ENGAGE WP2

Supporting Efficient Workflow Deployment of Federated Learning Systems on the Computing Continuum

PhD candidate
Cédric Prigent, Inria

Advisors
Gabriel Antoniu, Inria
Alexandru Costan, Inria
Loïc Cudennec, DGA MI

24-05-22

Who Am I ?

- **Cédric Prigent**
  - 24 years old

PLOUENAN
BREST
UBO
Université de Bretagne Occidentale
RENNES
Inria
Lannion
Morlaix
Landivisiau
Landerneau
Plouzané
Guingamp
Saint-Brieuc
Lamballe
Dinan
Saint-Malo
Granville
Carhaix-Plouguer
Loudéac
Pontivy
Douarnenez
Quimper
Concarneau
Quimperlé
Hennebont
Guidel
Ploërmel
Bruz

- **Education (BSc,MSc)**
  - University of Western Brittany, Brest

- **PhD thesis**
  - Inria of the University of Rennes, INSA, Rennes
  - Supporting Online Learning and Inference in Parallel across the Digital Continuum
  - Funded by ENGAGE project

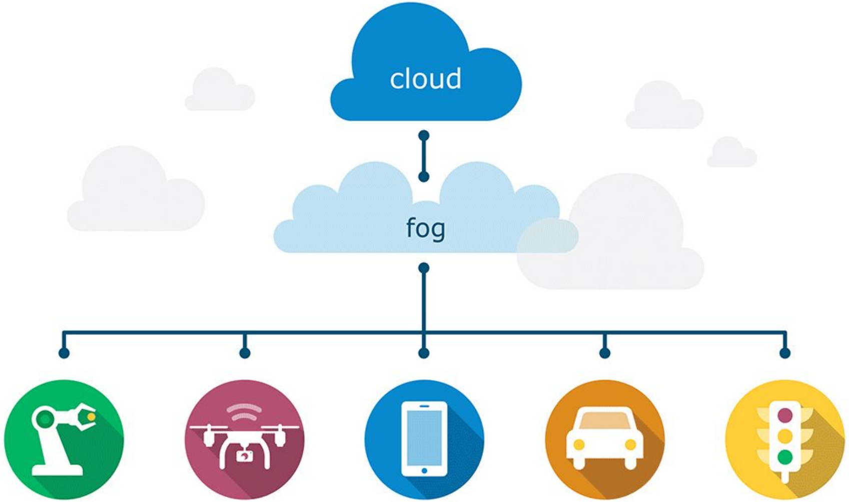
My Position in Work Package 2

Investigating various deployment strategies for complex AI workflows

- ○ How different deployment options impact performance metrics in a Digital Continuum
- ○ How can the available infrastructure can be best leveraged in this context
- ○ How the end-to-end performance of the application is correlated to various algorithmic-dependent and system-dependent factors

Computing Continuum

- **Interconnected ecosystem**
  - Allowing complex applications to be executed from IoT devices to HPC Cloud systems.
- **Emergence of a space**
  - In which complex data workflow systems operate over Cloud, Fog and Edge resources



Smart Living Use Case (Proposed by DFKI)

- **Non-intrusive load monitoring**
  - From the global power consumption of the house
  - Predict the consumption of each object with a fine granularity
  - Predict which object is used at a given moment

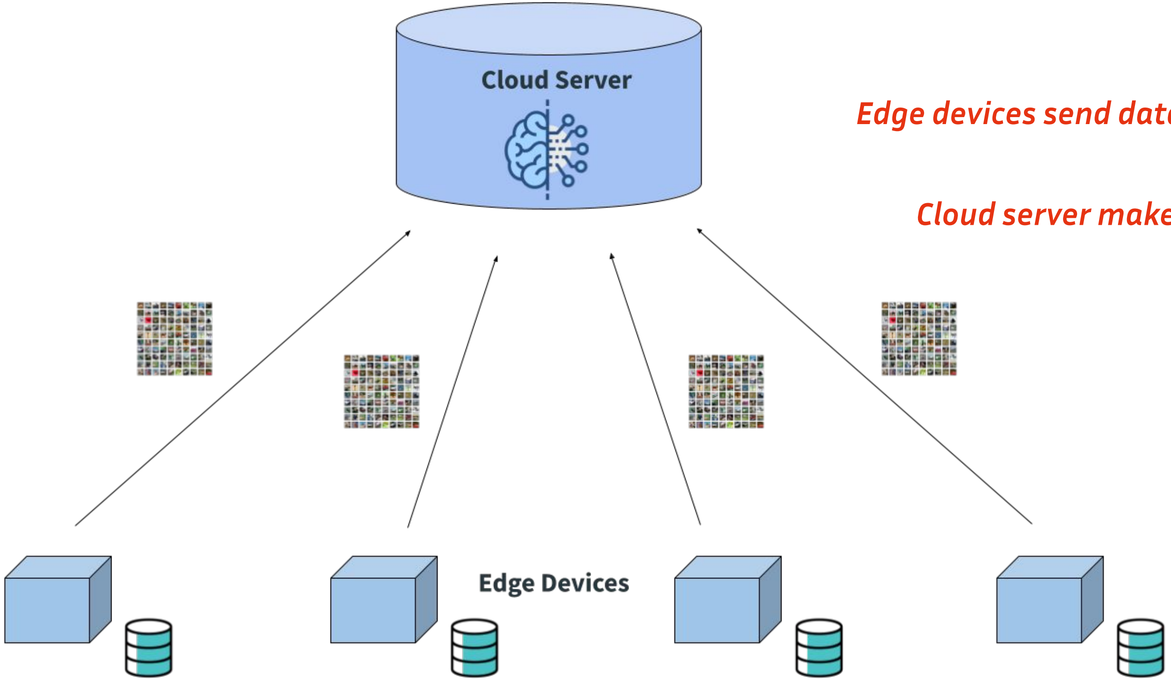
- **Investigating deployment strategies of AI workload in this context**

ML Settings We Want to Investigate

- **Centralized ML**
  - Computation on the cloud

- **Federated Learning**
  - Computation in the edge

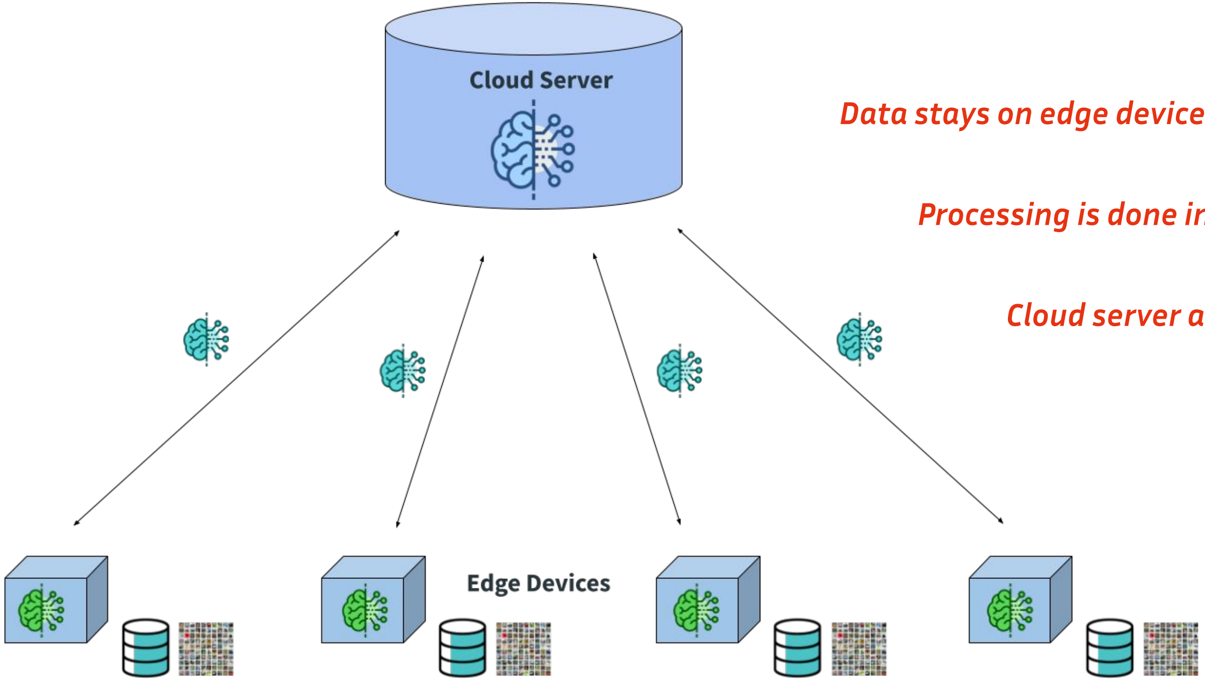
Centralized ML



Edge devices send data to the cloud

Cloud server makes all the processing

Federated Learning



Data stays on edge devices

Processing is done in the edge

Cloud server aggregates local models

ML Settings Pros & Cons

- **Centralized ML**
  - Taking advantage of the Cloud
    - Stable environment
    - Computing power
  - Bandwidth can be a bottleneck
- **Common tools**
  - TensorFlow
  - PyTorch
  - Kafka
  - Flink

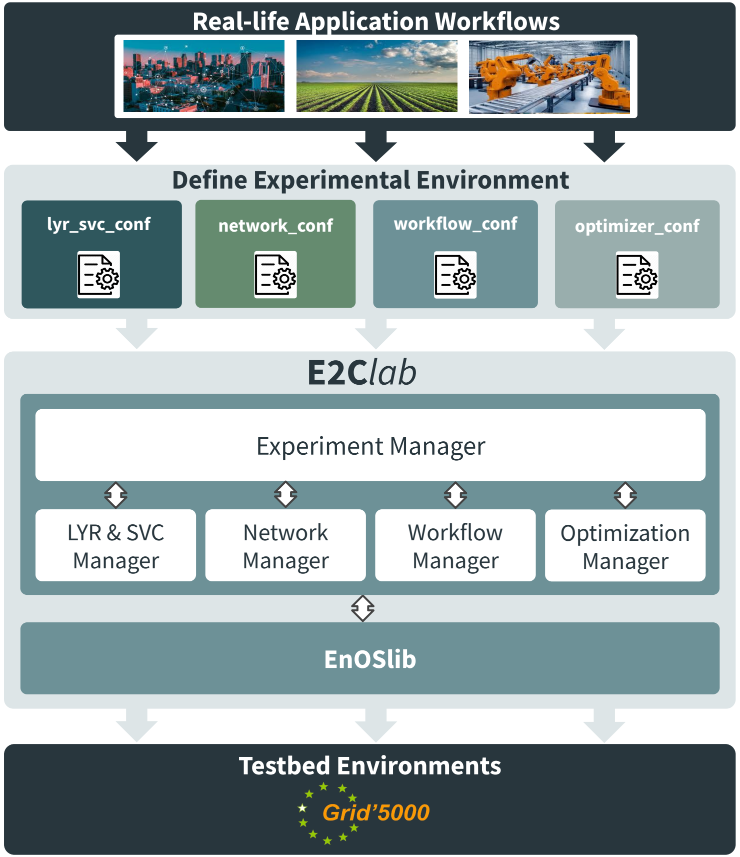
- **Federated Learning**
  - Taking advantage of Edge resources
    - Privacy preservation
    - Reducing bandwidth usage
  - Heterogeneous and unstable environment
- **FL frameworks**
  - TensorFlow Federated
  - Flower
  - FedML
  - FATE

Metrics We Want to Evaluate

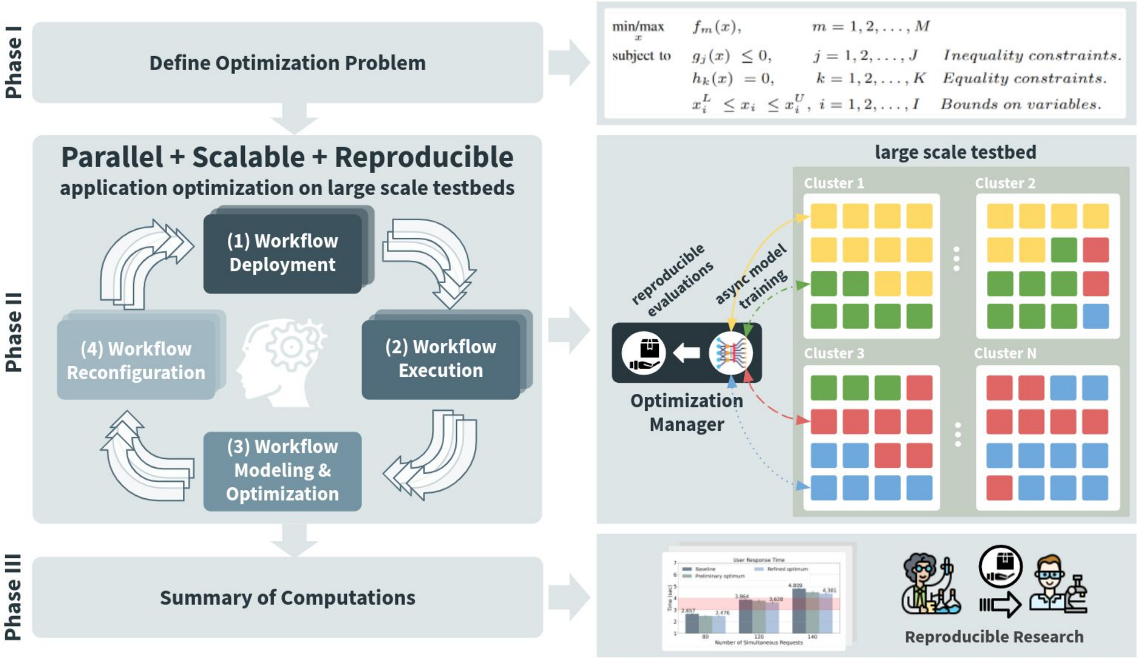
- **Execution Performance**
  - Execution time
  - Impact of scaling up
    - Starting from 2 households
    - Scale experiments with E2Clab
- **Model Precision**
  - Using Optimization tools
- **Energy Consumption**

E2Clab

- **Deployment tool**
  - Reproducible experiments
  - Testbed Environments
- **Components**
  - Layers and Services Manager
    - Reserving physical resources
    - Installing, configuring, launching services
  - Network Manager
    - Defining communication rules
  - Workflow Manager
    - Running the components of each service
  - Optimization Manager



Optimization Tool



Phase I

Define Optimization Problem

$$
\begin{array}{llll}
\min/\max_{x} & f_m(x), & m = 1, 2, \ldots, M & \\
\text{subject to} & g_j(x) \le 0, & j = 1, 2, \ldots, J & \textit{Inequality constraints.} \\
 & h_k(x) = 0, & k = 1, 2, \ldots, K & \textit{Equality constraints.} \\
 & x_i^L \le x_i \le x_i^U, & i = 1, 2, \ldots, I & \textit{Bounds on variables.}
\end{array}
$$

Phase II

Parallel + Scalable + Reproducible
application optimization on large scale testbeds

(1) Workflow Deployment

(2) Workflow Execution

(3) Workflow Modeling & Optimization

(4) Workflow Reconfiguration

large scale testbed

Optimization Manager

Phase III

Summary of Computations

Reproducible Research

Parallelize the optimization process

RAY tune

Supports SOTA Bayesian Optimization libraries

Scikit Optimize

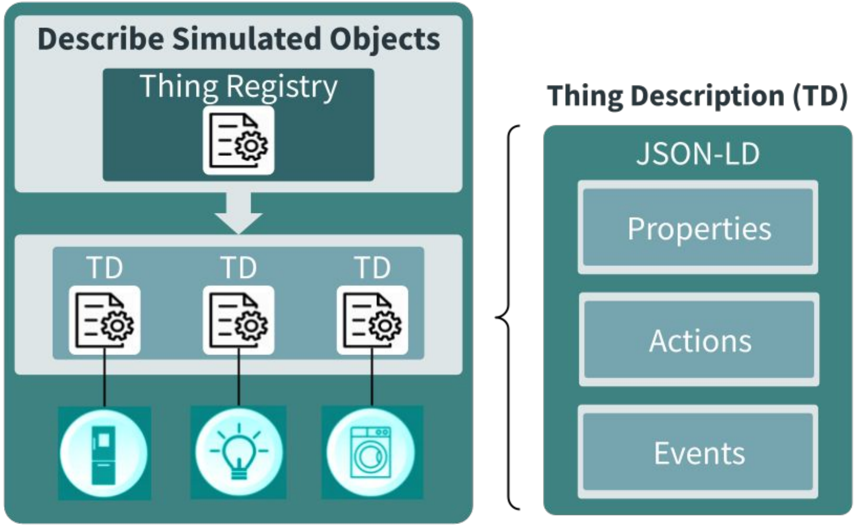
Ax

OPTUNA

Dragonfly
Scalable Bayesian Optimisation

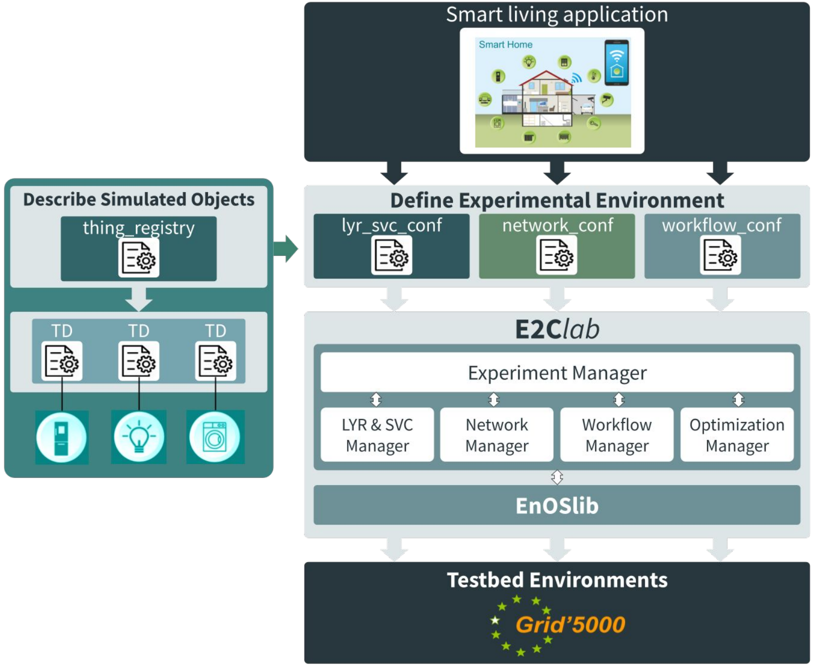
Thing Description/Registry

- **W3C Web of Things architecture**
  - Improve interoperability and usability across IoT platforms
- **Thing Description (TD)**
  - Entrypoint of a Thing
  - Metadata
  - Interactions
- **Thing Registry**
  - Manages TDs
  - Query interface



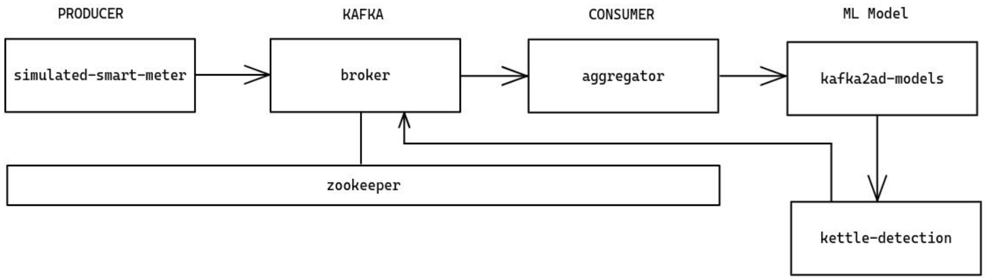
Our Approach

- **E2Clab (KerData Team - Inria)**
  - Deployment tool
  - Optimization tool
- **Thing Description/Registry (DFKI)**
  - Support semantic orchestration of IoT use cases
  - Describe simulated objects
- **Goal**
  - Describe/Orchestrate simulated devices with a same standard
  - Deploy/Optimize the application with E2Clab

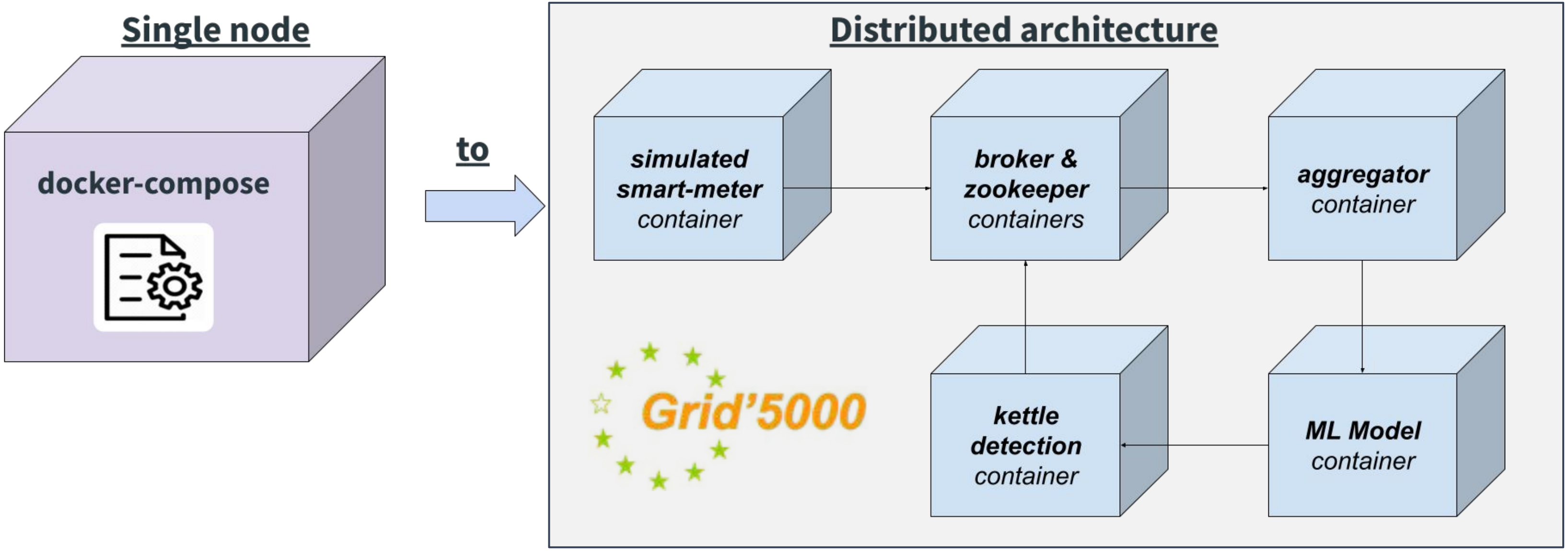


Toy Example Provided by DFKI

- **Docker-compose**
  - Services
    - Simulated smart-meter
    - Broker
    - Zookeeper
    - Aggregator
    - Kafka2ad-models
    - Kettle-detection
  - Running on a single device



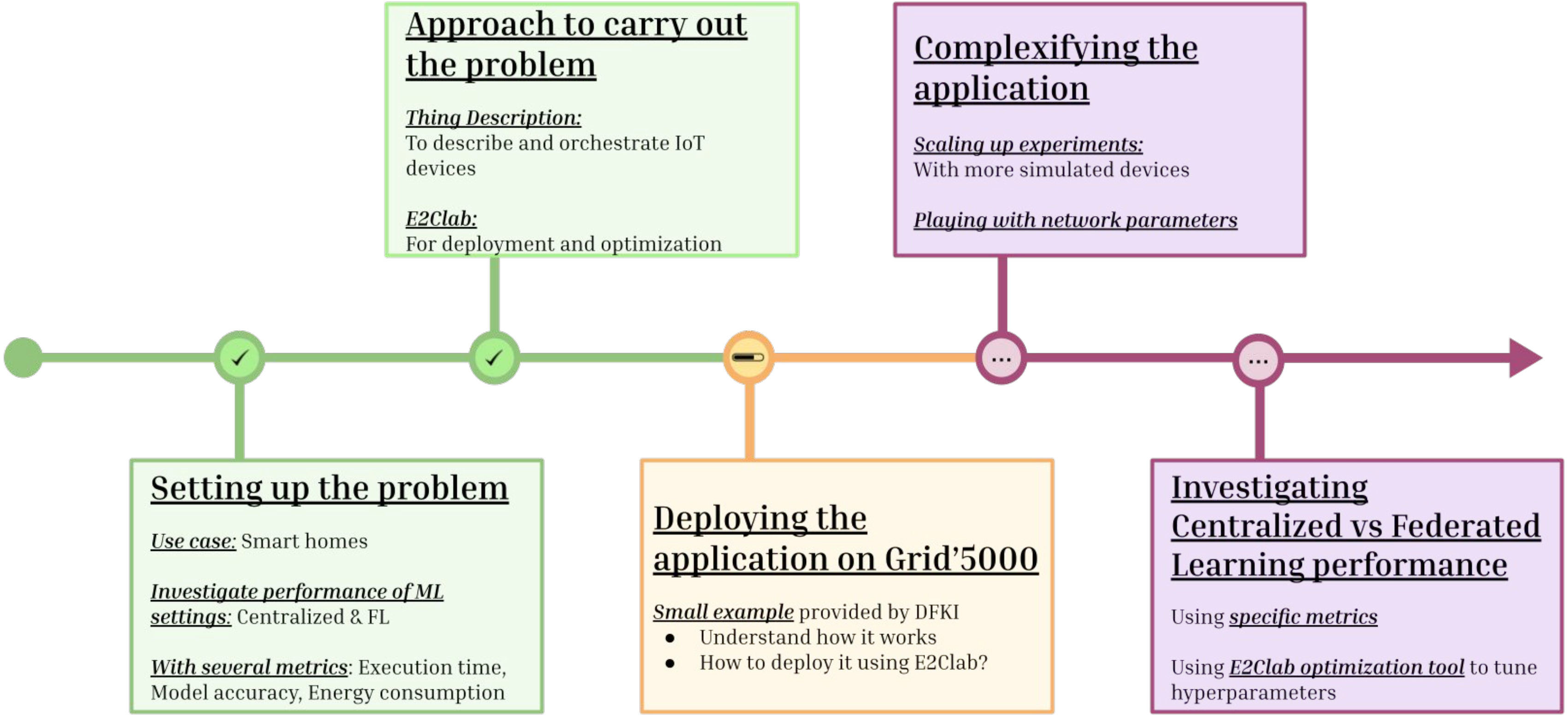
Work in Progress: Deployment on Grid'5000



Next steps: Investigate ML Deployment Strategies

- **Complexify the application**
  - Scaling up the application
    - Add simulated devices (described with TDs)
  - Varying network configurations
- **Investigate Centralized vs Federated Learning performance**
  - Depending on application settings
    - Scale of experiments
    - Network settings
  - Optimizing the model
    - Using E2Clab optimization tool

Progress Status



Setting up the problem

Use case: Smart homes

Investigate performance of ML settings: Centralized & FL

With several metrics: Execution time, Model accuracy, Energy consumption

Approach to carry out the problem

Thing Description:
To describe and orchestrate IoT devices

E2Clab:
For deployment and optimization

Deploying the application on Grid'5000

Small example provided by DFKI
- Understand how it works
- How to deploy it using E2Clab?

Complexifying the application

Scaling up experiments:
With more simulated devices

Playing with network parameters

Investigating Centralized vs Federated Learning performance

Using *specific metrics*

Using *E2Clab optimization tool* to tune hyperparameters