INDIGO - DataCloud

RIA-653549

The INDIGO-DataCloud Project

Giacinto Donvito

INFN
INDIGO-DataCloud Technical Director

giacinto.donvito@ba.infn.it

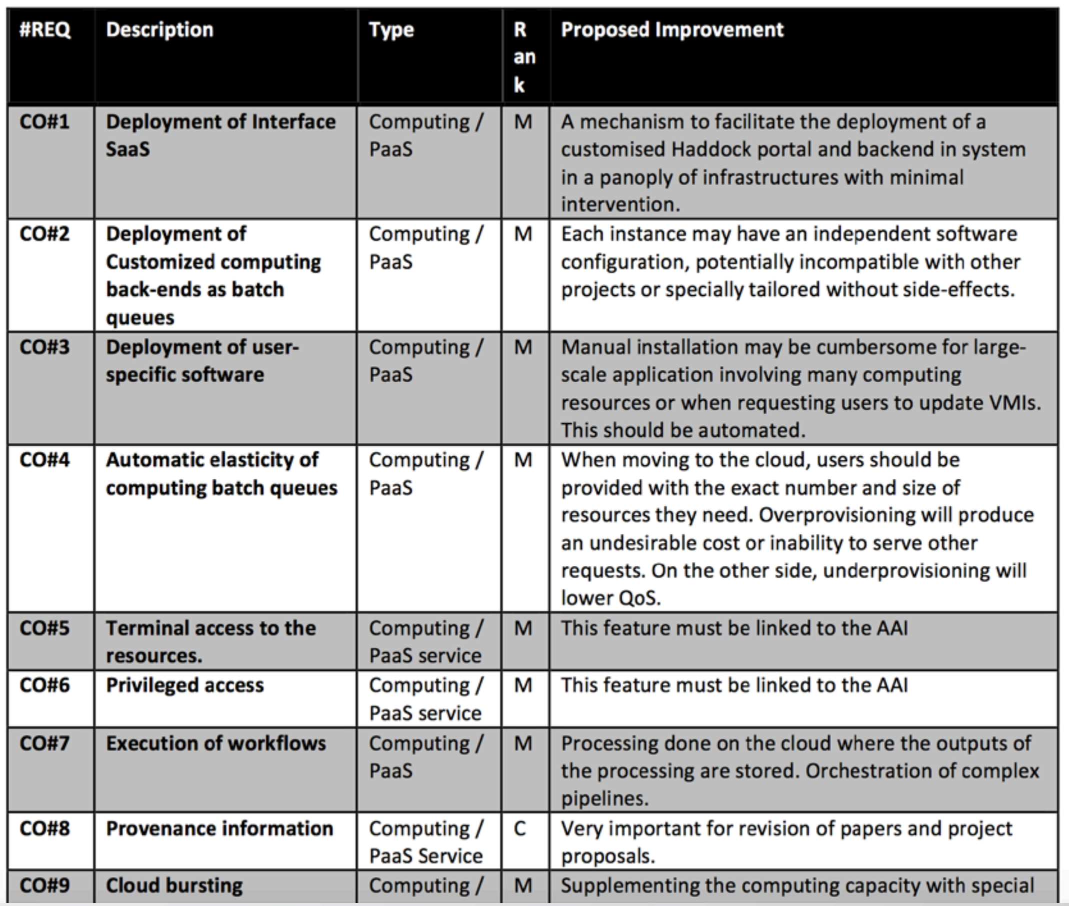
INDIGO-DataCloud is co-founded by the
Horizon 2020Framework Programme

INDIGO-DataCloud
https://www.indigo-datacloud.eu

- **An H2020 project** approved in January 2015 in the EINFRA-1-2014 call
  - 11.1M€, 30 months (until September 2017)
- **Who**: 26 European partners in 11 European countries
  - Including developers of distributed software, industrial partners, research institutes, universities, e-infrastructures
- **What**: develop an open source Cloud platform for computing and data ("DataCloud").
- **For**: multi-disciplinary scientific communities
  - E.g. structural biology, earth science, physics, bioinformatics, cultural heritage, astrophysics, life science, climatology
- **Where**: deployable on hybrid (public or private) Cloud infrastructures
  - INDIGO = **IN**tegrating **D**istributed data **I**nfrastructures for **G**lobal Expl**O**itation
- **Why**: see next slide

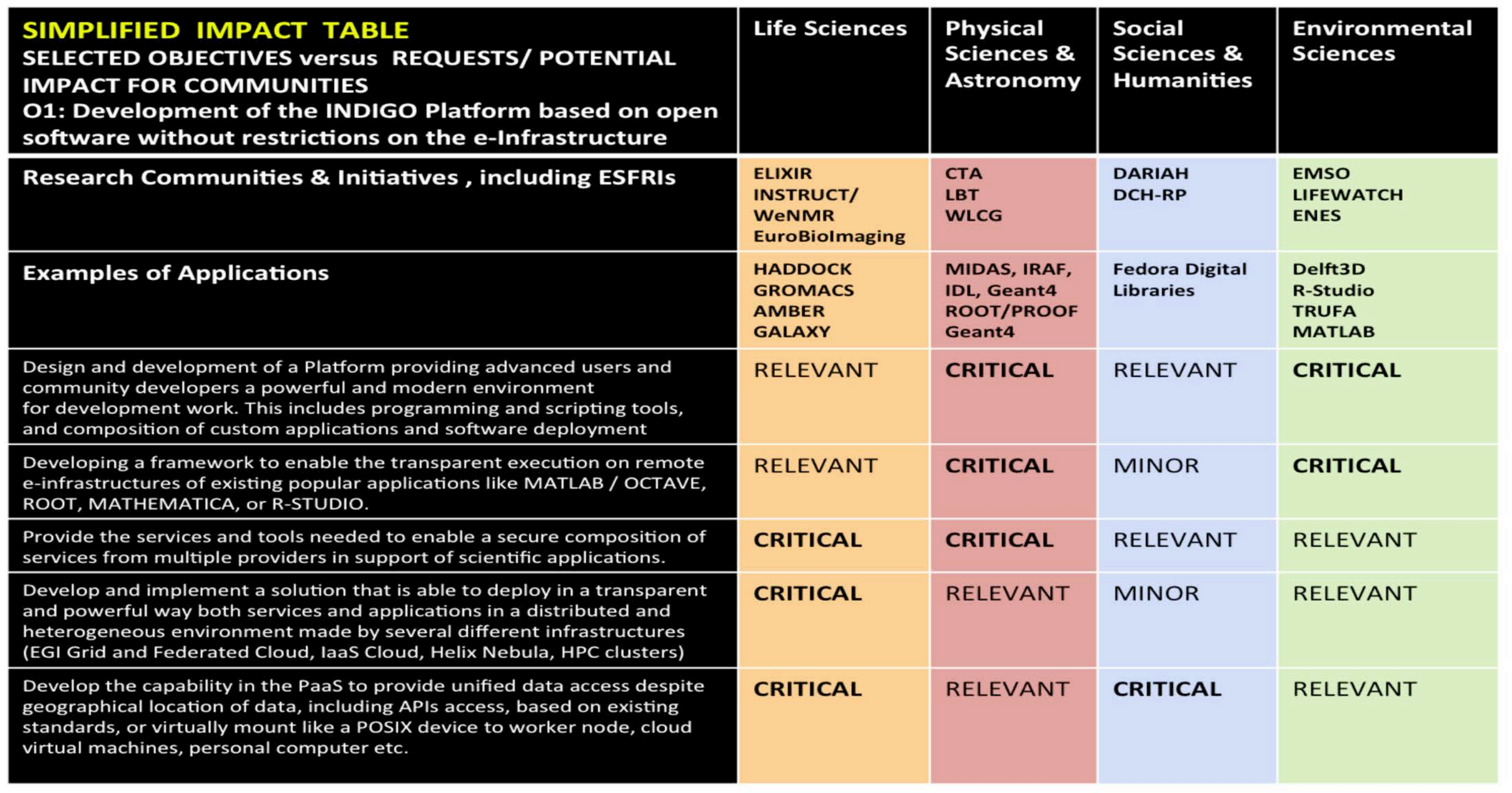
Users First

- **Requirements come from research communities**
  - “The proposal is oriented to support the use of different e-infrastructures by a wide-range of scientific communities, and aims to address a wide range of challenging requirements posed by leading-edge research activities conducted by those communities.” (INDIGO DoW)
- **We gathered use cases** from 11 scientific communities.
  - LifeWatch, EuroBioImaging, INSTRUCT, LBT, CTA, WeNMR, ENES, eCulture Science Gateway, ELIXIR, EMSO, DARIAH.
- Starting from about 100 distinct requirements we derived a much shorter list, grouped **into 3 categories: Computational requirements, Storage requirements, Requirements on infrastructures**. Each requirement has an associated ranking (mandatory / convenient / optional).
- See **https://www.indigo-datacloud.eu/pages/components/deliverables.html**



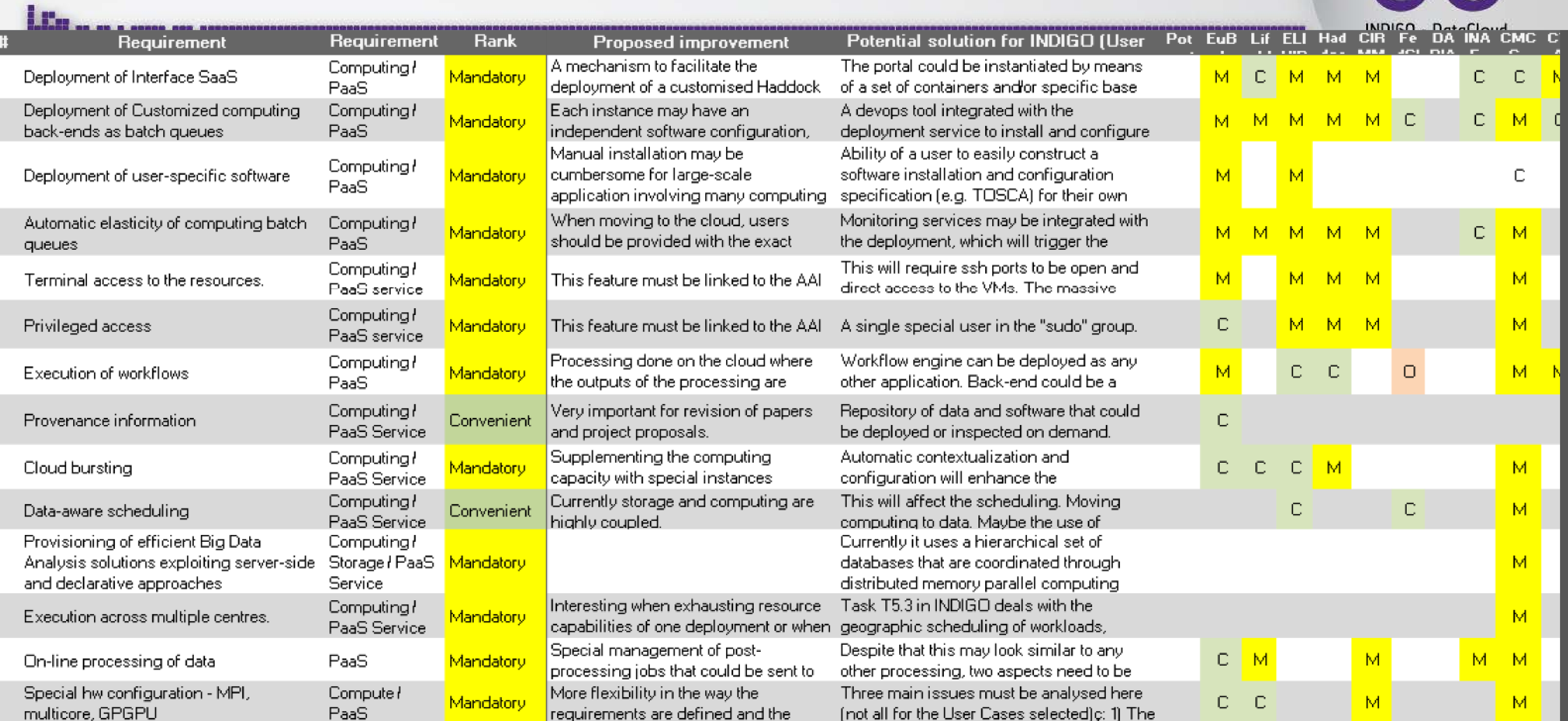
#REQ	Description	Type	Rank	Proposed Improvement
CO#1	**Deployment of Interface SaaS**	Computing / PaaS	M	A mechanism to facilitate the deployment of a customised Haddock portal and backend in system in a panoply of infrastructures with minimal intervention.
CO#2	**Deployment of Customized computing back-ends as batch queues**	Computing / PaaS	M	Each instance may have an independent software configuration, potentially incompatible with other projects or specially tailored without side-effects.
CO#3	**Deployment of user-specific software**	Computing / PaaS	M	Manual installation may be cumbersome for large-scale application involving many computing resources or when requesting users to update VMIs. This should be automated.
CO#4	**Automatic elasticity of computing batch queues**	Computing / PaaS	M	When moving to the cloud, users should be provided with the exact number and size of resources they need. Overprovisioning will produce an undesirable cost or inability to serve other requests. On the other side, underprovisioning will lower QoS.
CO#5	**Terminal access to the resources.**	Computing / PaaS service	M	This feature must be linked to the AAI
CO#6	**Privileged access**	Computing / PaaS service	M	This feature must be linked to the AAI
CO#7	**Execution of workflows**	Computing / PaaS	M	Processing done on the cloud where the outputs of the processing are stored. Orchestration of complex pipelines.
CO#8	**Provenance information**	Computing / PaaS Service	C	Very important for revision of papers and project proposals.
CO#9	**Cloud bursting**	Computing /	M	Supplementing the computing capacity with special

User Communities Say…



SIMPLIFIED IMPACT TABLE **SELECTED OBJECTIVES versus REQUESTS/ POTENTIAL IMPACT FOR COMMUNITIES** **O1: Development of the INDIGO Platform based on open software without restrictions on the e-Infrastructure**	**Life Sciences**	**Physical Sciences & Astronomy**	**Social Sciences & Humanities**	**Environmental Sciences**
Research Communities & Initiatives , including ESFRIs	ELIXIR INSTRUCT/ WeNMR EuroBioImaging	CTA LBT WLCG	DARIAH DCH-RP	EMSO LIFEWATCH ENES
Examples of Applications	HADDOCK GROMACS AMBER GALAXY	MIDAS, IRAF, IDL, Geant4 ROOT/PROOF Geant4	Fedora Digital Libraries	Delft3D R-Studio TRUFA MATLAB
Design and development of a Platform providing advanced users and community developers a powerful and modern environment for development work. This includes programming and scripting tools, and composition of custom applications and software deployment	RELEVANT	**CRITICAL**	RELEVANT	**CRITICAL**
Developing a framework to enable the transparent execution on remote e-infrastructures of existing popular applications like MATLAB / OCTAVE, ROOT, MATHEMATICA, or R-STUDIO.	RELEVANT	**CRITICAL**	MINOR	**CRITICAL**
Provide the services and tools needed to enable a secure composition of services from multiple providers in support of scientific applications.	**CRITICAL**	**CRITICAL**	RELEVANT	RELEVANT
Develop and implement a solution that is able to deploy in a transparent and powerful way both services and applications in a distributed and heterogeneous environment made by several different infrastructures (EGI Grid and Federated Cloud, IaaS Cloud, Helix Nebula, HPC clusters)	**CRITICAL**	RELEVANT	MINOR	RELEVANT
Develop the capability in the PaaS to provide unified data access despite geographical location of data, including APIs access, based on existing standards, or virtually mount like a POSIX device to worker node, cloud virtual machines, personal computer etc.	**CRITICAL**	RELEVANT	**CRITICAL**	RELEVANT

Common Requirements Analysis



#	Requirement	Requirement	Rank	Proposed improvement	Potential solution for INDIGO (User	Pot	EuB	Lif	ELI	Had	CIR	Fe	DA	INA	CMC	C
	Deployment of Interface SaaS	Computing / PaaS	Mandatory	A mechanism to facilitate the deployment of a customised Haddock	The portal could be instantiated by means of a set of containers and/or specific base		M	C	M	M	M			C	C	M
	Deployment of Customized computing back-ends as batch queues	Computing / PaaS	Mandatory	Each instance may have an independent software configuration,	A devops tool integrated with the deployment service to install and configure		M	M	M	M	M	C		C	M	C
	Deployment of user-specific software	Computing / PaaS	Mandatory	Manual installation may be cumbersome for large-scale application involving many computing	Ability of a user to easily construct a software installation and configuration specification (e.g. TOSCA) for their own		M		M						C	
	Automatic elasticity of computing batch queues	Computing / PaaS	Mandatory	When moving to the cloud, users should be provided with the exact	Monitoring services may be integrated with the deployment, which will trigger the		M	M	M	M	M			C	M	
	Terminal access to the resources.	Computing / PaaS service	Mandatory	This feature must be linked to the AAI	This will require ssh ports to be open and direct access to the VMs. The massive		M		M	M	M				M	
	Privileged access	Computing / PaaS service	Mandatory	This feature must be linked to the AAI	A single special user in the "sudo" group.		C		M	M	M				M	
	Execution of workflows	Computing / PaaS	Mandatory	Processing done on the cloud where the outputs of the processing are	Workflow engine can be deployed as any other application. Back-end could be a		M		C	C		O			M	M
	Provenance information	Computing / PaaS Service	Convenient	Very important for revision of papers and project proposals.	Repository of data and software that could be deployed or inspected on demand.		C									
	Cloud bursting	Computing / PaaS Service	Mandatory	Supplementing the computing capacity with special instances	Automatic contextualization and configuration will enhance the		C	C	C	M					M	
	Data-aware scheduling	Computing / PaaS Service	Convenient	Currently storage and computing are highly coupled.	This will affect the scheduling. Moving computing to data. Maybe the use of				C			C			M	
	Provisioning of efficient Big Data Analysis solutions exploiting server-side and declarative approaches	Computing / Storage / PaaS Service	Mandatory		Currently it uses a hierarchical set of databases that are coordinated through distributed memory parallel computing										M	
	Execution across multiple centres.	Computing / PaaS Service	Mandatory	Interesting when exhausting resource capabilities of one deployment or when	Task T5.3 in INDIGO deals with the geographic scheduling of workloads,										M	
	On-line processing of data	PaaS	Mandatory	Special management of post-processing jobs that could be sent to	Despite that this may look similar to any other processing, two aspects need to be		C	M			M			M	M	
	Special hw configuration - MPI, multicore, GPGPU	Compute / PaaS	Mandatory	More flexibility in the way the requirements are defined and the	Three main issues must be analysed here (not all for the User Cases selected): 1) The		C	C			M				M	

Key Challenges

- (The very hard task of) **Collecting and consolidating** evolving user requests
- Creation of a new **sustainable cloud competence in Europe for PaaS**, for both the scientific and industrial sectors, similar to what OpenStack and OpenNebula have done for IaaS
- **Many** technology gaps
    - For example: storage QoS, PaaS standardization, distributed AuthZ, static allocation of hardware resources, data sharing, customizable application portals

High-Level Gap Analysis (1)

- Provide support for **federated identities**; provide privacy and distributed authorization in **open Cloud platforms**
- **Orchestrate** and federate [public or private] **Cloud**, **Grid** and **HPC** resources
- Overcome barriers that **limit the adoption** of true **PaaS** solutions, such as the use of custom, **non-interoperable interfaces** and the limited availability of APIs for technology-independent storage access
- Overcome the lack of **flexible data sharing** between group members and the difficulty in obtaining **easy access to data** generated by collaborating users working with different infrastructures or sites
- Overcome **static allocation** and partitioning of both storage and computing **resources** in data centers

High-Level Gap Analysis (2)

- Avoid software and **vendor lock-in** → open source, use of standards
- Exploit **specialized hardware**, such as GPUs or low-latency interconnections
- Manage dynamic and **complex workflows** for scientific **data analysis**
- Provide **APIs** to exploit the capabilities of the infrastructure and write applications, **customizable portals and mobile views**
- Overcome the current **inflexible ways** of **distributing** and deploying applications
- Facilitate **porting of existing applications to the Cloud**, so they can exploit advanced features such as those mentioned above

INDIGO Focus On PaaS…

- Interoperable **description of Cloud applications and infrastructural services** through OASIS TOSCA applied and implemented at IaaS, PaaS and SaaS levels
  - Expressive service descriptions → try to customize the environment, not the application
- Interoperability with AuthN standards (OIDC, SAML, X.509) to create very much needed **flexible, scalable, distributed AuthZ policies**
- **Use of the Cloud Data Management Interface (CDMI)** to create, retrieve, update, and delete objects in a cloud.
  - Define extensions for storage QoS and data lifecycle management, standardizing definitions and specs within RDA, submitting a reference interface to SNIA
- **Unified data management**, e.g. through API for data and metadata management, optimized data access for federated data, gateway to external data repositories.

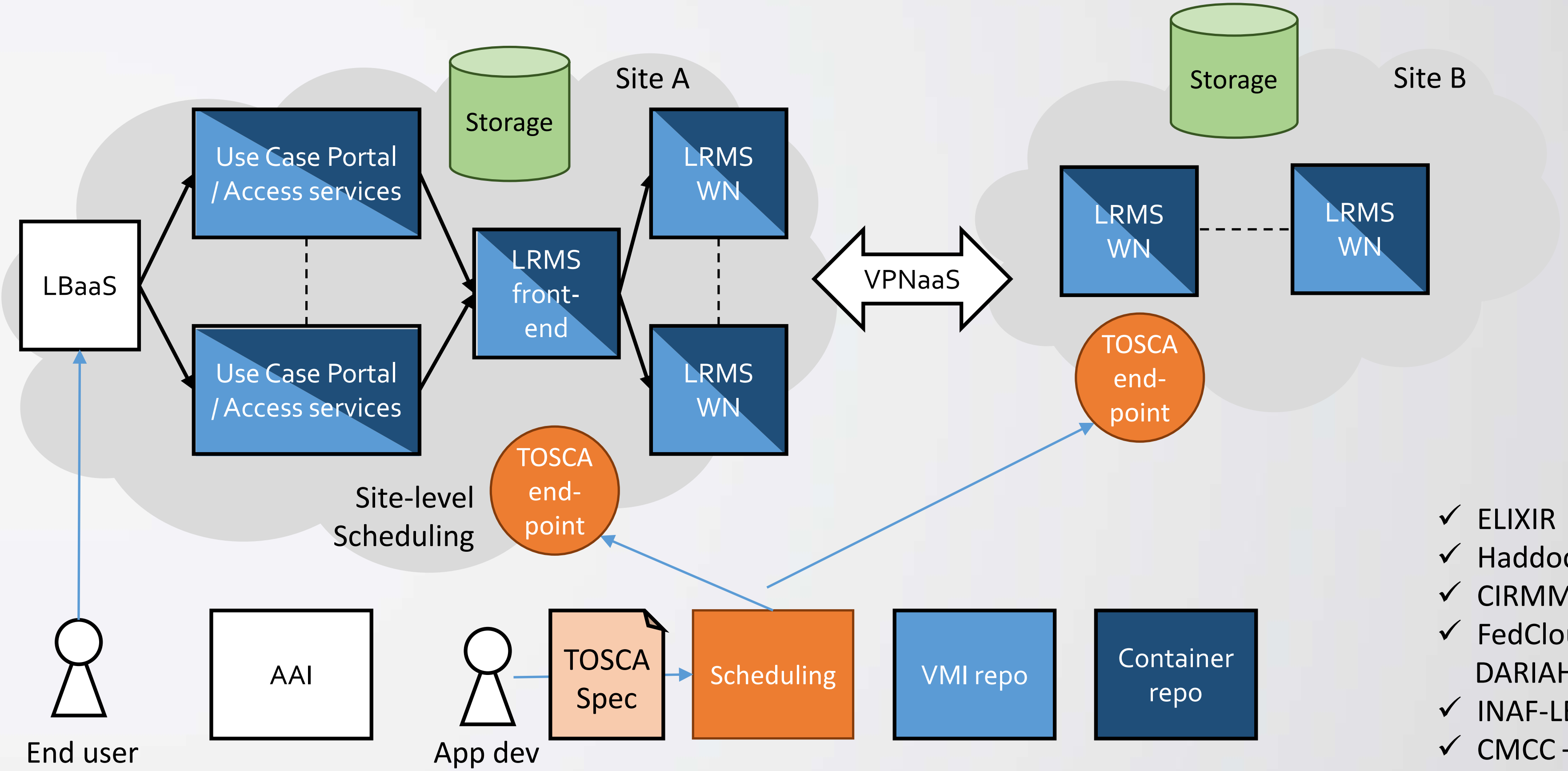
… Extensions to Site Virtualization Technologies…

- **Containers**
  - Initial work will leverage Docker, but we are open to consider any suitable technology
  - Each site will have a local container catalog, pulling containers from the main INDIGO repository
  - Allow the execution of containers in Batch Systems
  - PoC accessing InfiniBand and GPUs
- **Scheduling**
  - Scheduling with Cloud management frameworks right now too naïve
  - Provide two complementary mechanisms for scheduling
    - *Fair-share scheduling*
      - Instances can be queued.
      - Instances with fixed duration time.
      - Instances will be terminated after its wall time is exhausted.
    - *Support for spot instances*
      - Instances can be terminated by a higher priority task.
      - Flexible mechanism to support priorities or pricing.

… Support of Popular Scientific Applications…

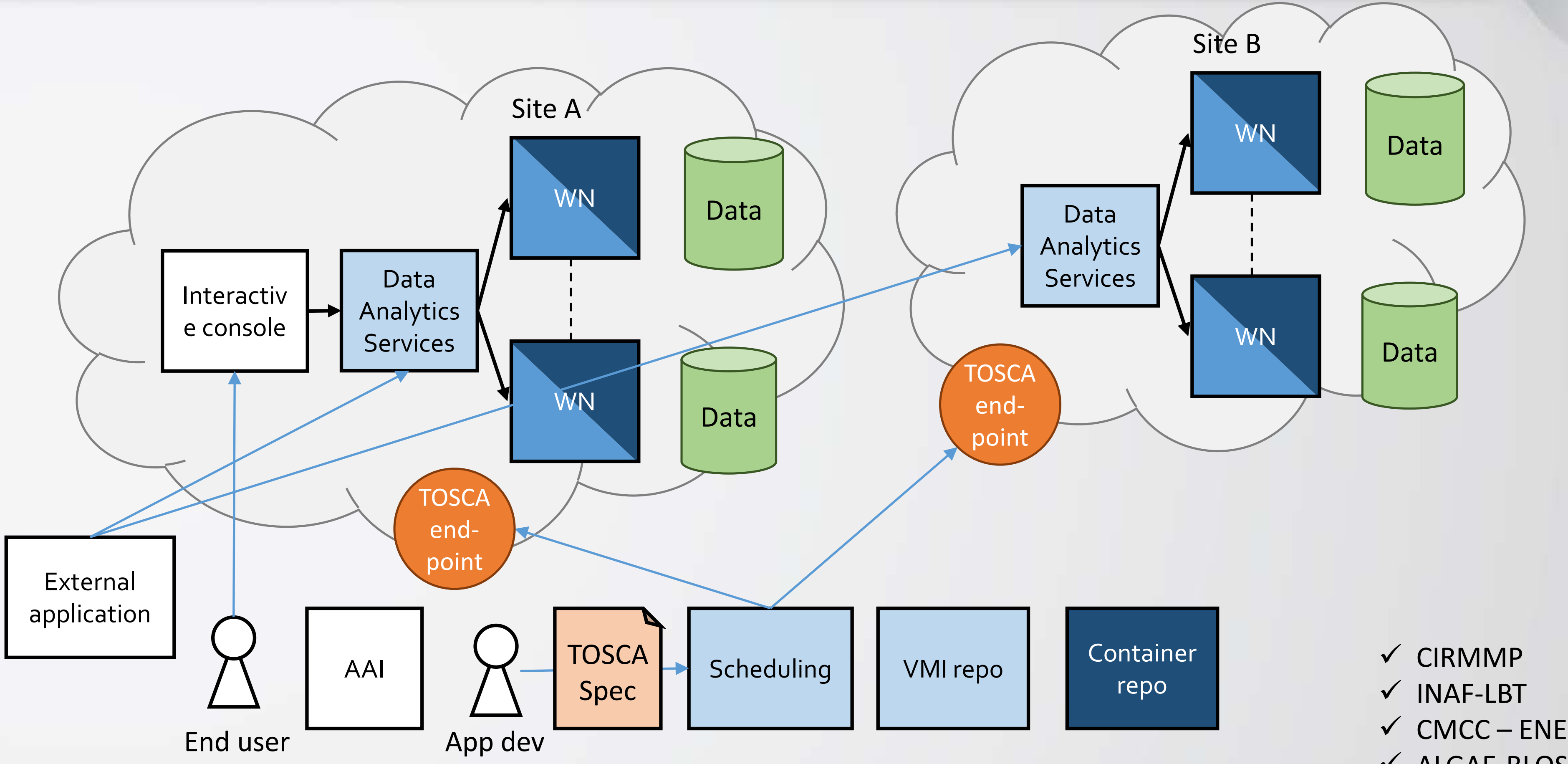
- Enable **transparent execution of existing popular applications**: MATLAB/Octave, Root, Mathematica, R-Studio
  - Option 1. Deploy a customized virtual infrastructure, dynamically stage the application and the dataset required and provide access to the user
    - SSH-based for CLI lovers
    - GUI-based via Remote Desktop/VNC
    - Web-based access (if applications support it)
  - Option 2. Modify existing applications to perform computing on a backend provisioned and configured by INDIGO.
    - Requires further development and maintenance of applications.
    - No need to provide an SDK-type of PaaS.
- Option 1 will maintain the same user interface of applications and will simplify development and maintenance.

… Scientific Computational Portal "as a Service"…



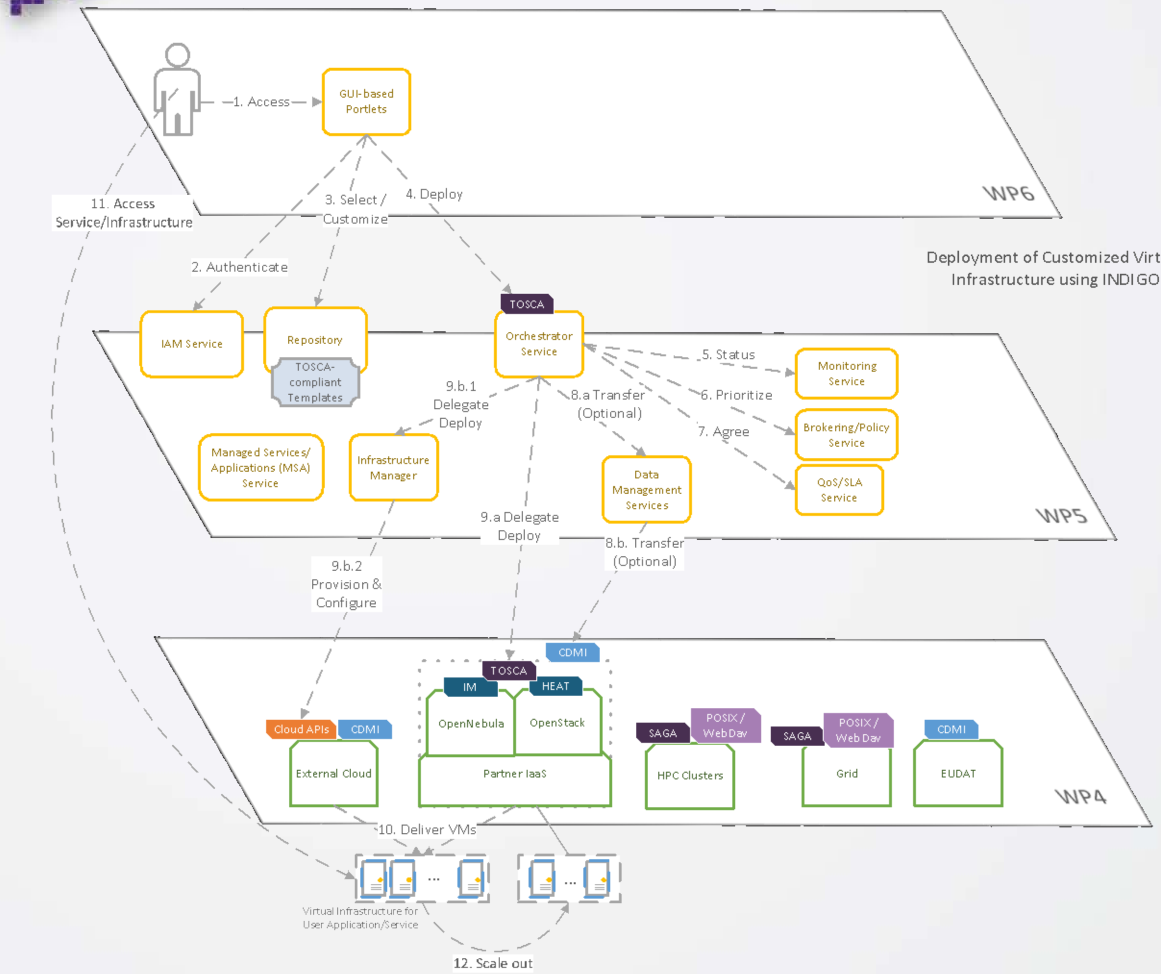
- ✓ ELIXIR
- ✓ Haddock
- ✓ CIRMMP
- ✓ FedCloud DARIAH
- ✓ INAF-LBT
- ✓ CMCC – ENES
- ✓ CTA
- ✓ ALGAE-BLOSSOM INGV - MOIST

… Software as a Service Use Case

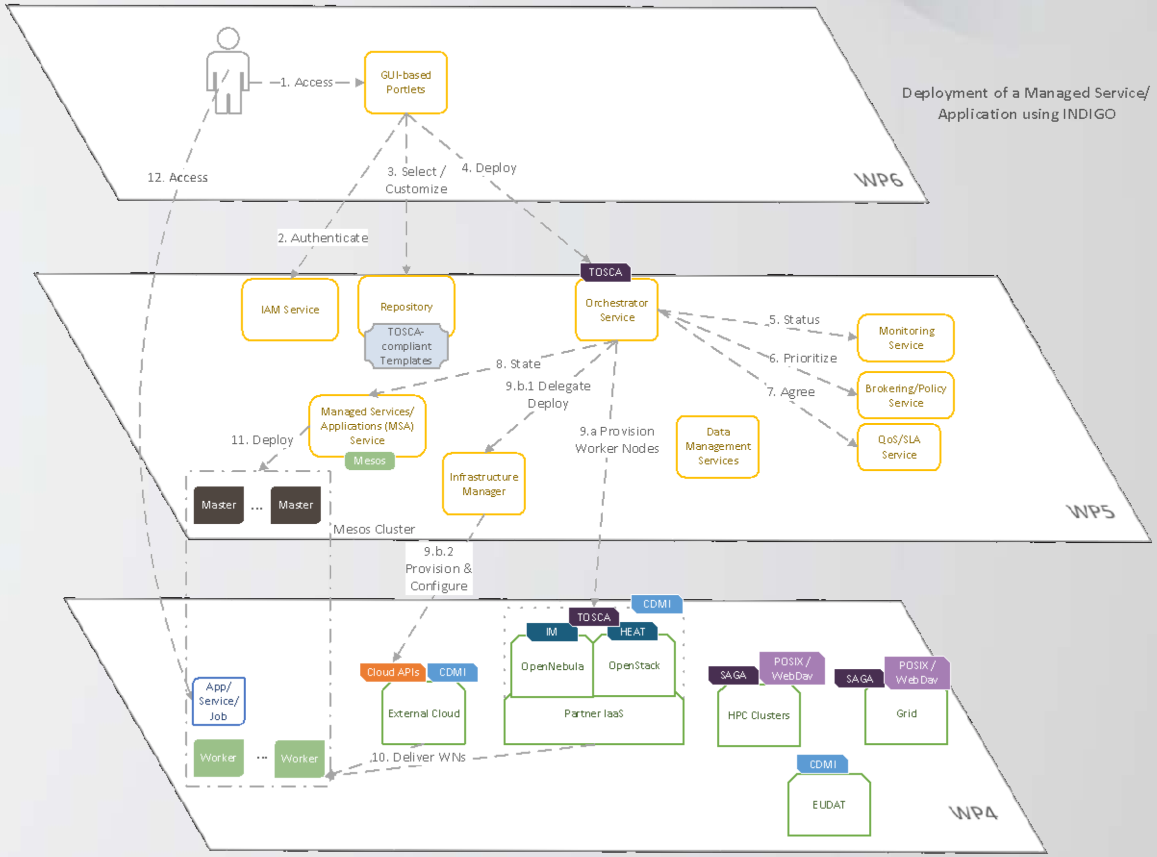


- ✓ CIRMMP
- ✓ INAF-LBT
- ✓ CMCC – ENES
- ✓ ALGAE-BLOSSOM
- ✓ INGV - MOIST

Deployment of Virtual Infrastructures or of Managed Services/Applications

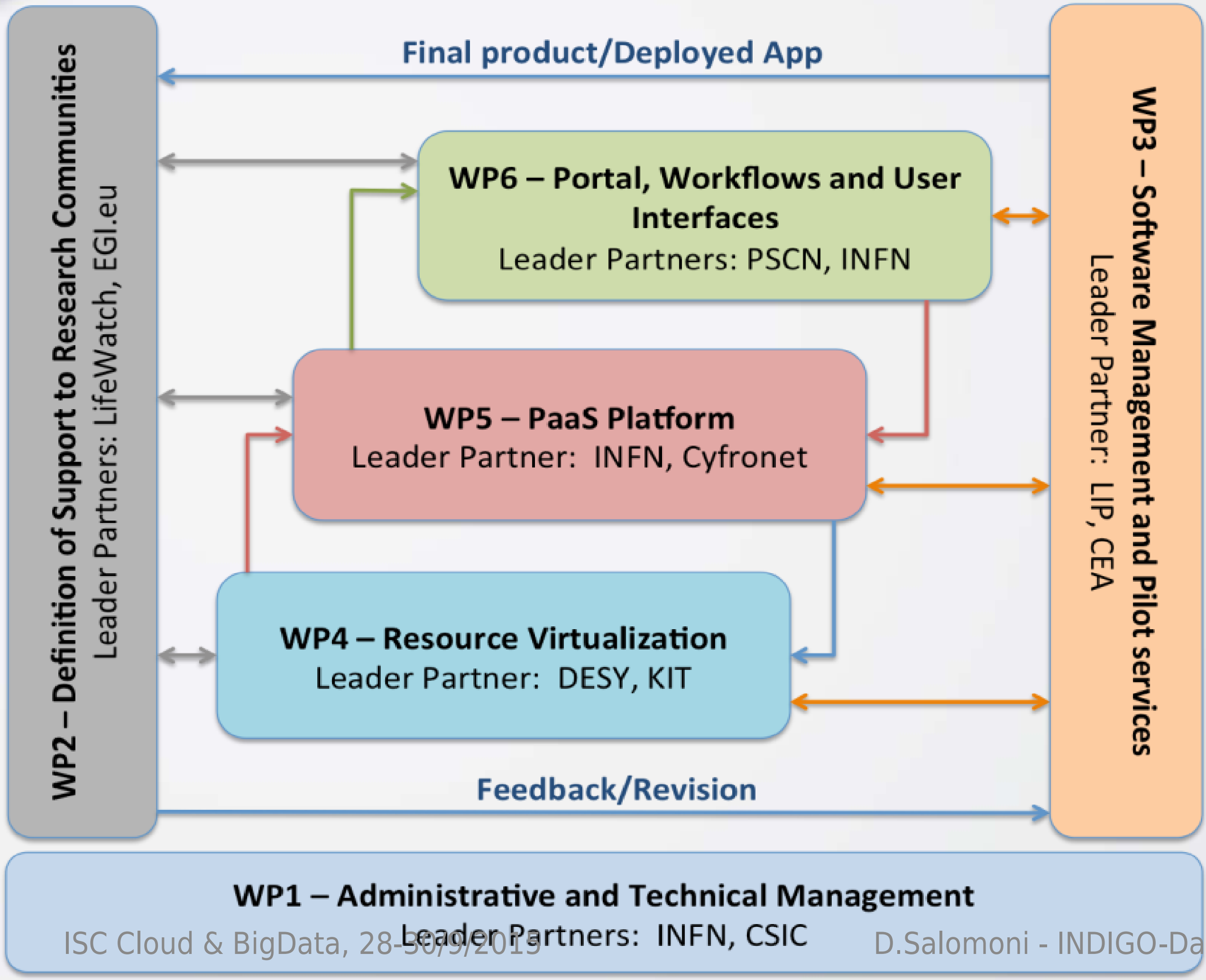


Deployment of Customized Virtual Infrastructure using INDIGO



Deployment of a Managed Service/ Application using INDIGO

INDIGO-DataCloud: Project Implementation



WP2 – Definition of Support to Research Communities
Leader Partners: LifeWatch, EGI.eu

Final product/Deployed App

WP6 – Portal, Workflows and User Interfaces
Leader Partners: PSCN, INFN

WP5 – PaaS Platform
Leader Partner: INFN, Cyfronet

WP4 – Resource Virtualization
Leader Partner: DESY, KIT

WP3 – Software Management and Pilot services
Leader Partner: LIP, CEA

Feedback/Revision

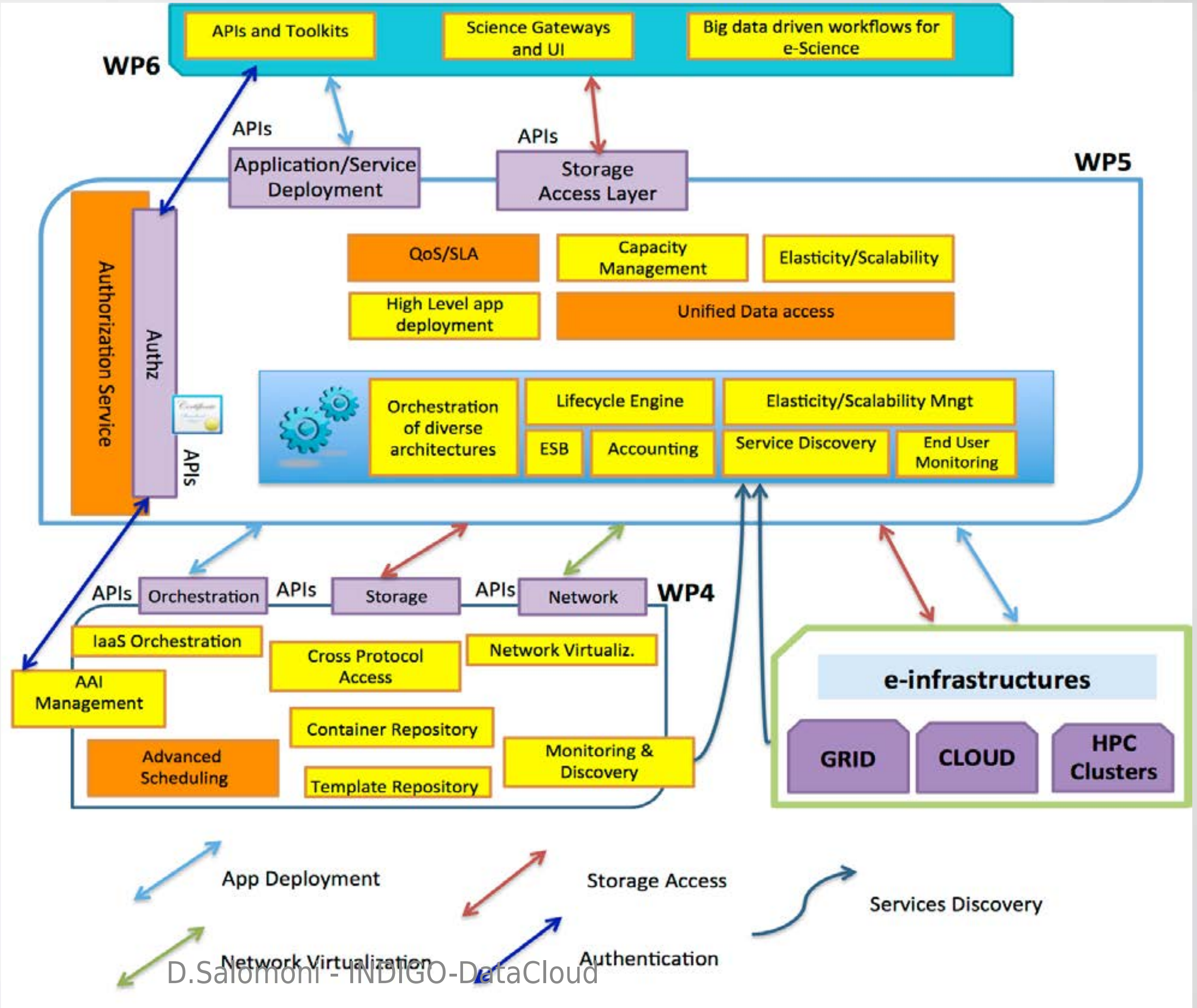
WP1 – Administrative and Technical Management
Leader Partners: INFN, CSIC

- ✓ **Community Requirements (WP2)**
- ✓ **Portal deployment → user access (WP6)**
- ✓ **Platform as a Service design (WP5)**
- ✓ **Infrastructure Oriented (WP4)**
- ✓ **Software management and pilot services (WP3)**

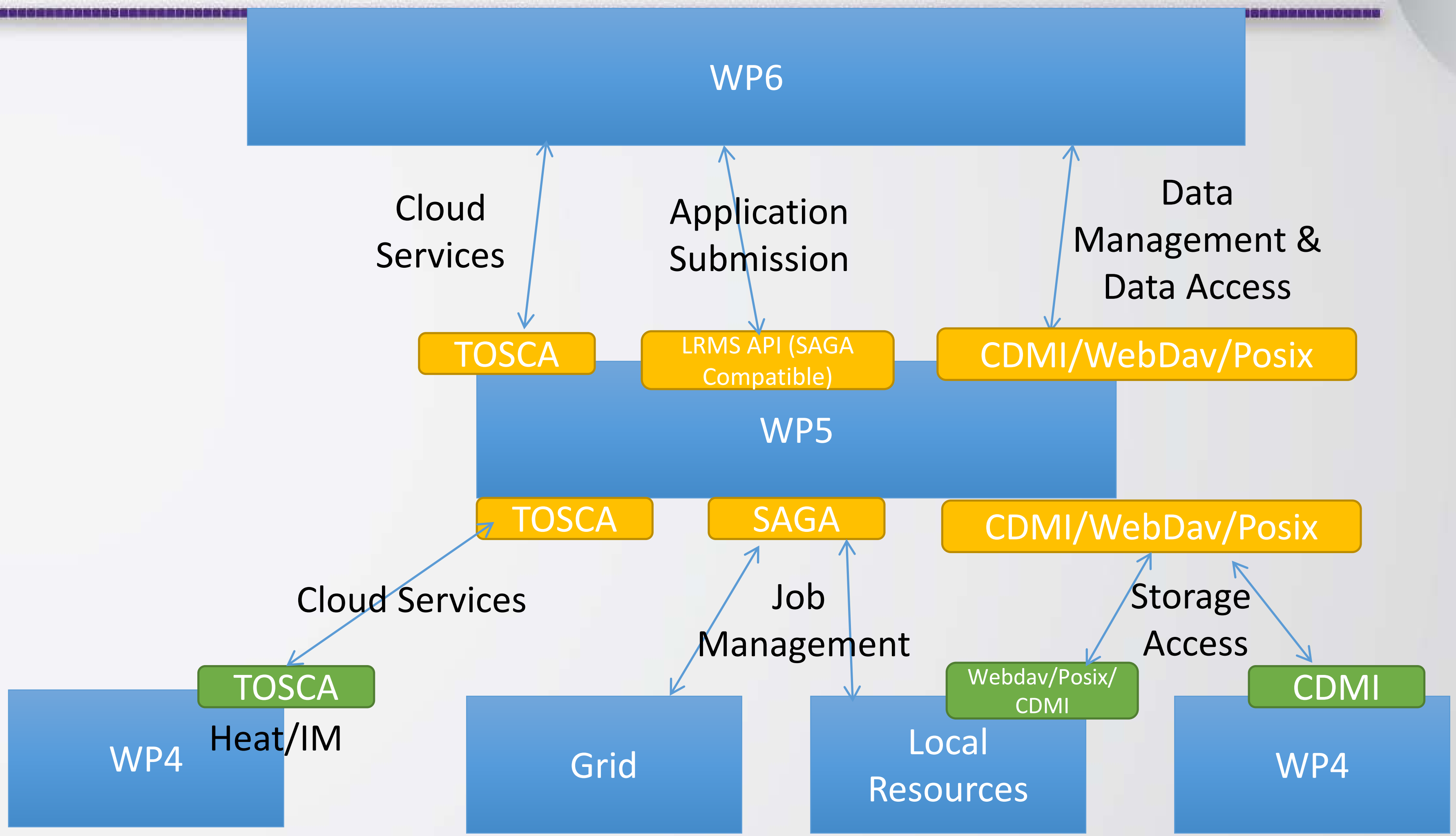
Overall Architecture

Color codes:
Yellow: implementation based on already available solution to be improved/changed;

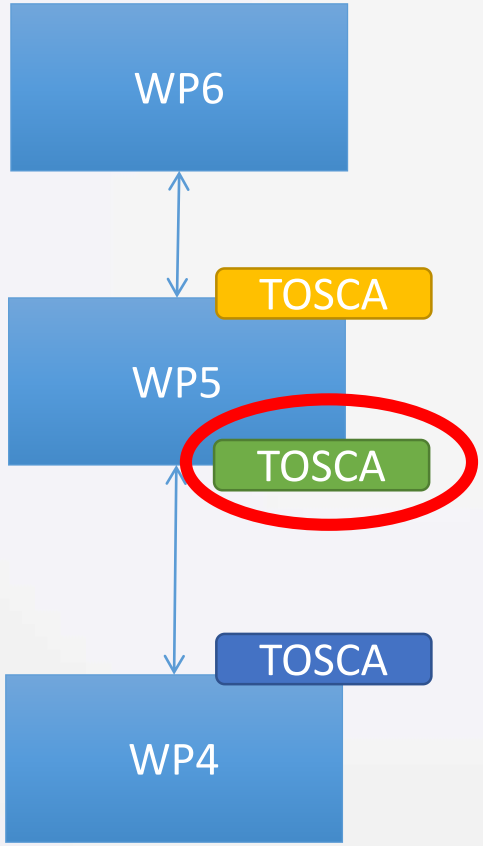
Orange:
Completely new services to be implemented



Interface among WPs



Use of Standards



- OASIS TOSCA (Topology and Orchestration Specification for Cloud Applications) 1.0 (11/2013)
  - Interoperable description of application and infrastructure cloud services, the relationships between parts of the service, and the operational behavior of these services (e.g., deploy, patch, shutdown) independent of the supplier creating the service, and any particular cloud provider or hosting technology.

100+ participants from 40+ companies:

Source: http://www.slideshare.net/OpenTOSCA/tosca-and-opentosca-tosca-introduction-and-opentosca-ecosystem-overview?ref=http://www.slideshare.net/OpenTOSCA/slideshelf

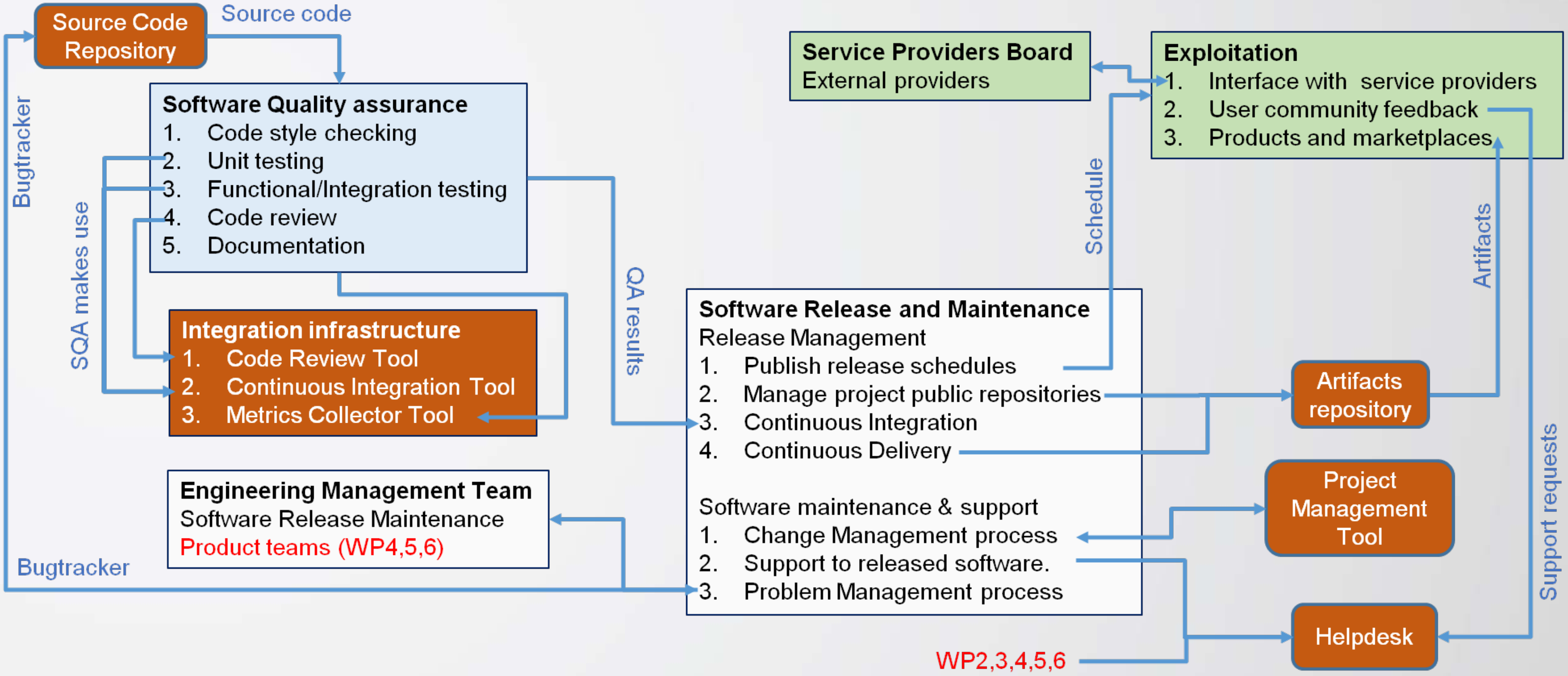
Unified Data Access

- Combining feature sets of products such as Onedata, FTS and DynaFed
- Data set registry - provides unified vision of geographically distributed data set
- API for data and metadata management including: registration, migration, replication, sharing
- Unified optimized data access for federated data based on APIs or POSIX
- Optimization of remote data access by remote block transfer on the fly and local caching
- Access Control Lists management for federated data
- Gateway to external data repositories (EGI, EUDAT, Virtual Observatory)
- Open data gateway

PaaS-level Unified Storage Interfaces

- Data access methods and protocols:
  - CDMI, Web GUI, WebDAV, S3
  - POSIX and mounted virtual volume
- Data locations:
  (some standardization effort required here)
  - Onedata locations via CDMI
  - DynaFed location API (or WebDAV)
- Data migrations and replications:
  (some standardization effort required here)
  - FTS data migration REST API
  - Onedata data migration REST API
  - Onedata CDMI-extended for replications based on metadata attributes

Software development, release and maintenance



Source Code Repository

Software Quality assurance
1. Code style checking
2. Unit testing
3. Functional/Integration testing
4. Code review
5. Documentation

Integration infrastructure
1. Code Review Tool
2. Continuous Integration Tool
3. Metrics Collector Tool

Engineering Management Team
Software Release Maintenance
Product teams (WP4,5,6)

Service Providers Board
External providers

Exploitation
1. Interface with service providers
2. User community feedback
3. Products and marketplaces

Software Release and Maintenance
Release Management
1. Publish release schedules
2. Manage project public repositories
3. Continuous Integration
4. Continuous Delivery

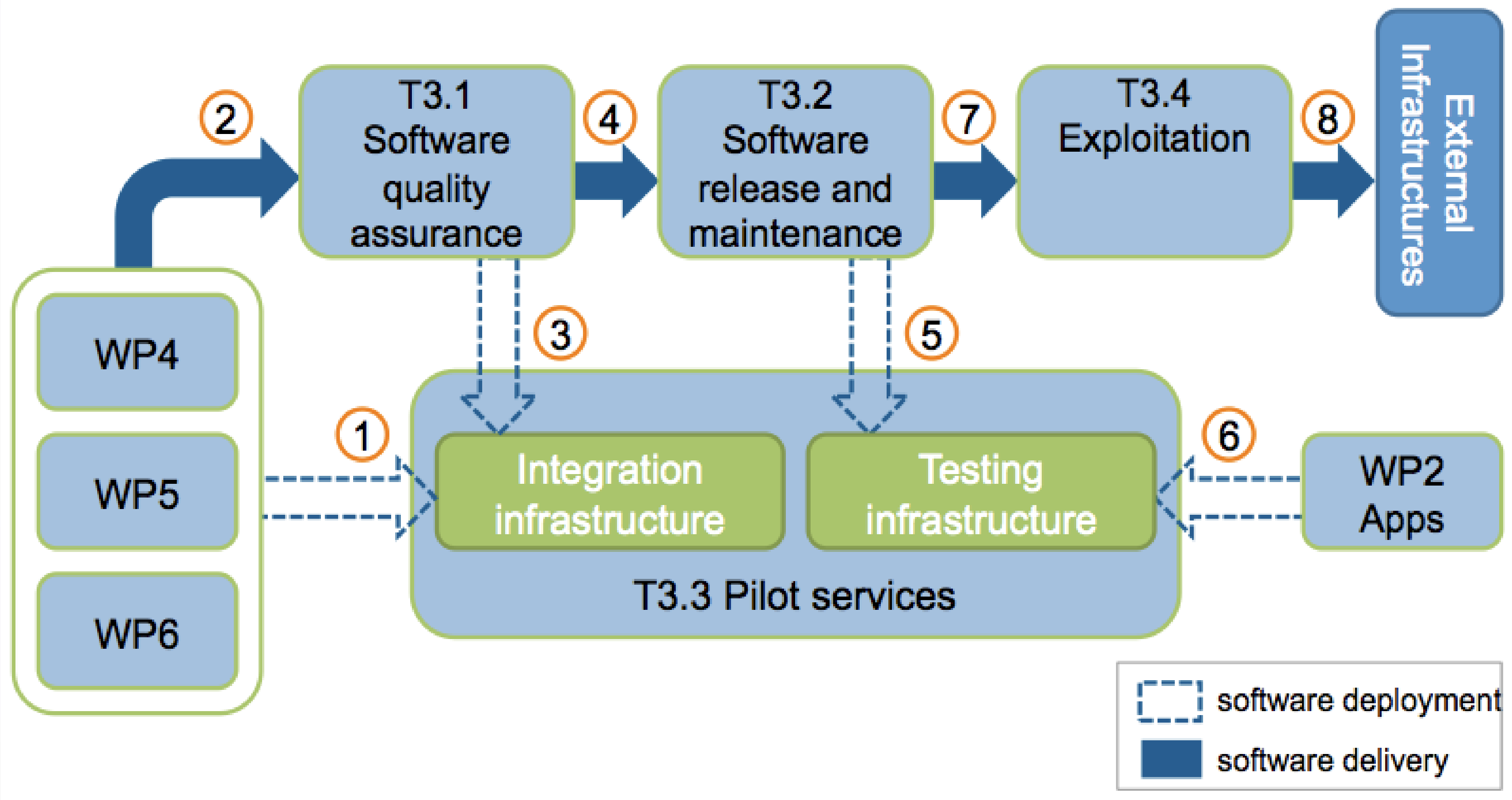
Software maintenance & support
1. Change Management process
2. Support to released software.
3. Problem Management process

Artifacts repository

Project Management Tool

Helpdesk

Testing Activities



2 T3.1 Software quality assurance

4 T3.2 Software release and maintenance

7 T3.4 Exploitation

8 External Infrastructures

3

5

WP4

WP5

WP6

1 Integration infrastructure

Testing infrastructure

6 WP2 Apps

T3.3 Pilot services

⬚ software deployment

■ software delivery

(Some of the) Next Steps

- **Bridging** the differences, exploiting distributed competence, joining forces across communities and projects moving toward a European Science Cloud / Digital Single Market
- **Early development and pilot test-beds** open to diverse communities
  - *Open* is the keyword here: open tools, standards, data, results
  - Continuous checks through QA & KPI tables
- **Communication and dissemination** activities
  - Tracking / helping people unaware of existing solutions or efforts born out of similar use cases
  - Effort on consolidating evolving requirements

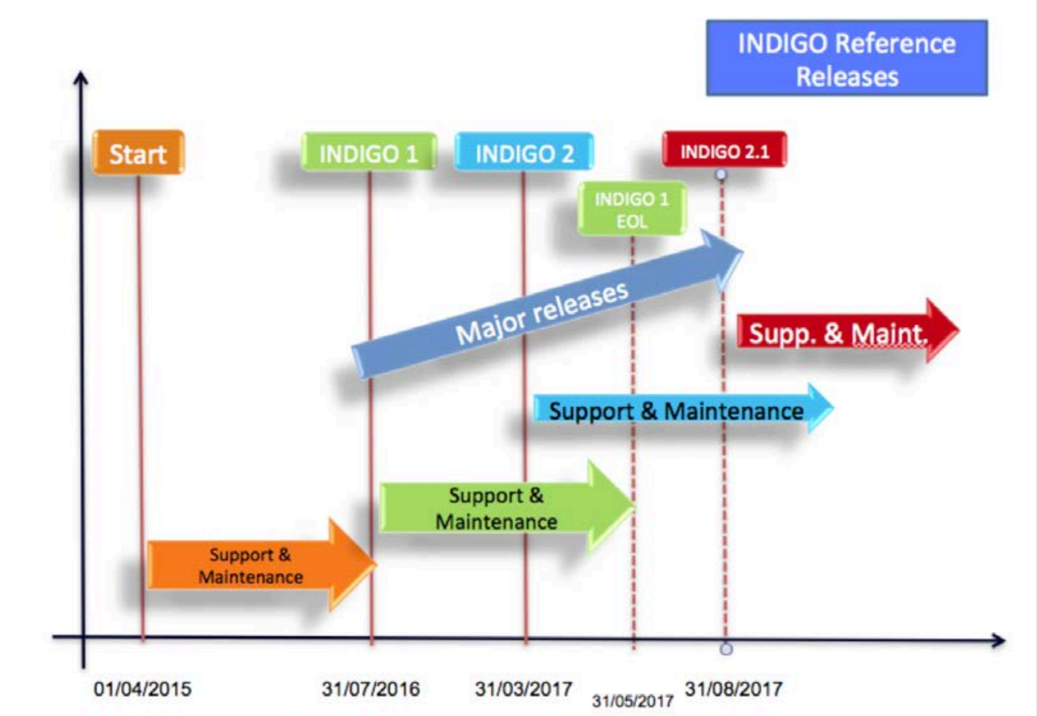


Figure 8: INDIGO-DataCloud Release Timeline

Thanks

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

davide.salomoni@cnaf.infn.it

giacinto.donvito@cnaf.infn.it