The Evolution of the INDIGO-DataCloud Architecture

Davide Salomoni
davide@infn.it

INFN-CNAF

ISGC 2019, Taipei
April 5, 2019

Outline

The long title of this talk

"The evolution of INDIGO-DataCloud: toward an advanced open source Cloud platform integrating AI-based workflows exploiting large-scale Big Data facilities"

Note the assorted use of several fancy keywords.

The long title of this talk

"The evolution of INDIGO-DataCloud: toward an advanced open source Cloud platform integrating AI-based workflows exploiting large-scale Big Data facilities"

Note the assorted use of several fancy keywords.

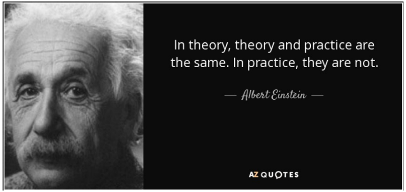
I will focus on:

- AI-based workflows (e.g. deep learning-based)
- Big data management
- Exploitation of large-scale facilities

Table of Contents

A typical data processing workflow

- In a naïve set of assumptions, I *have*:
  - A data set I want to analyze.
  - Some algorithms I want to apply to this data.
  - Some software that can use these algorithms.
  - Some computing resources that can run this software.
  - Some space where I can store my output.
- I *assemble everything together* and off I am.



In fact, there are several challenges

(which go well beyond the "FAIR data" mantra)

- **Accessing Data**:
  - Is the data open? For all? Always?
  - Is the data distributed? Where? How do I find and integrate it?

In fact, there are several challenges

(which go well beyond the "FAIR data" mantra)

- **Accessing Data**:
  - Is the data open? For all? Always?
  - Is the data distributed? Where? How do I find and integrate it?
- **Processing Data**:
  - Where can I find the resources I need for my workflow / data processing?
  - Does all my data require the same QoS? The same algorithms? How do I decide that?
  - How open are the tools that will process my data?
  - What happens if some services are not available? If there is a failure somewhere?

In fact, there are several challenges

(which go well beyond the "FAIR data" mantra)

- **Accessing Data**:
  - Is the data open? For all? Always?
  - Is the data distributed? Where? How do I find and integrate it?
- **Processing Data**:
  - Where can I find the resources I need for my workflow / data processing?
  - Does all my data require the same QoS? The same algorithms? How do I decide that?
  - How open are the tools that will process my data?
  - What happens if some services are not available? If there is a failure somewhere?
- **Post-Processing Data**:
  - What if I get new data? How do I re-train my model?
  - How can I reproduce, tweak and (re-)publish my work?

In fact, there are several challenges

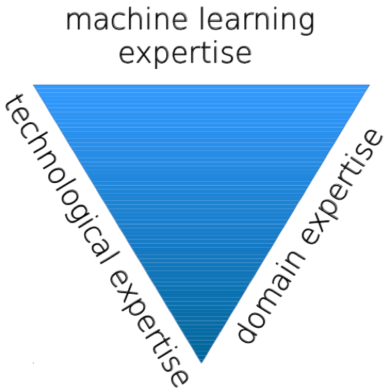
(which go well beyond the "FAIR data" mantra)

- **Accessing Data**:
  - Is the data open? For all? Always?
  - Is the data distributed? Where? How do I find and integrate it?
- **Processing Data**:
  - Where can I find the resources I need for my workflow / data processing?
  - Does all my data require the same QoS? The same algorithms? How do I decide that?
  - How open are the tools that will process my data?
  - What happens if some services are not available? If there is a failure somewhere?
- **Post-Processing Data**:
  - What if I get new data? How do I re-train my model?
  - How can I reproduce, tweak and (re-)publish my work?

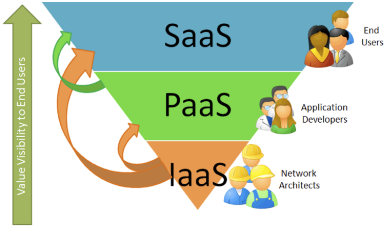
In the end, **how much effort and know-how** is needed to have all this in place?

Take machine learning. Which know-how do I need?

Istituto Nazionale di Fisica Nucleare

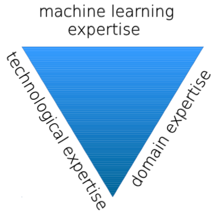


Choices...



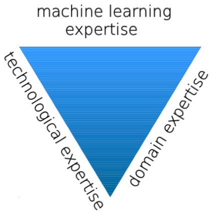
What matters, at the end, are the *applications*.
But how to properly get to the Application level?

Categorizing the know-how



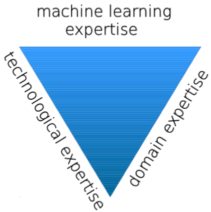
- **Category 1**: Deploy an already trained ML network for somebody else to use on his/her own data set.
  - Domain knowledge

Categorizing the know-how



- **Category 1**: Deploy an already trained ML network for somebody else to use on his/her own data set.
  - Domain knowledge
- **Category 2**: Retrain (parts of) an already trained ML network to make use of its inherent knowledge and solve a new learning task.
  - Domain + machine learning knowledge

Categorizing the know-how



- **Category 1**: Deploy an already trained ML network for somebody else to use on his/her own data set.
  - Domain knowledge
- **Category 2**: Retrain (parts of) an already trained ML network to make use of its inherent knowledge and solve a new learning task.
  - Domain + machine learning knowledge
- **Category 3**: Completely work through the ML / deep learning cycle with data selection, model architecture, training and testing.
  - Domain + machine + technological knowledge

How do we go about this?

- **Objective 1: build added value services on top of IaaS & PaaS infrastructures**
  - Due to the nature of many scientific endeavors (but also public services and industry), these infrastructures may often be hybrid, i.e. public + private.

How do we go about this?

- **Objective 1: build added value services on top of IaaS & PaaS infrastructures**
  - Due to the nature of many scientific endeavors (but also public services and industry), these infrastructures may often be hybrid, i.e. public + private.
- **Objective 2: lower the entry barrier for non-skilled scientists**
  - Transparent ("ZeroOps") execution on e-Infrastructures.
  - Offer ready-to-use modules, components or services through a catalog, or rather a configurable marketplace.
  - Enable flexible service composition.
  - Implement common software development techniques also for scientists' applications ("DevOps").

Table of Contents

The foundation project: **INDIGO-DataCloud**

INFN
Istituto Nazionale di Fisica Nucleare

INDIGO - DataCloud

- **A European Horizon2020** project which ran from April 2015 to September 2017. **26 European partners** located in 11 European countries, coordinated by the Italian National Institute for Nuclear Physics (INFN).

The foundation project: **INDIGO-DataCloud**

- **A European Horizon2020** project which ran from April 2015 to September 2017. **26 European partners** located in 11 European countries, coordinated by the Italian National Institute for Nuclear Physics (INFN).
- **Task**: develop an open source Cloud platform for computing and data ("DataCloud").
  - Applicable to multi-disciplinary communities, such as biology, physics, cultural heritage, astrophysics, life science, climatology.

The foundation project: **INDIGO-DataCloud**

INFN
Istituto Nazionale di Fisica Nucleare

INDIGO - DataCloud

- **A European Horizon2020** project which ran from April 2015 to September 2017. **26 European partners** located in 11 European countries, coordinated by the Italian National Institute for Nuclear Physics (INFN).
- **Task**: develop an open source Cloud platform for computing and data ("DataCloud").
  - Applicable to multi-disciplinary communities, such as biology, physics, 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

The foundation project: **INDIGO-DataCloud**

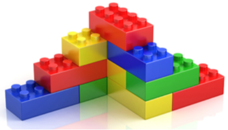
Istituto Nazionale di Fisica Nucleare

INDIGO - DataCloud

- **A European Horizon2020** project which ran from April 2015 to September 2017. **26 European partners** located in 11 European countries, coordinated by the Italian National Institute for Nuclear Physics (INFN).
- **Task**: develop an open source Cloud platform for computing and data ("DataCloud").
  - Applicable to multi-disciplinary communities, such as biology, physics, 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**: address technological needs of scientists wishing to easily exploit distributed compute and data resources.

The INDIGO-DataCloud main outcomes

- With INDIGO, we provided **open source tools and services** for:
  - A common, standards-based AAI model and implementation.
  - Independence from IaaS infrastructures.
  - Several PaaS modules, composable through a standard language (TOSCA).
  - Compute orchestration.
  - Web and mobile based interfaces.
- See the **ElectricIndigo software catalogue** (`https://www.indigo-datacloud.eu/service-component`):
  - 47 open source modular components, distributed via 170 software packages, 50 ready-to-use Docker containers.



This laid the foundation for the creation of new added value services.

A concrete example: **DODAS**

- DODAS is a service obtained by the composition of several INDIGO components.

A concrete example: **DODAS**

- DODAS is a service obtained by the composition of several INDIGO components.
- It provides deployment of complex cluster set-ups on "any cloud provider"† with almost zero effort ("ZeroOps").
  - As easy as creating a virtual machine on a IaaS: a simple one-click solution.
  - DODAS configuration details are stored in high-level TOSCA templates.
  - It allows to instantiate on-demand microservices and container-based clusters to execute software applications.

†Verified on OpenStack-based clouds (public and private), AWS, Google Compute Cloud, Microsoft Azure.

A concrete example: **DODAS**

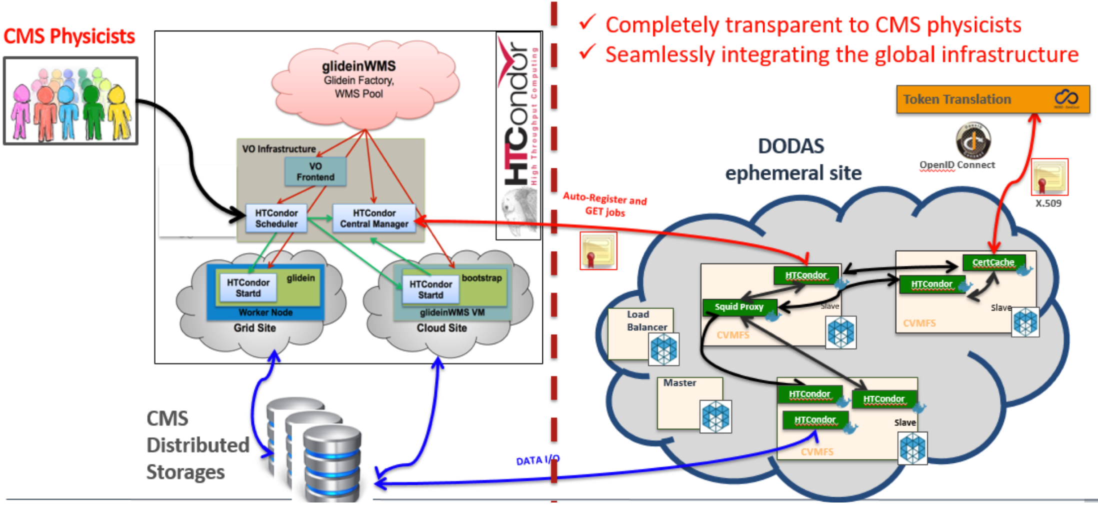
Istituto Nazionale di Fisica Nucleare

- DODAS is a service obtained by the composition of several INDIGO components.
- It provides deployment of complex cluster set-ups on "any cloud provider"† with almost zero effort ("ZeroOps").
  - As easy as creating a virtual machine on a IaaS: a simple one-click solution.
  - DODAS configuration details are stored in high-level TOSCA templates.
  - It allows to instantiate on-demand microservices and container-based clusters to execute software applications.
- DODAS currently provides support to generate:
  - An HTCondor-based **Batch System as a Service**
  - A big data platform for **Machine Learning as a Service**
  - An extension of these two, integrating community-specific services

†Verified on OpenStack-based clouds (public and private), AWS, Google Compute Cloud, Microsoft Azure.

DODAS for CMS

- ✓ Completely transparent to CMS physicists
- ✓ Seamlessly integrating the global infrastructure



DODAS for Machine Learning as a Service

INFN
Istituto Nazionale di Fisica Nucleare

- Analysis of “Data Cache” related metadata flow
  - To improve caching layer management: **Smart Cache**

1. Reading HDFS@CERN data
2. Data enrichment and reduction with Spark jobs
   - Storing of output data in HDFS
3. Analysis of structured data

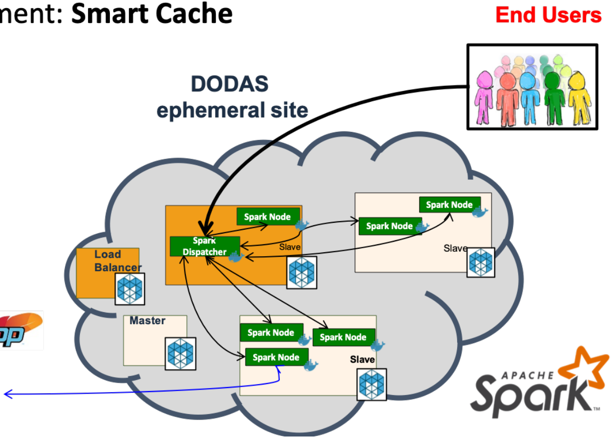
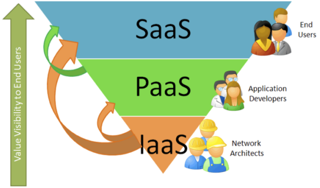


Table of Contents

What Next?



So, we now know how to:

- ✓ Dynamically instantiate clusters over several IaaS.
- ✓ Compose services through high-level languages.
- ✓ Integrate various types of AAI systems.
- ✓ Integrate various components in existing frameworks, monitor and auto-scale them.

But what about deep learning?

Right now...

- Scientists typically create a deep learning application on their personal computers.
- The deep learning model is trained in a GPU-based node (maybe locally) - if available.
- The work (architecture, configuration, results) is published (or not).

However:

- How can a scientist easily offer his results and workflows to a broad audience?
- What about dependencies?

DEEP-Hybrid DataCloud

- **DEEP-HybridDataCloud (DEEP)** is a software development and integration project.
- It promotes the use of intensive computing services by different research communities and areas.
  - It validates its results through use cases that benefit from using hardware accelerators, such as GPUs and low latency interconnects.
  - Pilot cases: diabetic rethinopaty detection, biodiversity applications, online analysis of data streams.
- Key objective: provide a general, distributed architecture and pipeline to **train**, **retrain** and **use** deep learning (and other machine learning) models.
- It deals with heterogeneous datasets, bringing to TRL8 services and prototypes initially at least at TRL6, including them into a unified service catalogue.

From service composition to reusable components

- An application consists on several components that need to be deployed, configured, etc. This leads to service composition.
- Service composition provides a way to re-deploy the same topology over different infrastructures.
- However, scientists should not need to deal with technologies and infrastructures.

The DEEP project has created a set of APIs and a collection of ML-related TOSCA-based templates, available through an open catalog of components. With that, scientists can easily create a deep learning application on their personal computers and later deploy it on a Cloud.

The DEEP Open Catalog

INFN
Istituto Nazionale di Fisica Nucleare

- Collection of ready-to-use modules (for inference, training, retraining, etc.)
  - Comprising machine learning, deep learning, big data analytics tools + corresponding TOSCA templates
  - ML/DL Marketplace: https://marketplace.deep-hybrid-datacloud.eu
  - GitHub: https://github.com/deephdc/
  - DockerHub: https://hub.docker.com/u/deephdc/
- Based on **DEEPaaS API** component
  - Expose underlying model functionality with a common API
  - Follows OpenAPI specifications
  - Minimal modifications to user applications.
- **Goal**: execute the **same module on any** platform and infrastructure:
  - Laptop, workstation, HPC, Kubernetes, Mesos, DEEPaaS, other FaaS frameworks etc.

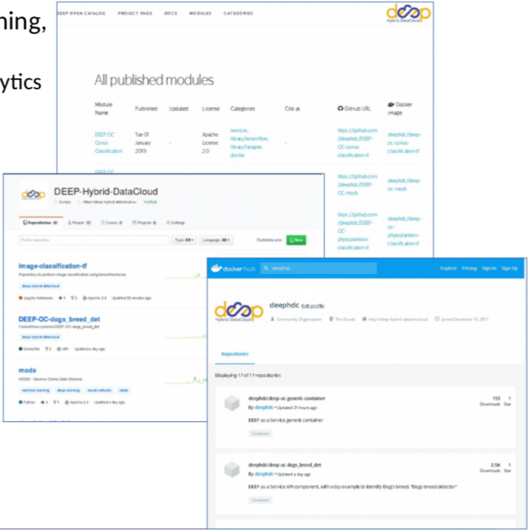
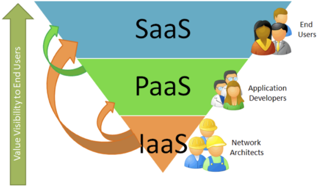


Table of Contents

What Next?



So, we now know how to:

- ✓ Dynamically instantiate clusters over several IaaS.
- ✓ Compose services through high-level languages.
- ✓ Integrate various types of AAI systems.
- ✓ Integrate various components in existing frameworks, monitor and auto-scale them.
- ✓ Re-use deep learning-based building blocks, customize and publish them in a high-level catalog of services.

But what about data management?

Right now...

- When handling data, scientists are typically oblivious to data distribution policies, in particular for:
  - QoS-based (e.g. disks vs. tape vs. SSD) data distribution policies, esp. cross-sites.
  - Data lifecycle management.

However:

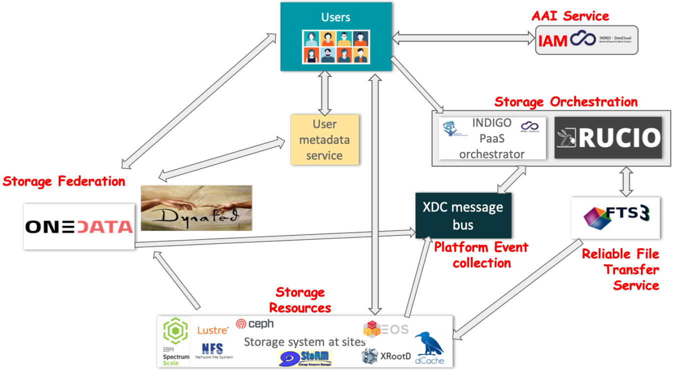
- How to perform data pre-processing during data ingestion?
- How to control how replica management is done?
- How to perform some *smart* data caching, or data management based e.g. on access patterns?
  - For example, automatically move *unused* data to some "glacier-like" storage, and conversely move *hot* data to some fast storage.

eXtreme-DataCloud (XDC)

- **eXtreme-DataCloud (XDC)** is a software development and integration project.
- It develops scalable technologies for federating storage resources and managing data in highly distributed computing environments.
  - Focus on efficient, policy-driven and QoS-based data management.
  - Pilot cases: XFEL, Lifewatch, CTA, WLCG/CMS.
- The target platforms are the current and next generation e-Infrastructures deployed in Europe.
  - The European Open Science Cloud (EOSC).
  - The e-infrastructures used by the represented communities.
- It deals with heterogeneous datasets, bringing to TRL8 services and prototypes initially at least at TRL6, including them into a unified service catalogue.

The XDC architecture

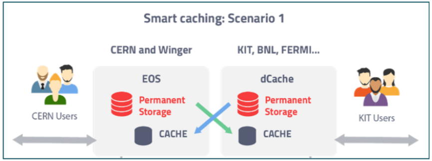
- **Main point**: improve already existing, production-quality Data Management services by <u>adding missing functionalities</u> (such as improved QoS support) requested by research communities.
- Based mainly on technologies provided by the partners and by the INDIGO-DataCloud project.

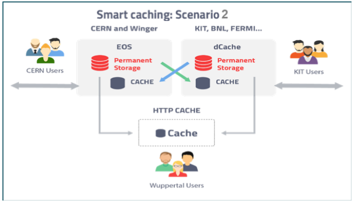


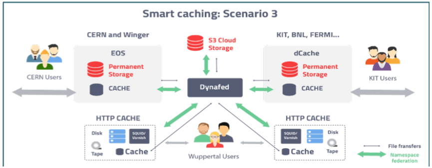
XDC smart caching

Develop a global caching infrastructure supporting the following building blocks:

- dynamic integration of satellite sites by existing data centers;
- creation of standalone caches modeled on current `http` and `xrootd` solutions;
- federation of the above to create a large scale, regional caching infrastructure.







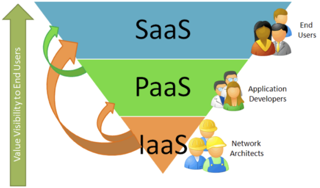
For more info:

- D. Cesini, The eXtreme-DataCloud project: advanced data management services for distributed e-infrastructures (ISGC 2019, April 4, 14:30)
- D. Ciangottini, Integration of the Italian cache federation in the CMS computing model (ISGC 2019, April 5, 09:00)

Table of Contents

What Next?

So, we now know how to:



- ✓ Dynamically instantiate clusters over several IaaS.
- ✓ Compose services through high-level languages.
- ✓ Integrate various types of AAI systems.
- ✓ Integrate various components in existing frameworks, monitor and auto-scale them.
- ✓ Re-use deep learning-based building blocks, customize and publish them in a high-level catalog of services.
- ✓ Perform some data management automation and optimization, as well as call some QoS functions on storage.

Is this it? Did we address all the challenges mentioned above?

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.
- Integrating and automating a full data & compute orchestration solution.

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.
- Integrating and automating a full data & compute orchestration solution.
- Being able to react to events, such as the insertion of new files in catalogues, DB, file systems.

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.
- Integrating and automating a full data & compute orchestration solution.
- Being able to react to events, such as the insertion of new files in catalogues, DB, file systems.
- Exploiting metadata-driven lambda-based processing models (function as a service).

(Some of) The still missing pieces

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.
- Integrating and automating a full data & compute orchestration solution.
- Being able to react to events, such as the insertion of new files in catalogues, DB, file systems.
- Exploiting metadata-driven lambda-based processing models (function as a service).
- Enabling user-centric service composition capable of triggering automated, extensible processing, deployment and monitoring for the above.

(Some of) The still missing pieces

Istituto Nazionale di Fisica Nucleare

We still need to invest effort in developing and integrating **open source modular solutions** capable of:

- Securely connecting to multiple data sources: e-infrastructures, HPC centers, opportunistic resources, devices, storage systems, data sets, sync & share services.
- Integrating and automating a full data & compute orchestration solution.
- Being able to react to events, such as the insertion of new files in catalogues, DB, file systems.
- Exploiting metadata-driven lambda-based processing models (function as a service).
- Enabling user-centric service composition capable of triggering automated, extensible processing, deployment and monitoring for the above.

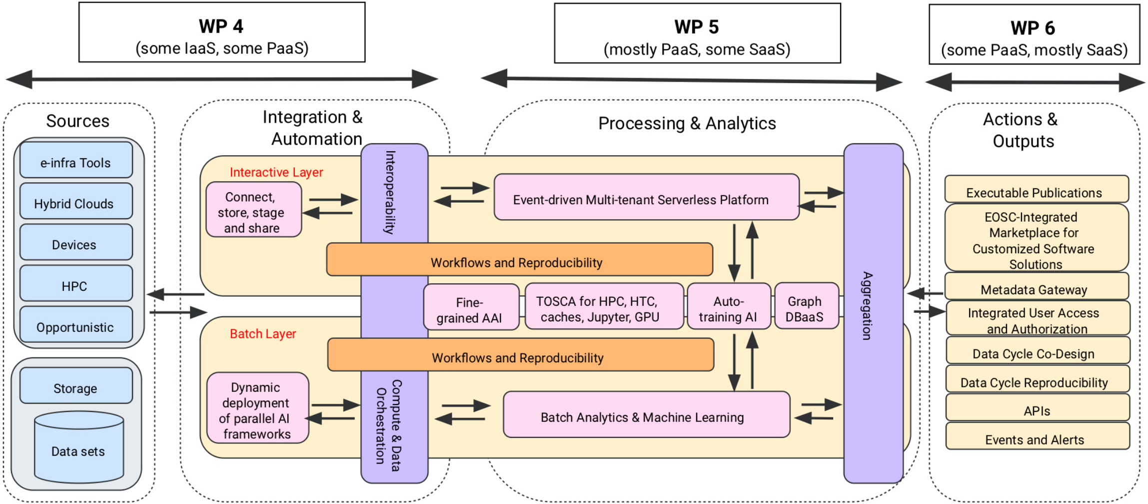
- Supporting effective **co-design** by interdisciplinary scientists, as well as many other potential stakeholders, through a configurable selection of components and reproducible workflows.

INDIGO-Next

- INFN is working toward the definition and development of new open source components that will make this possible.
- Together with several EU public and private institutions, we have recently submitted an H2020 proposal to get funding for this.

If you are interested in this effort and wish to contribute, you are very welcome to get in touch with us.

INDIGO-Next High-level Architecture



INDIGO-Next interdisciplinarity examples

- WLCG
- LSST
- Climate Change
- European XFEL
- Social Science
- ELIXIR
- Fusion
- Earth Observation
- EMSO
- EPOS
- Gravitational Waves
- Digital Repositories

⟷

- Transparent access to HTC and HPC resources to execute mixed workflows
  - *Exploitation also of specialized architectures (e.g. GPU or FPGA)*

INDIGO-Next interdisciplinarity examples

- WLCG
- LSST
- Climate Change
- European XFEL
- Social Science
- ELIXIR
- Fusion
- Earth Observation
- EMSO
- EPOS
- Gravitational Waves
- Digital Repositories

↔

- Transparent access to HTC and HPC resources to execute mixed workflows
  - *Exploitation also of specialized architectures (e.g. GPU or FPGA)*
- Auto-scalable platforms for enhanced event-triggered computing pipelines
  - *Continuous training of Machine Learning models*

INDIGO-Next interdisciplinarity examples

- WLCG
- LSST
- Climate Change
- European XFEL
- Social Science
- ELIXIR
- Fusion
- Earth Observation
- EMSO
- EPOS
- Gravitational Waves
- Digital Repositories

↔

- Transparent access to HTC and HPC resources to execute mixed workflows
  - *Exploitation also of specialized architectures (e.g. GPU or FPGA)*
- Auto-scalable platforms for enhanced event-triggered computing pipelines
  - *Continuous training of Machine Learning models*
- Combine declarative analysis model with TOSCA-based infrastructure description enabling reproducibility
  - *Reproducing workflows for data reprocessing*

INDIGO-Next interdisciplinarity examples

Istituto Nazionale di Fisica Nucleare

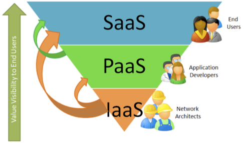
- WLCG
- LSST
- Climate Change
- European XFEL
- Social Science
- ELIXIR
- Fusion
- Earth Observation
- EMSO
- EPOS
- Gravitational Waves
- Digital Repositories

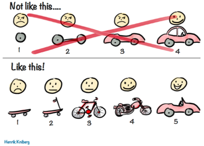
↔

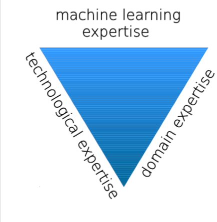
- Transparent access to HTC and HPC resources to execute mixed workflows
  - *Exploitation also of specialized architectures (e.g. GPU or FPGA)*
- Auto-scalable platforms for enhanced event-triggered computing pipelines
  - *Continuous training of Machine Learning models*
- Combine declarative analysis model with TOSCA-based infrastructure description enabling reproducibility
  - *Reproducing workflows for data reprocessing*
- Scalable infrastructure for data streaming
  - *Building MLaaS for signal and image processing*

Table of Contents

In summary

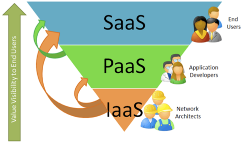


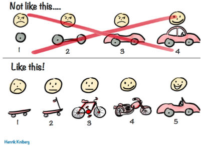


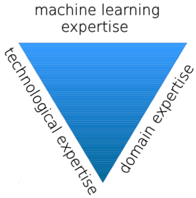


- It is naïve to think that silos-based, proprietary, monolithic solutions will address the explosion of data production and the related data analysis, esp. with complex requirements such as those encountered with AI and open science.

In summary

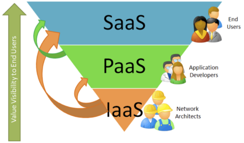


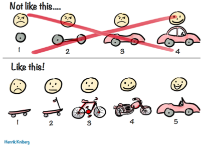


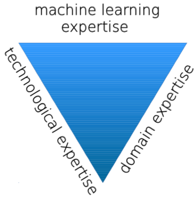


- It is naïve to think that silos-based, proprietary, monolithic solutions will address the explosion of data production and the related data analysis, esp. with complex requirements such as those encountered with AI and open science.
- Transparency, support of *de jure* and *de facto* standards, incremental, provider-agnostic, modular solutions and focus on actual needs across the entire Cloud stack are the way to go.

In summary

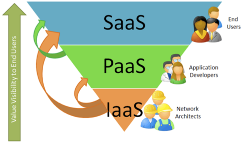


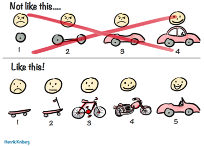


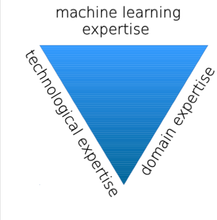


- It is naïve to think that silos-based, proprietary, monolithic solutions will address the explosion of data production and the related data analysis, esp. with complex requirements such as those encountered with AI and open science.
- Transparency, support of *de jure* and *de facto* standards, incremental, provider-agnostic, modular solutions and focus on actual needs across the entire Cloud stack are the way to go.
- We (research *and* industry) have the possibility, know-how and motivations to create an inclusive ecosystem to support this.

In summary







- It is naïve to think that silos-based, proprietary, monolithic solutions will address the explosion of data production and the related data analysis, esp. with complex requirements such as those encountered with AI and open science.
- Transparency, support of *de jure* and *de facto* standards, incremental, provider-agnostic, modular solutions and focus on actual needs across the entire Cloud stack are the way to go.
- We (research *and* industry) have the possibility, know-how and motivations to create an inclusive ecosystem to support this.

A number of steps have already been taken (some were mentioned here), but there are several exciting challenges ahead of us, and room for many to play a significant role!

Acknowledgments and additional resources

Clouds come floating into my life,
no longer to carry rain or usher storm,
but to add color to my sunset sky.
Rabindranath Tagore, Stray Birds

- **Acknowledgments**: D. Cesini, G. Donvito, Á. López García, D. Spiga.
- **For more info** on the projects mentioned here:
  - INDIGO-DataCloud: `https://www.indigo-datacloud.eu`
  - DEEP-Hybrid DataCloud: `https://deep-hybrid-datacloud.eu`
  - eXtreme-DataCloud: `http://www.extreme-datacloud.eu`
- **Contact**: `davide@infn.it`