Facilitating Scientific Models Sharing

Architectures and Technologies for Enabling Access and Use of MaaR

Mattia Santoro (CNR-IIA)

01

KNOWLEDGE GENERATION

Introduction and Context

02

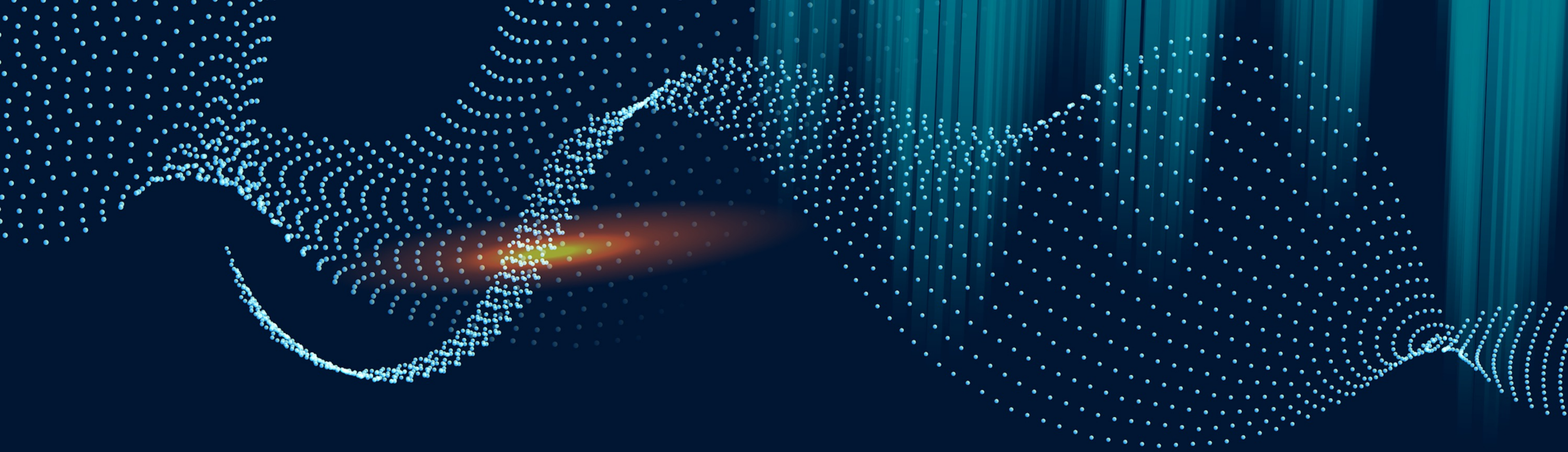
ARCHITECTURE

Conceptual Approach and Components

03

TECHNOLOGY

A Possible Technology Stack

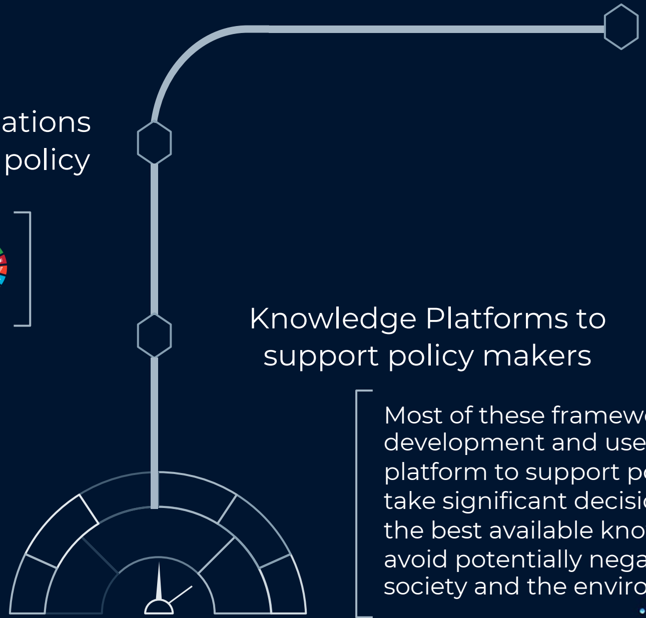


01

Knowledge Generation

Introduction and Context

Evidence-Based Decision Making



Addressing global environmental and social challenges

Food, water and energy security, resilience to natural hazards, population growth, pandemics of infectious diseases, sustainability of natural ecosystem services, poverty, etc.

International organizations have defined a list of policy actions

SENDAI FRAMEWORK
FOR DISASTER RISK REDUCTION 2015-2030

PARIS 2015

SUSTAINABLE DEVELOPMENT GOALS

Knowledge Platforms to support policy makers

Most of these frameworks require the development and use of a knowledge platform to support policy makers to take significant decisions (providing the best available knowledge) and avoid potentially negative impacts on society and the environment

Indicators/Indexes to evaluate policy targets

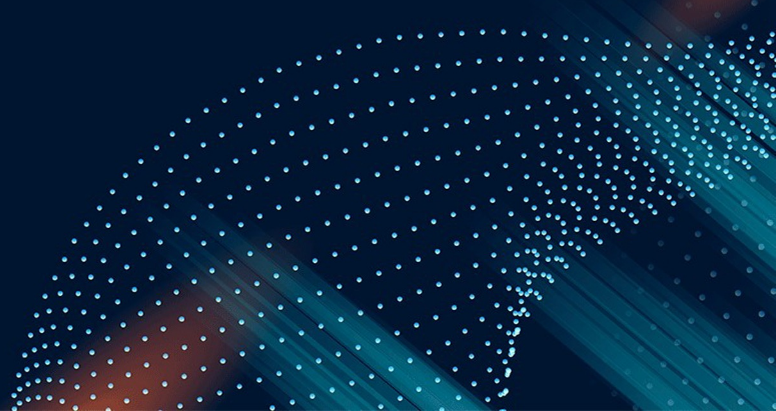
Big Earth Data Science

Big Earth Data (BED) science (Guo et al., 2020) aims to provide the methodologies and instruments to generate knowledge from numerous, complex, and diverse data sources

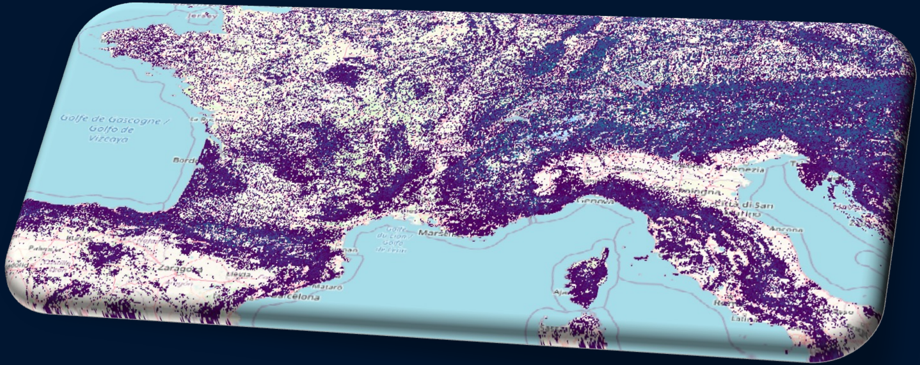
This data deluge (Big Earth Data) offers a great opportunity for **extracting the required knowledge** from the large amount of data produced every day, including the recognition of changes over time

A key tool for transforming the huge amount of data currently available into knowledge is represented by **scientific models**

The **observation of our planet** is implemented by using remote sensing (e.g., sensors flying on satellites, drones, airplanes, etc.), in-situ measurements (e.g., meteo stations, crowdsourcing, etc.), and socio-economic statistical data, (e.g., social networks and official statistical data).



Types of Resources



- DATA
- ANALYTICAL
- INFRASTRUCTURAL

Data Resources

Data to be processed or generated by computational models (e.g., model input and output)

HETEROGENEITY

Geospatial datasets are characterized by a high level of variety in terms of spatial and temporal characteristics, coordinate reference systems, encoding formats, etc.

VOLUME

Earth observation and Earth science products are now available in a great amount. Thanks to high-resolution sensors and new platforms, innovative solutions for sensor networks (IoT), social networks, passive crowdsourcing (e.g., from mobile phones), a lot of data is available to deliver relevant insights for decision-makers.

STANDARDS

Several (standard) data schemas and data access protocols exist for data sharing including metadata and data typologies, access service interfaces, APIs, etc.

Infrastructural Resources

This category of resources includes networking, storage, computing, and other fundamental infrastructural resources, which are commonly used to execute a scientific model

SCALABILITY

Scientific models encompass a wide scale of complexity. They range from simple models generating indicators to very complex simulation models.

IaaS

Different solutions exist to provide scalable infrastructure resources. Although such solutions differ in terms of technical capabilities and philosophical approaches (e.g. resources availability, costs, privacy, and property rights conditions), they can all be characterized as IaaS (Infrastructure as a Service) solutions.

MULTICLOUD

There is the need to be able to execute models on different platforms and, when possible, choose the right platform considering many aspects including user preference, cost, resource availability

Analytical Resources

These resources represent the implementation(s) of scientific models to analyze/process one or more datasets.

HETEROGENEITY

Scientific models are developed in many different programming environments (e.g., Python, Java, R, etc.) or simulation frameworks (e.g., NetLogo, Simulink, etc.).

EXECUTION ENVIRONMENT

In addition to programming languages, model executions require specific environments configurations (e.g., availability of necessary software libraries).

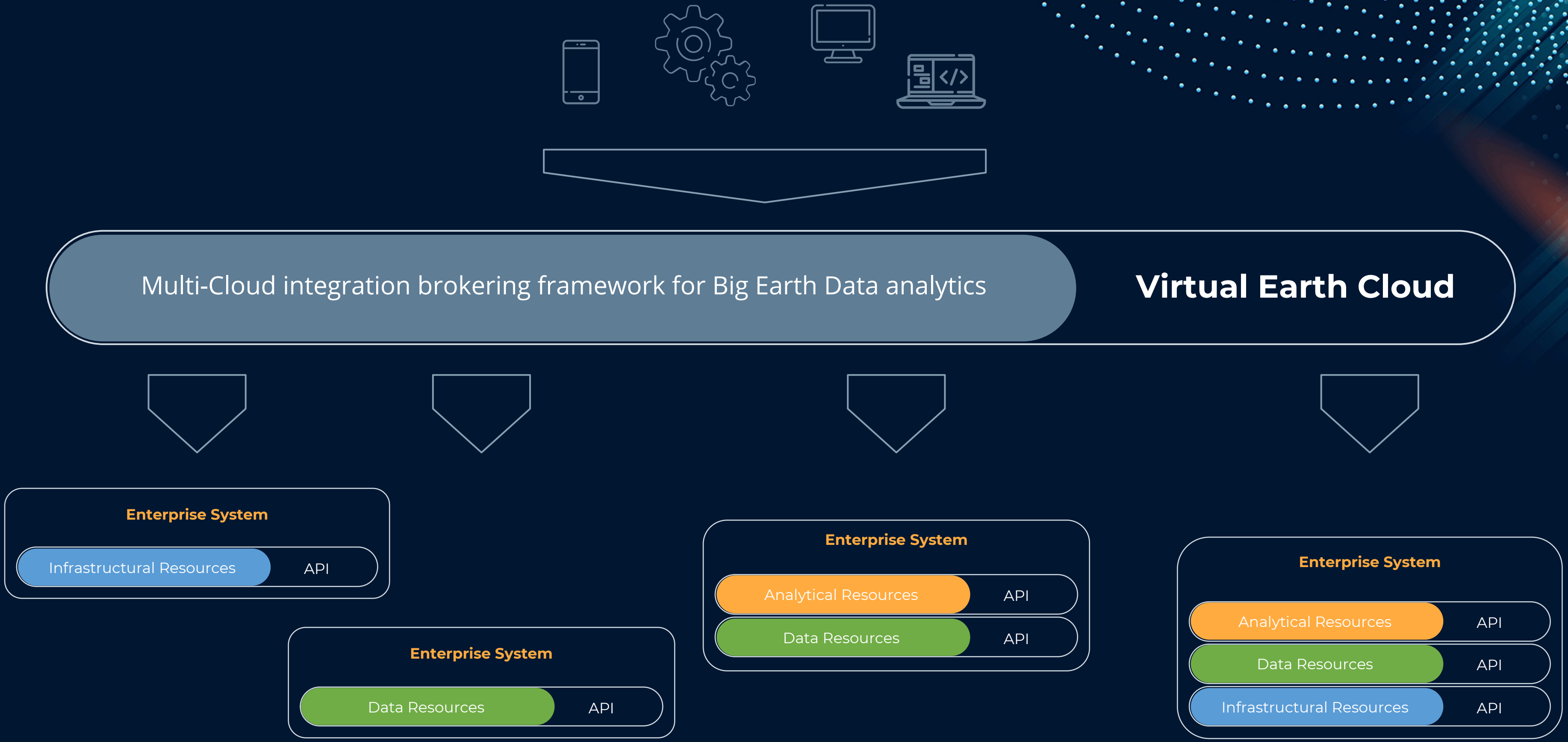
SHARING APPROACHES

Model as a Tool (MaaT): users interact with a software tool

Model as a Service (MaaS): a given implementation runs on a specific server and APIs are exposed to interacting with the model

Model as a Resource (MaaR): the source code (or executable binary) of analytical model itself is shared

Building on Existing Systems



Multi-Cloud integration brokering framework for Big Earth Data analytics

Virtual Earth Cloud

Enterprise System

Infrastructural Resources | API

Enterprise System

Data Resources | API

Enterprise System

Analytical Resources | API

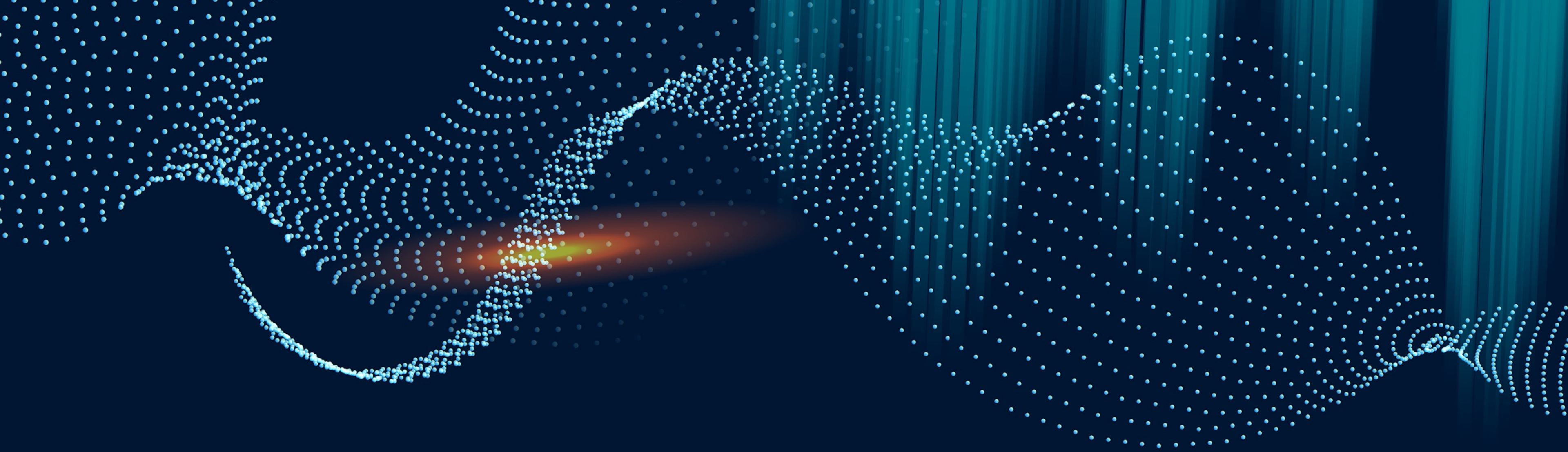
Data Resources | API

Enterprise System

Analytical Resources | API

Data Resources | API

Infrastructural Resources | API



02 Architecture

Conceptual Approach and Components

High Level Requirements

The main goal of the Virtual Earth Cloud is to allow the execution of analytical software (i.e., scientific models) on the most appropriate of the different systems, in a seamless way for the requester (i.e., users via Client Applications)

- Implementation of the workflow required for model execution: configuring the environment (programming languages, software libraries, etc.), ingesting input data, etc.;
- Provisioning of computational resources from the available underlying enterprise systems;
- Discovery of and access to the necessary data and model resources, from the available underlying enterprise systems;
- Optimization of the model execution, e.g. based on availability of computational resources, latency time, and required data.

Conceptual Approach

CONTAINERIZATION

To provide an environment that supports the required programming language and the software libraries utilized by the model

ORCHESTRATION

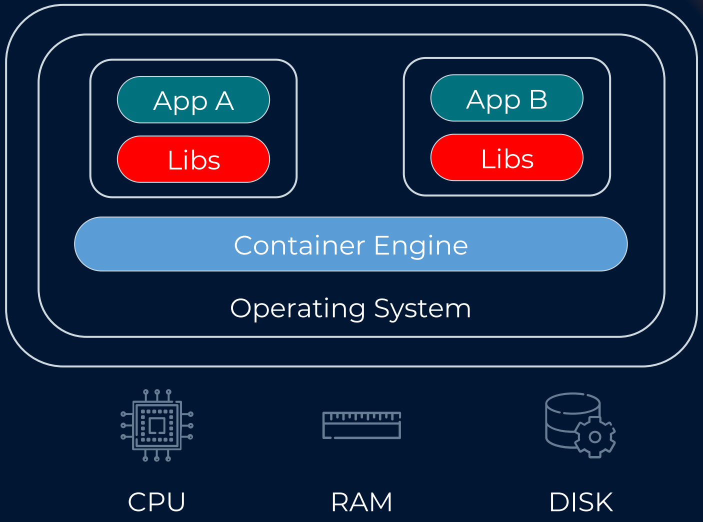
Automated configuration, management, and coordination of computer systems, applications, and services

BROKERING

Interoperability is implemented by dedicated components (the brokers) that oversee connecting to the participant systems

Containerization

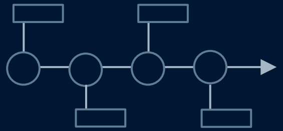
Containerization is the **packaging together of software code with all its necessary components** (e.g., libraries, frameworks, and other dependencies). This creates a **container**, i.e., a single fully packaged and portable executable, which can be run on any infrastructure compatible with the specific containerization technology (i.e., container engine)



Orchestration

The Virtual Earth Cloud must provide orchestration functionalities at three different levels

Model Execution Orchestration



- Management of input data access/ingestion
- Configuration of model execution
- Triggering of model execution
Storage of the generated output

Container Orchestration

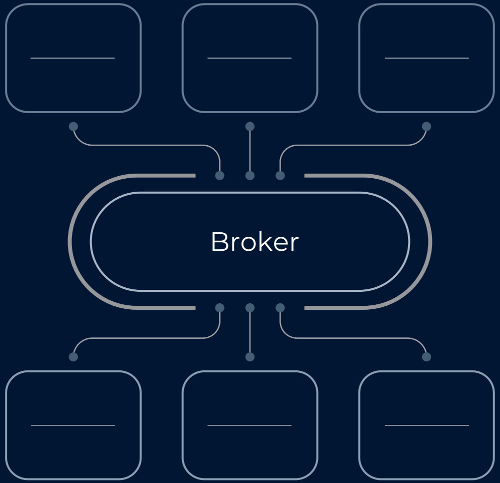
- Selection of computing node to use, according to the capacities (memory, CPU, etc.) required by the container
- Execution of all container-level configurations (e.g. links to persistent storage if requested)
- Triggering the container
- Monitoring the resource allocations and the state of the containers

Infrastructure Orchestration

When new computational resources are requested on a specific enterprise system, it is necessary to coordinate and invoke the instantiation of the processing, storage, networks, and other fundamental computing resources. Finally, the newly instantiated computational resources must be properly configured to support the containerized execution of models

Brokering

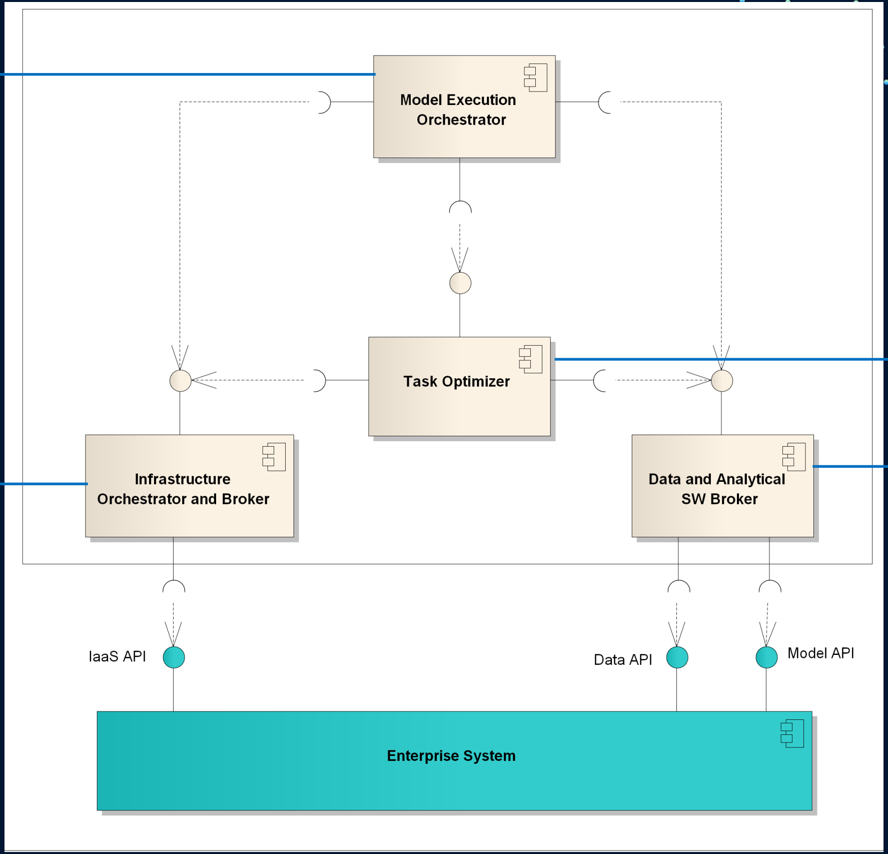
In the brokered approach participating systems can adopt or maintain their preferred interfaces, metadata and data models. Interoperability is implemented by dedicated components (the brokers) that oversee connecting to the participant systems, by implementing all the required mediation and harmonization artifacts.



Architecture Components

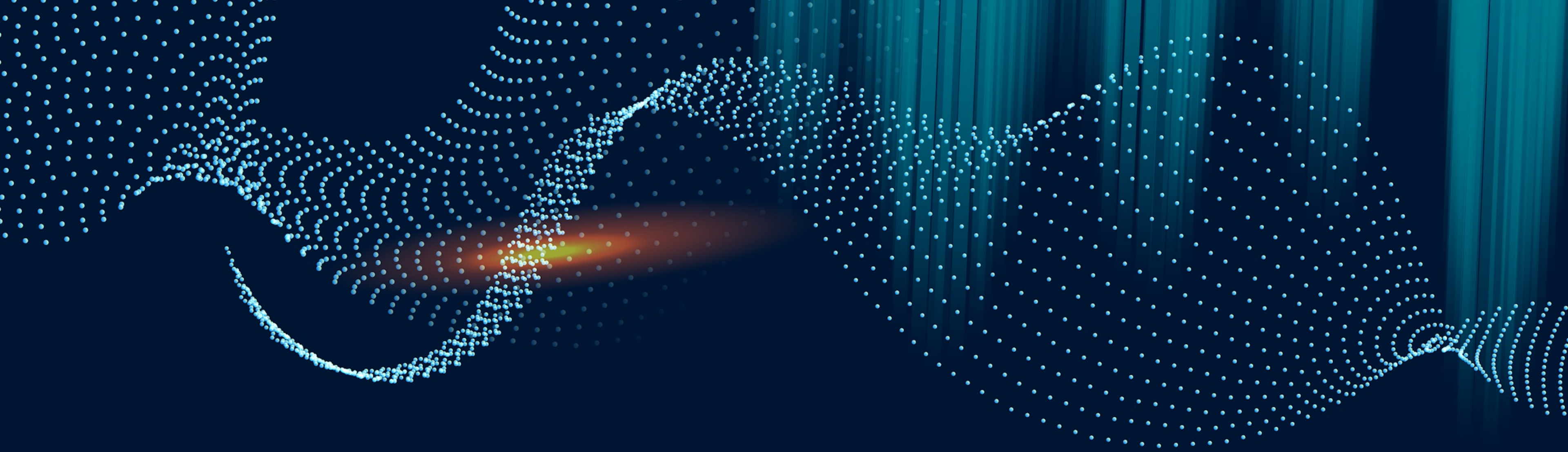
The main entry point of the Virtual Earth Cloud component, i.e. it publishes the APIs which can be invoked to request the execution of a model. Upon a model execution requests, this module implements the business logic needed to execute a model.

This component is in charge of connecting to the systems which share computational resources in order to discover and allocate the required computational resources needed for the execution of the model.


Model Execution Orchestrator
Task Optimizer
Infrastructure Orchestrator and Broker
Data and Analytical SW Broker
IaaS API
Data API
Model API
Enterprise System

Responsible for selecting the optimal system for the execution of the specified model (e.g., according to computing resources availability, data availability, etc.).

Enables the discoverability and access of data and analytical resources (models). It provides a unique entry point to discover and access such resources; i.e., it exposes a set of APIs which can be used to discover and access the (data and models) resources from the different participating systems.



03 Technology

A Possible Technology Stack

A Possible Technology Stack

The GEO Discovery and Access Broker (GEO DAB) implements a brokering framework for data discovery and access. The GEO DAB is a component of the GEOSS (Global Earth Observation System of Systems) Platform and provides the necessary mediation, harmonization and distribution functionalities to allow data providers to share resources without having to make major changes to their technology or standards.

The Virtual Earth Laboratory (VLab) is a framework that implements all required model orchestration functionalities to automate the technical tasks required to execute a model on cloud infrastructures, minimizing the possible interoperability requirements for both model developers and users.

Kubernetes is an open-source container-as-a-service (CaaS) framework created by Google developers more than a decade ago. The system automates application deployment, scaling, and operations.

Docker, now often referred to as "Docker Engine", is one of the most widely known and utilized containerization technologies. Its initial release was a monolith application which covered all aspects of containers' creation and execution, including pulling images from registries, managing storage and data, etc.

Cluster Autoscaler is a software that automatically adjusts the size of a Kubernetes Cluster so that all pods have a place to run and there are no unused computational resources

The Cluster API is a Kubernetes project to bring declarative, Kubernetes-style APIs to cluster creation, configuration, and management. It provides optional, additive functionality on top of core Kubernetes to manage the lifecycle of a Kubernetes cluster. Cluster API supports a variety of cloud providers APIs.

GEO DAB

Data Brokering

VLAB

Model Execution Orchestration

Container Orchestration

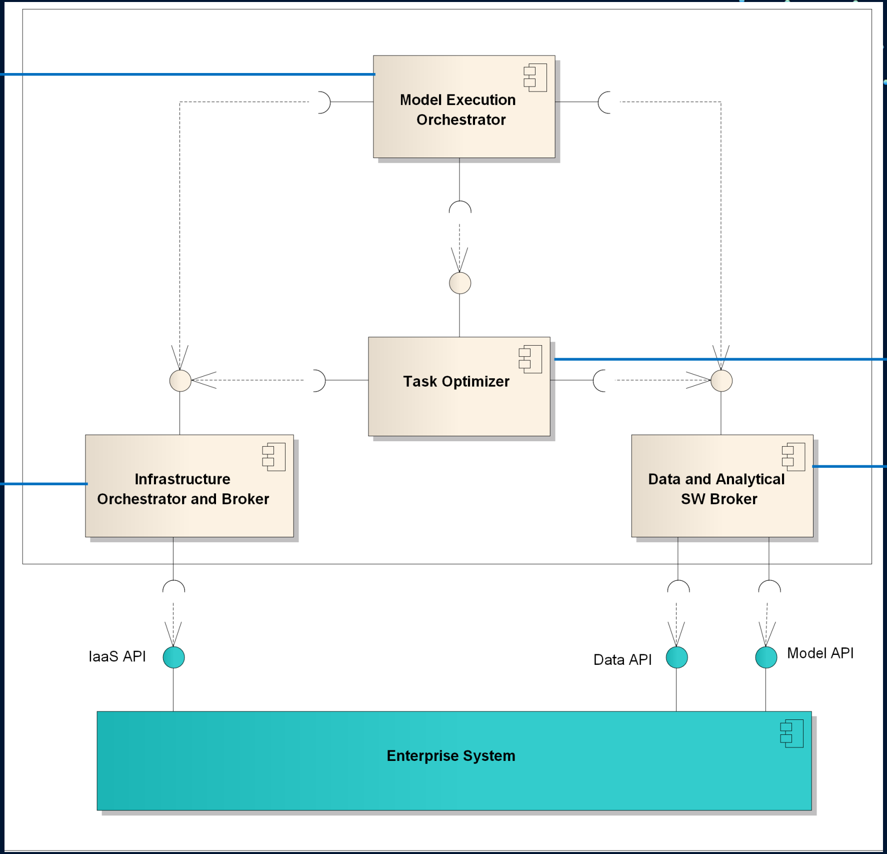
docker

Containerization

CLUSTER AUTO SCALER

IaaS Brokering + Infrastructure Orchestration

Architecture Implementation


Model Execution Orchestrator
Task Optimizer
Infrastructure Orchestrator and Broker
Data and Analytical SW Broker
IaaS API
Data API
Model API
Enterprise System

Simple Optimizer

Future Work

- Support of **real-time capabilities**: this is part of planned future development, starting with example implementations in the Hydrology modeling domain.
- **Knowledge Base integration**: capturing scientific experts' knowledge about a sound process for knowledge generation (e.g. the choice of appropriate datasets to be used as inputs for existing models, which model to use for a specific use-case, etc.) is a key element to build a sound Knowledge Platform in line with Open Science principles of reproducibility, replicability and re-usability.
- **Data quality/uncertainty**: while not considered in the scope of this manuscript, data quality/uncertainty is of paramount importance; future enhancements will investigate how to include this aspect, also considering its implications in possible chaining of models.
- **AI and Digital Twins of the Earth**: extension of the presented architecture focusing on specific support for the implementation of Digital Twins of natural environments.

THANK YOU

Do you have any questions?
mattia.santoro@cnr.it

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik.

INTERNATIONAL JOURNAL OF DIGITAL EARTH
2023, VOL. 16, NO. 1, 43–65
https://doi.org/10.1080/17538947.2022.2162986

OPEN ACCESS Check for updates

Virtual earth cloud: a multi-cloud framework for enabling geosciences digital ecosystems

Mattia Santoro[a], Paolo Mazzetti [a] and Stefano Nativi[a,b]*

[a]National Research Council of Italy, Institute of Atmospheric Pollution Research – Unit of Florence, Roma, Italy; [b]Joint Research Centre of the European Commission, B6 Unit, Ispra, Italy

ABSTRACT
Humankind is facing unprecedented global environmental and social challenges in terms of food, water and energy security, resilience to natural hazards, etc. To address these challenges, international organizations have defined a list of policy actions to be achieved in a relatively short and medium-term timespan. The development and use of knowledge platforms is key in helping the decision-making process to take significant decisions (providing the best available knowledge) and avoid potentially negative impacts on society and the environment. Such knowledge platforms must build on the recent and next coming digital technologies that have transformed society – including the science and engineering sectors. Big Earth Data (BED) science aims to provide the methodologies and instruments to generate knowledge from numerous, complex, and diverse data sources. BED science requires the development of Geoscience Digital Ecosystems (GEDs), which bank on the combined use of fundamental technology units (i.e. big data, learning-driven artificial intelligence, and network-based computing platform) to enable the development of more detailed knowledge to observe and test planet Earth as a whole. This manuscript contributes to the BED science research domain, by presenting the Virtual Earth Cloud: a multi-cloud framework to support GDE implementation and generate knowledge on environmental and social sustainability.

ARTICLE HISTORY
Received 11 September 2022
Accepted 20 December 2022

KEYWORDS
Earth observation; geosciences digital ecosystem; virtual cloud; big earth data; multi-cloud; interoperability science

1. Introduction

Humankind is facing unprecedented global environmental and social challenges in terms of food, water and energy security, resilience to natural hazards, population growth and migrations, pandemics of infectious diseases, sustainability of natural ecosystem services, poverty, and the development of a sustainable economy (Nativi, Mazzetti, and Craglia 2021). Addressing these challenges is crucial for our planet preservation and the future development of human society. To this aim, international organizations have defined a list of policy actions to be achieved in a relatively short and medium-term time framework. Notably, the United Nations (UN) defined 17 Sustainable Development Goals (SDGs)[1] along with an implementation agenda. Such an effort is supported by other relevant international initiatives and programmes, including the UNFCCC Process-and-meetings[2] (see for example the Conference of Parties 2015 on Climate: COP21)[3] and the Sendai Framework for Disasters Risk Reduction[4] overseen by the United Nations Office for Disaster Risk Reduction

CONTACT Mattia Santoro mattia.santoro@cnr.it
*Disclaimer: the views expressed are purely those of the authors and may not in any circumstances be regarded as stating an official position of the European Commission.

https://doi.org/10.1080/17538947.2022.2162986