

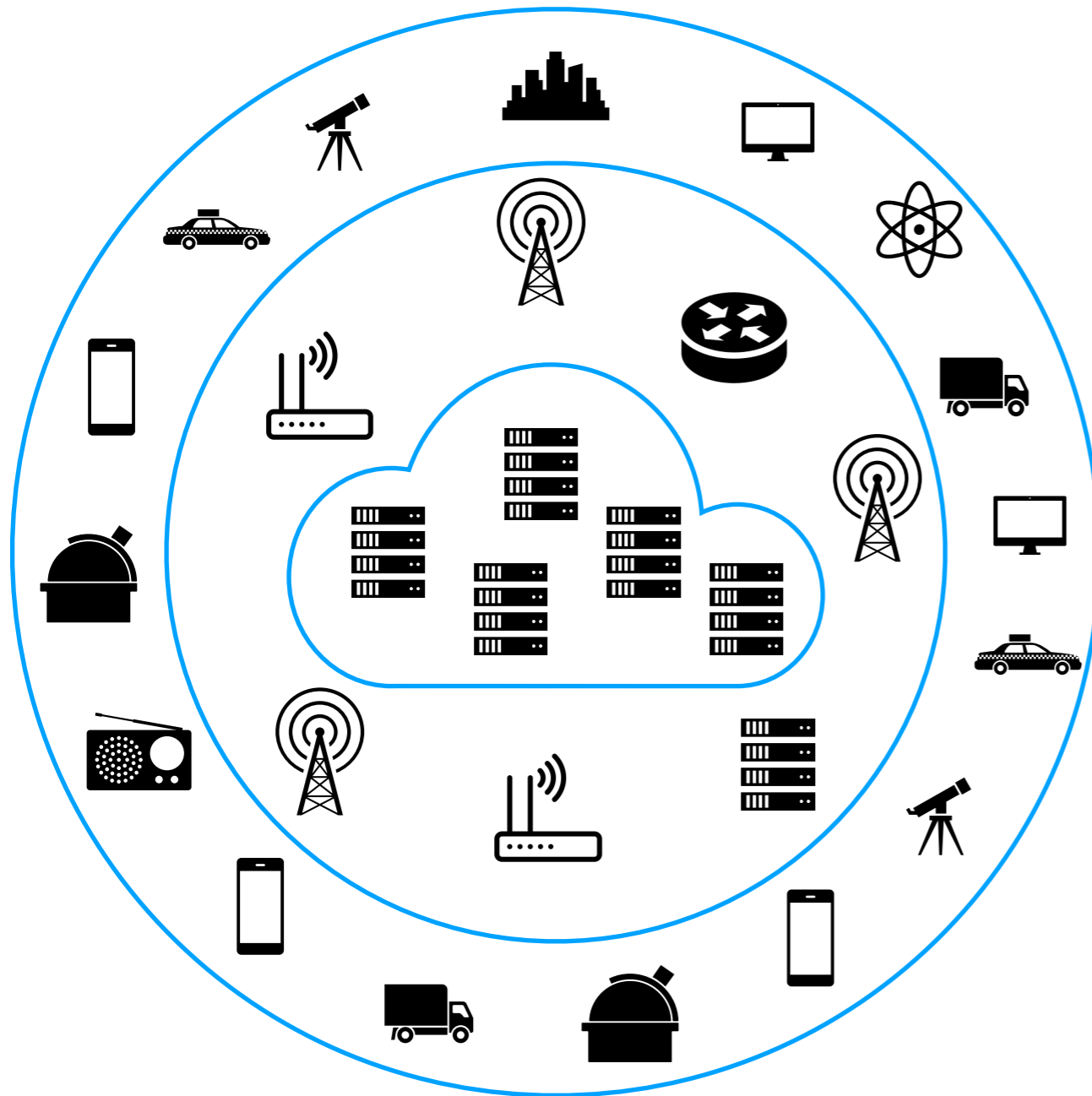
Improving the performance of AI-based Big Data applications using Edge computing

Pedro Silva
Kerdata Team - Inria de Rennes
pedro.silva@inria.fr

Agenda

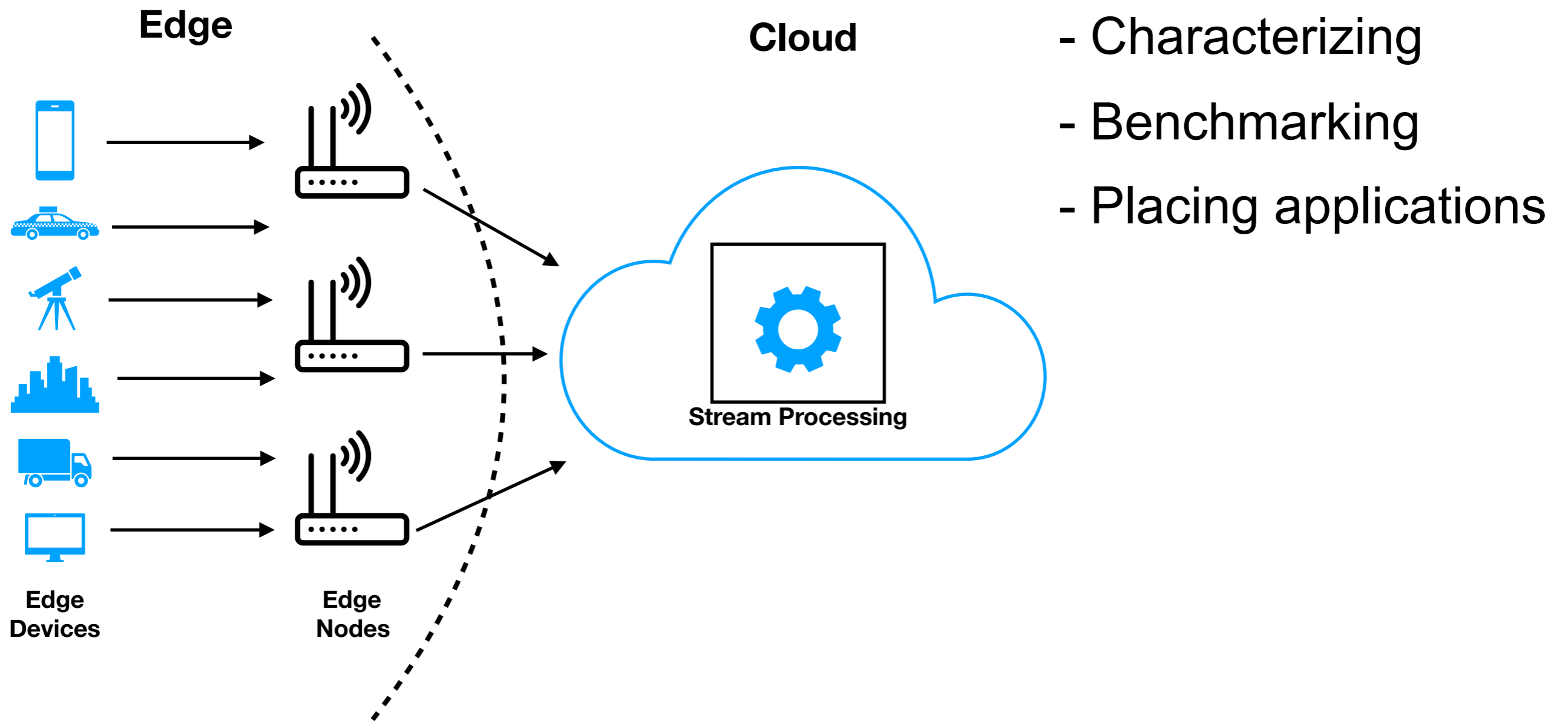
- Edge computing
- Earthquake early warning systems
- PI@ntnet

(Edge what?)

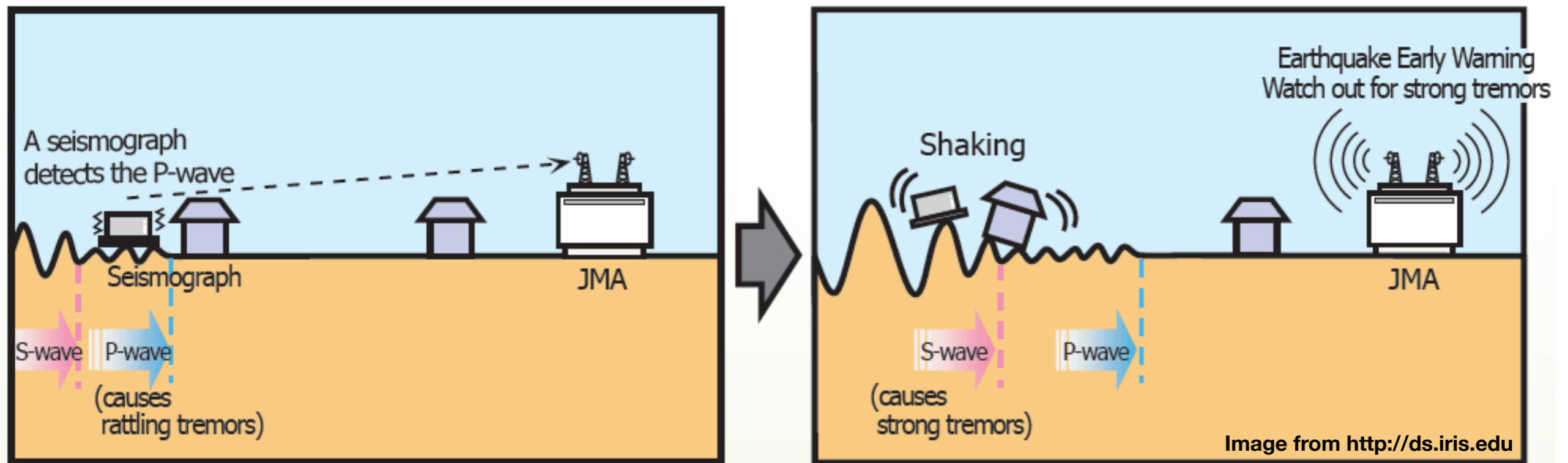


- **Edge**: everything between the Cloud and data sources;
- Edge computing advantages:
 - easier **access** to data;
 - **bandwidth** saving;
 - “**privacy**”;
 - potential high **parallelism**.

Characterizing Edge Processing

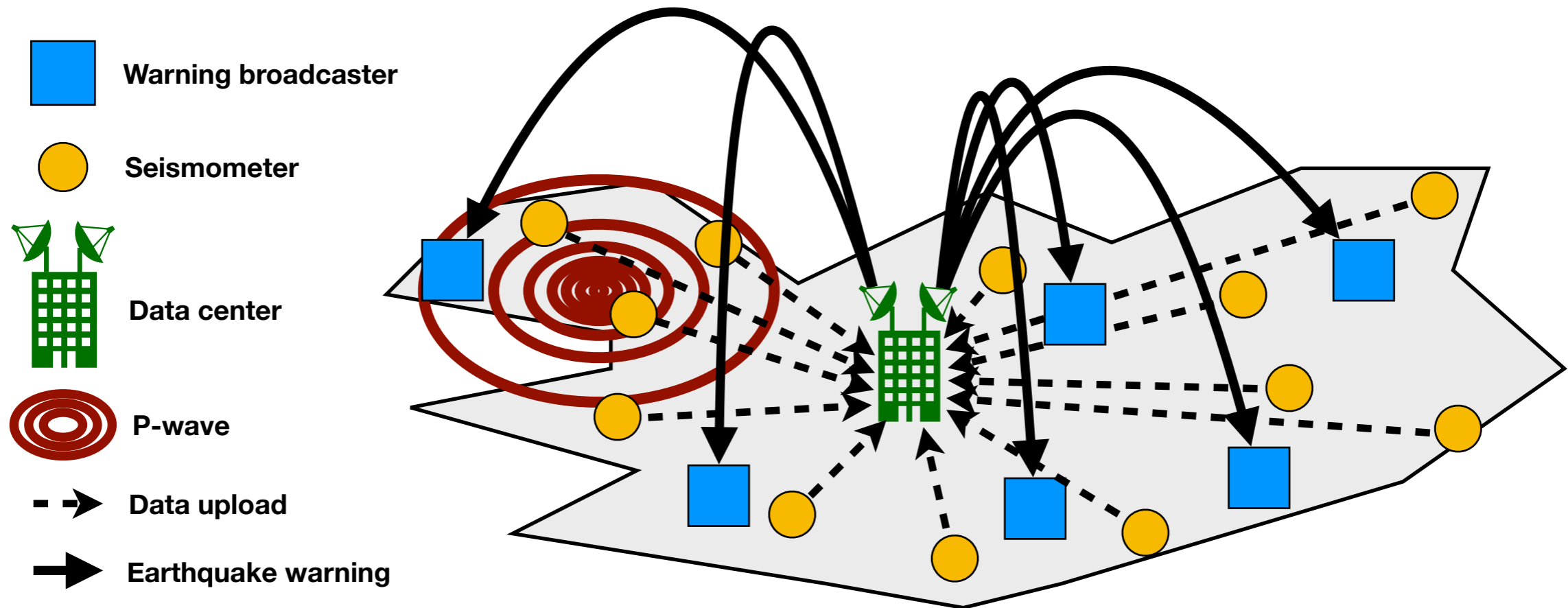


Earthquake Early Warning System (EEW)



- P-waves are faster than S-waves but much weaker;
- It is possible to characterize S-waves by analyzing P-waves;
- **Objective: process P-waves and characterize earthquakes before they start.**

Earthquake Early Warning System (EEW)

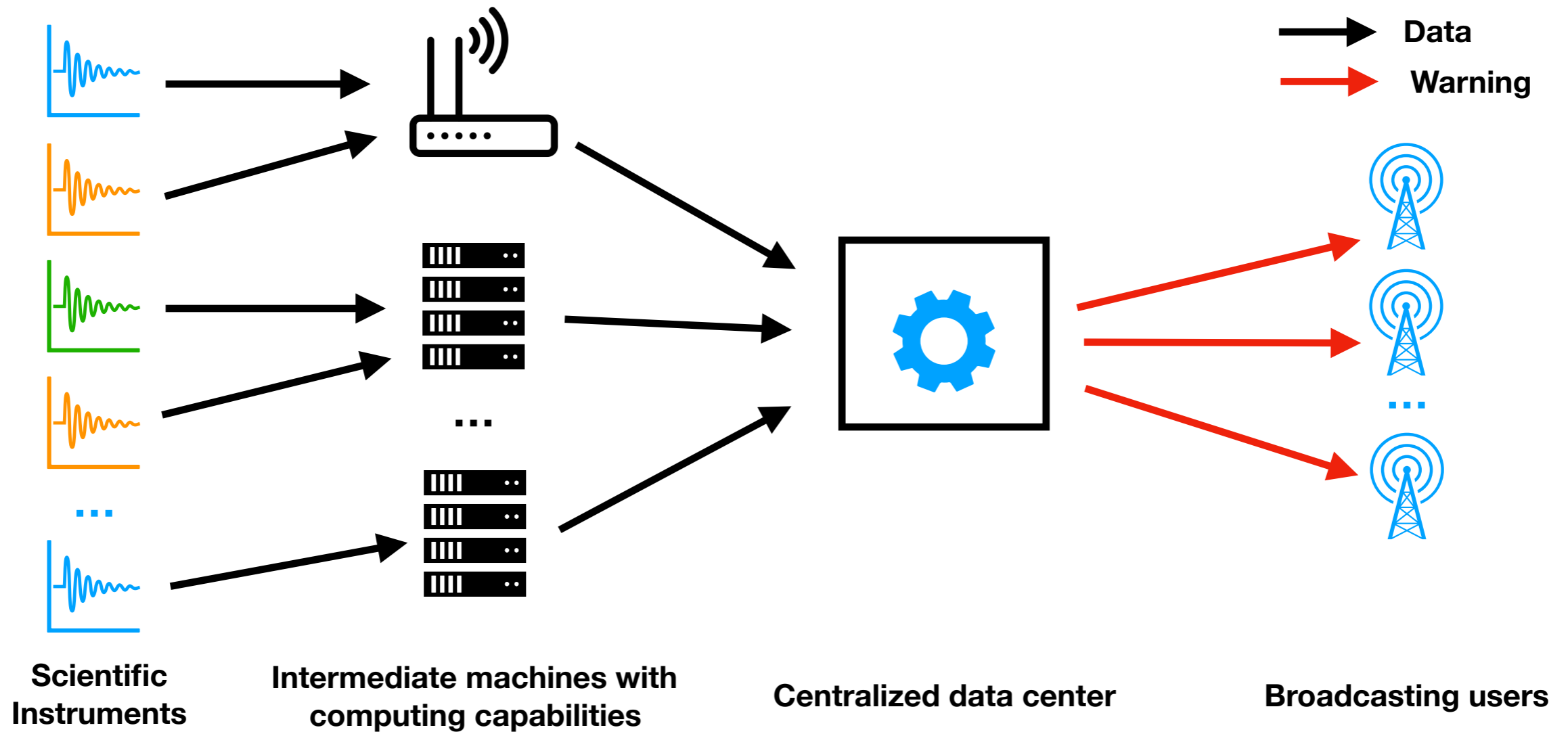


- Centralized data processing;
- Based on seismometers (JMA has more than 4k)

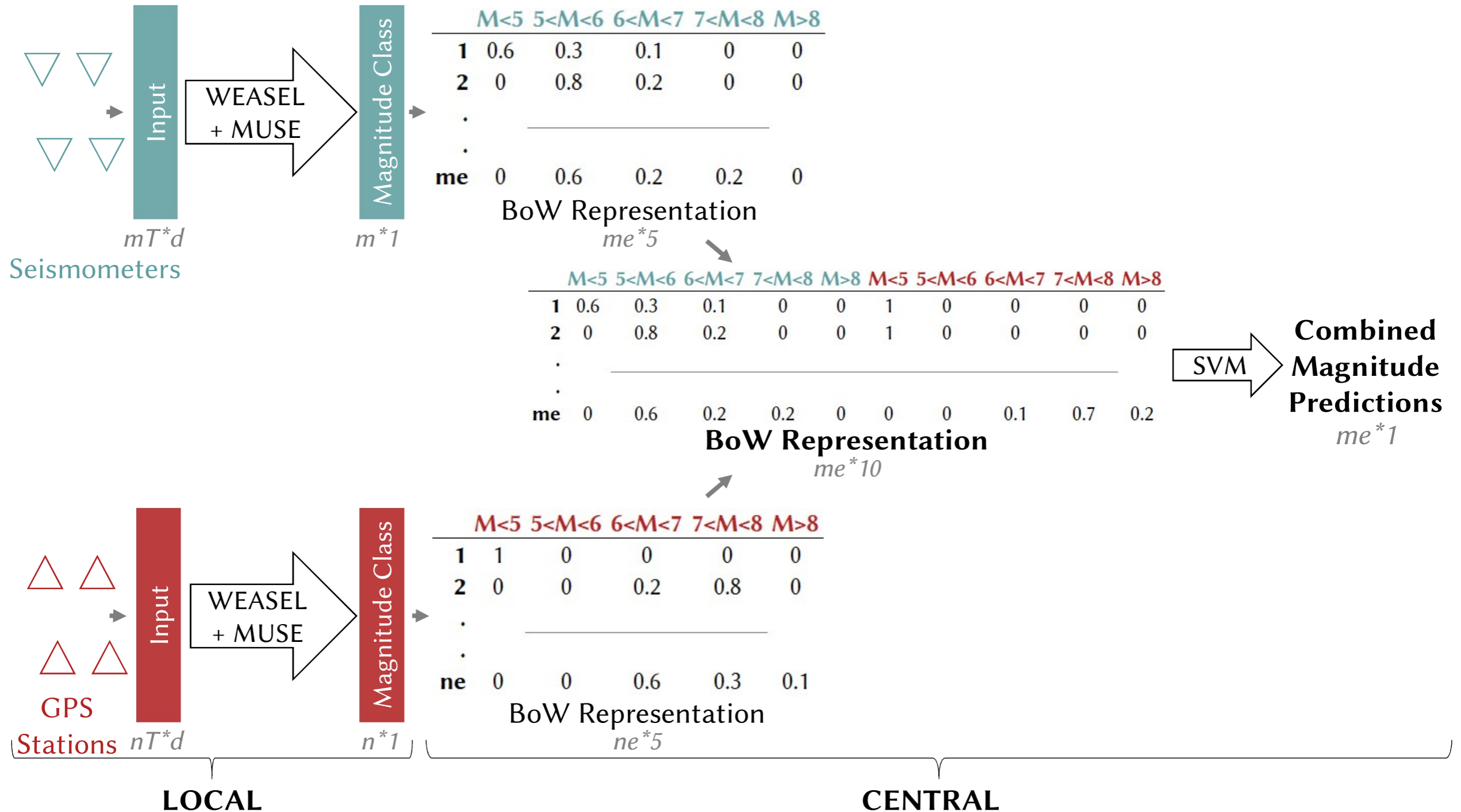
Current EEW issues:

- Sensor **homogeneity**:
 - GPS stations;
 - Submarine seismometers;
 - Seabed pressure detectors.
- Seismometer **“saturation”**:
 - Can't precisely identify $6 < M < 8$ earthquakes.
 - Very expensive.
- **Centralized processing** related issues:
 - 2011's M-9 earthquake in Japan.

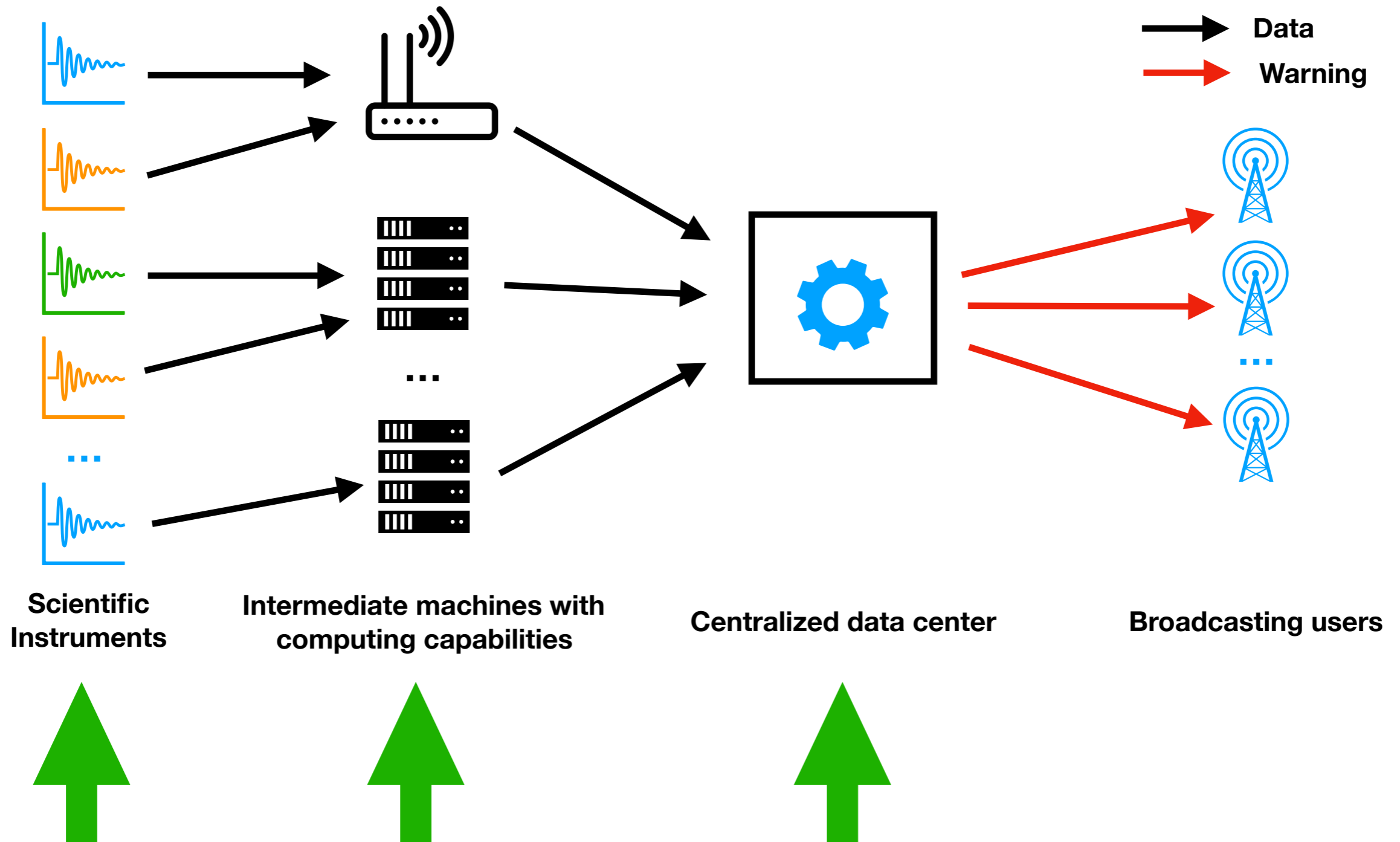
Edge-based Decentralized Infrastructure



Decentralized Early Earthquake Magnitude Algorithm (DEEM)



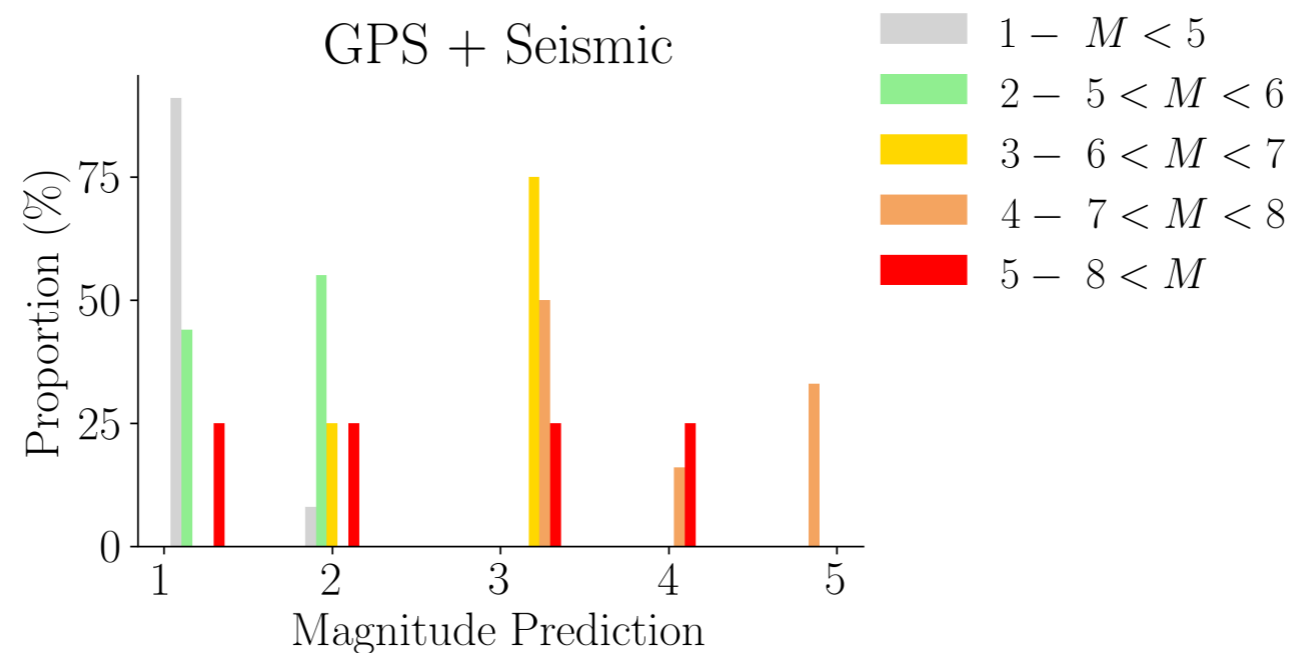
Edge-based Decentralized Infrastructure



DEEM: Performance

Paper submitted to
ECML PKDD 2019

- **moderate** F1-score: 64% vs. 56%
- **strong** F1-score: 88% vs. 37%
- **normal** activity F1-score: 85% vs. 86%

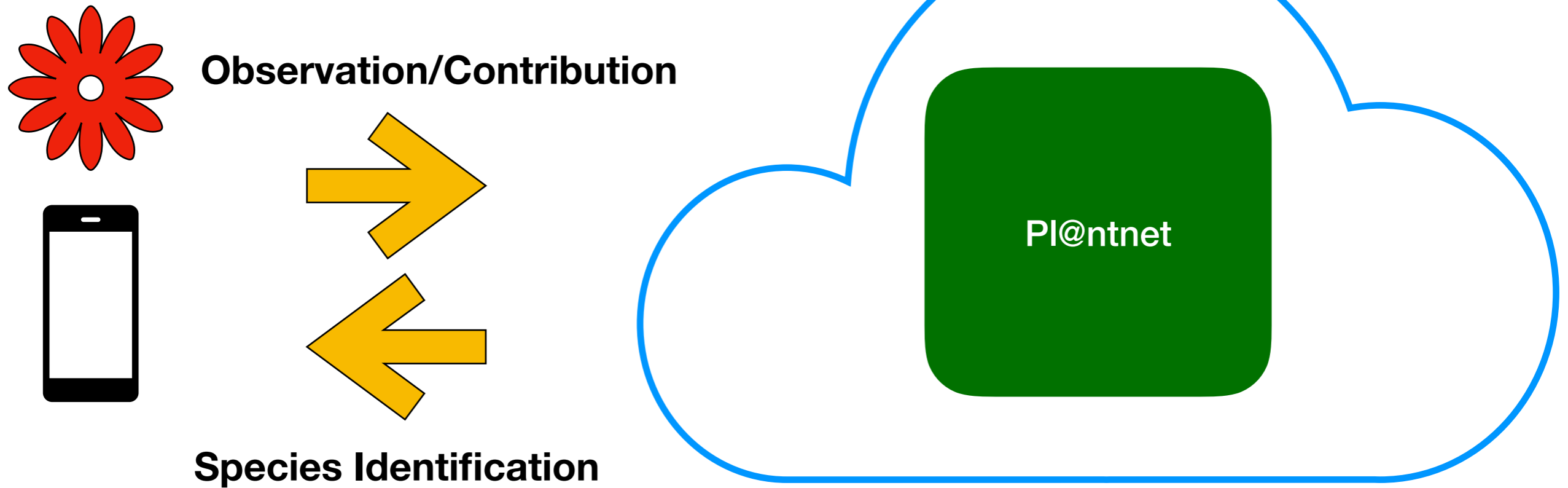


GPS stations and seismometers predictions vs. the ground truth.

EEW: Next Steps

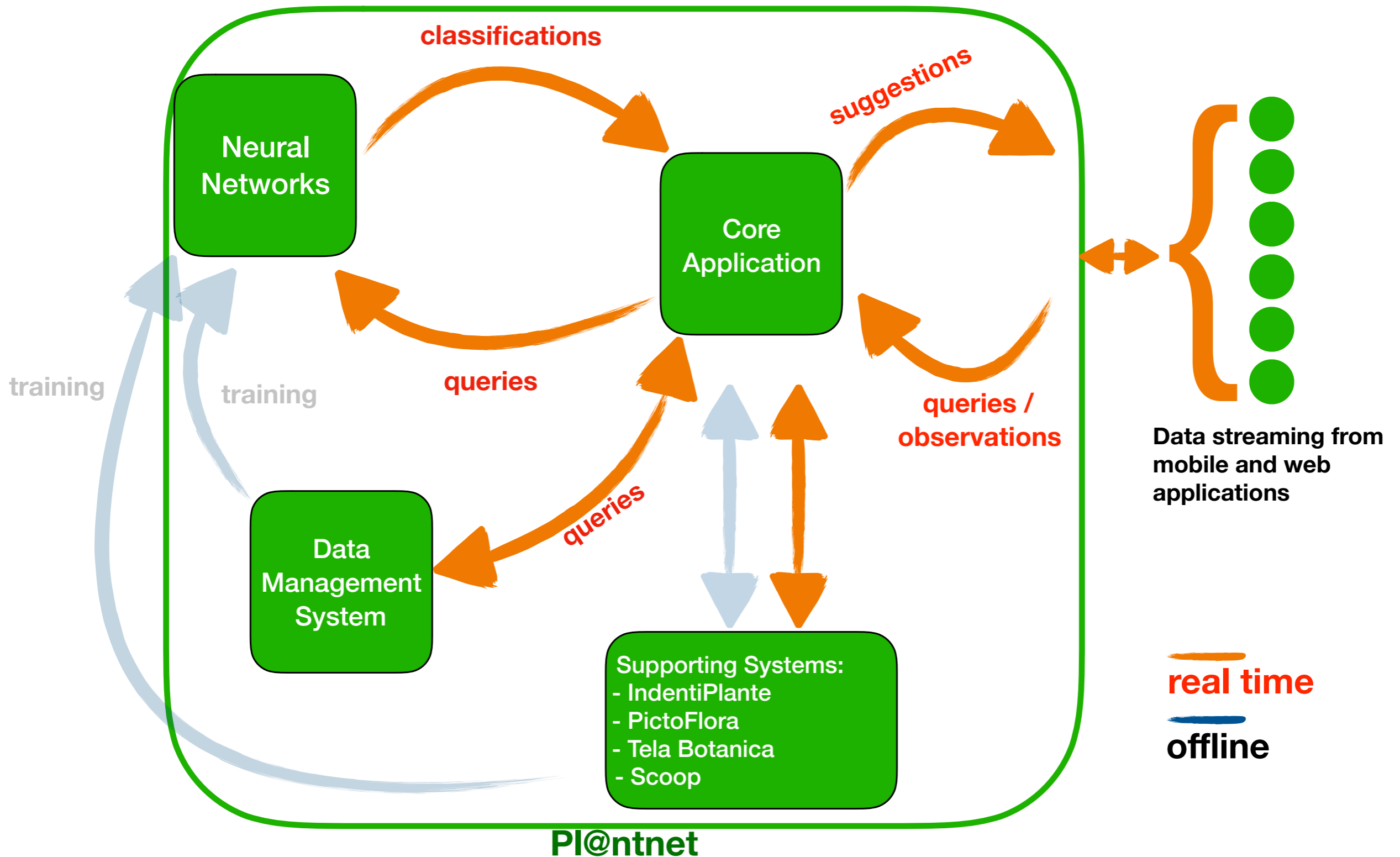
- Improving **DEEM's** accuracy (algorithm and training datasets)
- **Simulating** an infrastructure using Grid'5000;
- **Benchmarking** different Edge/Fog configurations;
- Earthquake **epicenter** identification;
- **Tsunami** characterization.

PI@ntnet



Collaboration between Inria teams' Kerdata and Zenith

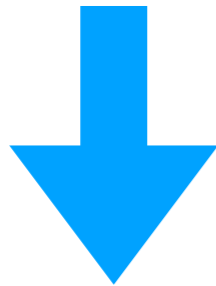
PI@ntnet: Big Data Aspects



PI@ntnet: Observation Query Performance

Query (200ms):

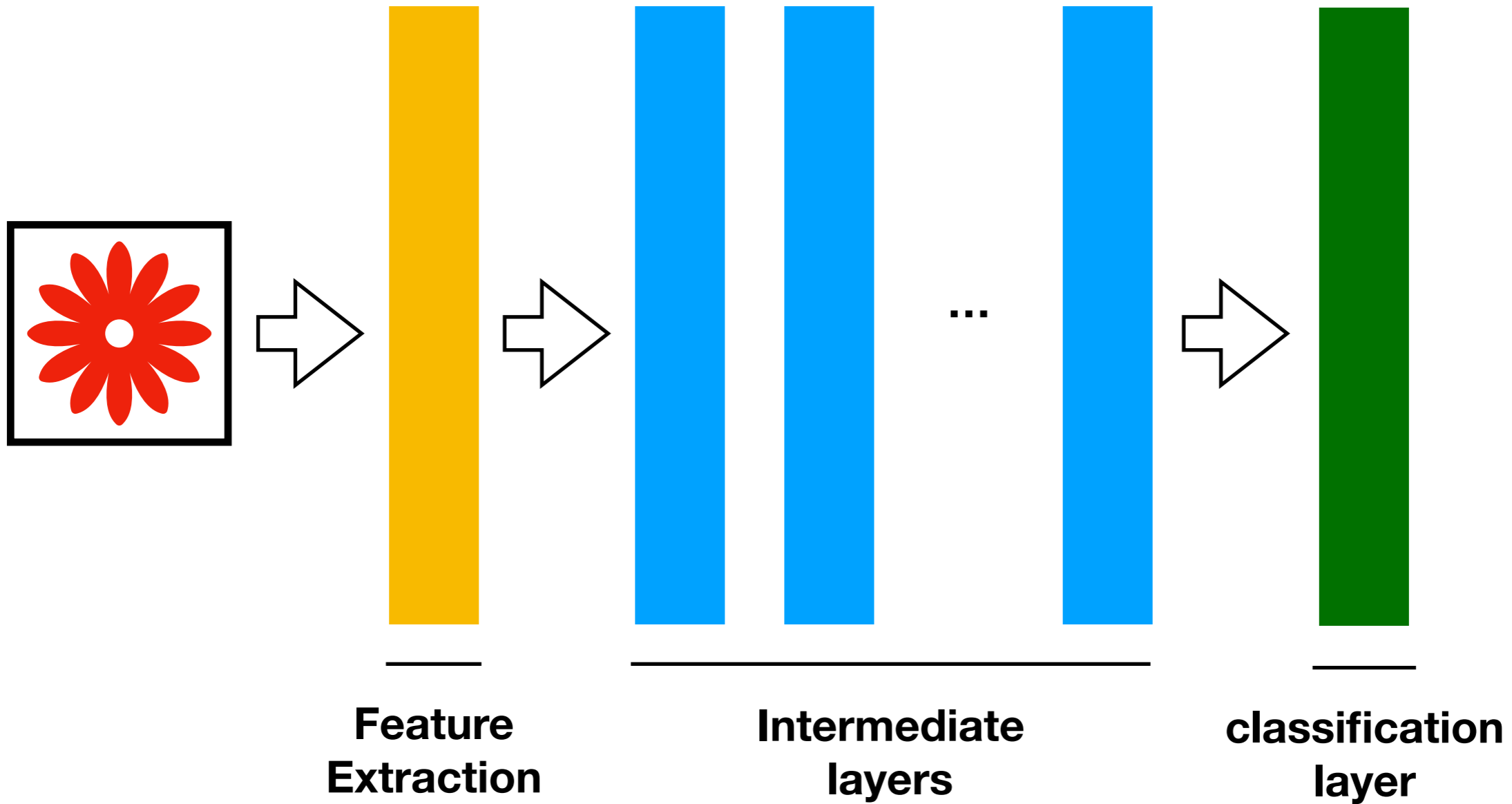
- Image reduction;
- Image upload: 80ms;
- Feature extraction: 90ms;
- Image classification: 1ms;
- Similarity search.



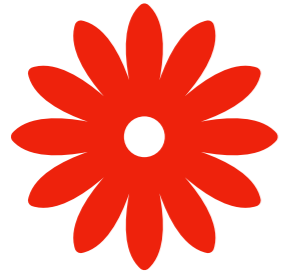
**Reduce Data Traffic &
improve performance of
Feature extraction**

Pl@ntnet: Neural Network Overview

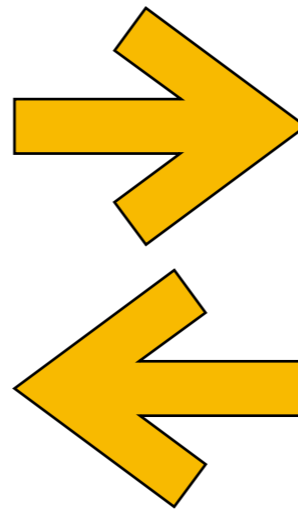
Layers



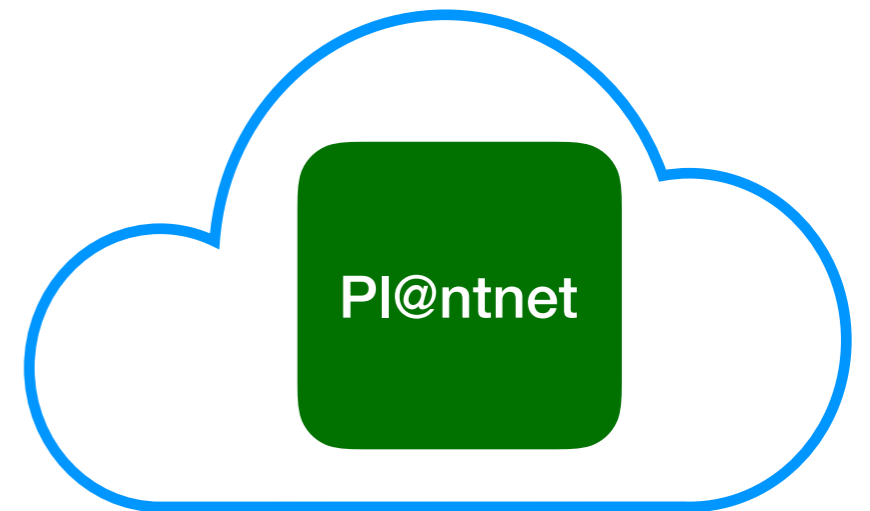
Pl@ntnet: Leveraging Edge processing



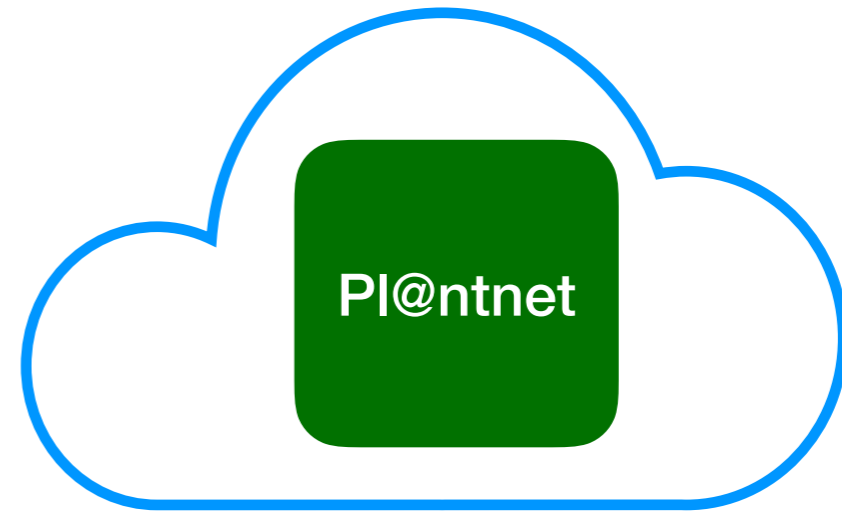
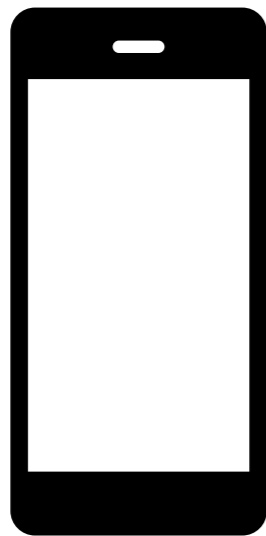
Observation/Contribution



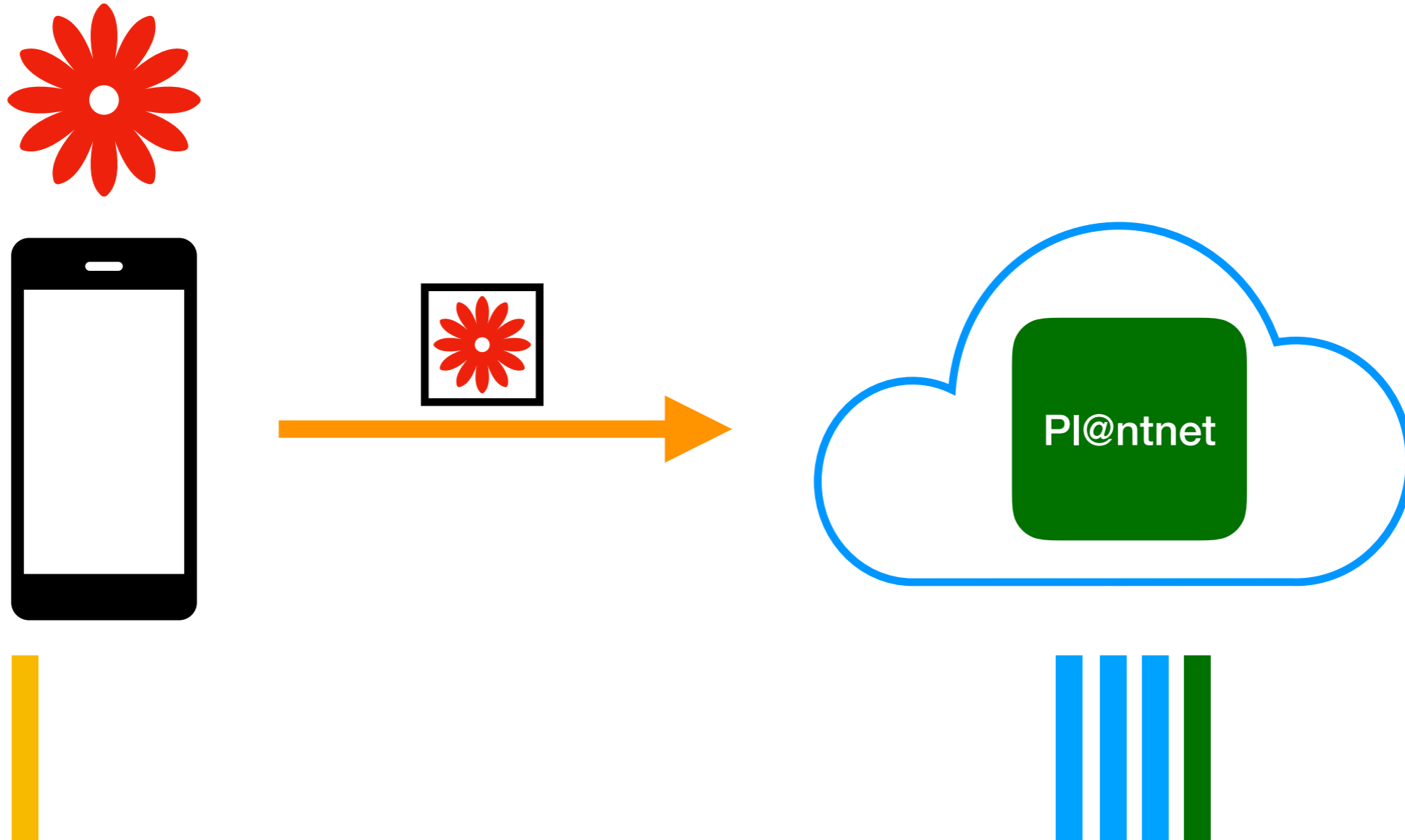
Species Identification



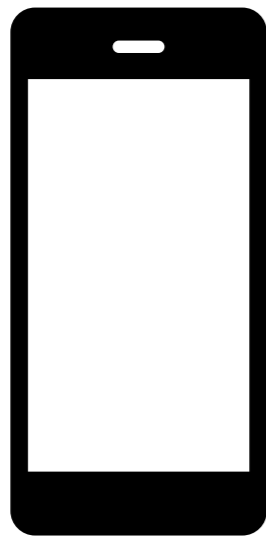
PI@ntnet: Feature Extraction @ Edge



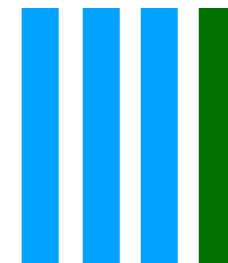
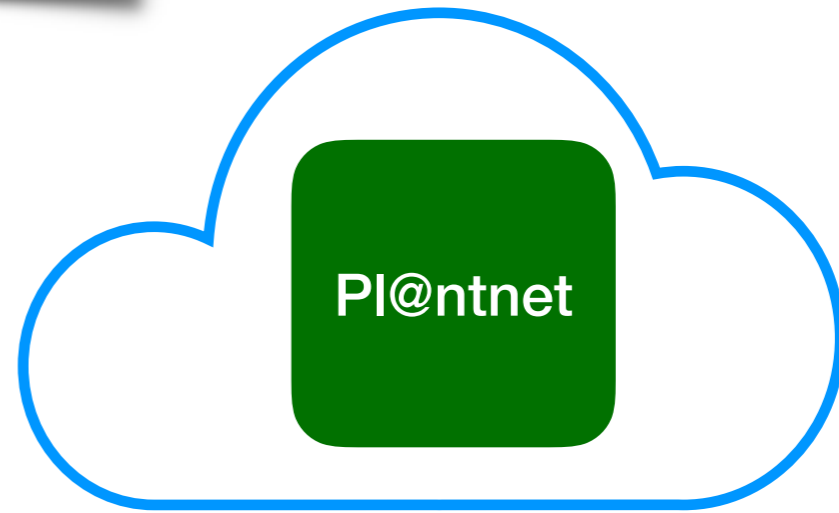
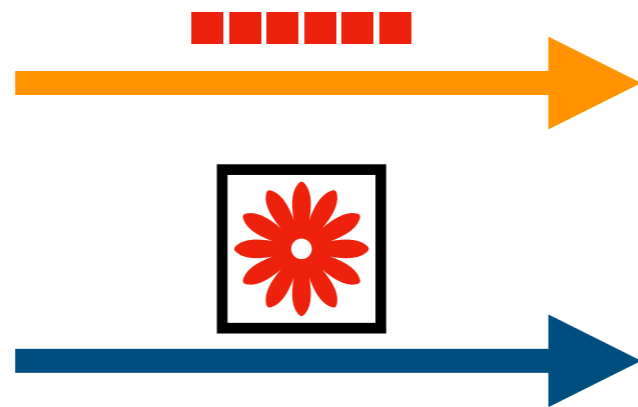
PI@ntnet: Feature Extraction @ Edge



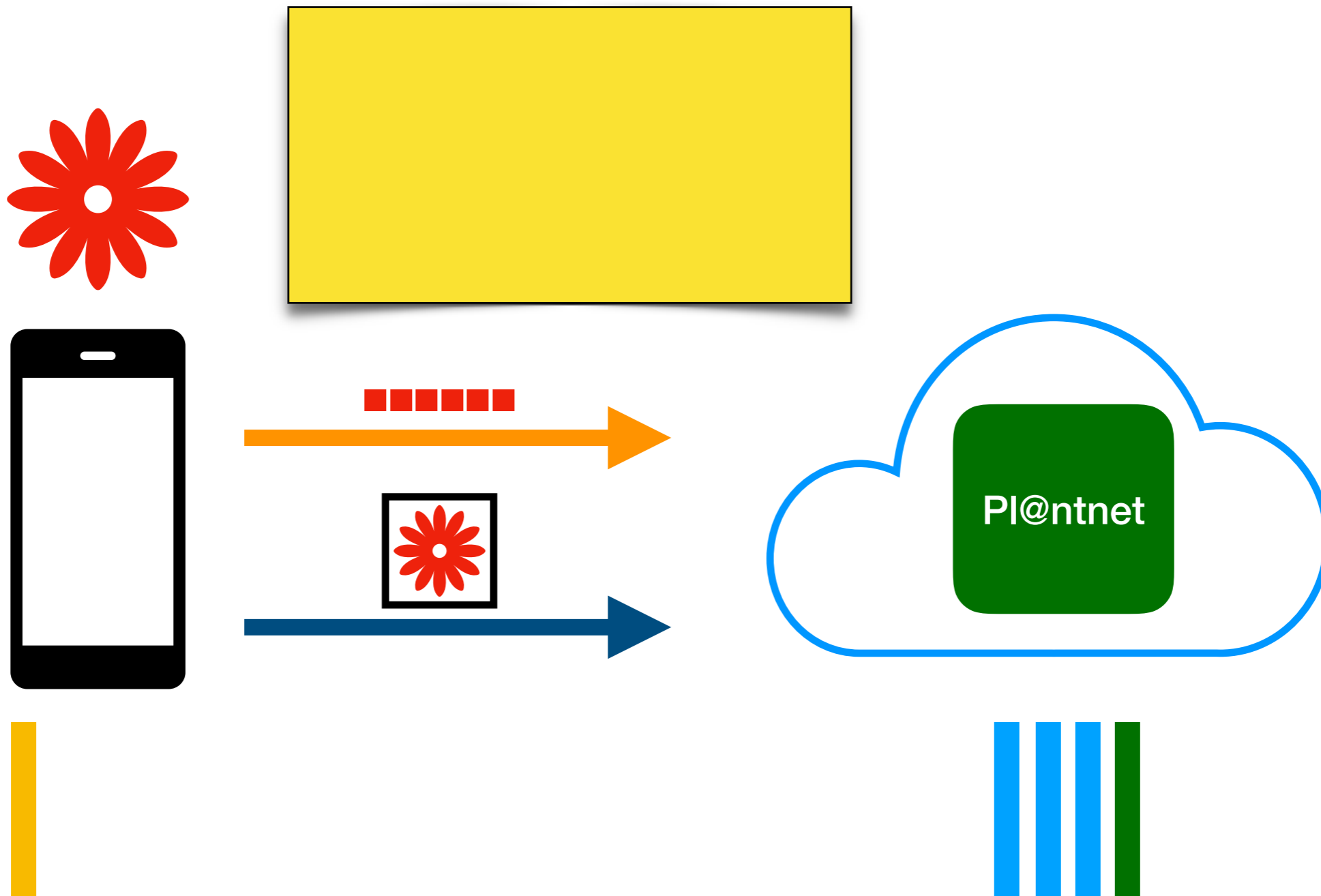
PI@ntnet: Feature Extraction @ Edge



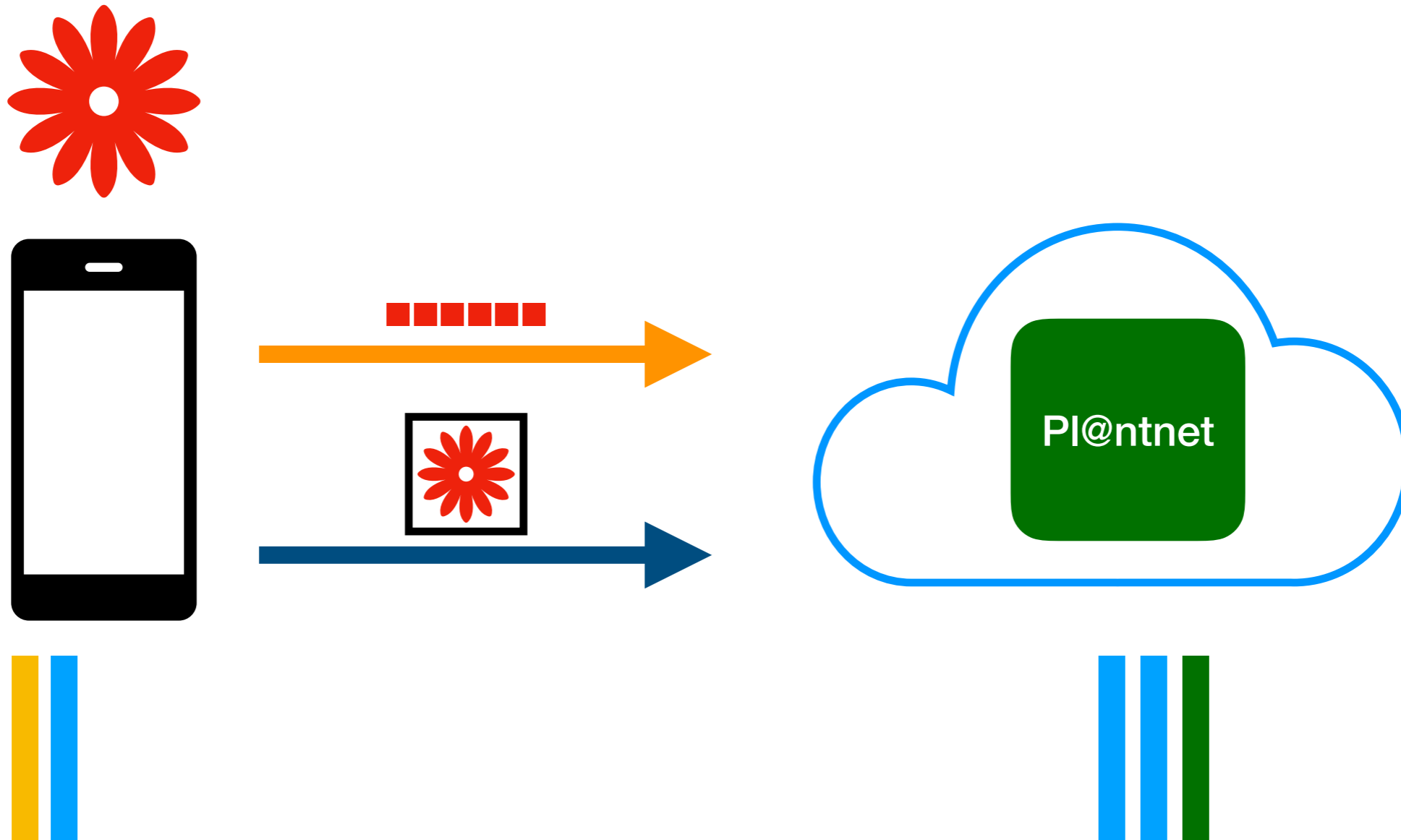
- Feature Vector Upload
- Batch Image Upload



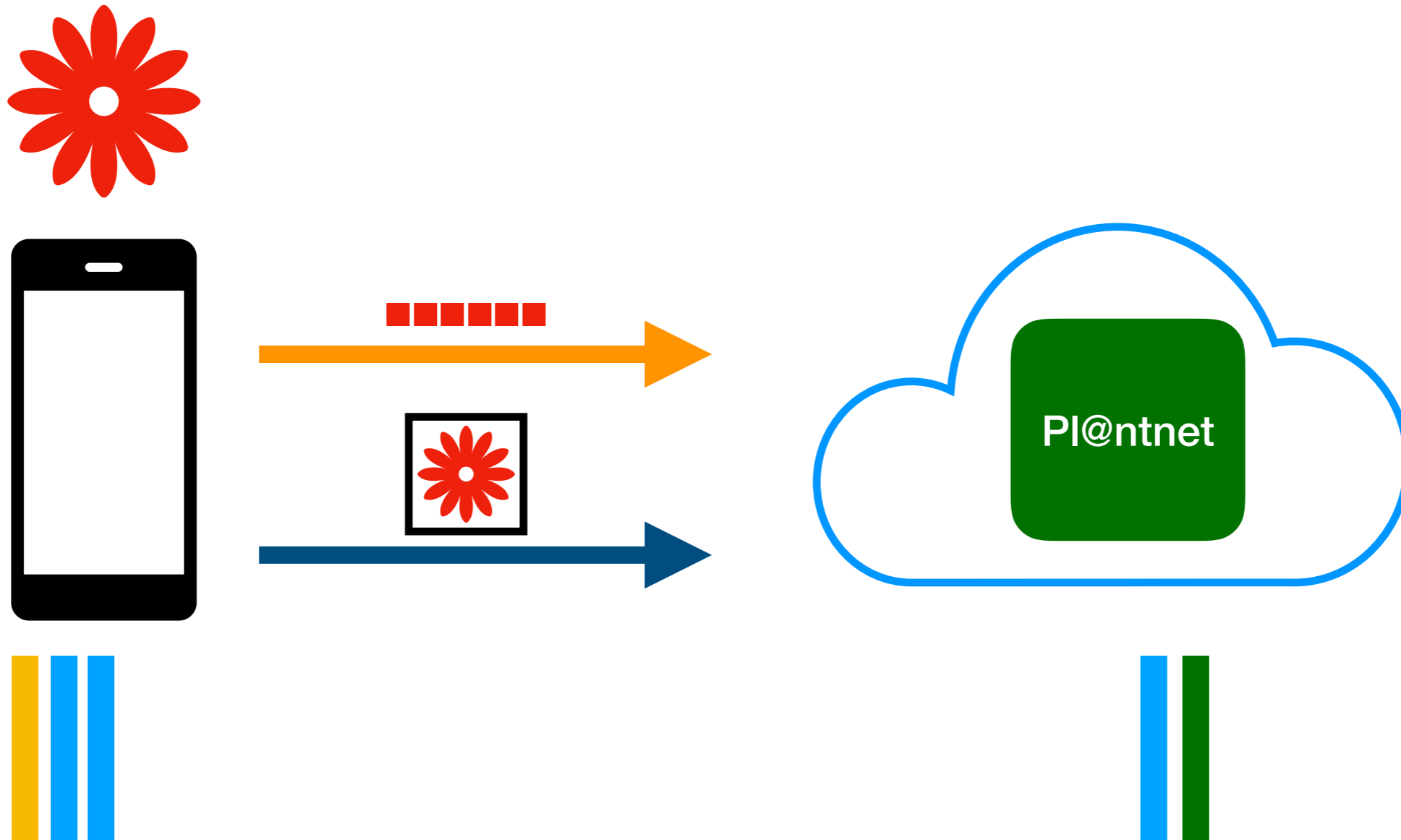
PI@ntnet: Feature Extraction @ Edge



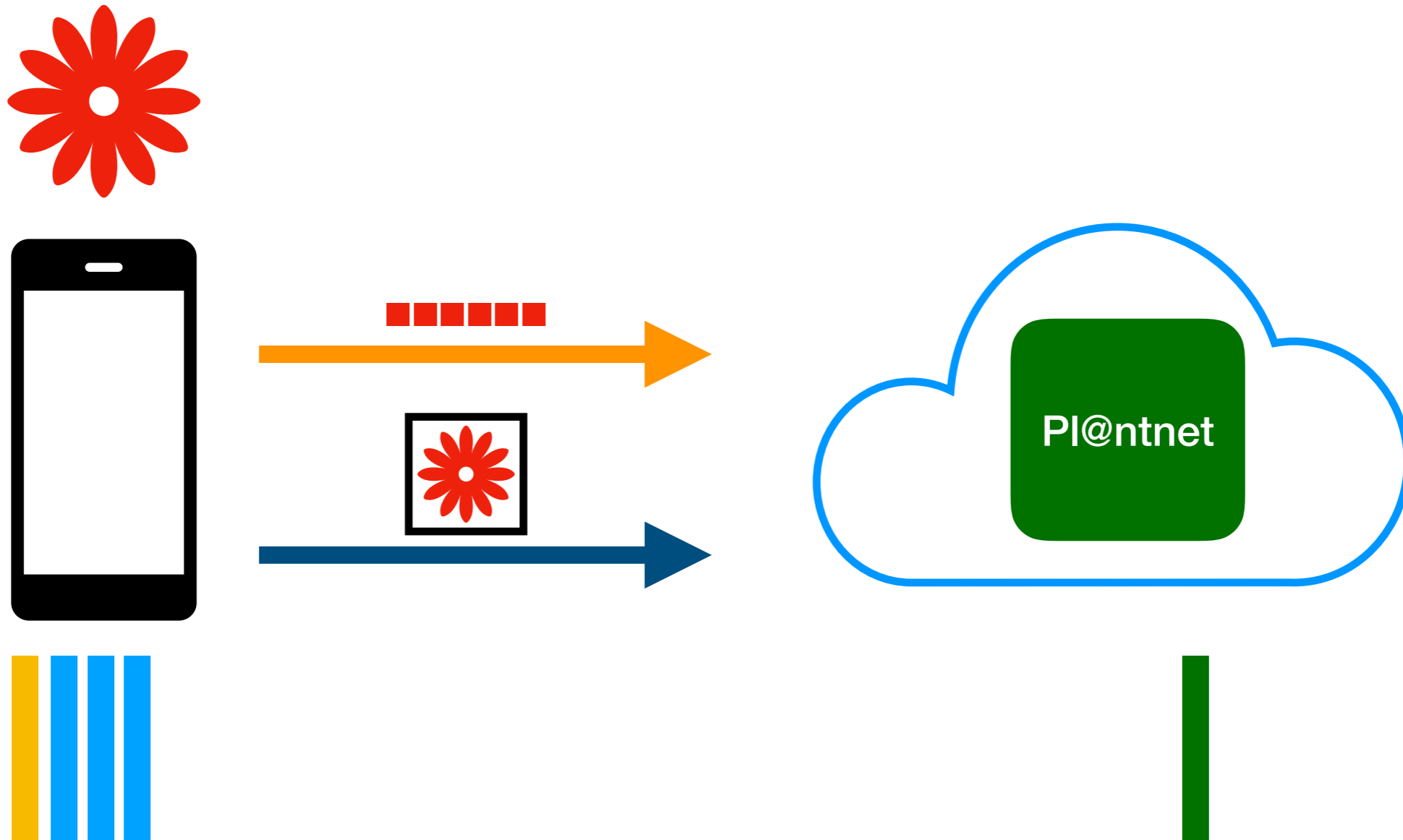
PI@ntnet: Feature Extraction @ Edge



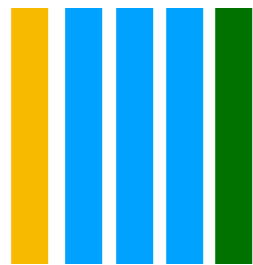
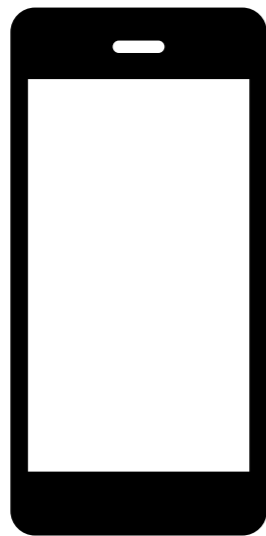
PI@ntnet: Feature Extraction @ Edge



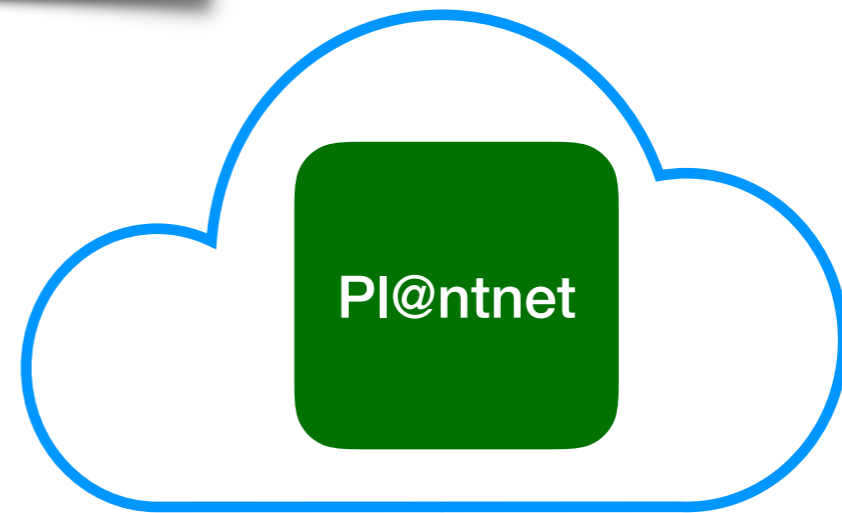
PI@ntnet: Feature Extraction @ Edge



PI@ntnet: Feature Extraction @ Edge



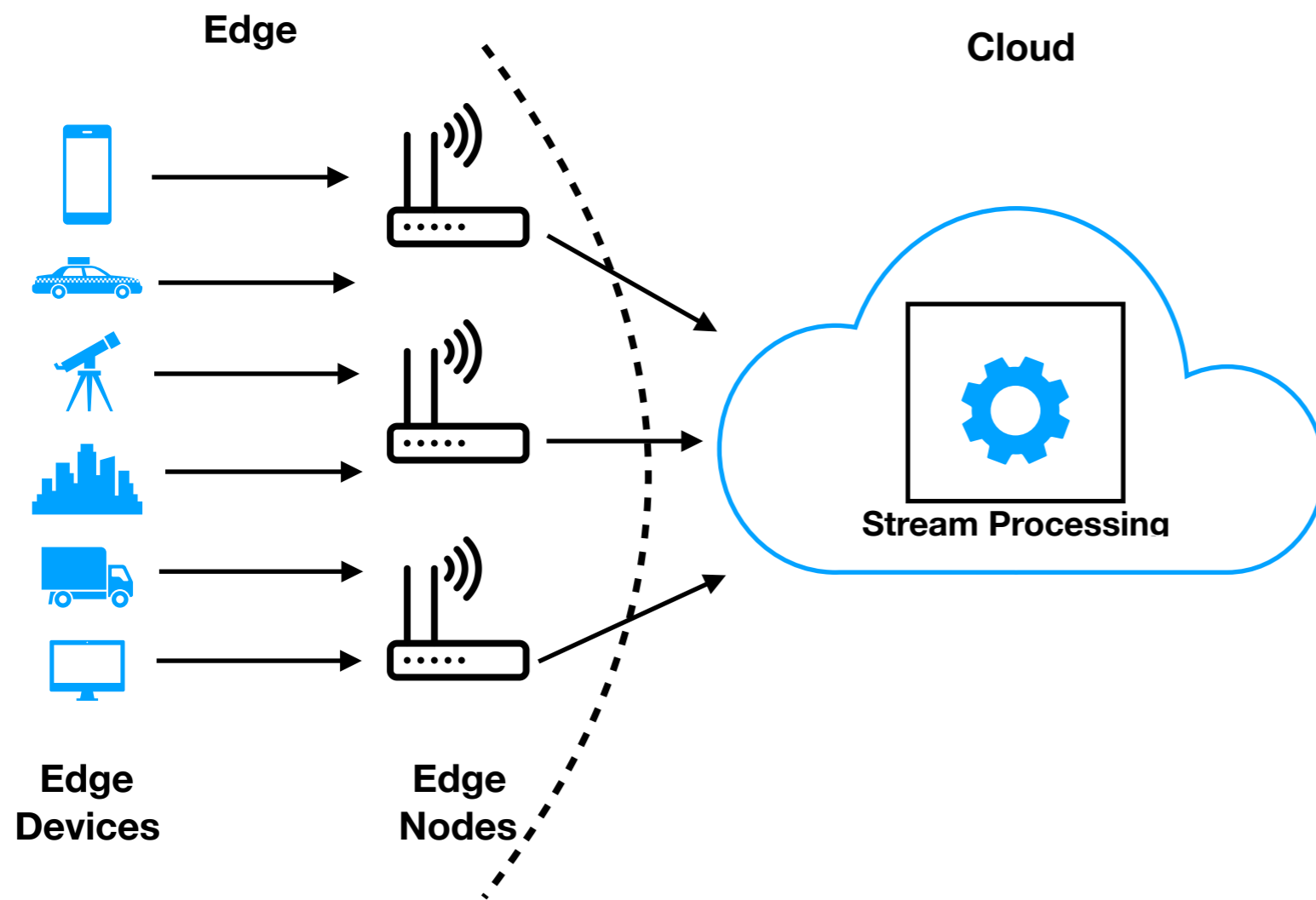
- Fake Image Classification
- Towards Offline Mode



PI@ntnet: Next Steps

- **Replicate** PI@ntnet scenarios on Grid5k in order to experiment our ideas:
 - **Feature extraction** on the Edge;
 - Effects of **NN distribution** on app performance;
 - Performance of **NN on the Edge**;

Thanks!



- Characterizing
- Benchmarking
- Placing applications

pedro.silva@inria.fr