DynPart)	Component to ease management of a hybrid data center, where both Batch System based and cloud based services are provided.
udocker	A basic user tool to execute simple docker containers in user space without requiring root privileges. Can be used in conventional batch systems, HPC, grid or interactive Linux clusters.

TableA 1.2: WP5 - PaaS components



INDIGO - DataCloud

Component	Description
Infrastructure Manager (IM)	Tool that deploys complex and customized virtual infrastructures on IaaS Cloud deployments (such as AWS, OpenStack, etc).
CLUES (CLUster Energy Saving) INDIGO Extensions	New plugins provided as extensions for CLUES [R27] to support the PaaS Orchestrator and to introduce elasticity capabilities to both HTCondor [R28] and Apache Mesos [R29] (including its frameworks Chronos and Marathon).
CloudProviderRanker	A standalone REST WEB Service which ranks cloud providers basing on rules implemented with the Drools framework [R30].
CloudInfoProvider	Service generating a representation of cloud resources, to be published inside INDIGO-DataCloud Configuration Management DB.
QoS/SLA Management Service	Service allowing the handshake between a user and a site on a given SLA. Gives the Orchestrator/Ranker useful information to make decisions on tasks scheduling according to valid SLAs.
Orchestrator	Orchestrator of the PaaS layer, core component of the INDIGO-DataCloud project. It receives high-level deployment requests and coordinates the deployment process over the CMFs and Mesos.
Onedata	Global data management system, providing easy access to distributed storage resources, supporting wide range of use cases from personal data management to data-intensive scientific computations.
Accounting	REST API for APEL
Kubernetes	Provides documentation on how to setup a Kubernetes Cluster [R31] together with a shell



INDIGO - DataCloud

	script and Ansible playbook for installation.
Mesos Cluster	Documentation, Ansible roles and TOSCA templates for cluster set-up featuring high-availability, service-discovery and load-balancing; Integration with the INDIGO-DataCloud Orchestrator.
Zabbix Probes	Part of the INDIGO-DataCloud Monitoring Framework - Zabbix probes for OCCl and Heapster.
Identity and Access Management Service (IAM)	Service Providing a layer where identities, enrollment, group membership and other attributes and authorization policies on distributed resources can be managed in an homogeneous way, supporting the federated authentication mechanisms supported by the INDIGO-DataCloud AA.
TOSCA Parser	OpenStack project - developed to parse TOSCA Simple Profile [R32] in YAML.
TOSCA Types	New types/TOSCA templates added in INDIGO-DataCloud to extend TOSCA Simple Profile in YAML Version 1.0 to add high level entities.

Table A1.3: WP6 - SaaS components

Component	Description
LiferayIAM	Set of modules for authentication using INDIGO-DataCloud IAM service.
Ophidia	Research project on big data analytics for eScience. It provides a framework for parallel I/O and data analysis, an array-based storage model and a hierarchical storage organization to partition and distribute multidimensional scientific datasets.



INDIGO - DataCloud

FutureGateway components	Set of software components able to build, or assist existing web portals or other community oriented interfaces to become Science Gateways: API Server, API Server Daemon, Portal Setup, jSAGA adaptors (TOSCA and rOCCI), fgTools.
jSAGA ResourceManager API	An extension of the SAGA specification / JSAGA implementation [R33] - it allows to acquire and manage resources (compute, storage, network) and use them to run job or manage data.
INDIGO-DataCloud Plugins for scientific workflow systems	Plugins to be used in order to allow, already available workflow manager, to exploit the INDIGO-DataCloud WP6 APIs: INDIGO-DataCloud Kepler.
INDIGO-DataCloud OpenMobile Toolkit (OMT)	Android Library Project which simplifies access to INDIGO-DataCloud API.



INDIGO - DataCloud

A2. APPENDIX 2

A2.1 Integration Infrastructure

The tables A2.1.1 to A2.1.4 show the definition of tests performed in the integration infrastructure. The tests address integration or interaction between components and/or services. Each table correspond to a given development WP, or set of WPs when the tests target interactions between IaaS and PaaS layers.

Table A2.1.1: The table show the tests performed between WP4 components.

Name	Test description	Task(s)	Product	Provider	Tester
nova-docker deployment with OpenStack Liberty	Ensure that nova-docker integrates correctly with OpenStack Liberty	4.1.1	nova-docker	CSIC	Alvaro Lopez
				LIP	Mario David
ONEDock deployment with OpenNebula	Ensure that ONEDock integrates correctly with OpenNebula	4.1.1	OneDock	UPV	Carlos de Alfonso, German Molto
				LIP	Mario David
ooi support for containers	Ensure that ooi is able to manage containers with no differences with VMs	4.1.3	ooi	INFN-PD	Alvise Dorigo, Alvaro Lopez, Massimo Sgaravato
				CSIC	Alvaro Lopez
				LIP	Mario David, Jorge Sevilla
rOCCI support for containers	Ensure that rOCCI is able to manage containers with no differences with VMs	4.1.3	rOCCI	UPV	Carlos de Alfonso, German Molto
Docker-Hub OpenStack synchronization	Ensure that java-syncrepos is pulling the containers whenever a hook is sent, and that it is uploading it to OpenStack glance	4.1.2	java-syncrepos	LIP + Atos	Jose Sanchez (Atos)
				CSIC	Alvaro Lopez
				CSIC	Fernando Aguilar
				CSIC	Alvaro Lopez
				CSIC	Fernando Aguilar
Docker-Hub OpenNebula synchronization	Ensure that manual sync (triggered by admin) is possible			LIP + Atos	Jose Sanchez (Atos)
				LIP + Atos	Jose Sanchez (Atos)
				UPV + Atos	Jose Sanchez (Atos)
				LIP + Atos	Jose Sanchez (Atos)
				UPV + Atos	Jose Sanchez (Atos)
Synergy support for containers	Ensure that Synergy is able to manage containers as well as VMs	4.1.4	Synergy / nova-docker	INFN-PD	Lisa Zangrando, Massimo Sgaravato
OPIE support for containers	Ensure that OPIE can manage containers as well as VMs	4.1.4	OPIE	CSIC	Alvaro Lopez
IAM integration with OpenStack Keystone	Ensure that OpenStack Keystone can consume the INDIGO IAM OpenID credentials and return a proper Keystone token	4.0.2	Keystone	CSIC LIP	Alvaro Lopez Mario David



INDIGO - DataCloud

IAM integration with OpenStack Keystone libraries	Ensure that the OpenStack CLI tools are able to authenticate users against Keystone using the IAM	4.0.2	keystoneauth	CSIC	Alvaro Lopez
---	---	-------	--------------	------	--------------

Table A2.1.2: The table show the tests performed between WP4 and WP5 components.

Name	Test description	Task(s)	Product	Provider	Tester
Ensure that the user AuthN/AuthZ works properly	Ensure that the AuthN/AuthZ works properly. User credentials from WP5 should end in WP4 as well. This test needs to be performed for all the services that interact with WP4 resources.		IM	UPV / REPLY	Miguel Caballer / Alberto Brigandi
			Orchestrator	REPLY	Alberto Brigandi
			Keystone		
java-syncrepos OpenStack metadata generation	Ensure that java-syncrepos is putting all the required metadata into the uploaded images	4.1.3	java-syncrepos	ATOS	Fernando Aguilar
Information system image metadata publishing for OpenNebula	Ensure that the Information System is publishing the required metadata for all the WP5 components	4.0.X/5.X	Information System		
IM integration with public Clouds	Ensure that IM is able to deploy on public Clouds	5.1	IM	UPV	Miguel Caballer
Orchestrator - IM (OpenNebula)	Ensure that the Orchestrator is able to deploy a TOSCA Template on an OpenNebula site via the IM	5.1 / 4.1	Orchestrator	UPV / REPLY	Alberto Brigandi / Miguel Caballer
Gathering Accounting Information from Sites (Open Stack)	Ensure that Accounting obtains proper expected information from the testbed sites	5.1 / 4.X	Accounting	STFC	Greg Corbett
IAM + OneData Integration	Authenticate to OneData with IAM	5.1 / 5.xxx	IAM, OneData	Cyfronet	Lukas Dutka
CDMI + OneData Integration	Files stored with oneData can be QoSsed (query/modify) with CDMI	4.2 / 5.xxx	CDMI, OneData	KIT, Cyfronet	Benjamin Ertl Lukas Dutka
IAM + TTS Integration	Authenticate TTS (Web + REST) with IAM	4.0 / 5.1	IAM, TTS	KIT	Bas Wegh
IAM + CDMI QoS Integration	Authenticate to CDMI-QoS with IAM	4.2 / 5.1	IAM, CDMI	KIT	Benjamin Ertl

Table A2.1.3: The table show the tests performed between WP5 components.

Name	Test description	Task(s)	Product	Provider	Tester
Orchestrator - CloudProviderRanker	Ensure that the Orchestrator obtains a ranked list of providers from the CloudProviderRanker	T5.1/T5.3	Orchestrator / CloudProviderRanker	REPLY	Alberto Brigandi
TOSCA + OneData	Ensure that a user-specified OneData space in a TOSCA template gets mounted on a customised virtual infrastructure (of VMs)	T5.1/T5.4	OneData	UPV	Miguel Caballer / OneData Team
Orchestrator - Mesos	Ensure that the submitted job request is correctly handled: data are staged-in/out	T5.1/T5.3/T5.4	Orchestrator/Mesos/Onedata	REPLY/INF N	Alberto Brigandi/Marica



INDIGO - DataCloud

	using the OneData spaces provided by the user				Antonacci/OneData Team
Orchestrator - Mesos	Ensure that the submitted job request is correctly handled: data are fetched from public URLs and stored on a repository (e.g. Swift container) with credentials provided by the user. OneData "service" space is used as temporary cache.	T5.1/T5.3/T5.4	Orchestrator/Mesos/Onedata	REPLY/INFN	Alberto Brigandi/Marica Antonacci/OneData Team

Table A2.1.4: The table show the tests performed between WP6 components.

Name	Test description	Task(s)	Product	Provider	Tester
indigoclient communicates with FutureGateway	Make sure that two-way communication between Java REST client (indigoclient) and some testing FutureGateway instance is working	WP6	indigoclient	PSNC	Tomasz Zok
				INFN	
indigokepler example workflows run on Docker(?) and communicates with FutureGateway	Make sure that example Kepler workflows run correctly with some testing FutureGateway instance. Kepler can work in non-gui mode and verification can be made automatically. Some workflows test could also involve tasks in Ophidia	WP6	indigokepler	PSNC	Tomasz Zok
				INFN	
				CMCC	Sandro Fiore
testing communication with FG API from mobile client	Make sure that two-way communication between Android mobile client (library&sample-app) and some testing FutureGateway instance is working	WP6	indigo-omt	PSNC	Michał Urbaniak
				INFN	

A2.2 Pilot Preview Infrastructure

The tables A2.2.1 to A2.2.7 show the Pilot Preview Infrastructure services their endpoints and corresponding contacts. Each table correspond to a given category of services or components.

Table A2.2.1: AAI (Cross WP4-WP6)

Partner	Endpoint	Components (version)	Contact	Status Comments
CNAF/INFN	https://iam-test.indigo-datacloud.eu/	IAM (0.0.4)	Enrico Vianello, Marco Caberletti, Andrea Ceccanti	Deployed

Table A2.2.2: Cloud computing (IaaS WP4)

Partner	Endpoint	Components (version)	Contact	Status Comments
---------	----------	----------------------	---------	-----------------



INDIGO - DataCloud

LIP/NCG	https://nimbus.ncg.ingrid.pt/dashboard https://nimbus.ncg.ingrid.pt:5000/v3	OpenStack (mitaka) <ul style="list-style-type: none">- ooi- IAM connector- nova-docker- OS Identity Auth lib	Mario David	Deployed
LIP/NCG	https://one01.ncg.ingrid.pt:9869	OpenNebula (4.14) <ul style="list-style-type: none">- ONEDock- rOCCI server	Milos Mulac	Deployed
LIP/NCG	https://one01.ncg.ingrid.pt	TTS → OpenNebula LIP/NCG	Mario David	Deployed
LIP/NCG	infra01.ncg.ingrid.pt	java-syncrepos	Mario David	Deployed
LIP/NCG	infra01.ncg.ingrid.pt	cloud-info-provider	Mario David	Deployed
LIP/NCG	https://infra01.ncg.ingrid.pt:8800	IM → OpenStack at LIP/NCG	Mario David	Deployed
IFCA/CSIC	https://cloud.ifca.es	OpenStack (Mitaka) <ul style="list-style-type: none">- ooi- IAM connector- OS Identity Auth lib	Pablo Orviz	Deployed
INFN-Padova	https://egi-cloud.pd.infn.it:8787/occi1.1	OpenStack (Liberty) <ul style="list-style-type: none">- ooi- Synergy	Marco Verlato, Matteo Segatta, Federica Fanzago	Deployed

Table A2.2.3: Cloud storage (IaaS WP4)

Partner	Endpoint	Components (version)	Contact	Status Comments
KIT	https://cdmi-qos.data.kit.edu	CDMI-QOS	Benjamin Ertl	Connected to the OneProvider and gridftp endpoints.
KIT	https://one.data.kit.edu	OneProvider	Bas Wegh	All endpoints use the same mountpoint for all protocols
INFN-Bari	https://indigo-wp3-onep.recas.ba.infn.it	OneProvider	Alessandro Italiano	Deployed
LIP/NCG	https://oprov01.ncg.ingrid.pt	OneProvider	Mario David	Deployed
DESY	http://dcache-onedata-01.desy.de	OneProvider, dCache	Juergen Starek	Deployed

Table A2.2.4: Data and storage management (PaaS WP5)

Partner	Endpoint	Components (version)	Contact	Status Comments
LIP/NCG	https://ozone01.ncg.ingrid.pt	Onedata (3.0.0-rc9): OneZone	Mario David	Deployed



INDIGO - DataCloud

Table A2.2.5: Orchestration, Docker management (PaaS WP5)

Partner	Endpoint	Components (version)	Contact	Status /Comments
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it:8888	Orchestrator	Diego Michelotto, Cristina Aiftimiei	Deployed
UPV	http://servproject.i3m.upv.es/im/	IM- CLUES	Miguel Caballer	Deployed

Table A2.2.6: PaaS services (WP5)

Partner	Endpoint	Components (version)	Contact	Status Comments
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it:8444/rank	CloudProviderRanker	Diego Michelotto, Cristina Aiftimiei	Deployed
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it:8082/	zabbix-wrapper	Diego Michelotto, Cristina Aiftimiei	Deployed
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it:8081/slam	SLAManager	Diego Michelotto, Cristina Aiftimiei	Deployed
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it:7777/cmdb	CMDB	Diego Michelotto, Cristina Aiftimiei	Deployed
CNAF/INFN	http://orchestrator.cloud.cnaf.infn.it	Zabbix-server	Diego Michelotto, Cristina Aiftimiei	Deployed

Table A2.2.7: Scientific Portals (SaaS WP6)

Partner	Endpoint	Components (version)	Contact	Status Comments
LIP/NCG	https://fgw01.ncg.ingrid.pt	Future Gateway <ul style="list-style-type: none"> - FG API server - FG Portal - LiferayIAM - Indigo Kepler - Ophidia - Tosca templates 	Tomasz Żok, Marco Fargetta , Mario David	Deployed But still some work on issues

A3. APPENDIX 3

A3.1 Metrics values

Table A3.1.1: Metrics I

INDIGO-DC product	M_STAND_1 Number of	M_STAND_2 Adherence to	M_SECUR_1 Number of	M_ACCUR_1 Number of	M_ACCUR_2 Functional
-------------------	------------------------	---------------------------	------------------------	------------------------	-------------------------



INDIGO - DataCloud

	documented code style guidelines	documented code style guidelines	security vulnerabilities	functionalities tested	testing coverage
Applicability	M	M	M	M	R
dCache	6	100%	0	2	100%
Token Translation Service (TTS)	22	100%	0	9	100%
Identity Harmonization Service (IDH)	54	100%	0	7	100%
CDMI Server (QOS)	54	100%	0	17	100%
rOCCI	62	100%	0	3	100%
pOCCI	79	83% (5 exceptions)	0	1	100%
Infrastructure Manager (IM)	71	97% (2 exceptions)	0	6	100%
Infrastructure Manager Java API (IM-Java-API)	54	100%	0	9	100%
Synergy Service	111	100%	0	3	100%
Synergy Scheduler Manager	111	100%	0	3	100%
OpenStack OCCI Interface (ooi)	111	100%	0	5	100%
OpenStack Preemptible Instances	111	100%	0	29	100%



INDIGO - DataCloud

Extension (OPIE)					
ONEDock	73	100%	0	1	100%
Partition Director (dynpart)	73	100%	0	3	100%
python-OpenStack client	111	100%	0	1	100%
python-novaclient	111	100%	0	1	100%
udocker	73	100%	0	22	100%
reposync	54	100%	0	4	100%
tosca-parser	111	100%	0	26	100%
heat-translator	111	100%	0	14	100%
cdmi-s3-qos	54	100%	0	<not-provided>	100%
CLUSTER Energy Saving (CLUES)	72	99% (1 exception)	0	3	100%
SLA Manager (SLAM)	54	100%	0	<not-provided>	100%
Orchestrator	54	100%	0	2	100%
Onedata	18	100%	0	8	100%
Zabbix Probes/Monitoring	54	100%	0	7	100%
INDIGO IAM	54	100%	0	2	100%
Accounting	101	100%	0	3	100%
CloudProviderRanker	54	76% (13 exceptions)	0	1	100%
CloudInfoProvider	84	100%	0	4	100%



INDIGO - DataCloud

FutureGateway API Server	73	100%	0	3	100%
INDIGO OMT	82	100%	0	4	100%
Ophidia	32	100%	0	3	100%
INDIGO-kepler	63	100%	0	9	100%
JSAGA Resource Management API	54	100%	0	40	100%
Liferay IAM	63	100%	0	2	100%
orchent	3	100%	0	8	100%

Table A3.1.2: Metrics II

INDIGO-DC product	M_MATUR_1 Trend analysis	M_MATUR_2 Defects by severity	M_CHANG_3 Lines of code	M_CHANG_4 Comments frequency
Applicability	M	M	M	M
dCache	0	0	388200	91733
Token Translation Service (TTS)	0	0	4846	1873
Identity Harmonization Service (IDH)	0	0	2975	705
CDMI Server (QOS)	0	0	7390	3232
rOCCI	0	0	9667	3639
pOCCI	0	0	2800	767
Infrastructure Manager (IM)	0	0	21562	5784
Infrastructure Manager Java API	0	0	3722	2018



INDIGO - DataCloud

(IM-Java-API)				
Synergy Service	0	0	1996	536
Synergy Scheduler Manager	0	0	3445	321
OpenStack OCCI Interface (ooi)	0	0	10093	1876
OpenStack Preemptible Instances Extension (OPIE)	0	0	1158	350
ONEDock	0	0	4344	1340
Partition Director (dynpart)	0	0	671	151
python-OpenStackclient	0	0	51834	8947
python-novaclient	0	0	23222	5864
udocker	0	0	4153	832
reposync	0	0	2714	222
tosca-parser	0	0	14532	1683
heat-translator	0	0	10093	1065
cdmi-s3-qos	0	0	2183	979
CLUster Energy Saving (CLUES)	0	0	1942	128
SLA Manager (SLAM)	0	0	25705	3918
Orchestrator	0	0	14494	1977
Onedata	0	0	6614	759
Zabbix Probes/M	0	0	26733	5359



INDIGO - DataCloud

onitoring				
INDIGO IAM	0	0	70827	6371
Accounting	0	0	1155	308
CloudProviderRanker	0	0	1438	260
CloudInfoProvider	0	0	3792	228
FutureGateway API Server	0	0	5542	1323
INDIGO OMT	0	0	2027	213
Ophidia	0	0	69124	4609
INDIGO-kepler	0	0	1419	332
JSAGA Resource Management API	0	0	104552	51921
Liferay IAM	0	0	5595	3889
orchent	0	0	920	349

Table A3.1.3: Metrics III

INDIGO-DC product	M_DOCUM_1 Developer documentation completeness	M_DOCUM_2 Deployment and administration documentation completeness	M_DOCUM_3 CLI/API documentation completeness	M_TEST_1 Unit testing coverage
Applicability	M	M	M	M
dCache	2	2	1	22%
Token Translation Service (TTS)	2	2	2	74%



INDIGO - DataCloud

Identity Harmonization Service (IDH)	2	2	2	54%
CDMI Server (QOS)	2	2	2	35%
rOCCI	0	2	2	48%
pOCCI	2	2	2	20%
Infrastructure Manager (IM)	0	2	2	68%
Infrastructure Manager Java API (IM-Java-API)	2	2	0	76%
Synergy Service	2	2	0	13%
Synergy Scheduler Manager	2	2	0	9%
OpenStack OCCI Interface (ooi)	2	2	2	87%
OpenStack Preemptible Instances Extension (OPIE)	2	2	0	92%
ONEDock	0	2	0	NA
Partition Director (dynpart)	0	2	0	0%
python-OpenStackclient	2	2	2	73%
python-novaclient	2	2	2	78%
udocker	0	2	2	25%
reposync	2	2	2	29%
tosca-parser	2	2	2	90%
heat-translator	2	2	2	17%



INDIGO - DataCloud

cdmi-s3-qos	NA	NA	NA	NA
CLUster Energy Saving (CLUES)	0	2	0	89%
SLA Manager (SLAM)	0	2	2	88%
Orchestrator	0	2	2	28%
Onedata	0	2	2	NA
Zabbix Probes/Monitoring	2	2	2	NA
INDIGO IAM	2	2	2	64%
Accounting	2	2	2	70%
CloudProviderRanker	0	2	2	70%
CloudInfoProvider	2	2	0	94%
FutureGateway API Server	0	2	0	0%
INDIGO OMT	2	0	2	14%
Ophidia	2	2	2	70%
INDIGO-kepler	2	2	0	86%
JSAGA Resource Management API	0	2	2	46%
Liferay IAM	2	2	2	9%
orchent	0	2	2	NA

A3.2 KPIs values - SQA process

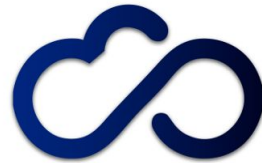
Table A3.2.1: SQA Progress I



INDIGO - DataCloud



Product	W1 13/06 17/06	W2 20/06 24/06	W3 27/06 01/07	W4 04/07 08/07	W5 11/07 15/07	W6 18/07 22/07	W7 25/07 29/07	W8 01/08 05/08 INDIGO Release	W9 22/08 26/08 <i>Docum. review</i>
dCache	67%	67%	82%	82%	82%	82%	82%	82%	82%
Token Translation Service (TTS)	67%	82%	92%	98%	98%	98%	98%	98%	87%
Identity Harmonization Service (IDH)	50%	50%	57%	65%	65%	77%	90%	90%	83%
CDMI Server (QOS)	50%	50%	50%	65%	65%	80%	87%	87%	80%
rOCCI	75%	92%	92%	98%	98%	98%	98%	98%	78%
pOCCI	72%	87%	87%	87%	87%	87%	87%	87%	78%
Infrastructure Manager (IM)	72%	87%	95%	95%	95%	95%	95%	95%	87%
Infrastructure Manager Java API (IM-Java-API)	97%	98%	98%	98%	98%	98%	98%	98%	90%
Synergy Service	52%	68%	68%	68%	68%	83%	83%	83%	75%
Synergy Scheduler Manager	67%	77%	77%	77%	77%	85%	85%	85%	85%
OpenStack OCCI Interface (ooi)	48%	48%	82%	97%	97%	97%	98%	98%	90%
OpenStack Preemptible Instances Extension (OPIE)	48%	48%	65%	65%	65%	65%	65%	97%	88%



INDIGO - DataCloud

ONEDock	58%	73%	90%	90%	92%	92%	92%	92%	83%
Partition Director	0%	25%	25%	27%	40%	42%	42%	58%	58%
python-OpenStackclient	58%	58%	58%	58%	58%	93%	93%	93%	93%
python-novaclient	63%	63%	63%	58%	63%	97%	97%	97%	97%
udocker	72%	72%	72%	72%	72%	78%	78%	78%	78%
reposync (java-syncrepos)	40%	72%	72%	83%	83%	83%	83%	83%	75%
tosca-parser	-	-	-	-	-	-	NA	NA	NA
heat-translator	-	-	-	-	-	-	-	NA	NA
cdmi-s3-qos	-	-	-	-	-	-	-	-	-
Infrastructure Manager (IM)	72%	87%	95%	95%	95%	95%	95%	95%	87%
CLUster Energy Saving (CLUES)	72%	72%	72%	72%	83%	88%	97%	98%	90%
SLA Manager	0%	0%	0%	0%	0%	0%	0%	58%	63%
Orchestrator	72%	88%	88%	88%	88%	88%	88%	88%	80%
Onedata	0%	47%	47%	58%	58%	67%	75%	75%	67%
Zabbix Probes	25%	25%	48%	48%	67%	67%	73%	82%	73%
INDIGO IAM	0%	67%	67%	73%	82%	93%	93%	93%	95%
Accounting	25%	35%	35%	68%	70%	70%	78%	87%	87%



INDIGO - DataCloud

CloudProviderRanker	67%	78%	78%	78%	78%	78%	87%	87%	78%
CloudInfoProvider	-	-	-	-	-	-	57%	98%	90%
FutureGateway API Server	17%	17%	55%	62%	67%	70%	70%	70%	62%
Token Translator Service Client (TTSc)	67%	82%	92%	98%	98%	98%	98%	98%	87%
INDIGO OMT	17%	25%	37%	47%	67%	75%	80%	87%	78%
Ophidia	8%	32%	32%	52%	75%	83%	95%	95%	87%
iINDIGO-kepler	25%	33%	55%	67%	77%	77%	98%	98%	90%
JSAGA Resource Management API	-	-	20%	37%	37%	38%	38%	92%	83%
Liferay IAM	-	-	47%	57%	67%	82%	82%	82%	77%
orchent	-	-	-	-	-	-	-	-	-

Table A3.2.2: SQA Progress II

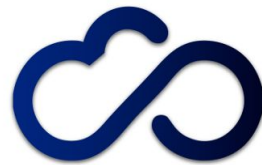
Product	W10 29/08 02/08 <i>Docum. review</i>	W11 05/09 09/09 <i>Docum. review</i>	W12 19/09 23/09 <i>END of Docum. review</i>	W13 26/09 30/09	W14 03/10 07/10	W15 24/10 28/10	W16 30/10 04/11	W17 07/11 01/11
dCache	85%	85%	85%	85%	85%	85%	85%	85%
Token Translation Service (TTS)	87%	95%	95%	95%	95%	88%	88%	88%



INDIGO - DataCloud



Identity Harmonization Service (IDH)	83%	83%	83%	83%	83%	83%	83%	83%
CDMI Server (QOS)	80%	80%	80%	80%	80%	80%	80%	80%
rOCCI	78%	78%	98%	98%	98%	98%	98%	98%
pOCCI	78%	78%	78%	83%	87%	85%	87%	85%
Infrastructure Manager (IM)	87%	95%	95%	95%	95%	95%	95%	95%
Infrastructure Manager Java API (IM-Java-API)	90%	90%	90%	90%	90%	90%	90%	90%
Synergy Service	83%	83%	83%	83%	83%	85%	85%	85%
Synergy Scheduler Manager	85%	85%	85%	85%	85%	85%	85%	85%
OpenStack OCCI Interface (ooi)	90%	90%	98%	98%	98%	98%	98%	98%
OpenStack Preemptible Instances Extension (OPIE)	88%	88%	97%	97%	97%	97%	97%	97%
ONEDock	83%	83%	92%	92%	92%	92%	92%	92%
Partition Director	58%	58%	58%	67%	67%	67%	67%	67%
python-OpenStackclient	93%	93%	93%	93%	93%	93%	93%	93%
python-novaclient	97%	97%	97%	97%	97%	97%	97%	97%
udocker	78%	78%	78%	78%	78%	78%	78%	78%
reposync (java-syncrepos)	75%	75%	75%	83%	83%	83%	83%	83%



INDIGO - DataCloud



tosca-parser	NA	NA	NA	NA	NA	NA	NA	NA
heat-translator	NA	NA	NA	NA	NA	NA	NA	NA
cdmi-s3-qos	-	-	-	-	-	-	33%	43%
Infrastructure Manager (IM)	87%	95%	95%	95%	95%	95%	95%	95%
CLUster Energy Saving (CLUES)	90%	90%	98%	98%	98%	98%	98%	98%
SLA Manager	63%	63%	63%	63%	63%	63%	63%	63%
Orchestrator	80%	80%	88%	88%	88%	88%	88%	88%
Onedata	67%	67%	75%	75%	75%	75%	75%	75%
Zabbix Probes	73%	73%	82%	82%	82%	65%	72%	80%
INDIGO IAM	95%	95%	95%	95%	95%	95%	95%	95%
Accounting	87%	87%	95%	95%	95%	98%	98%	98%
CloudProviderRanker	78%	78%	87%	87%	87%	87%	87%	87%
CloudInfoProvider	98%	98%	98%	98%	98%	98%	98%	98%
FutureGateway API Server	62%	62%	75%	75%	75%	75%	75%	75%
Token Translator Service Client (TTSc)	87%	95%	95%	95%	95%	88%	88%	88%
INDIGO OMT	78%	78%	87%	87%	87%	87%	87%	87%
Ophidia	87%	87%	95%	95%	95%	93%	93%	93%



INDIGO - DataCloud

INDIGO-kepler	90%	98%	98%	98%	98%	98%	98%	98%
JSAGA Resource Management API	83%	83%	92%	92%	92%	88%	88%	88%
Liferay IAM	85%	85%	85%	85%	85%	85%	85%	85%
orchent	-	-	-	-	-	75%	75%	75%

Table A3.2.3: Automation testing index

INDIGO-DataCloud product	Automation Testing Index (CS: code style; UT: unit testing; FT: functional testing; IT: integration testing)
dCache	75% (CS, UT, FT)
Token Translation Service (TTS)	75% (CS, UT, FT)
Identity Harmonization Service (IDH)	75% (CS, UT, FT)
CDMI Server (QOS)	75% (CS, UT, FT)
rOCCI	75% (CS, UT, FT)
pOCCI	50% (CS, UT)
Infrastructure Manager (IM)	75% (CS, UT, FT)
Infrastructure Manager Java API (IM-Java-API)	75% (CS, UT, FT)
Synergy Service	50% (CS, UT)
Synergy Scheduler Manager	50% (CS, UT)
OpenStack OCCI Interface (ooi)	75% (CS, UT, FT)
OpenStack Preemptible Instances Extension (OPIE)	75% (CS, UT, FT)
ONEDock	75% (CS, UT, FT)



INDIGO - DataCloud

Partition Director (dynpart)	25% (CS)
python-OpenStackclient	75% (CS, UT, FT)
python-novaclient	75% (CS, UT, FT)
udocker	75% (CS, UT, FT)
reposync	50% (CS, UT)
tosca-parser	75% (CS, UT, FT)
heat-translator	75% (CS, UT, FT)
cdmi-s3-qos	50% (CS, UT)
CLUster Energy Saving (CLUES)	50% (CS, UT)
SLA Manager (SLAM)	50% (CS, UT)
Orchestrator	75% (CS, UT, FT)
Onedata	50% (CS, FT)
Zabbix Probes/Monitoring	75% (CS, UT, FT)
INDIGO IAM	75% (CS, UT, FT)
Accounting	50% (CS, UT)
CloudProviderRanker	50% (UT, FT)
CloudInfoProvider	75% (CS, UT, FT)
FutureGateway API Server	25% (CS)
INDIGO OMT	50% (CS, UT)
Ophidia	75% (CS, UT, FT)
indigo-kepler	50% (CS, UT)
JSAGA Resource Management API	75% (CS, UT, FT)
Liferay IAM	50% (CS, UT)



INDIGO - DataCloud



orchent	25% (CS)
---------	----------