Geographically distributed PaaS and User Interface toolkit for scientific applications in the INDIGO-DataCloud

Giacinto Donvito (INFN-Bari)

Lukasz Dutka (CYFRONET)

Marcin Plociennik (PSNC)

Roberto Barbera (INFN-Catania)

Ignacio Blacher (UPV)

German Molto (UPV)

Andrea Ceccanti (INFN-Bari)

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

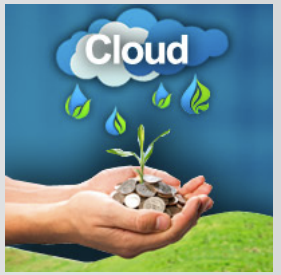
Agenda

- Brief introduction about INDIGO-DataCloud
- INDIGO PaaS general overview
  - PaaS core
  - AAI service
  - Scheduling/Deploying application/services
  - Data access/transfers
- Gateways, Workflows and frontend APIs
  - Libraries and Toolkits
  - Science Gateways
  - Scientific Workflow
- Conclusions

How do we see distributed computing in the future

INDIGO - DataCloud

1. Ease of access and use for small and big collaborations alike.
2. Software and economic sustainability.
3. Robustness (no single points of failure).
4. Modular, scalable architecture.
5. Open source software, vendor independence, hybrid infrastructures.



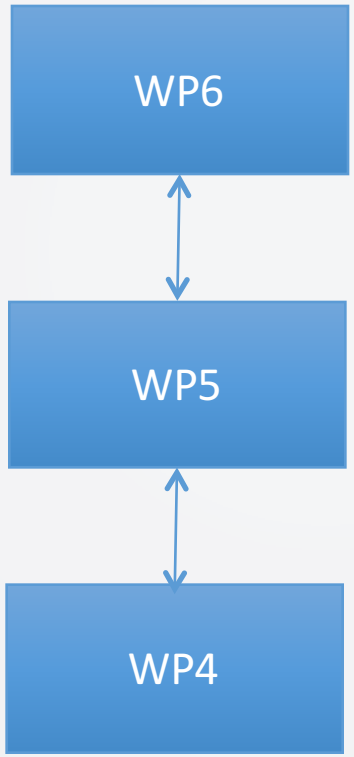
INDIGO approach

INDIGO - DataCloud

- Based on **Open Source** solutions
- widely **supported** by big communities
- whenever possible exploit **general solutions** instead of specific tools/services
  - or put effort in **increasing the generality** of tools developed in a given community
  - this will be important for **sustainability** of the architecture
- ensure that the framework offered to final users, as well as to developers, will have a **low learning curve**
  - **existing software suites** like ROOT, OCTAVE/MATLAB, MATHEMATICA or R-STUDIO, **will be supported** and offered in a transparent way

On the Boundaries between Work Packages

Summary of the JRA WPs



APIs, GUIs
Libraries and Toolkits (T.6.1), Science Gateways and User Interfaces (T6.2), Big data e-science workflows (T6.3)

PaaS
Architecture & Implementation (T5.1), Security & Authorization (T5.2), High-level Geographical Scheduling (T5.3), Unified Data Access (T5.4)

IaaS: CMPs* on steroids
Cloud Computing Virtualization/containers (T4.1), Cloud Storage Virtualization (T4.2), Network Virtualization (T4.3)

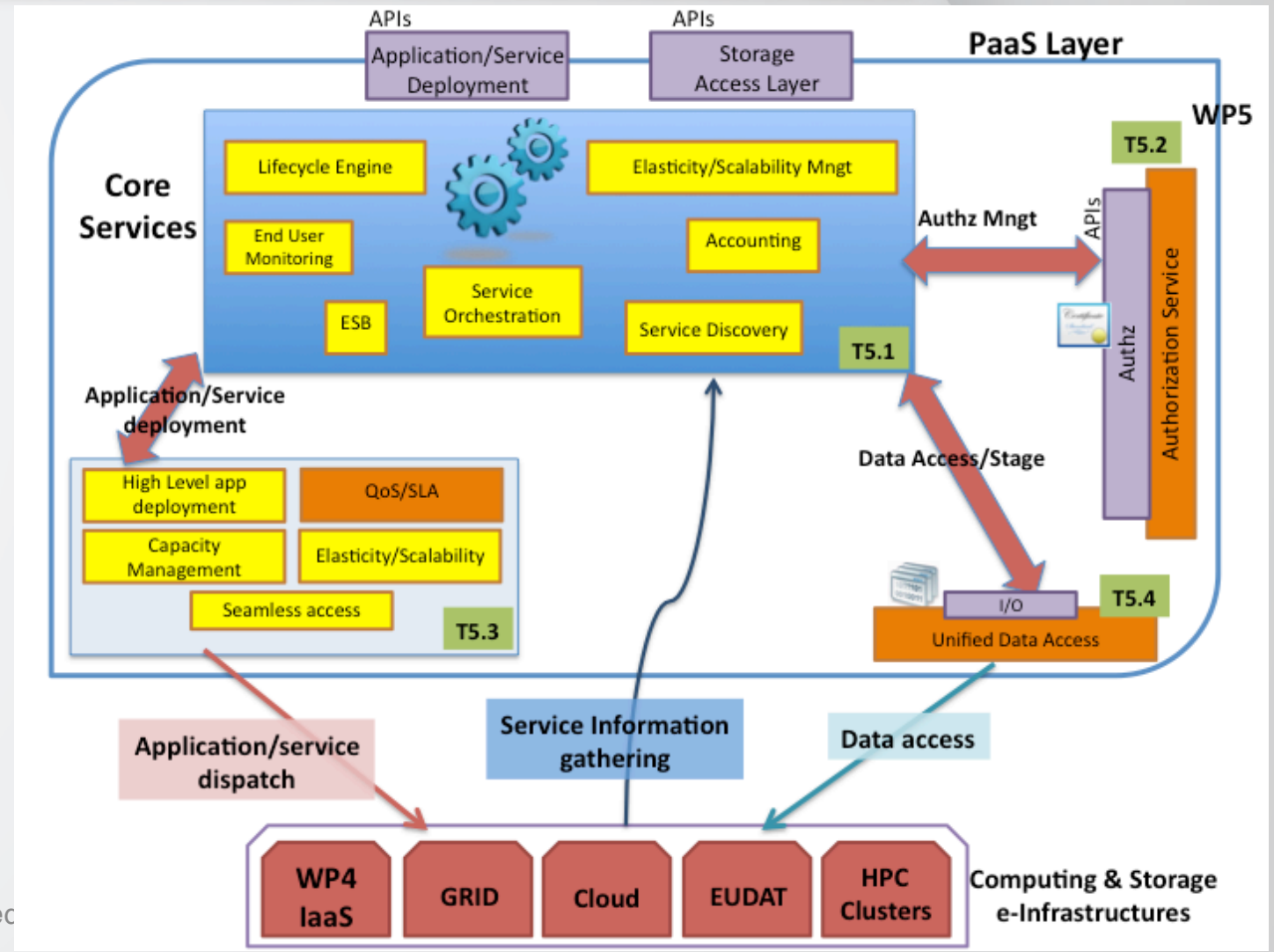
* CMP: Cloud Management Platform (e.g. OpenNebula, OpenStack)

WP5: PaaS Platform Development

INDIGO - DataCloud

- **Task 5.1**: PaaS core architecture and implementation
- **Task 5.2**: Security and Authorization
- **Task 5.3**: High-level geographical application/service deployment
- **Task 5.4**: Unified Data Access

- Focus on using standards as much as possible in order to guarantee the interoperability



Task 5.1: PaaS core architecture and implementation

- The PaaS platform will provide application developers with a way to describe and interact with a set of meta-models that define functional requirements and characteristics of the applications.
  - Application developers will express their needs as services (e.g. processing worker, synchronization service, database, storage, etc.) delegating to the IaaS orchestrator (Heat and similar) the task of configuring, contextualizing and deploying them.
- This task aims to design and develop 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.
- To this aim, high-level programming models will be supported by implementing an abstraction layer specifically designed to access programmatically a set of core services.

Task 5.1: PaaS core architecture and implementation

- Such services will provide basic functionalities to both execute scientific application in a distributed environment as Cloud, and to access federated Cloud-based and physical computing resource.
- we will consider the current available standards such as OASIS CAMP and TOSCA for application management and automation at the level of the PaaS abstraction layer.
- The core will be an orchestration system that will receive standard descriptions of services or applications (e.g. TOSCA) and their interactions with other services or applications.
- This orchestration system will also coordinate the services and layers coming from the other WP5 tasks
  - coordinating the capabilities of the application deployment (provided by Task 5.3) with the stage-in and stage-out features (provided by Task 5.4) in order to fulfill the user-requests.

Task 5.1: PaaS core architecture and implementation

- There is a wide range of general solutions available (OpenShift, CloudFoundry, WSO2, Cloudify 3.0) and specific implementations addressing additional requirements and features in national projects (PRISMA, OCP, PLGrid, PLATON, CODECLOUD) and European initiatives (Helix Nebula, EGI, VENUS-C). The orchestrator will reuse and extend these technologies.
- Other key components of the PaaS platform implemented in the T5.1 will be:
  - A scalable system to manage the accounting of the computing and storage resources usage over heterogeneous resources in a geographically distributed environment on a single combined repository.
  - The life cycle management engine for granting the management/update/configuration of the deployed services
  - A powerful and scalable information system

Task 5.3: Goals

- Managing both job-like scheduling and classic deployment for the standard PaaS-level services (DBaaS, Application as a Service, etc)
- Executing applications in different computational infrastructures, by hiding the real underlying computing architecture
- Matching user/service requirements against the availability of the resources
- Deploying both user-provided applications and already available application/services
- Scheduling application or service execution based on the data location
- Managing the resource providers capacity, according to the users requests and agreed SLAs

Task 5.3: Goals

INDIGO - DataCloud

- Providing the automatic scalability needed for the efficient execution of the specific service or application
- Dealing with complex clusters of services on demand such as Hadoop, batch cluster on demand, cluster of diverse services (Database+application server, etc).
- Exploiting the available resources based on high-level priority algorithms: priority scheduling, deadline scheduling, estimated cost, QoS, FairShare, etc.
- Providing automatic instantiation of the pilot factory to better exploit the available computational resources
- Taking care of the failure and needed retries for both the application execution and service deployment.

Task 5.1 -- 5.3: Technologies

INDIGO - DataCloud

- OASIS TOSCA (Topology and Orchestration Specification for Cloud Applications) 1.0 (11/2013)
  - Interfacing with HEAT at the level of IaaS will be “easy”
- OGF OCCI (Open Cloud Computing Interface) 1.1 (06/2011)
- Apache Brooklyn
  - Framework for modeling, monitoring, and managing applications through autonomic blueprints. Brooklyn blueprints conform to CAMP v1.1 Public Review Draft 01.
- PaaS Open Source solutions:
  - OpenShift, WSO2, Cloudify, Mesos, SlipStream
  - The partners has very deep knowledge of the topic

The AAI problem

INDIGO - DataCloud

- Heterogeneous infrastructures use heterogeneous authentication/authorization mechanisms
  - Hard to integrate resources from distributed infrastructures without common AAI ground
- Even where a single authentication technology is used, managing user and privileges on distributed resources in a **dynamic** and secure way is complex
- DCIs are not easily and securely accessible from common users
  - Federated identity support lacking or very limited

AAI: main challenges

- How can we have common auhtN and auhtZ primitives that “just work” across several distributed infrastructures?
- Which tools should we provide to our users so that they have complete control on how authN and authZ is configured and performed on the resources (assembled from distributed providers) they will use for their research?
- How do we avoid reinventing the wheel? How do we exploit what is already available, leverage existing standards and ensure that what we develop is sustainable?

AAI: Technical challenges (I)

- Provide a layer where identities provided by different sources can be can be managed in a uniform way
- Define how attributes linked to this identities (on which authorization decisions are based) are represented and understood at lower and higher levels of the INDIGO stack
- Define a cryptografically strong token used to carry these attributes around in a secure way
- Define how the token carrying the attributes is exchanged between services
- Define how controlled delegation of privileges can happen

AAI: Technical challenges (II)

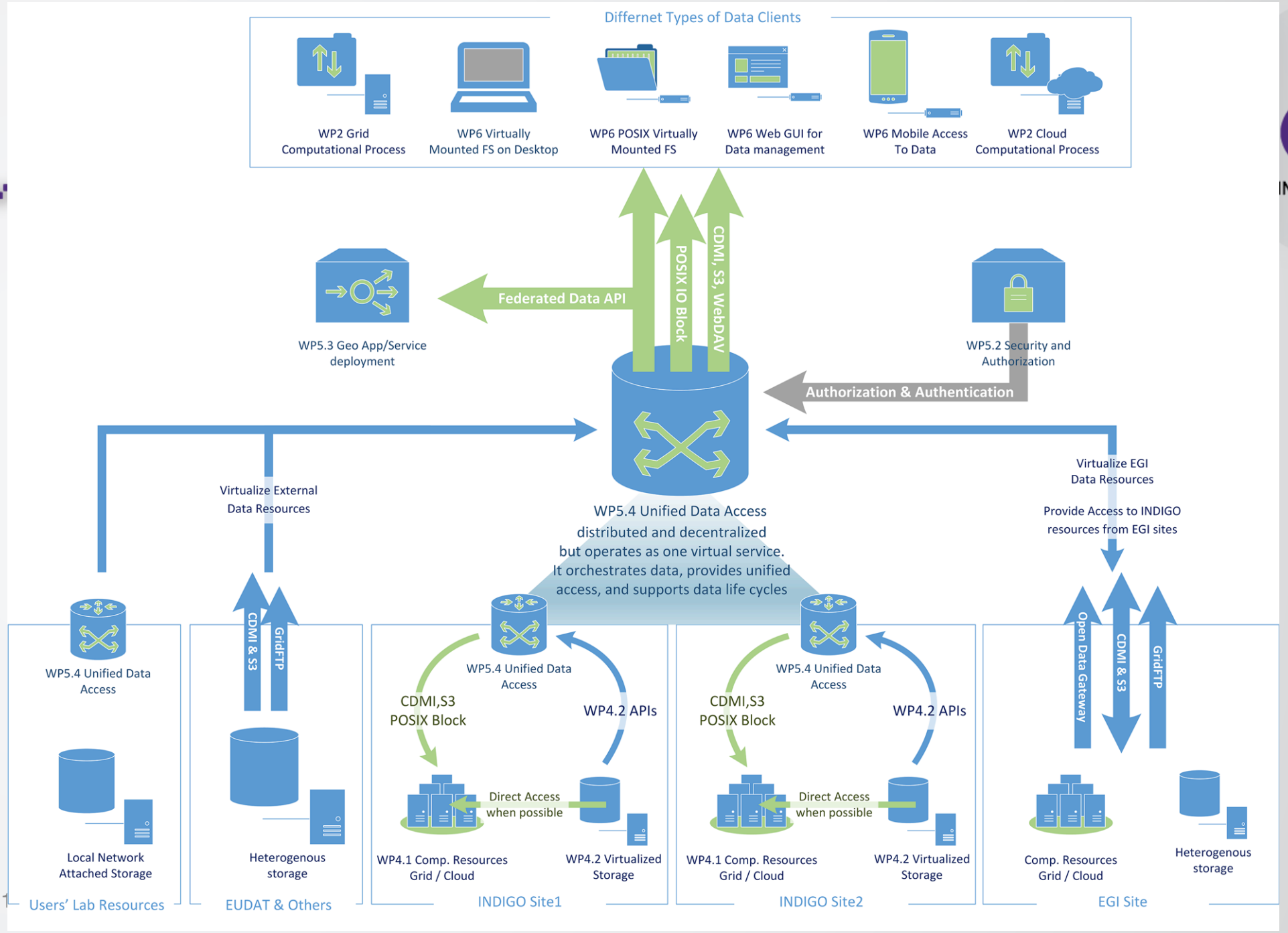
- Provide the tools to support cross-organizational user and privilege management
  - Group management
  - Enrollment flows management
- Provide tools to define, propagate, compose and enforce authorization policies based on these attributes at various levels of the INDIGO stack
  - Uniform and consistent authZ over resources assembled from multiple, heterogeneous providers

AAI: IAM service technologies

- Standard APIs/protocols for user/group management
  - SCIM, VOOT
- Federated AuthN support
  - SAML, OpenID connect
- Attribute authority/token service
  - SAML, OAuth
- Policy definition and composition
  - XACML

Task 5.4: Objectives

- UNIFICATION - Provide a set of services for uniform data access despite geographical location and storage technology;
- FEDERATED DATA ACCESS - Provide capabilities of data access federated access
- INTEROPERABILITY and OPEN DATA - Provide an interoperability and enabling cross e-infrastructures data storage
- OPTIMIZATION and DATA ON THE FLY - Manage access to data


Differnet Types of Data Clients
WP2 Grid Computational Process
WP6 Virtually Mounted FS on Desktop
WP6 POSIX Virtually Mounted FS
WP6 Web GUI for Data management
WP6 Mobile Access To Data
WP2 Cloud Computational Process
Federated Data API
POSIX IO Block
CDMI, S3, WebDAV
WP5.3 Geo App/Service deployment
WP5.2 Security and Authorization
Authorization & Authentication
Virtualize External Data Resources
Virtualize EGI Data Resources
Provide Access to INDIGO resources from EGI sites
WP5.4 Unified Data Access
distributed and decentralized
but operates as one virtual service.
It orchestrates data, provides unified
access, and supports data life cycles
CDMI & S3
GridFTP
Open Data Gateway
CDMI,S3
POSIX Block
WP4.2 APIs
Direct Access when possible
Local Network Attached Storage
Heterogenous storage
WP4.1 Comp. Resources Grid / Cloud
WP4.2 Virtualized Storage
Comp. Resources Grid / Cloud
Users' Lab Resources
EUDAT & Others
INDIGO Site1
INDIGO Site2
EGI Site

Clients and Interfaces

- Multiple types of clients: grid, cloud, virtual access from users desktops, high-throughput POSIX access, HTTP Based Interfaces
- Several type of protocols:
  - REST for replica and data location management
  - CDMI, S3, WebDAV for data access
- AAI integrated with Indigo generic methods of authentication
- High-grain access control to data.

WP6: Science Gateways, Workflows and Toolkits

WP6 overview – Objectives (1/2)

- WP6: Science Gateways, Workflows and Toolkits
  - **Develop Toolkits** (libraries) that will allow the platform usage from the level of the Scientific Gateways, desktop and mobile applications
  - Provide and develop the **Open Source Mobile Application Toolkit** for the iOS, Android and Windows Phone platform that will be the base for development of the Mobile Apps
  - Provide the **User Friendly front end's**, that will prove the usability of the PaaS proposed
  - Provide both a **general-purpose multi-domain Science Gateway** and customized examples for selected user communities/scenarios, that will make use of the proposed Toolkits, including **Data Analytics Gateways for e-Science**

WP6 overview – Objectives (2/2)

- Develop example cross platform native Mobile Apps for selected use cases, based on the Mobile App Toolkit;
- Manage the execution of complex workflows using PaaS layers;
- Support for both interactive and batch parallel data analytics workflows.

- Provide the dynamic scientific workflows services in a Workflows-as-a-Service model
- Provide workflow interfaces extensions for distributed and parallel data analytics on large volume of scientific, multidimensional data

Task 6.1 - Libraries and Toolkits

- **Develop Toolkits** (libraries) that will allow the platform usage from the level of the Scientific Gateways, desktop and mobile applications
- The overall goal of this task is to build a set of libraries and toolkits on the REST APIs developed by WP5
- The aim of these libraries and toolkits is to simplify the development process and speed up the creation of science gateways and desktop and mobile applications
- The lower layer will map the REST APIs provided by the underneath services and this will be shared in all the components. Several implementation (e.g. Java, Python, Objective-C) will be evaluated and developed taking into account VRCs requirements

Task 6.1 - Libraries and Toolkits

- To obtain authentication tokens (AuthN) that enable the user to authenticate to the PaaS layer
- Providing high-level APIs to enable seamless access (from the perspective of user and its application) to heterogenous storage systems and data
- Extend the concept of SAGA adaptors to pilot-job-based commercial cloud middleware stacks and grid middleware stacks
- Add PaaS interfaces (WP5) to JSAGA

Task 6.2.1 - Science Gateways

- The development of the Science Gateways will be driven by the requirements from VRC collected by WP2
- Will be based on the APIs provided by WP5 through T6.1 libraries
- Will provide example implementation for T6.1, and general purpose SG for small communities
- Enabling data analytics functionalities like data sub-setting, data reduction/aggregation, statistical data analysis, pivoting, time series analysis, along with data summaries, multiple charts, tables, reports, etc., to support big data analytics over large multi-dimensional datasets in different scientific domains (e.g. Earth Sciences, Life Sciences)

Task 6.2.2 - Mobile Apps

- Basing on the initial analysis of the applications/communities requirements several selected cross-platform native apps for mobile appliances will be developed
- Such apps will make use of the toolkits developed in T6.1
- The provided functionality and presentation layer will differ from case to case and will include the authentication, authorization, access to data, monitoring to status of data processing, application management

Task 6.3.1 - Provide scientific workflow support in a “Workflows as a Service” model

- Dynamic support for scientific workflows management according to a “Workflow as a Service” (WaaS) model
- Two different levels of workflow engines will be supported to address different needs:
  - “coarse grain”, targeting distributed (loosely coupled) experiments, through workflows orchestration across heterogeneous set of services;
  - “fine grain”, targeting high performance (tightly coupled) data analysis through workflows orchestration on big data analytics frameworks
- Components/modules to allow using WP5 PaaS within selected scientific workflows currently used by the user communities will be also developed: Galaxy, Ophidia, Taverna, Kepler, etc.
- Targeted use cases: climate change scientists, bioinformatics, biophysicists, astrophysics, structural and molecular biologists

Task 6.3.2 - Workflow interface extensions for big data analytics

- Cross-domain extensions supporting:
  - massive data analytics operators (single operator, multiple datasets) through application-level search and filter capabilities able to define/characterize (for a single task) very large inputs (from tens to potentially millions of datasets) with a single declarative statement
  - automated data processing ("rule-based") for triggering specific, user-defined actions whenever some events occur. An Event-Condition-Action (ECA) logic will be implemented
  - interactivity to address supervised scenarios for scientific data analysis. Such a feature will allow interacting with Science Gateways, Desktop applications and Mobile applications in more dynamic/reactive manner, for intermediate results delivery and visualisation. Interactivity will also allow the users a real-time control/tuning of the experiment parameters
  - interleaved support to run multiple, interleaved analytics chains at the same time (e.g. pipeline approach), to increase the workflow efficiency and minimize the time to solution

Lots of work ahead of us …

… tight deadlines and relatively scarce effort

but

we do not start from scratch!

and the partners involved in the implementation has very high level of expertise in the field and we are already hardly working together…