Living on the edge, sailing through the cloud

Orchestrating Applications in the Edge to Cloud Computing Continuum

Prof Dr Tamas Kiss

University of Westminster

t.kiss@westminster.ac.uk

www.digitbrain.eu

www.linkedin.com/groups/12439191

www.twitter.com/digitbrain_eu

www.facebook.com/DIGITbrainProject

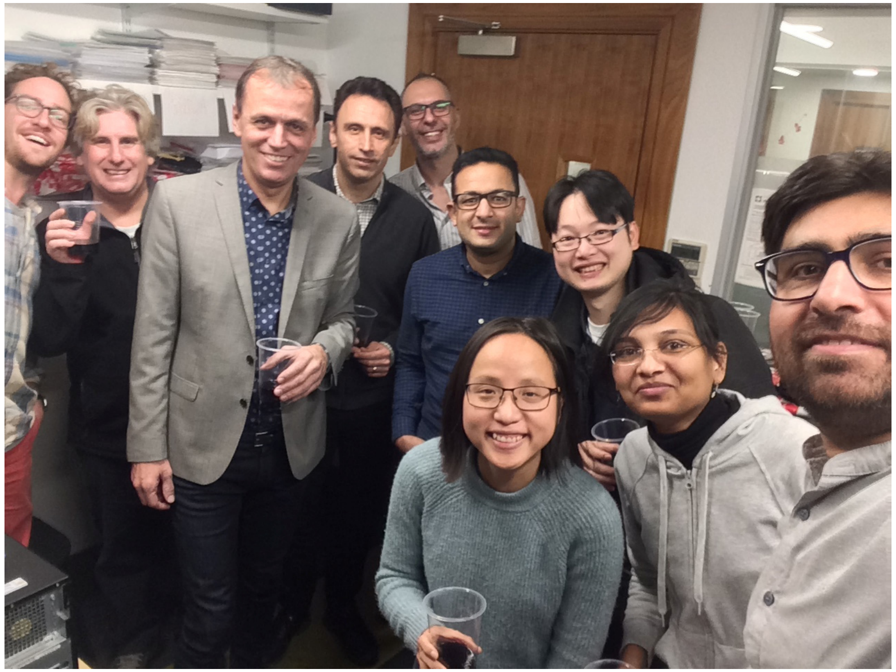
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The Centre for Parallel Computing

Established in the 1990s

Focus on cloud/fog/edge computing

- Orchestration and autoscaling in the Cloud to Edge continuum
- Secure management of multi-cloud environments
- Cloud-native technologies – Docker, Kubernetes
- Standardised description of cloud-native applications - TOSCA
- Cloud/fog/edge based simulation environments for SMEs, research and the public sector
- Science/business gateways - user friendly interfaces for clouds and high performance computing

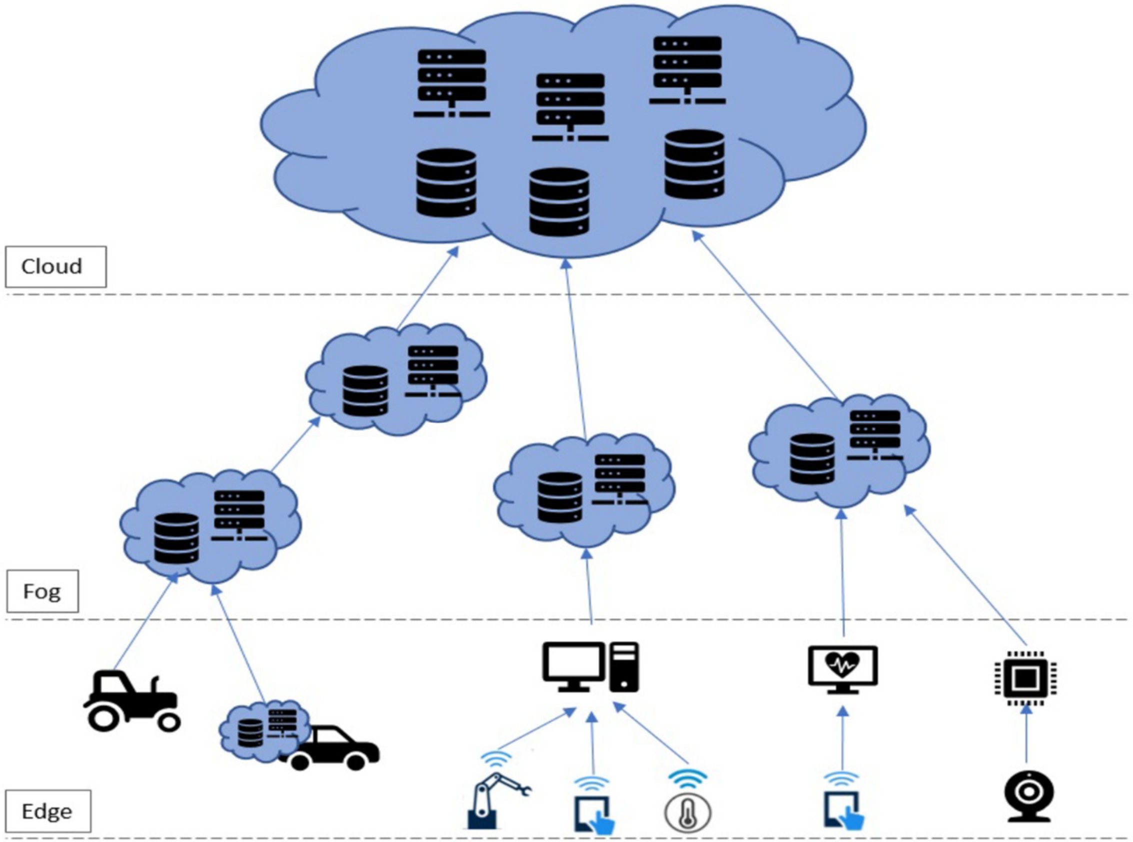


Main research outputs with impact in the past 6 years:

- MiCADO – application-level cloud to edge orchestrator - https://micado-scale.eu/
- emGORA workspace – a commercial digital marketplace for manufacturing simulation, data analytics and AI-based digital twins - https://www.emgora.eu/
- PITHIA e-Science Centre – Science Gateway for the space physics community - https://esc.pithia.eu/

DIGITbrain has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 952071

The Cloud-to-Edge Compute Continuum

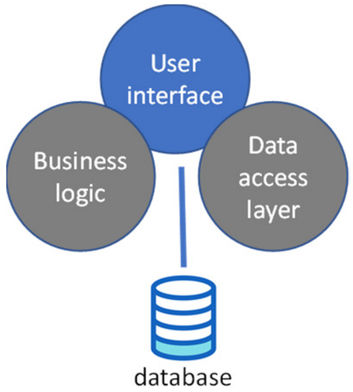


- Application areas:
  - Smart manufacturing/Industry 4.0
  - Smart cities
  - Precision agriculture
  - Self-driving cars
  - Augmented reality
  - Etc.

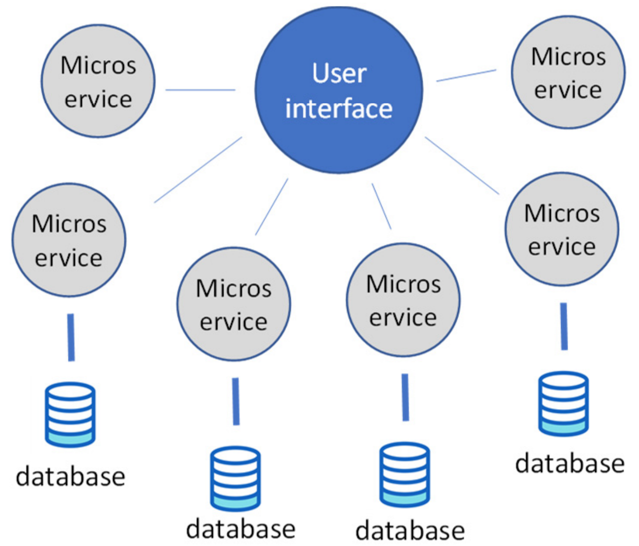
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Monolithic vs Microservice architectures

- Monolithic
  - All application components together
    - Single host, single DB, single server



- Microservice
  - Each application component individual
    - Containers make it possible!
    - Individual hosts, databases and servers
    - All working together



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Cloud-native approach

- Cloud Native Computing Foundation - CNCF Cloud Native Definition v1.0: https://github.com/cncf/toc/blob/master/DEFINITION.md

- Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds.

- Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

CLOUD NATIVE
COMPUTING FOUNDATION

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Cloud-to-Edge orchestration

- **Orchestration** is the automated configuration, management, and coordination of computer systems, applications, and services.
- **Cloud orchestration** can be used to provision or deploy servers, assign storage capacity, create virtual machines, manage networking, deploy complex microservice architectures, auto-scale applications at run-time etc.
- **Cloud-to-Edge orchestration** extends this beyond the cloud layer; e.g. offload tasks from fog/edge servers to the cloud

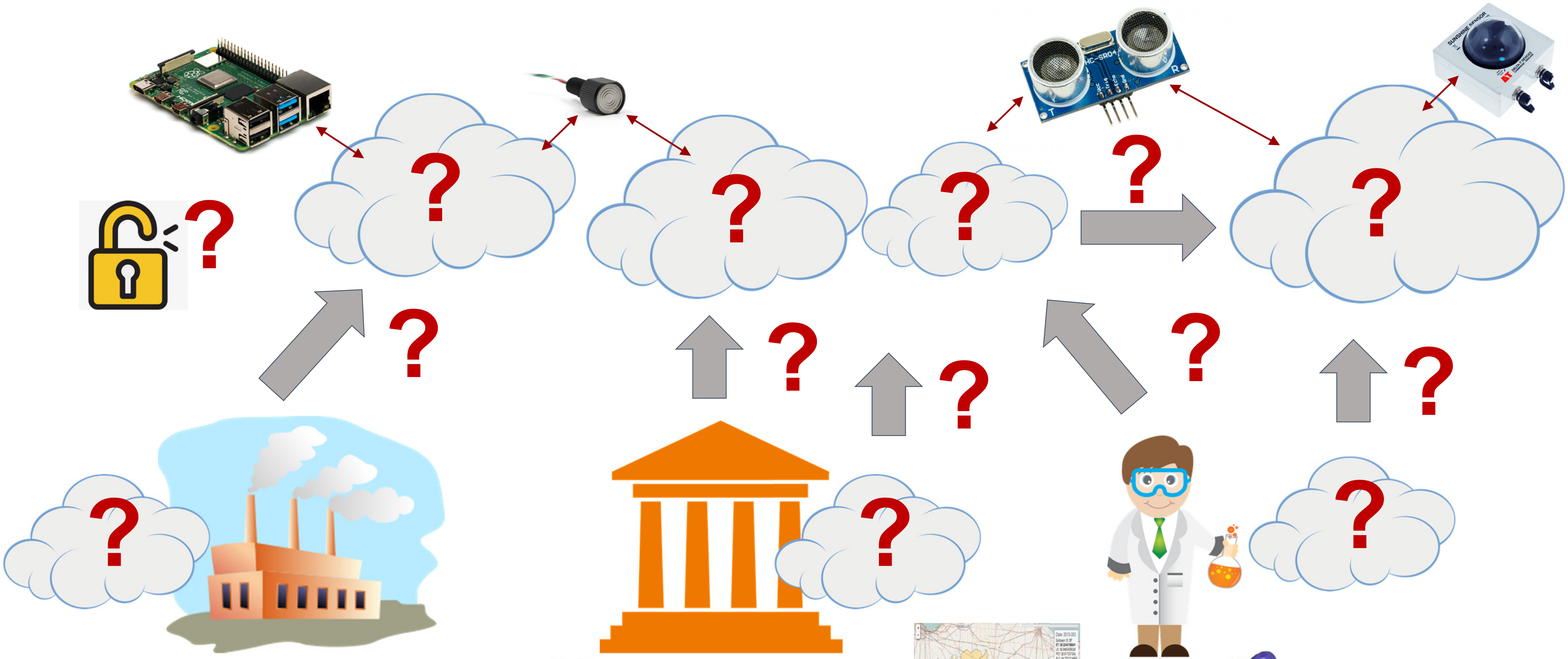
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

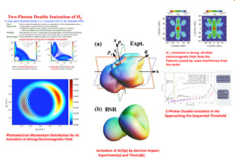
What challenges are we addressing?

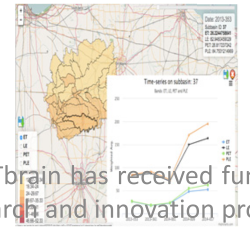
- Cloud computing is gaining more and more attention in **various industrial sectors**
- **Private** clouds have significant benefits in terms of security and integrability into the enterprise environment but **hybrid** and multi-clouds are also widespread.
- New solutions are needed to support **Internet of Things** (IoT) and **Big Data** application areas -> cloud-fog-edge computing is a new challenge
- Growing demand for **cloud ochestrators** and **brokering** tools.

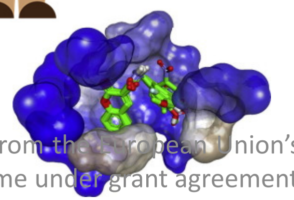
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

What challenges are we addressing?





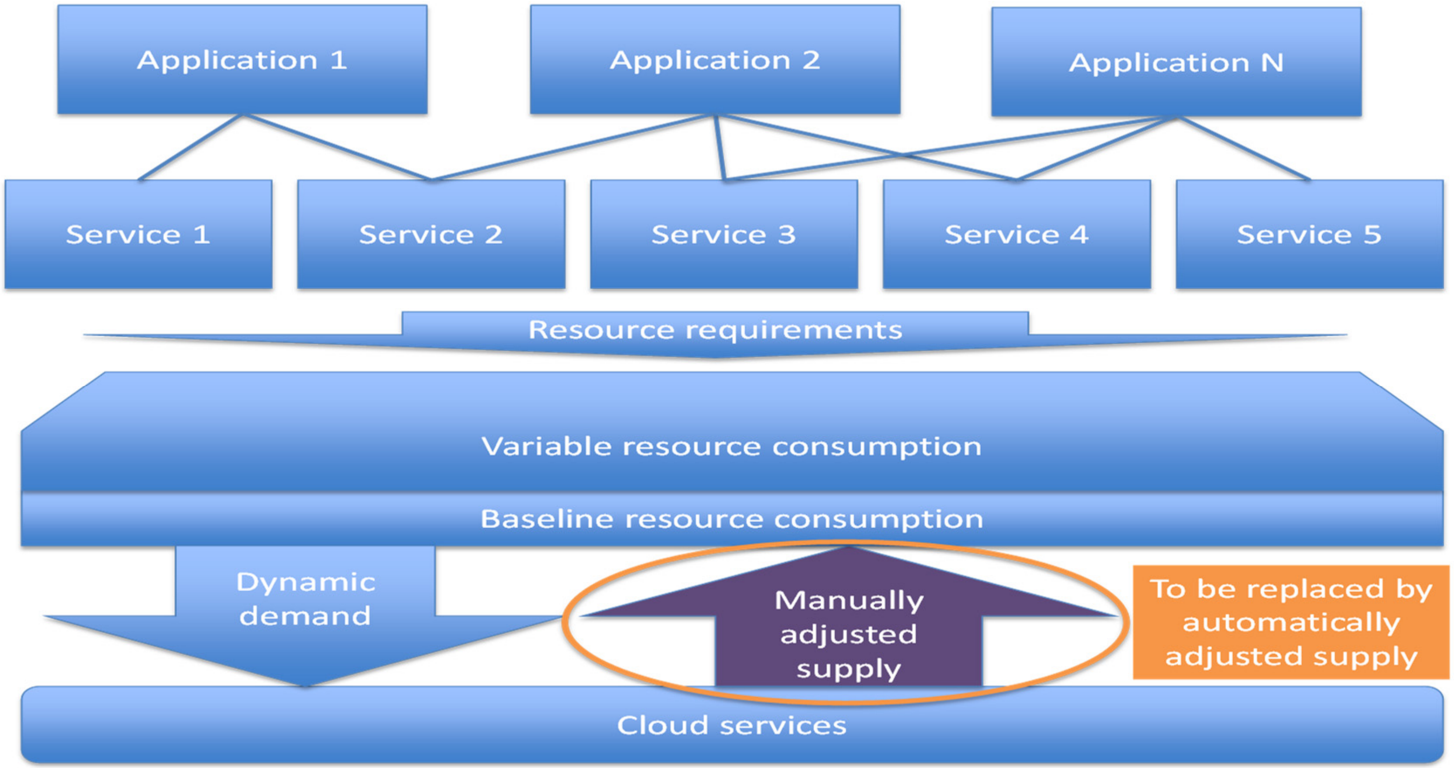




DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The beginnings – application-level cloud orchestration

- deploying complex sets of application microservices in a cloud agnostic way
- supporting a wide range of scaling policies and intelligent scaling decisions
- supporting cloud application developers
- providing advanced policy-based security solutions



COLA – Cloud Orchestration at the Level of Application
https://project-cola.eu/

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The COLA project

- EU H2020 project
- 1st January 2017 - 30th October 2019
- Funding: 4.2 million Euros
- 14 project partners from 6 European countries
- 10 companies and 4 academic/research institutions
- More information: https://project-cola.eu

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

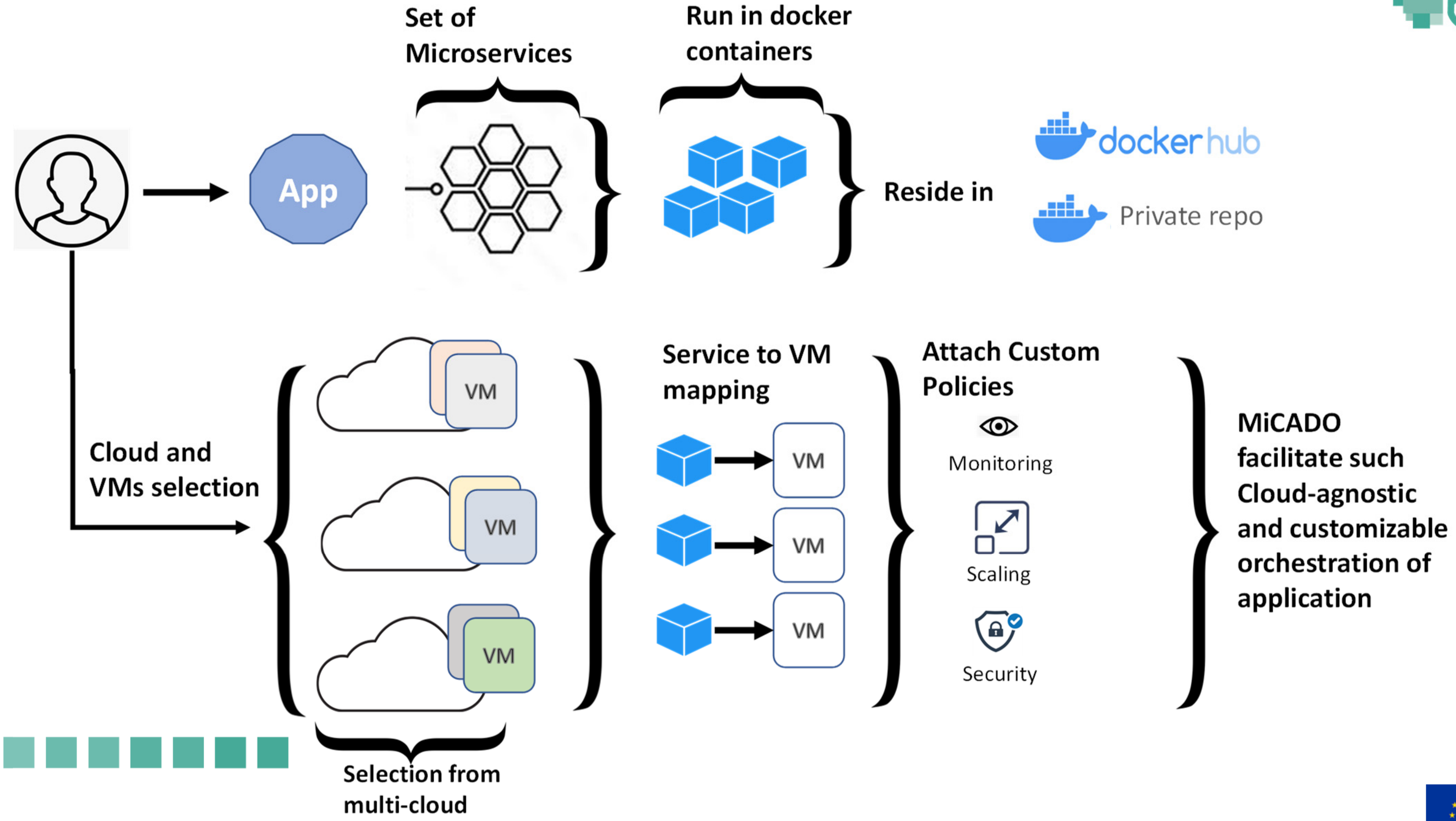
MiCADO – Microservices-based Cloud Application-level Dynamic Orchestrator

- Result of the **H2020 COLA** project
  - currently actively developed in the H2020 DIGITbrain project,
  - is/was used in ASCLEPIOS, PITHIA-NRF, CO-VERSATILE, Harpocrates and ARCAFF
- Automated application **deployment** based on TOSCA-based application description templates
- Automated **scaling** based on highly customisable scaling policies
  - scaling at both **container and virtual machine** levels
- **Multi-cloud** support – application portability
- Policy driven **security** settings
- Open source - https://micado-scale.eu/

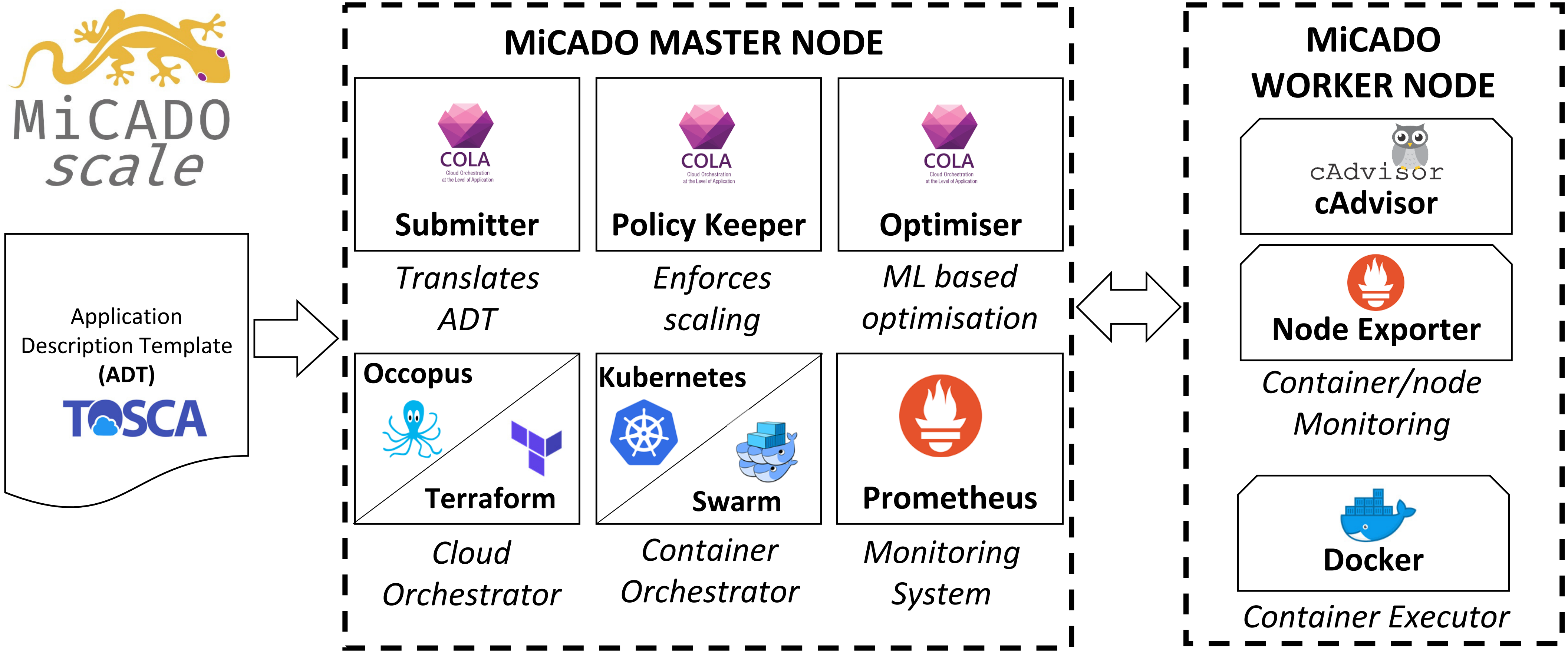
COLA
Cloud Orchestration at the Level of Application

MiCADO scale

The overall concept



MiCADO – Microservices-based Cloud Application-level Dynamic Orchestrator



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Application description in MiCADO

- Application Description Templates

- Written in Oasis Standard TOSCA
  - A cloud language in YAML
  - www.oasis-open.org/committees/tosca/

- Authored by developers
  - Understand application, metrics and scalable components

- Finalised by application operators
  - Provide Compute details for a Cloud Service Provider

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Application description

- Three sections
  - Container Infrastructure
    - Containers, volumes, configurations
  - Cloud Infrastructure
    - Instance size, SSH keys, opened ports, VM image
  - Monitoring & Scaling Policy
    - Metric collection
    - Queries, alerts, thresholds, scaling logic


```yaml
stressng:
  type: tosca.nodes.MiCADO.Container.Application.Docker.Deployment
  properties:
    image: lorel/docker-stress-ng
    args: ['--cpu', '0', '--cpu-method', 'pi', '-l', '20']
    resources:
      requests:
        cpu: "900m"

  worker-node:
    type: tosca.nodes.MiCADO.CloudSigma.Compute.Occo.small
    properties:
      vnc_password: secret
      libdrive_id: ADD_LIBRARY_DRIVE_ID (e.g. 0d2dc532-39f5
      public_key_id: ADD_PUBLIC_KEY_ID (e.g. d7c0f1ee-40df-
      nics:
      - firewall_policy: ADD_FIREWALL_POLICY_ID (e.g. fd97e
        ip_v4_conf:
          conf: dhcp
```


DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

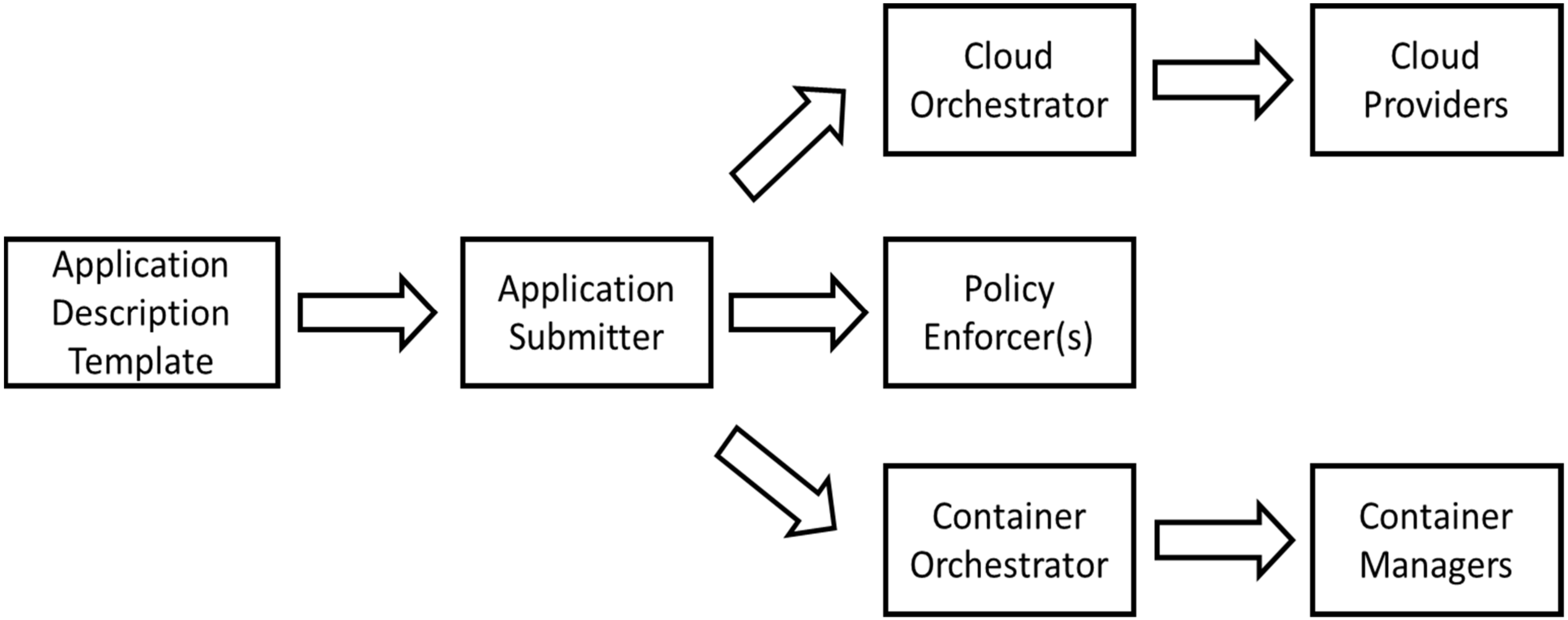
Monitoring and autoscaling in MiCADO

- Highly customizable monitoring subsystem
  - Monitored metrics are collected by dynamically attachable data collectors (Prometheus exporters)
- Highly customizable scaling logic
  - Scaling logic is fully programmable (using Python)
- Many different scaling policies are supported
  - Application types (job execution, web applications, …)
  - Different metrics (cpu, network, number of jobs, …)
  - Various strategies (load-based, deadline-based, event-based, …)
- Scaling of containers and virtual machines are supported
  - Container-only and VM-only scaling
  - Scaling at both levels in parallel, independently or cooperatively
  - Possible to use predefined scaling policy or own-developed

```yaml
policies:
  - monitoring:
      type: tosca.policies.Monitoring.MiCADO
      properties:
        enable_container_metrics: true
        enable_node_metrics: true
  - scalability:
      type: tosca.policies.Scaling.MiCADO.Container.CPU
      targets: [ stressng ]
      properties:
        constants:
          SERVICE_NAME: 'stressng'
          SERVICE_TH_MAX: '60'
          SERVICE_TH_MIN: '25'
        min_instances: 1
        max_instances: 3
```


DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

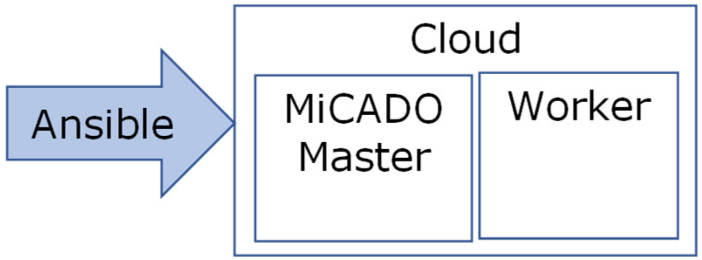
How does MiCADO work?



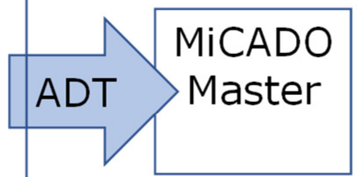
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How to use MiCADO?

1. Deploy MiCADO by customizing Ansible configuration files
2. Describe your application (for e.g. virtual machine, scaling policy, etc.) by creating/ customizing TOSCA-based ADT file (Application Description Templates)
3. Submit your ADT file visa REST API call
4. Tracking MiCADO master and worker nodes in Dashboard during run-time




```
topology_template:
  inputs:
    app_image:
      type: string
      description: Docker image
to run for the application
      required: yes
      ...
```



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Supports a large variety of clouds

amazon web services

Azure

ORACLE

openstack.

Google Cloud

egi

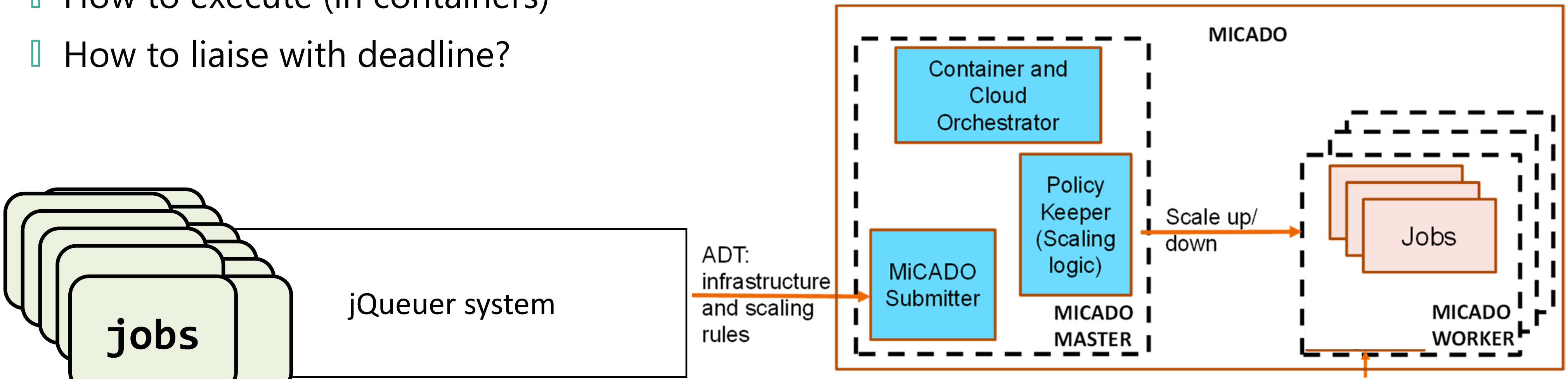
CloudBroker

CloudSigma

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

MiCADO and job execution

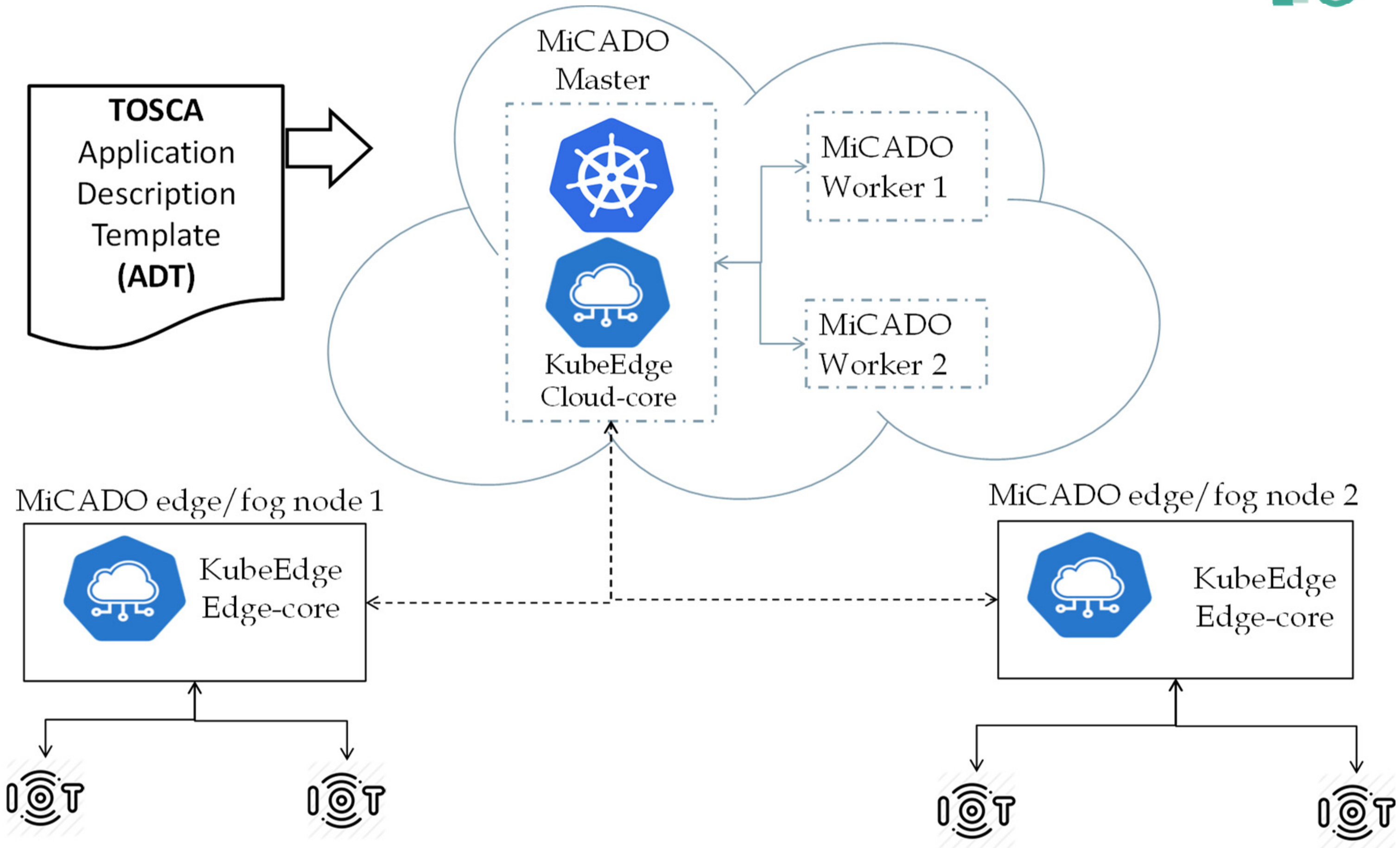
- Large number of jobs results in significant overall execution time
- Usually Restricted to complete all jobs by a deadline
  - Where to put the jobs?
  - How to distribute?
  - How to execute (in containers)
  - How to liaise with deadline?



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How to extend orchestration to the Edge

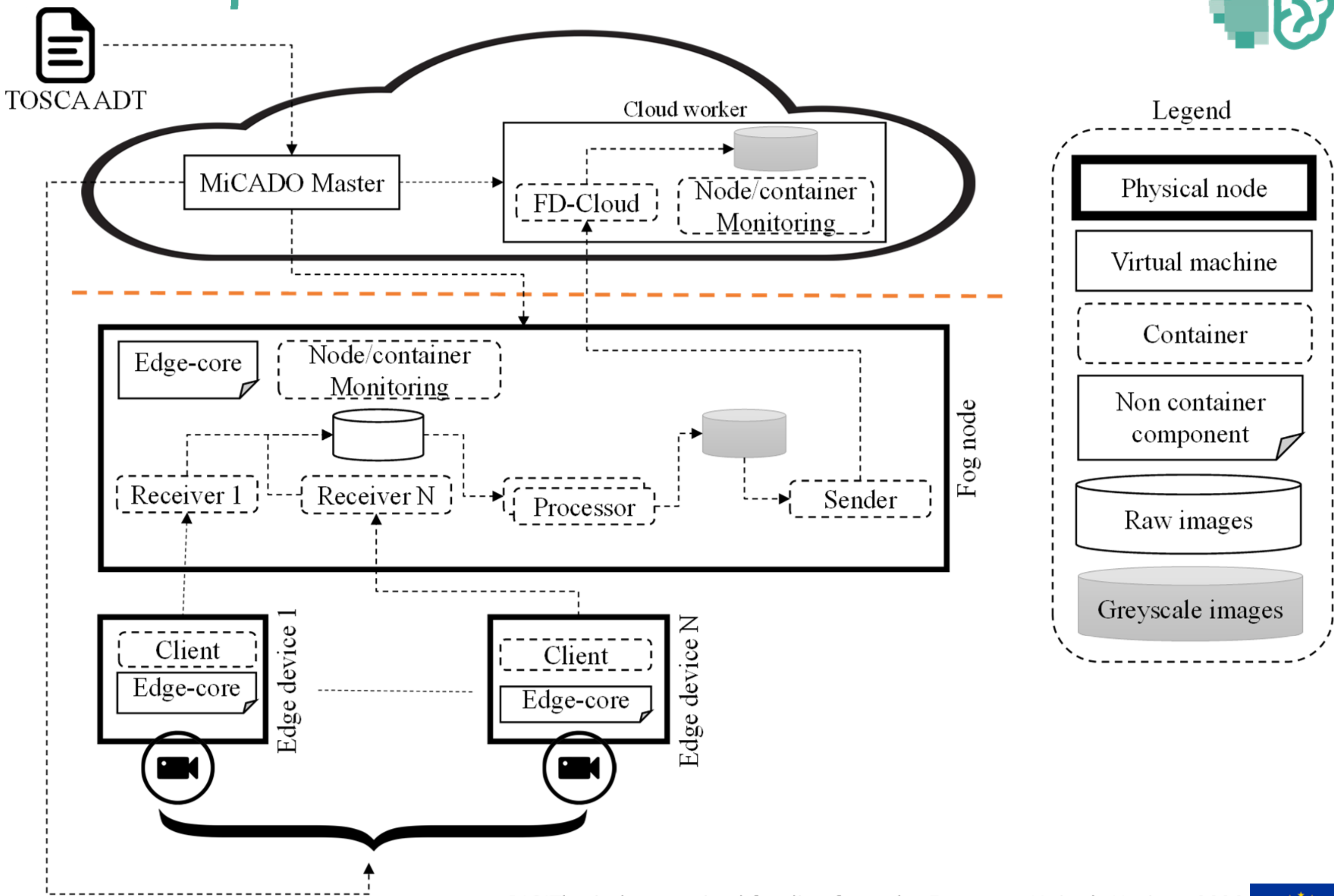
- Solution using KubeEdge
- Automated deployment of microservices extended to edge nodes
- Monitoring information collected from edge workers
- Scaling/reconfiguration policies extended towards edge



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

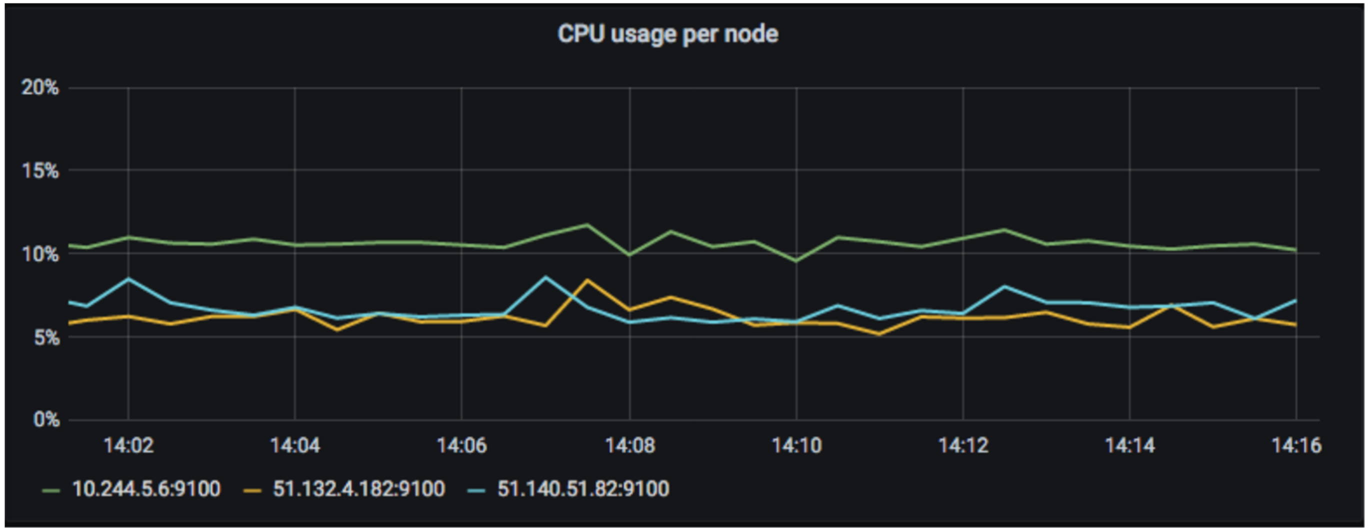
Face detection example

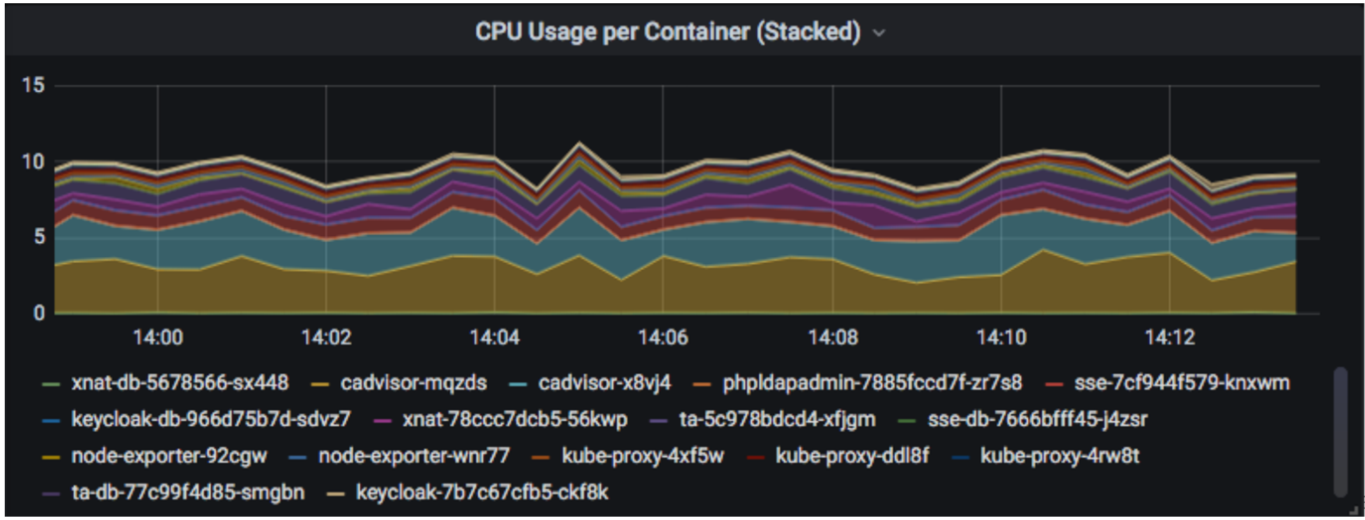
- Edge device captures video stream
- Fog node recognises faces in images
- Cloud server stores images with faces



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

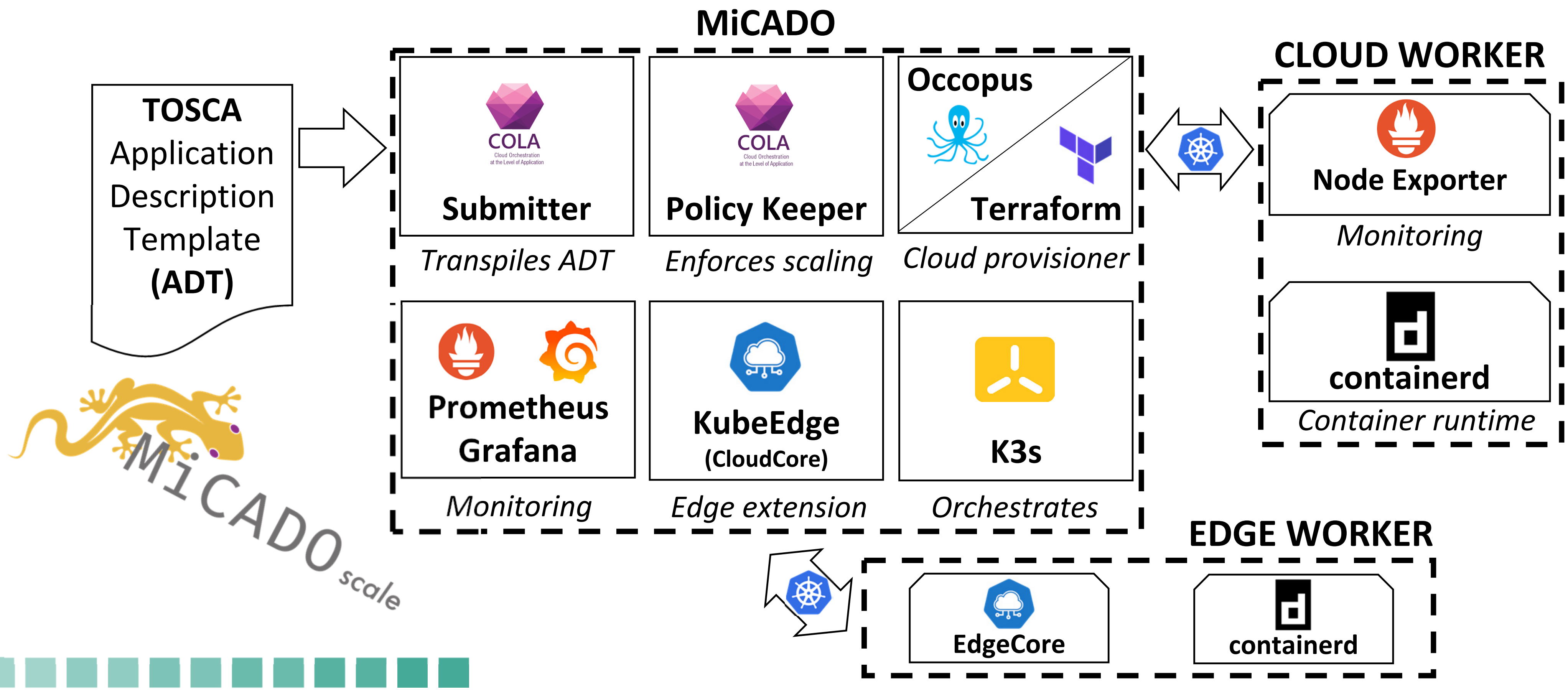
Face detection example MiCADO dashboard





an Union's Horizon 2020
agreement No 952071

Latest MiCADO architecture with Edge extension



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

MiCADO in the DIGITbrain project

- Start date: 1st July 2020 - Duration: 42 months - Number of partners: 36

Core technical and administrative partners

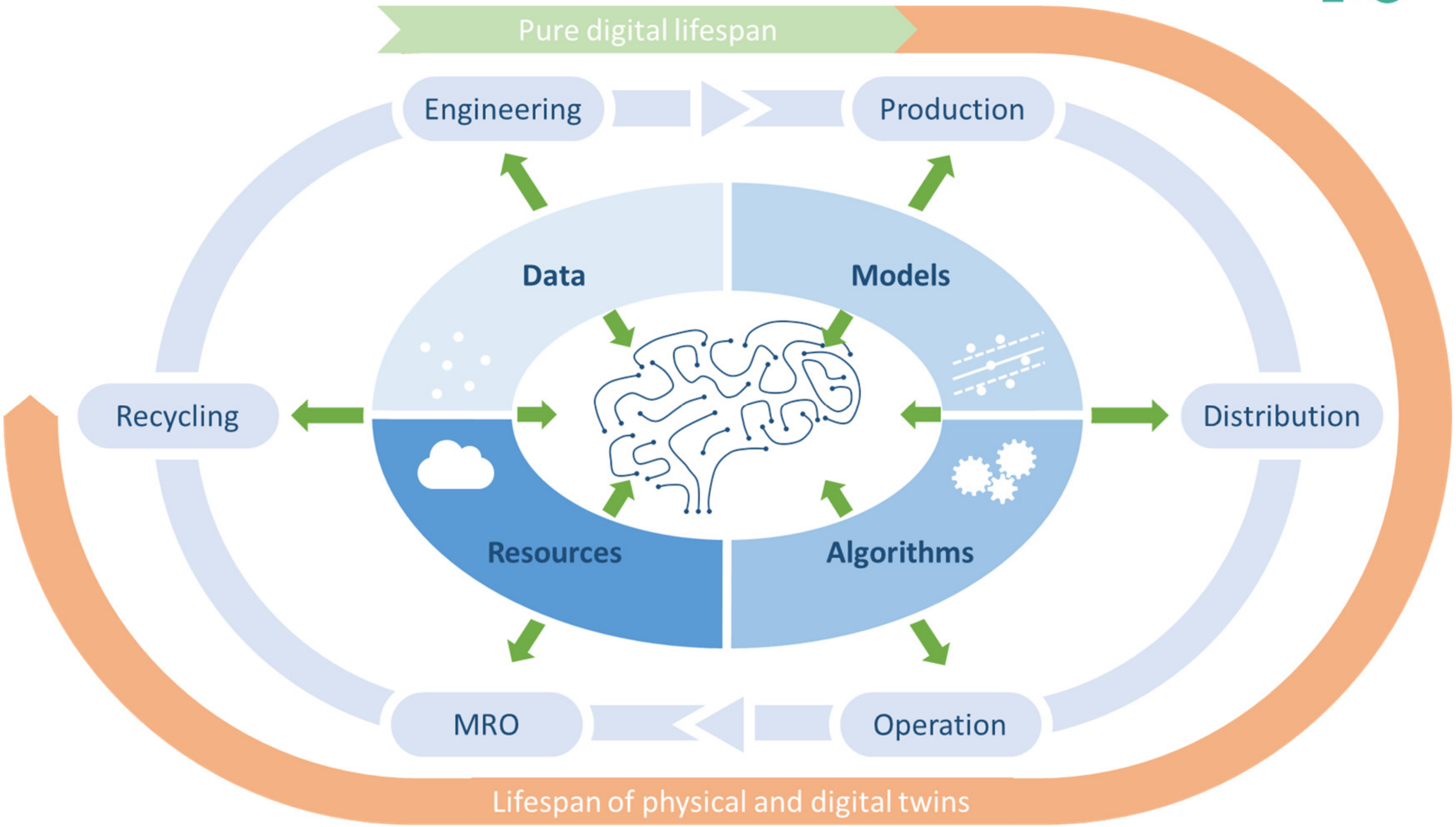
Experiment partners

Network of DIHs

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

MiCADO in the DIGITbrain project

- Compose Digital Twins from their basic building blocks (Data, Model, Algorithm)
- Execute Digital Twins in the Cloud-to-Edge continuum
- Collect, store and analyse data related to events in the Digital Product Brain



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

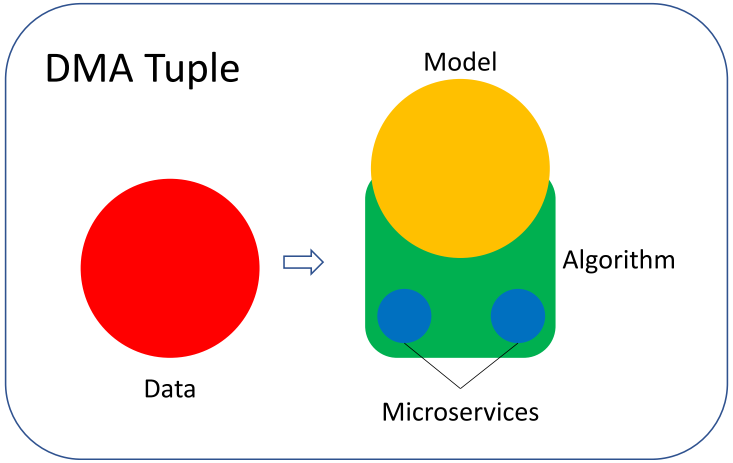
Key DIGITbrain concepts

- **Industrial Products:** Mechatronic systems (or components) that are operated by manufacturing companies to support the production of other products (e.g. a manufacturing line).
- **Digital Twin:** Represented by static and dynamic models that can evaluate historical and actual data to assess the condition and to simulate the behaviour of the industrial product.
- **Digital Product Brain:** Guides the behaviour and performance of the industrial product by coalescing its physical and digital dimensions and by memorising the occurred (physical and digital) events over a significant part of its lifecycle.

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Data – Model – Algorithm – Microservices

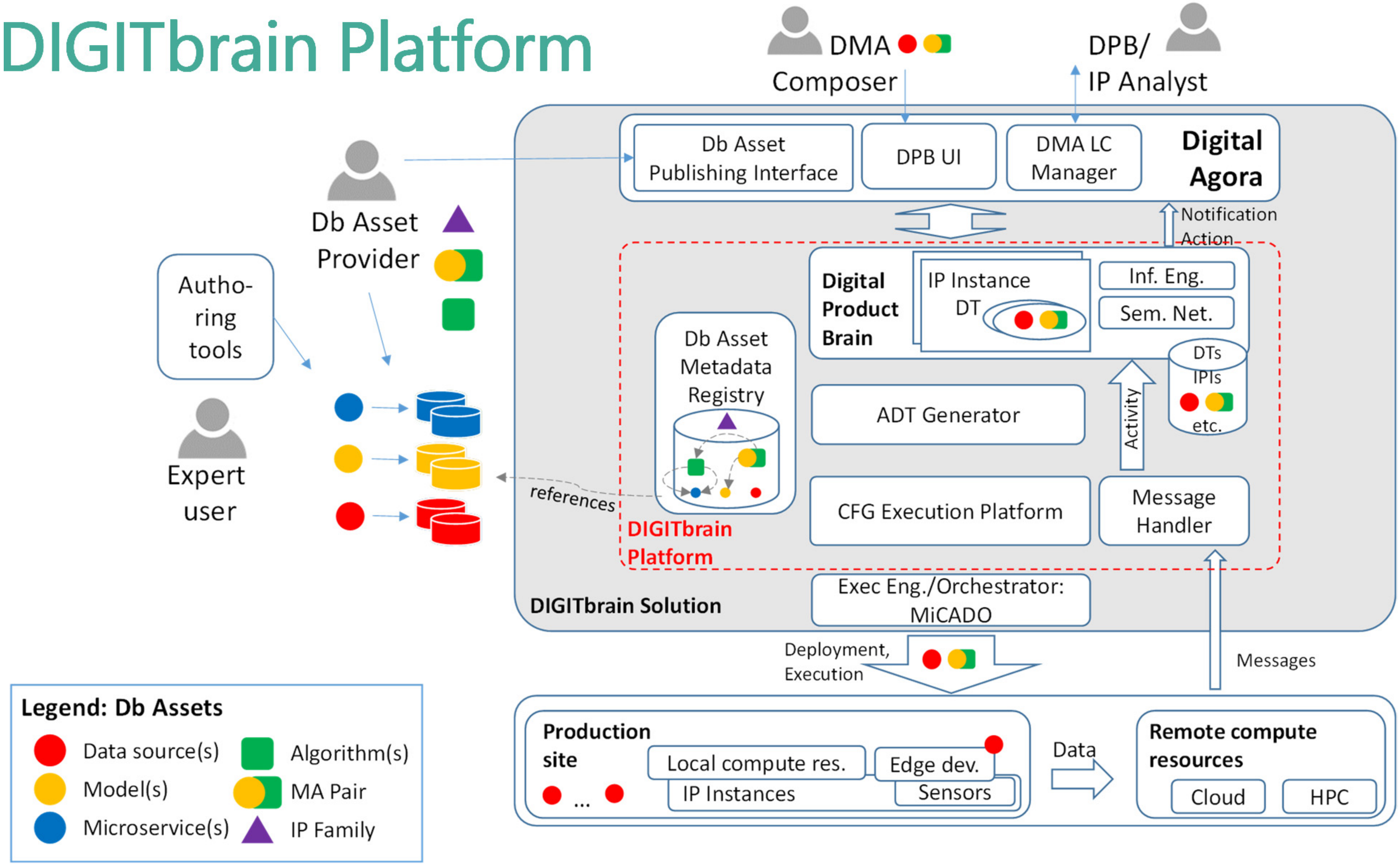
- **Data:** Refers to data sources in the form of files, streams or databases; data sources are considered to be external to the Db Platform (typically residing at the factory)
- **Model:** A Model is a description of a certain behaviour of/for an Industrial Product according to the given characteristics and operation conditions. A model is typically a file.
- **Microservice:** Encapsulates an executable procedure in a containerized form, e.g. in a Docker container.
- **Algorithm:** An Algorithm consists of one or more Microservice(s) and evaluate a model. Algorithms (its microservices) can be deployed into different resources (e.g. edge, cloud, HPC) to be executed, depending on the needs of the Model.



- Digital Twin: a DMA tuple

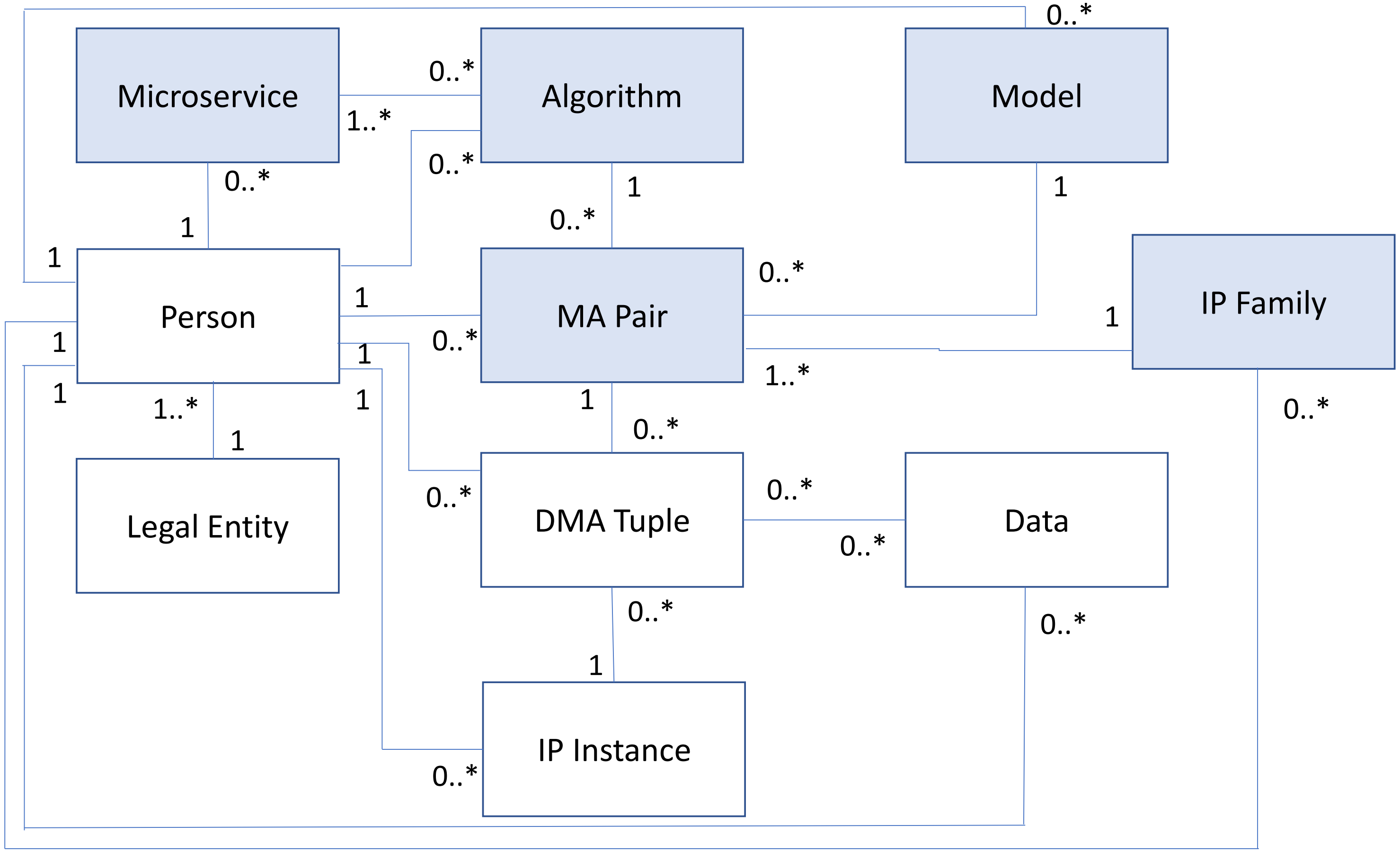
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The DIGITbrain Platform



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

DIGITbrain Metadata



https://digitbrain.github.io/

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

DIGITbrain Metadata

Microservice

Back to top

This page has been auto-generated based on an OpenAPI Specification
Most users will prefer this view which may include examples and additional info

Name	Type	Details
name	string character varying	description: human readable short, yet descriptive name of the Microservice.
version	string character varying	description: version as defined by the user.
description	string character varying	description: human readable short description of the Microservice's capabilities.
classification_schema	string public.classification_schema	enum: ['Simulation', 'ML', 'others'] description: fine-granular classification of the Microservice
type	string ARRAY	description: detailed type of the microservice, list of keywords
deployment_format	string public.deployment_format	enum: ['docker-compose', 'kubernetes-manifest'] description: identifier of the deployment environment required to deploy the Microservice's container
deployment_data	string json	description: JSON of docker-compose or kubernetes manifest required to run the container
configuration_data	string ARRAY	description: List of objects specifying configuration file(s) content required by the service

Algorithm

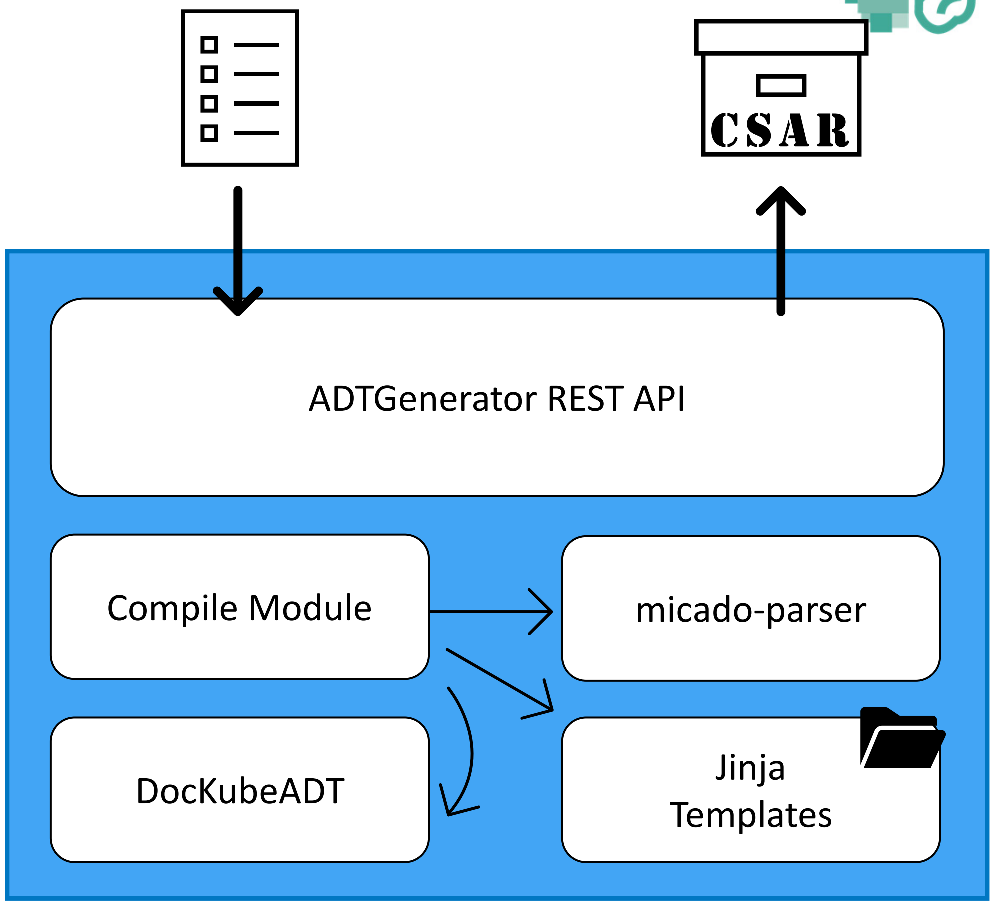
This page has been auto-generated based on an OpenAPI Specification
Most users will prefer this view which may include examples and additional info

Name	Type	Details
name	string character varying	description: a human-readable name to ease identification and discoverability for human users
description	string character varying	description: a short, human-readable description of the Algorithm to aid a human user in analysing the Algorithm's capabilities and its applicability to a certain problem
classification_schema	string public.classification_schema	enum: ['Simulation', 'ML', 'others'] description: the classification of the Algorithm, to describe the specialization area
type	string ARRAY	description: a detailed list of attributes to describe the Algorithm's field of application
version	string character varying	description: the version, as defined by the provider
list_of_microservices	string ARRAY	description: a list of Microservice Asset IDs, which are contained in the algorithm
deployment_mapping	string json	description: a mapping specifying which microservice should run on which host. By default each microservice is assigned a respective host, but this behaviour is not always ideal (eg. when two or more Microservices may need to run on the same host)

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The ADT Generator

- Aim:
  - Translate metadata key/value pairs into a TOSCA application descriptor
  - Compose various metadata (Ms, D, M, A, MA, DMA) into a single descriptor file
- Challenge:
  - all created by different authors at different phases



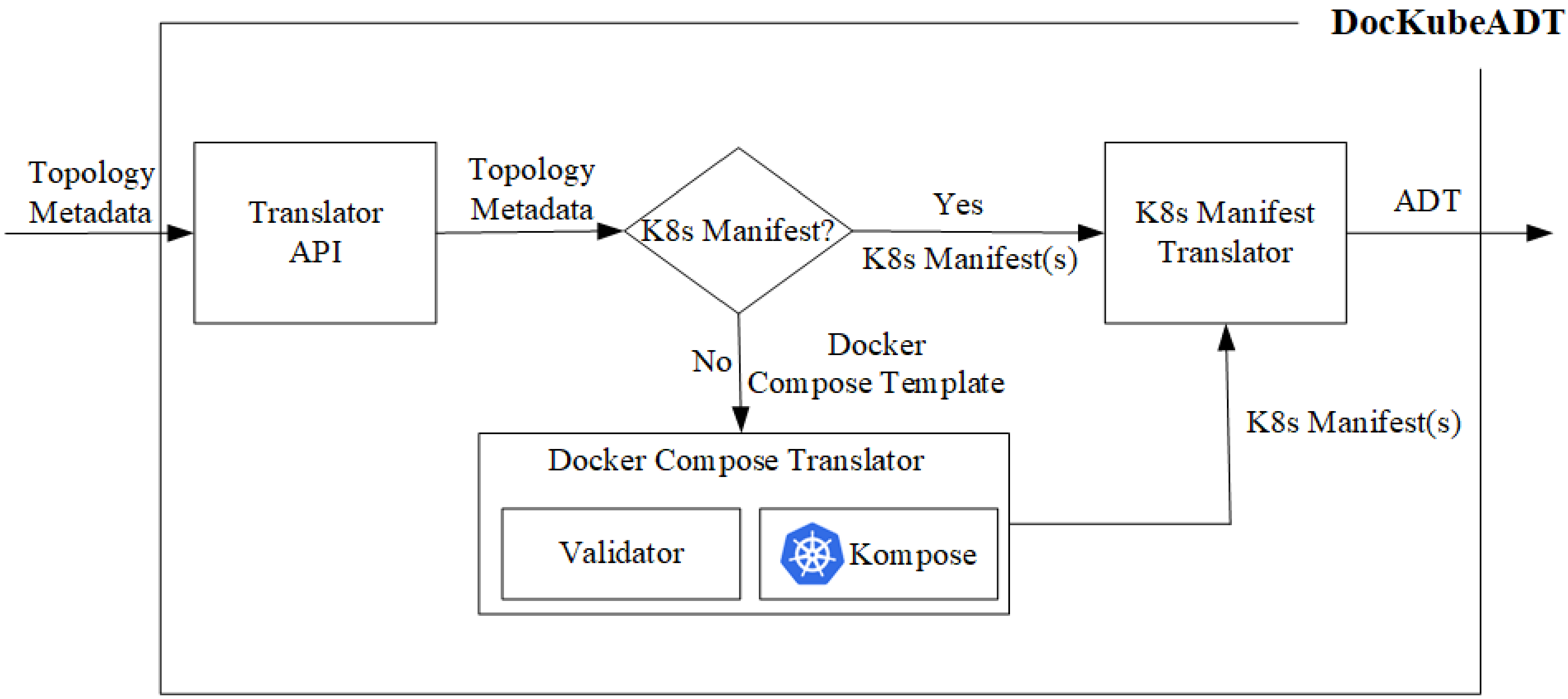
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

ADT generation in multiple steps

- 1. prepare
  - 1.1 save inputs
  - 1.2 validate inputs
- 2. generate deployments
  - 2.1 compile deployment
  - 2.2 save deployment
- 3. generate algorithm
  - 3.1 compile algorithm
  - 3.2 save algorithm
- 4. generate microservices
  - 4.1 substitute data asset references
  - 4.2 substitute model asset references
  - 4.3 checking unresolved references
  - 4.4 compile microservice
  - 4.5 save microservice
- 5. pack
  - 5.1 insert micado type imports
  - 5.2 create csar
  - 5.3 validate csar
- 6. store
  - 6.1 uplodad to s3

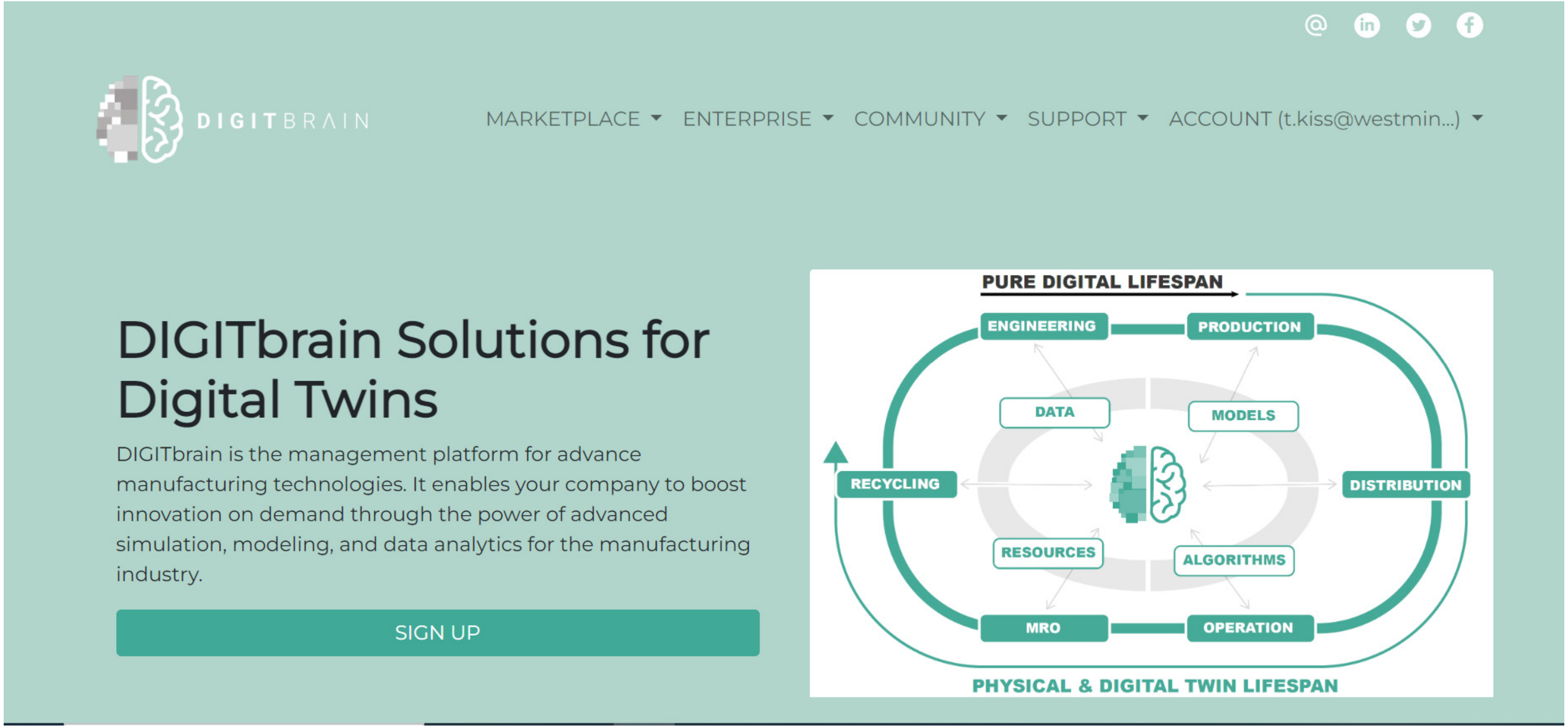
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

DocubeADT



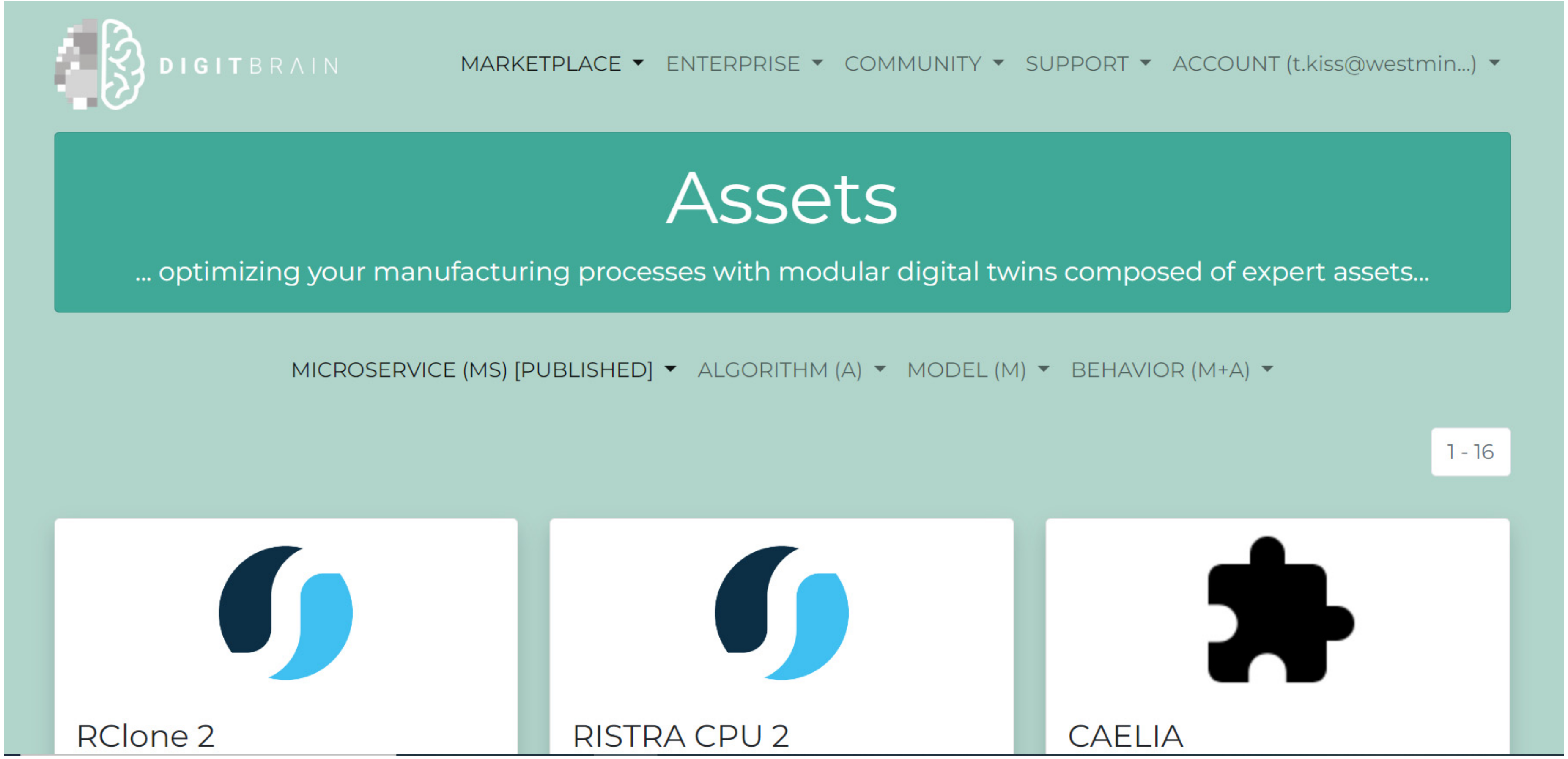
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How does it all work?



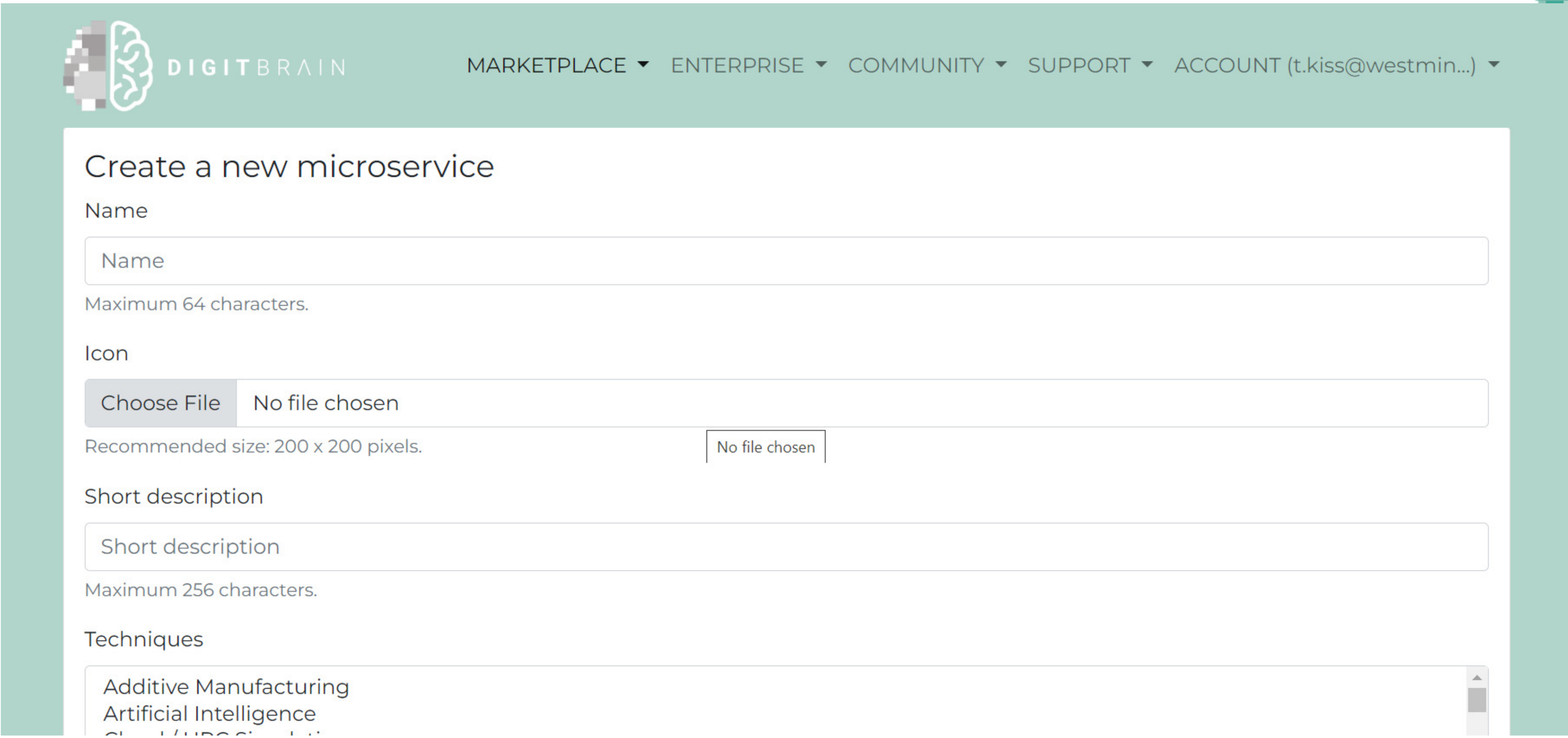
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Publish assets, build and execute DMA tuples



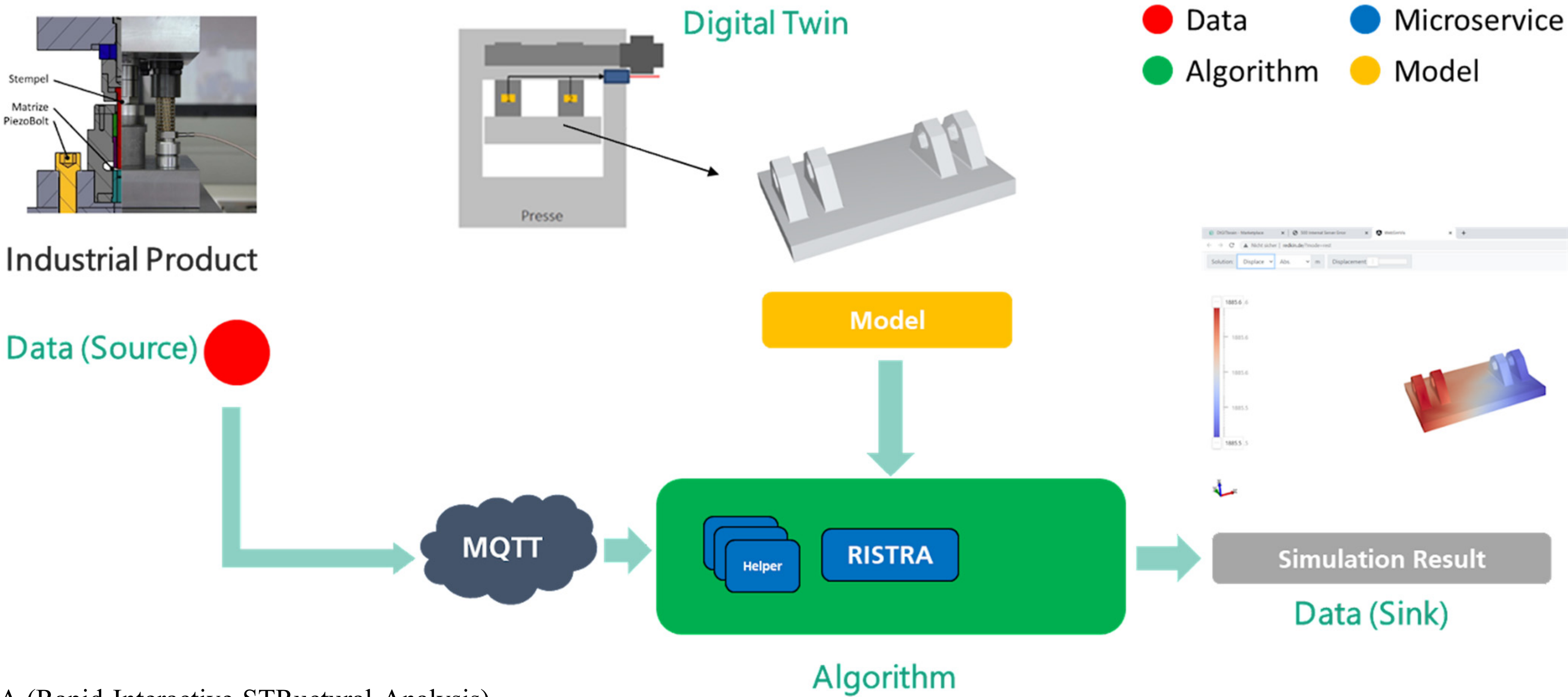
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Publish assets, build and execute DMA tuples



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Example: Digital Twin of a Punching Machine



RISTRA (Rapid Interactive STRuctural Analysis),
a software library for fast structural analysis.

Example provided Fraunhofer IGD – Andre Stork, Daniel Weber, Johannes Mueller-Roemer, Lars Lotter, Maxim Redkin

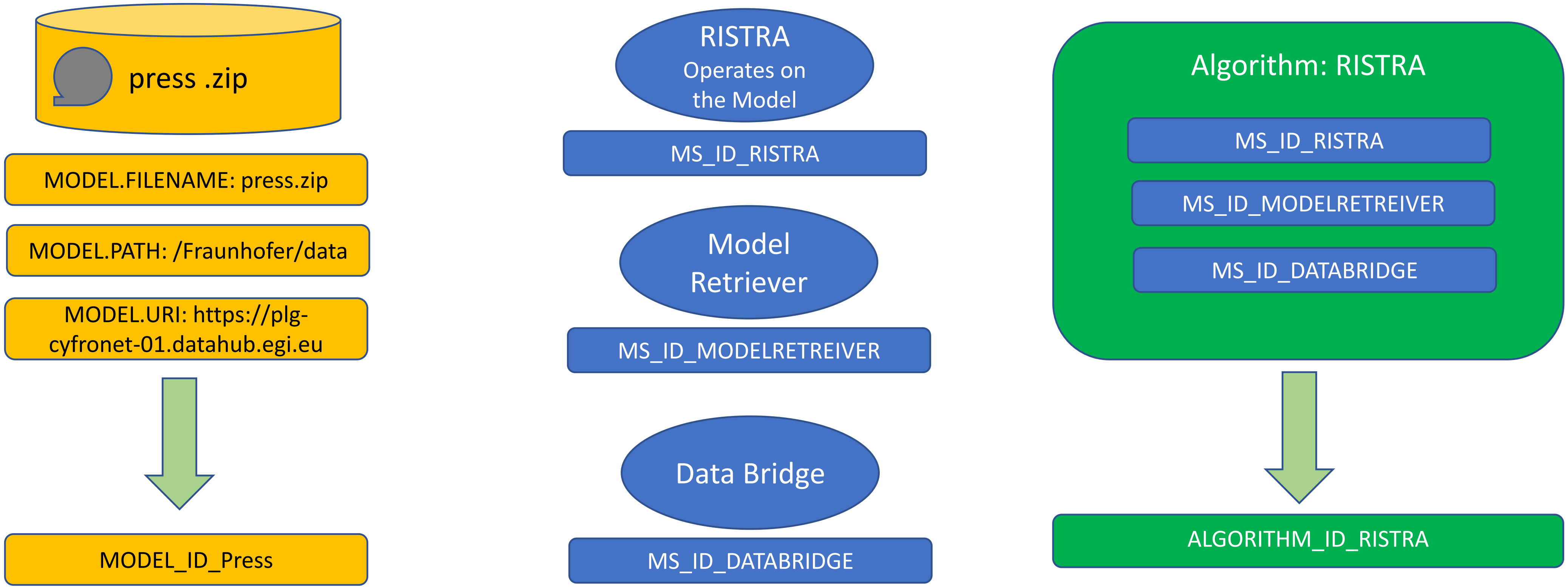
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How to build the DMA Tuple?

Model: Created with RISTRA using a CAD model

Microservices: Core and additional functionality

Algorithm: Includes three Microservices



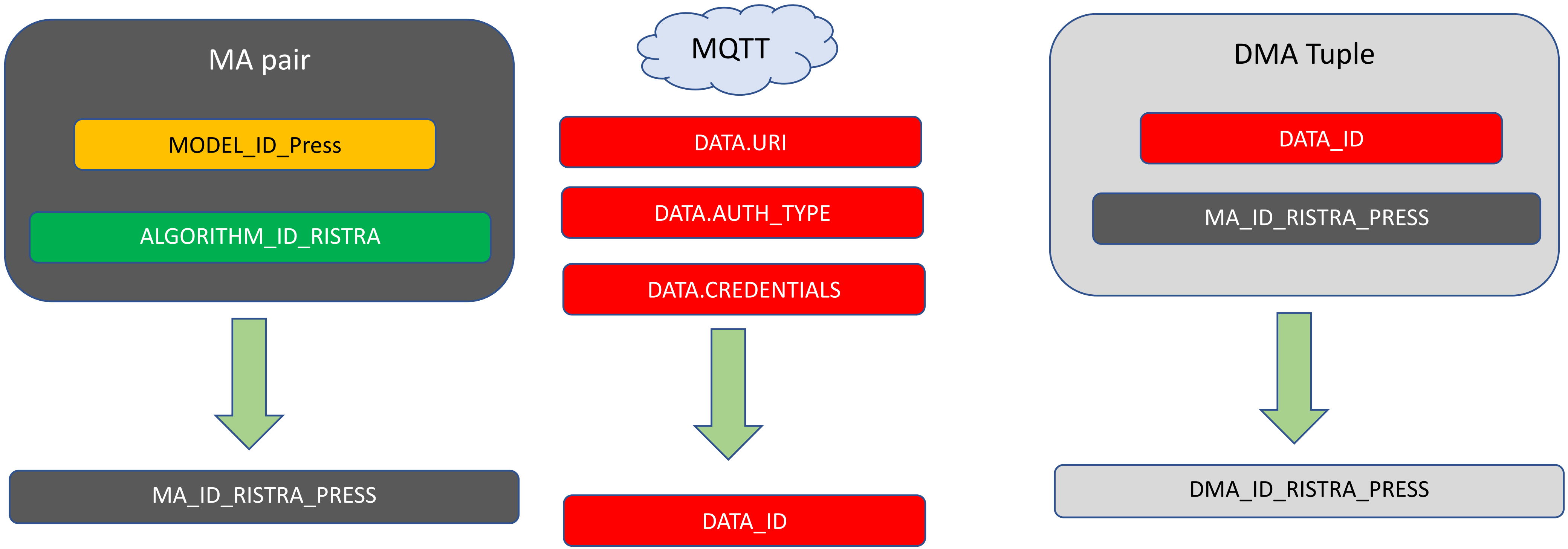
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How to build the DMA Tuple?

MA Pair: Describes the behaviour of an IP Family

Data source: From IoT sensors via MQTT

DMA Tuple: The executable Digital Twin



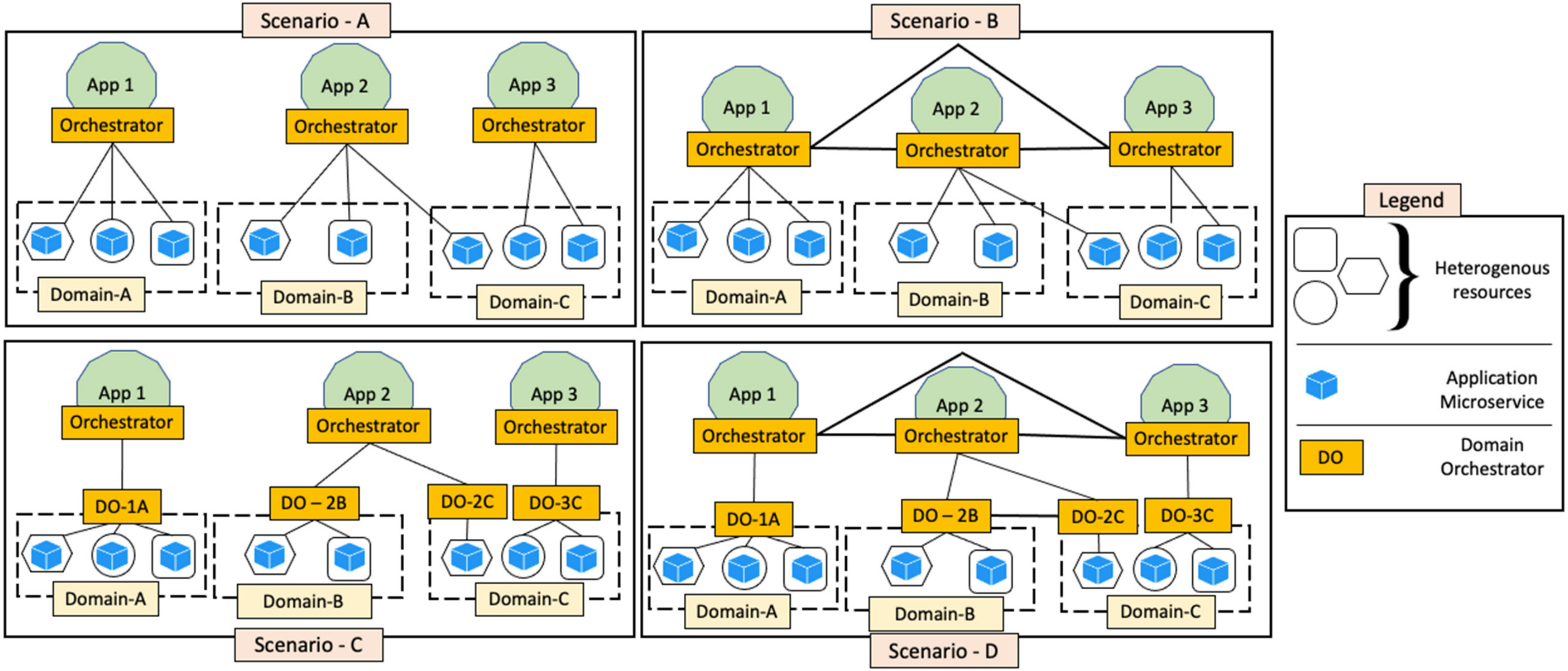
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

How to execute the DMA Tuple?

- Once DMA Tuple is built ADT generator can create TOSCA ADT for execution
- Copy of MiCADO is deployed by the platform
- ADT is passed to MiCADO to execute
- Resources are allocated based on ADT - RISTRA core simulation executes on edge

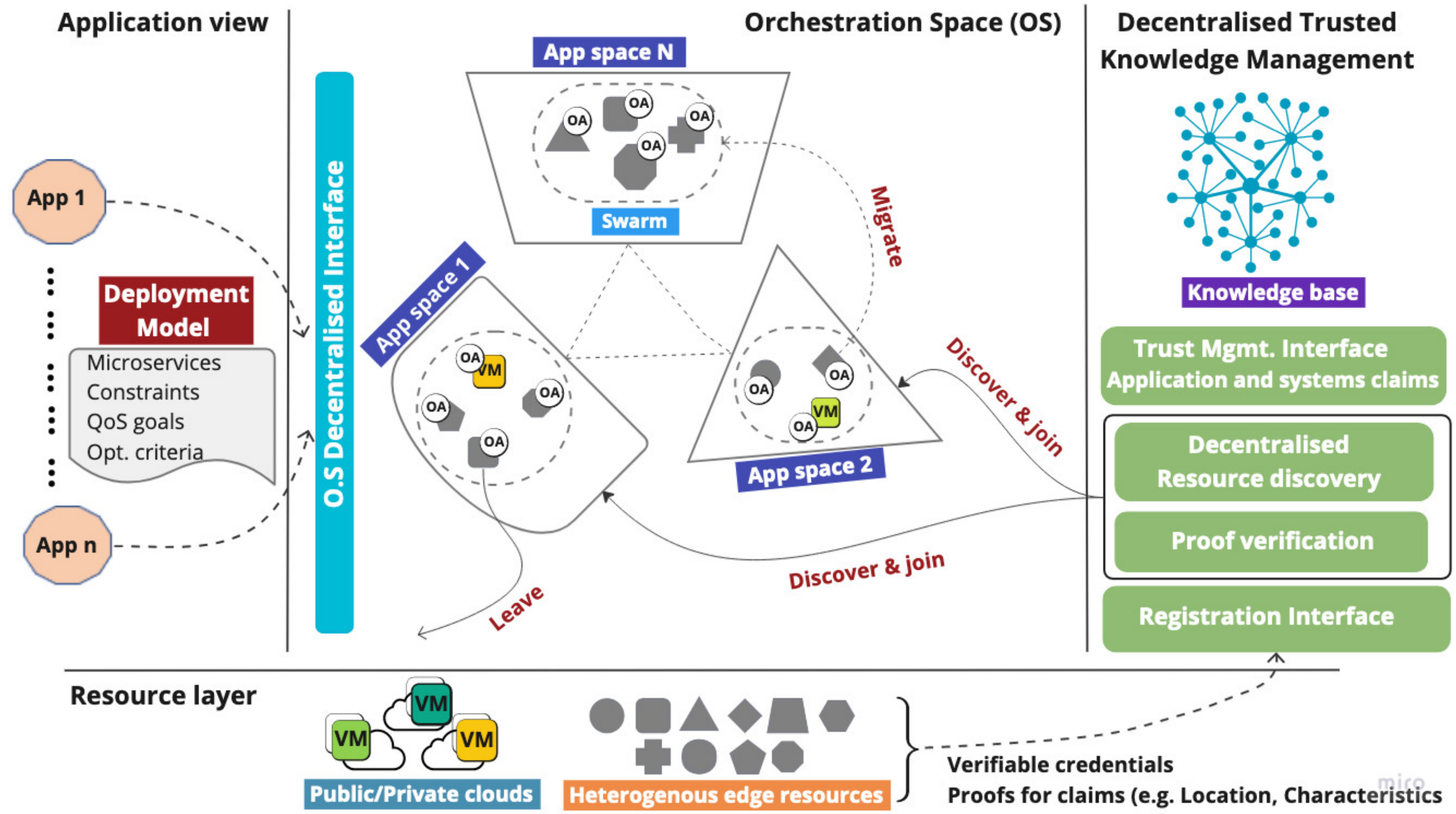
DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Where to go next? – Decentralised orchestration



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

Where to go next? - AI-Driven Swarm-based orchestration



DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071

The current CPC team behind all of this

Tamas Kiss

Gabor Terstyanszky

Gabriele Pierantoni

Huseyin Dagdeviren

Francesco Tusa

Hamed Hamzeh

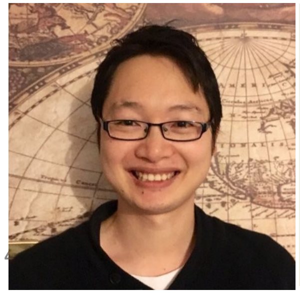
James DesLauriers

Antonis Michalas

Amjad Ullah

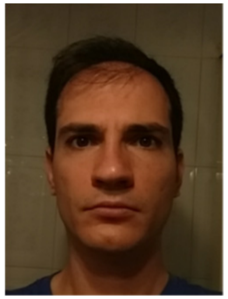
Jozsef Kovacs

Alim Gias



Huankai Chen

Yang Ma



Dimitris Kagialis

David Chan You Fee

DIGITbrain has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 952071

Thank you for your attention!

Prof Dr Tamas Kiss

University of Westminster

t.kiss@westminster.ac.uk

www.digitbrain.eu

www.linkedin.com/groups/12439191

www.twitter.com/digitbrain_eu

www.facebook.com/DIGITbrainProject

DIGITbrain has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952071