Funded by
the European Union

CODECO
Cognitive Decentralised
Edge Cloud Orchestration

CODECO - PDLC DEMO SESSION

Alejandro Espinosa Polo

AI project Engineer

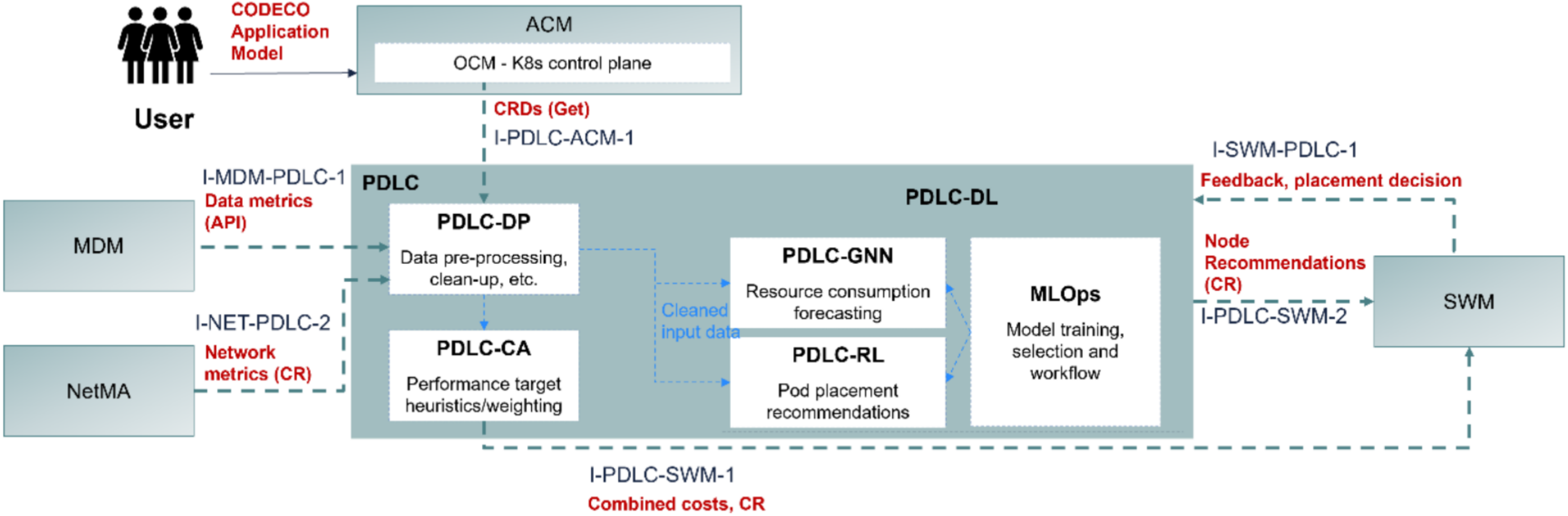
I2CAT Foundation

Date: 22/01/25

Venue: HIPEAC ML4CS Workshop

Funded by the European Union under Grant Agreement 101092696

PDLC GENERAL ARCHITECTURE

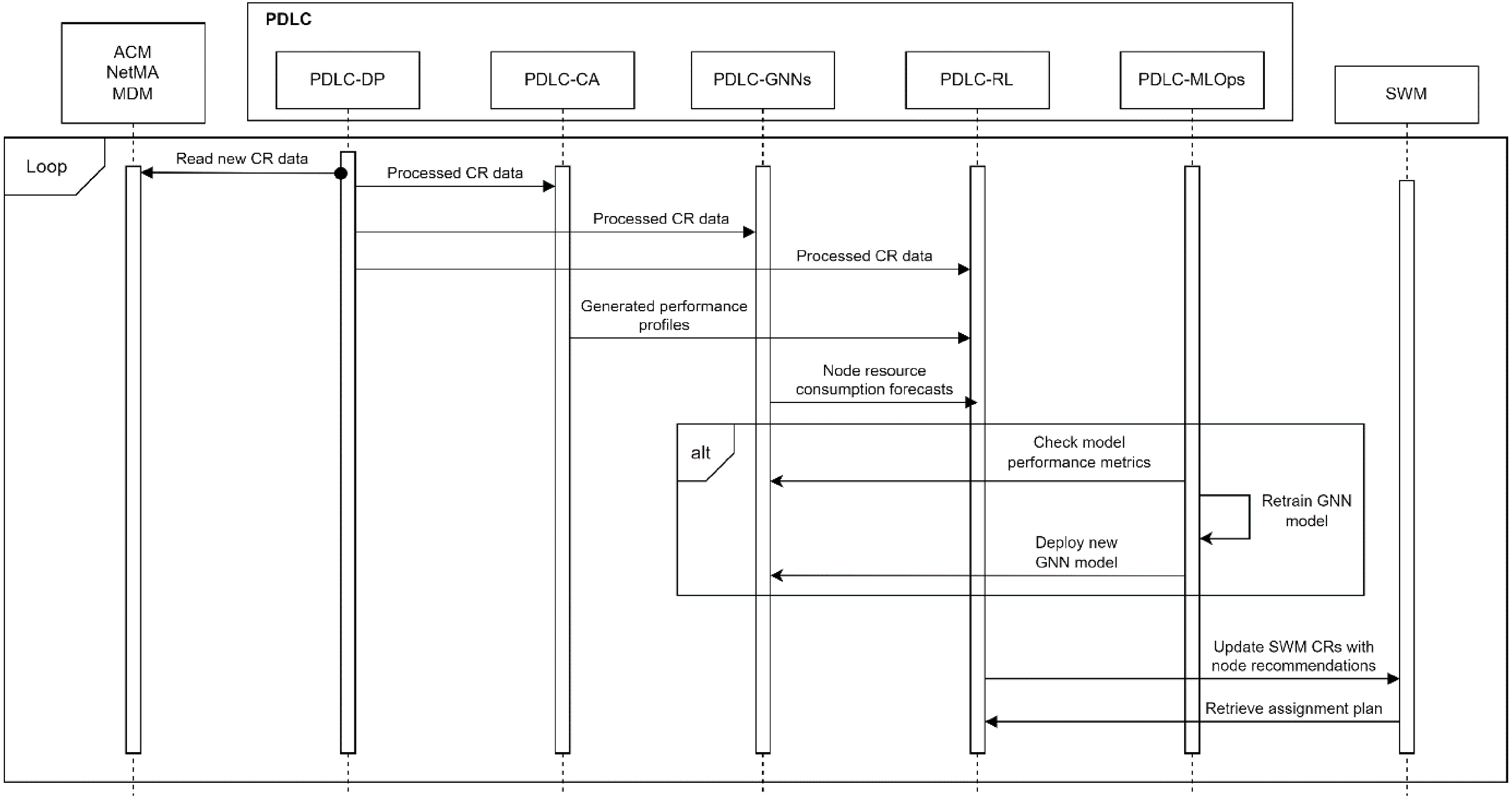


PDLC high-level architecture and interfaces.

Funded by the European Union

PDLC SEQUENCE DIAGRAM

Sequence diagram of PDLC and its sub-components within a single CODECO cluster.



Funded by the European Union

PDLC-CA

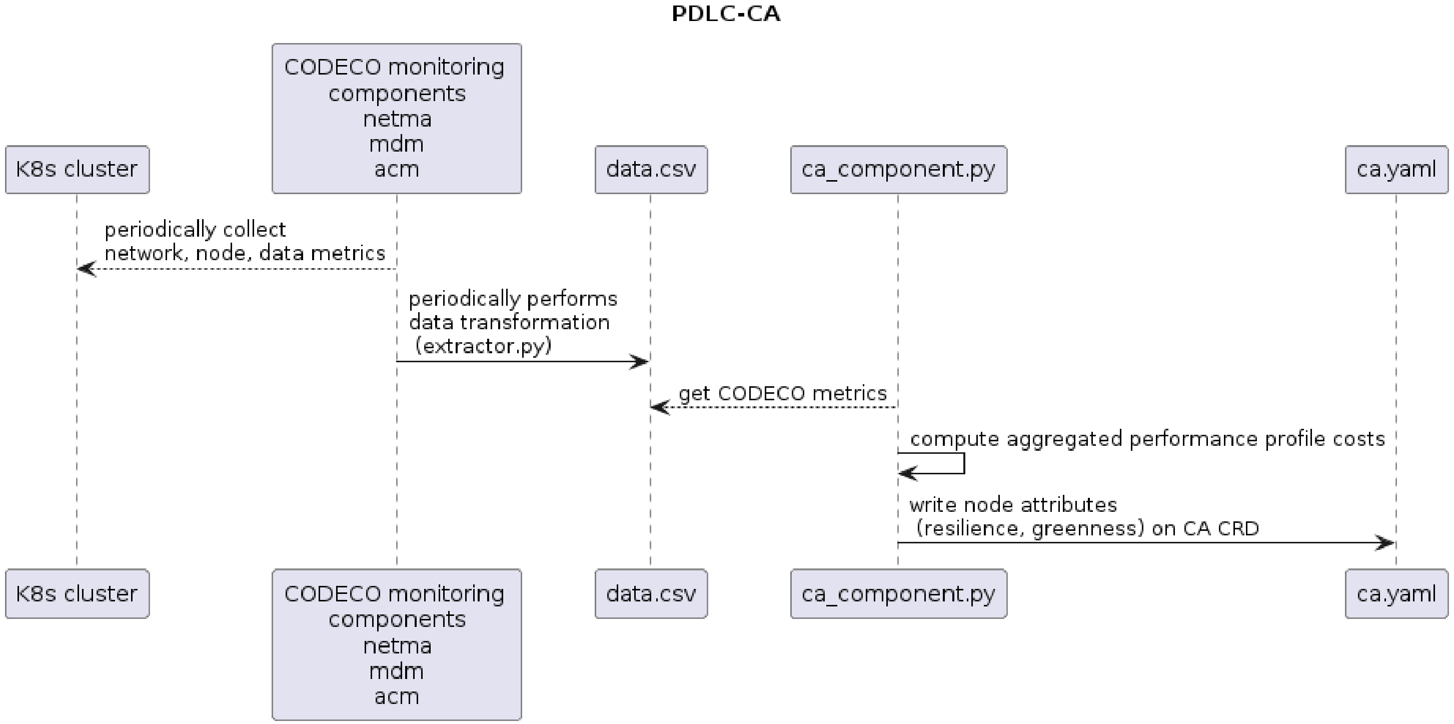
PDLC-CA creates aggregated metrics known as performance profiles to optimize the number of parameters passed in between CODECO components and to optimize pod placement using performance profiles such as Greeness or Resilience. Based on input provided by PDLC-DP it provides metrics to be consumed by PDLC-DL-RL.

Greenness profile: focused on creating minimal energy consumption

```
GreennessCost(n) = Node_Energy(n) x Links_energy(n)
```

Resilience profile: focused on maximizing overall infrastructure resilience

```
Resilience(n) = node_net_failure * 1/node_degree
                * sumlink_failure
```


PDLC-CA
K8s cluster
CODECO monitoring components netma mdm acm
data.csv
ca_component.py
ca.yaml
periodically collect network, node, data metrics
periodically performs data transformation (extractor.py)
get CODECO metrics
compute aggregated performance profile costs
write node attributes (resilience, greenness) on CA CRD
K8s cluster
CODECO monitoring components netma mdm acm
data.csv
ca_component.py
ca.yaml

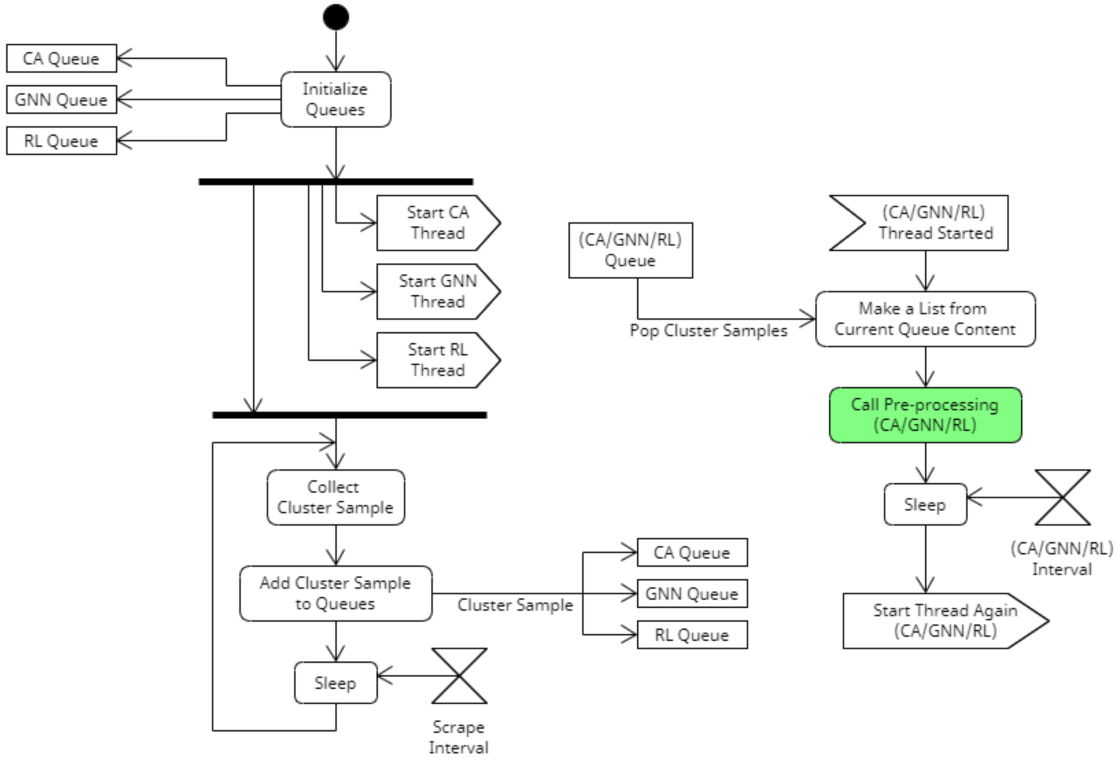
PDLC-CA internal sequence diagram

Funded by the European Union

PDLC-DP

PDLC-DP aims to provide processed data treated with a a range of techniques aimed to improved the quality and interpretability of data used by the rest of the components of PDLC.

1. **Gatherer:**
   a. Gathers data from ACM – MDM and NetMA periodically.
2. **Pre-processing operations per PDLC subcomponent:**
   a. Data Cleaning
   b. Data Integration
   c. Data Transformation
   d. Data Normalization
3. **Outputs:**
   a. Each subcomponent receives CSV file(s) with the information needed ready to use.



PDLC subcomponent	Final PDLC-DP output
PDLC-CA	- data_CA.csv
PDLC-GNN	- data_gnn.csv
PDLC-RL	- test_nodes.csv - test_integracio.csv

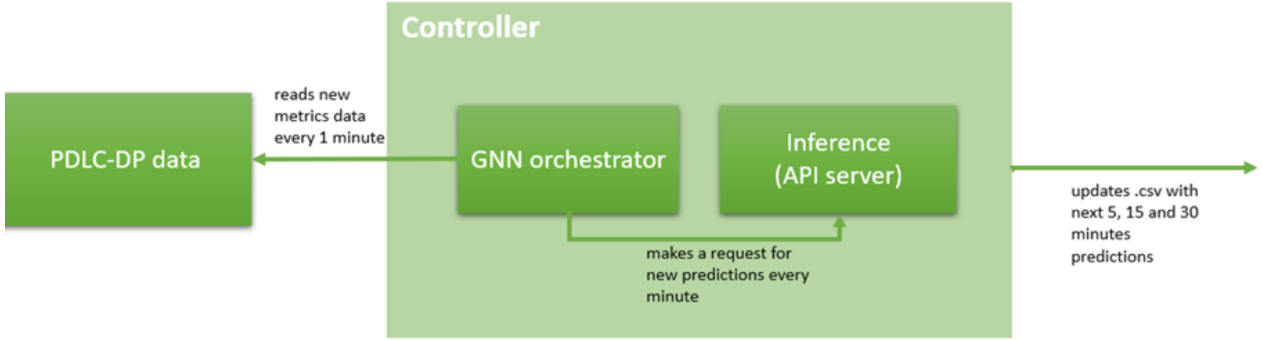
Output files of PDLC-DP

Funded by the European Union

PDLC-DL – GNN

PDLC- DL – GNNs uses the state of the system in order to create forecasts of CPU and RAM consumption across the cluster so that they can be used as input of the PDLC-DL-RL subcomponent.

1. **Input data:**
   a. Gathers data from PDLC-DP csv file, obtaining consumption of CPU and RAM across all nodes in the cluster.
2. **Functioning:**
   a. Queries API with historical data.
   b. Model is selected dynamically depending on the number of nodes in the cluster.
   c. Predicts future values on CPU and RAM consumption based on historical observations by using spatio-temporal graphs.
3. **Outputs:**
   a. Forecasted values are written in csv file depending on the forecast time (e.g. 5,15 or 30 minutes) so that RL can use them.



Controller

PDLC-DP data

reads new metrics data every 1 minute

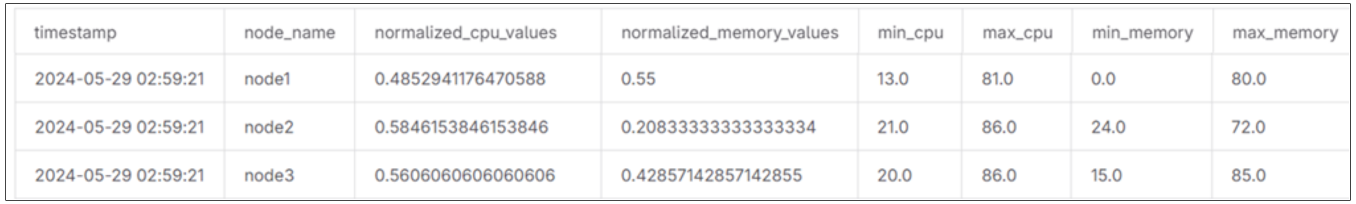
GNN orchestrator

Inference (API server)

makes a request for new predictions every minute

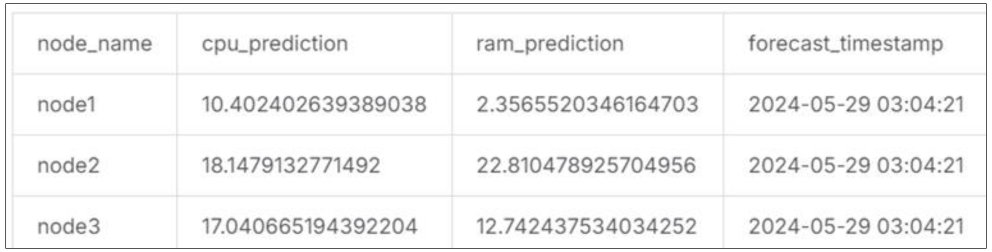
updates .csv with next 5, 15 and 30 minutes predictions

Input from PDLC-DP



timestamp	node_name	normalized_cpu_values	normalized_memory_values	min_cpu	max_cpu	min_memory	max_memory
2024-05-29 02:59:21	node1	0.4852941176470588	0.55	13.0	81.0	0.0	80.0
2024-05-29 02:59:21	node2	0.5846153846153846	0.20833333333333334	21.0	86.0	24.0	72.0
2024-05-29 02:59:21	node3	0.5606060606060606	0.42857142857142855	20.0	86.0	15.0	85.0

Output to PDLC-RL



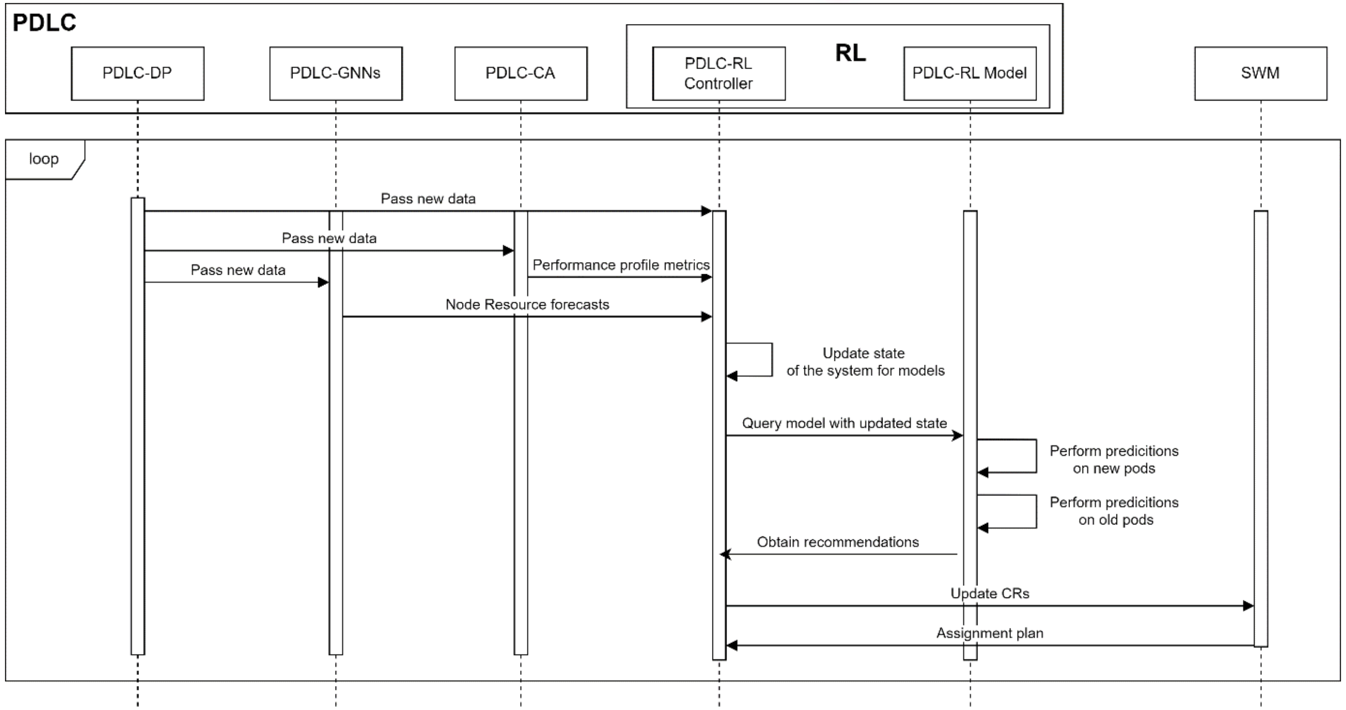
node_name	cpu_prediction	ram_prediction	forecast_timestamp
node1	10.402402639389038	2.3565520346164703	2024-05-29 03:04:21
node2	18.1479132771492	22.810478925704956	2024-05-29 03:04:21
node3	17.040665194392204	12.742437534034252	2024-05-29 03:04:21

Funded by the European Union

PDLC-DL – RL

PDLC- DL – RL uses the current (or forecasted if available) state of the system together with custom metrics such as greenness in order to provide node recommendations to optimize the chosen performance profile (e.g. reduce energy consumption in the system).
It serves as the central component of PDLC by having 4 inputs and 1 output.

1. **Input data:**
   a. Gathers data from PDLC-DP csv files.
   b. Gathers data from PDLC-CA csv file.
   c. Gathers state forecasts (if available) from GNNs csv file.
   d. It periodically obtains the assignment plan created by SWM (future usage) via K8s API.
2. **Functioning:**
   a. Model is selected dynamically depending on the number of nodes in the cluster.
   b. Creates node recommendations based on data received
3. **Outputs:**
   a. Writes node recommendations in SWM CRs via K8s python API.



```
nodeRecommendations:
        codeoco-master: 1.0
        codeco-worker1: 0.9
        codeco-worker2: 0.0
```

Funded by the European Union

LIVE DEMO

Funded by
the European Union

THANK YOU!

Questions?

Alejandro Espinosa Polo – **alejandro.espinosa@i2cat.net**

Funded by the European Union