

Investigating Job Allocation Policies in Edge Computing Platforms

Anderson Andrei DA SILVA

Advisor: Prof. Denis TRYSTRAM

Jury:

Prof. Martin HEUSSE

Prof. Christophe CÉRIN

Prof. Hubert GARAVEL

Summary

1. Introduction
2. An use case: The Qarnot Computing
3. Job Allocation
4. Batsim / SimGrid
5. Experiments
6. Analyses of Results
7. Conclusion
8. Further Remarks

Internet of Things, Cloud and Edge Computing

W. Shi, et al [1,2]:

- We will arrive in the post-cloud era, where there will be:
 - **a large quality of data generated by things** that are immersed in our daily life,
 - a lot of applications will also **be deployed at the edge** to consume these data.
- By 2019:
 - Data produced by people, machines, and things will **reach 500 zettabytes**, as estimated by Cisco Global Cloud Index,
 - However, the global data center **IP traffic will only reach 10.4 zettabytes** by that time.
 - **45% of IoT-created data** will be stored, processed, analyzed, and acted upon **close to, or at the edge of, the network**.

Y. Mao, et al. [3] :

- **Mobile devices tends to growth in terms of usability and processing of data, implicating the decentralization from the Cloud's presence.**

Luiz Bittencourt, et al. [4] :

- As a combination of
 - Run small, localized **applications at the edge**
 - **High-capacity from the cloud,**
 - **Support heterogeneous requirements**
 - **Resources from the edge of the network** and from the Cloud.

Internet of Things, Cloud and Edge Computing

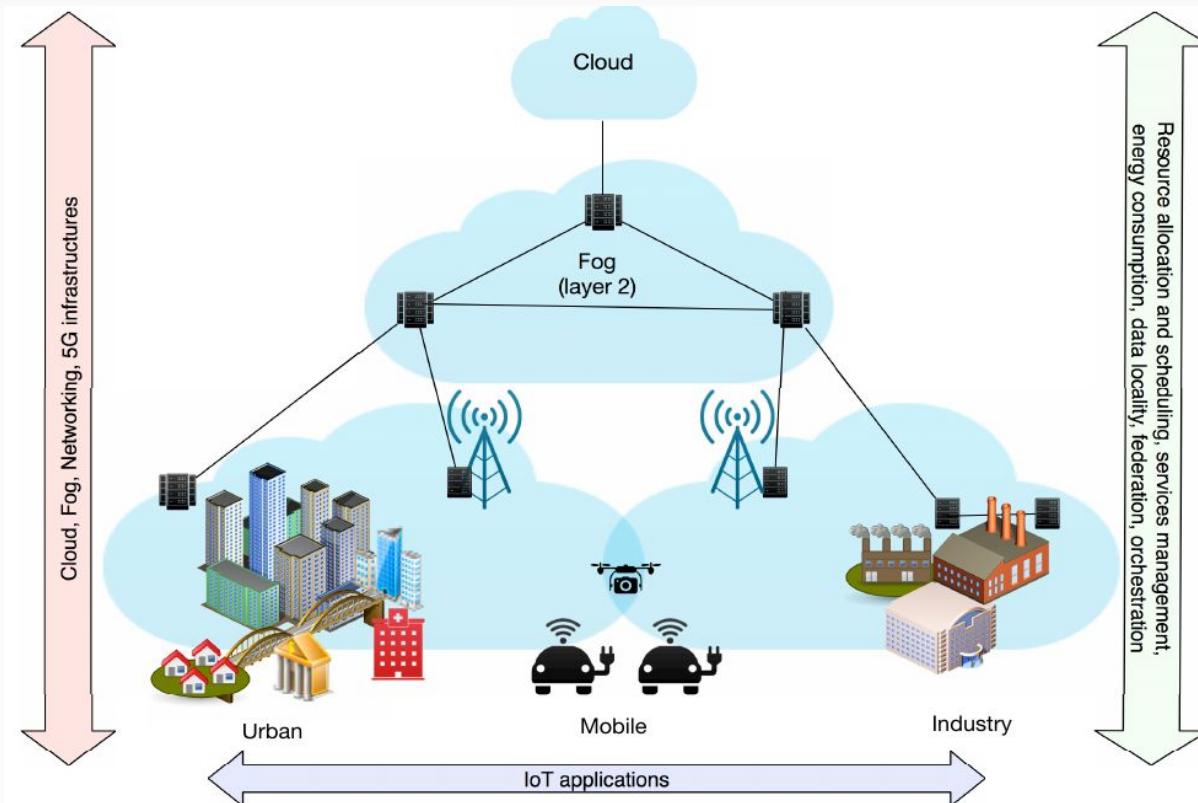


Figure: Illustrative overview, within the IoT-Fog-Cloud infrastructure [1]

Use case: The Qarnot Computing

Incorporated in 2010, the **Qarnot Computing used IT waste heat in a viable heating solution** for buildings with a distributed infrastructure in housing buildings, offices and warehouses across several geographical areas in France and Europe.

The whole platform is composed of about:

- **1,000 computing devices hosting**
- **3,000 diskless machines.**

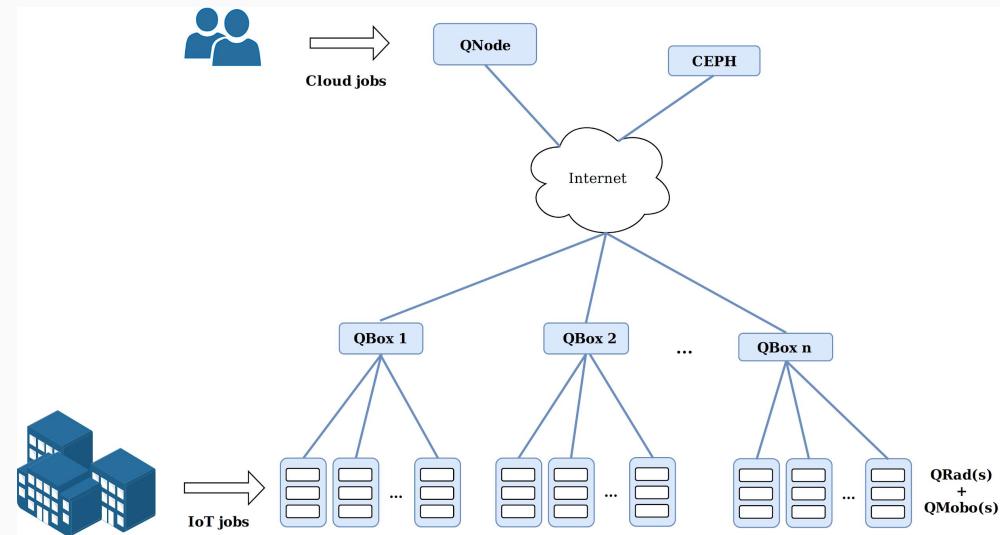


Figure: <https://www.qarnot.com/>

Use case: The Qarnot Computing



Infrastructure: QWare



Job Allocation: Importance and Impacts

How to manage jobs and resources, in order to fit the jobs among the resources in the best way.

- S. M. Parikh [6] points that the **management of flexible resources allocation is a problem emerged in the context of Cloud/ Edge Computing, due to heterogeneity** in hardware capabilities, workload estimation and a variety of services, also as the maximization of the profit for cloud providers and the minimization of cost for cloud consumers.
- Lu Huang et.al [7] affirm that **to make appropriate decisions** when allocating hardware resources to the tasks and dispatching the computing tasks to resource pool has become **the main issue in cloud computing**.
- According to Hameed Hussain et.al [8] the resource management mechanism **determines the efficiency of the used resources and guarantees the Quality of Service (QoS)** provided to the users.

Job Allocation: Implemented Scheduling Policies in an Use Case

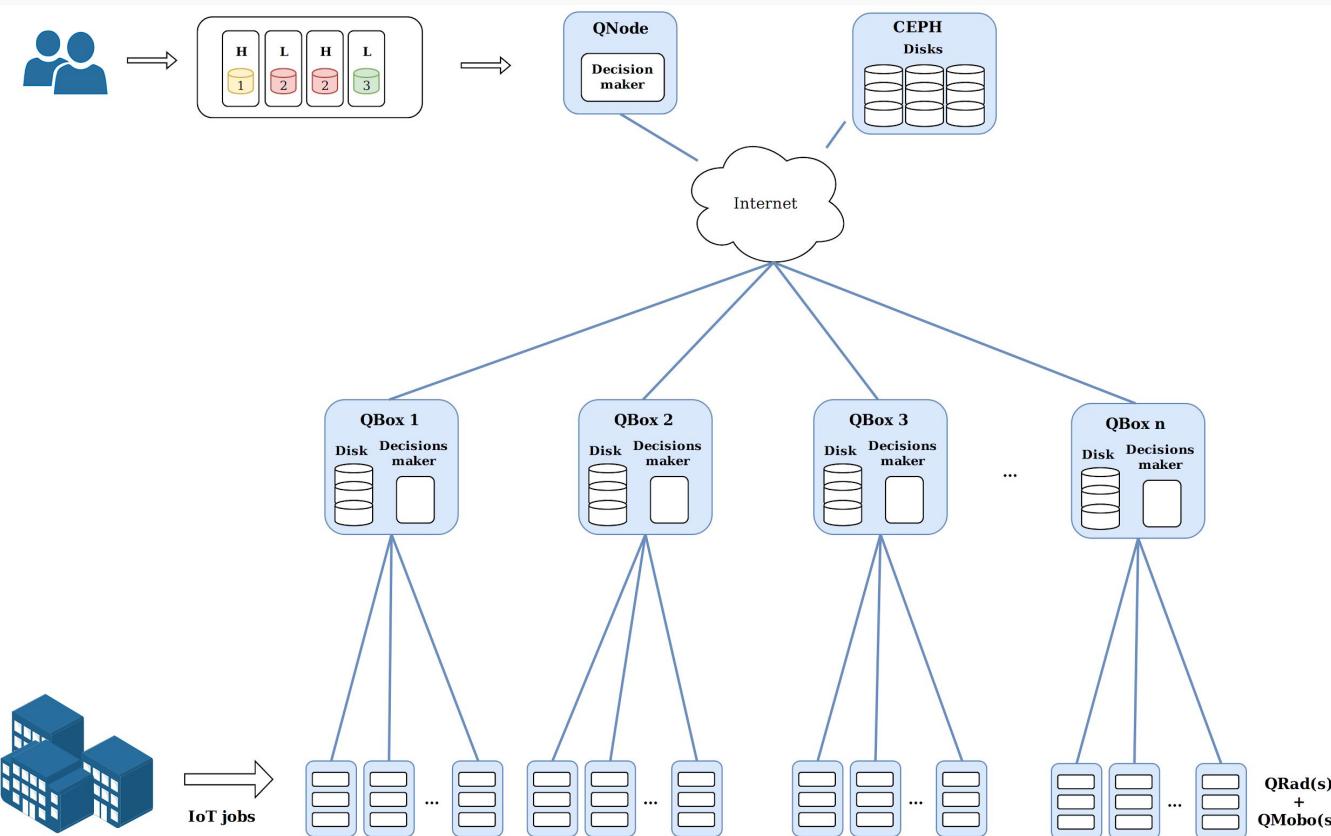
Policies implemented and compared:

- Standard (current Qarnot policy)
- Locality Based
- Full Replicate
- 3 Replicate
- 10 Replicate

Job's detail:

- Priorities: Background, Low, High
- Data sets dependencies

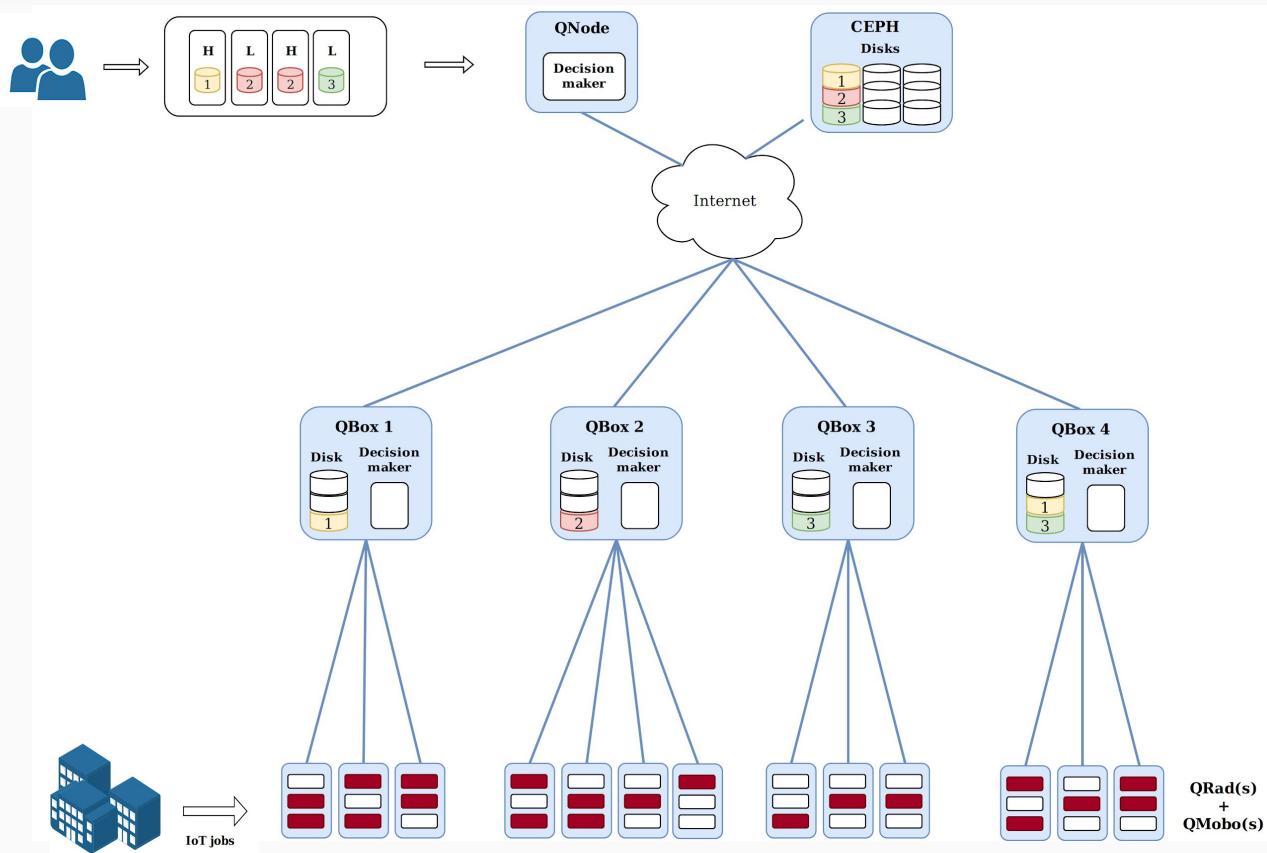
Qarnot Infrastructure: The QWare in Details



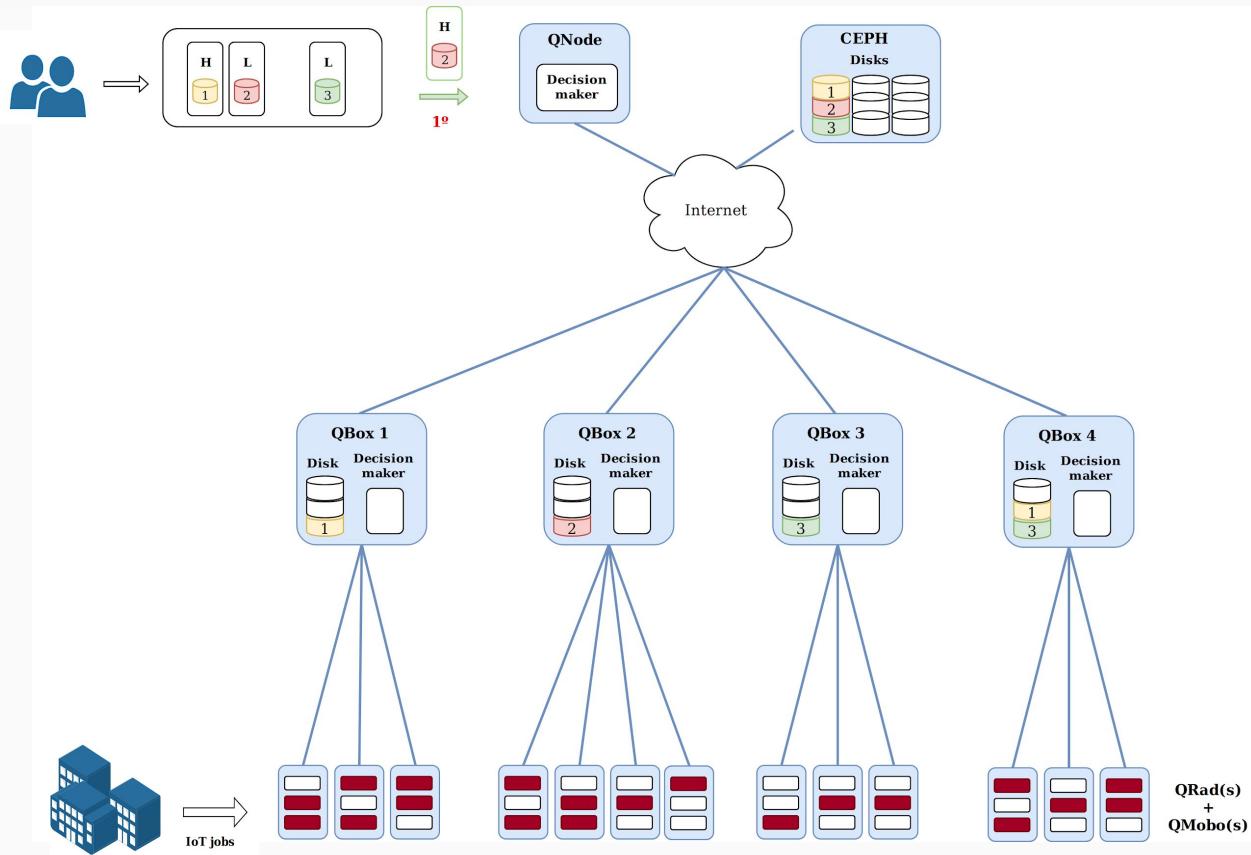
Standard

- Current Qarnot's policy.
- It dispatches instances, ordered by their priorities, to the QRads that need more heating.

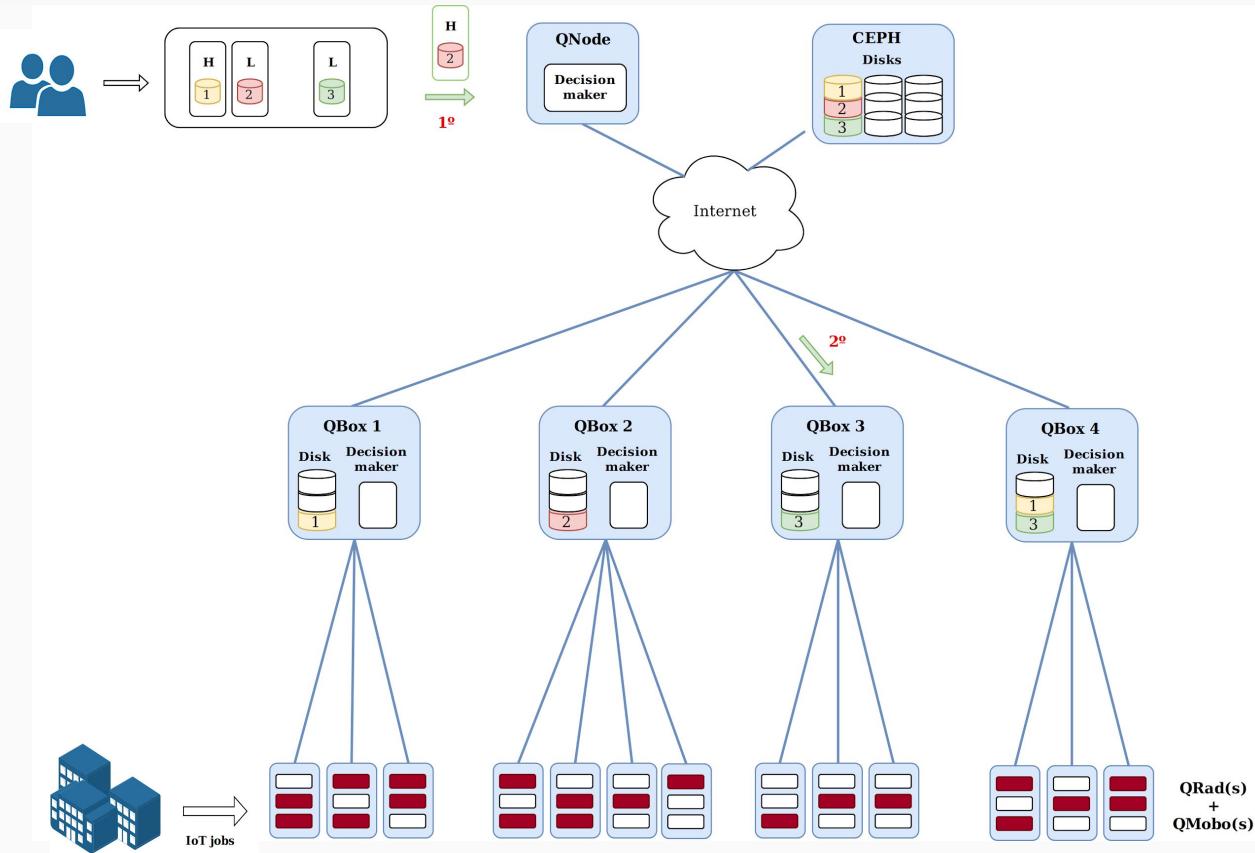
Job Allocation Policies - Standard



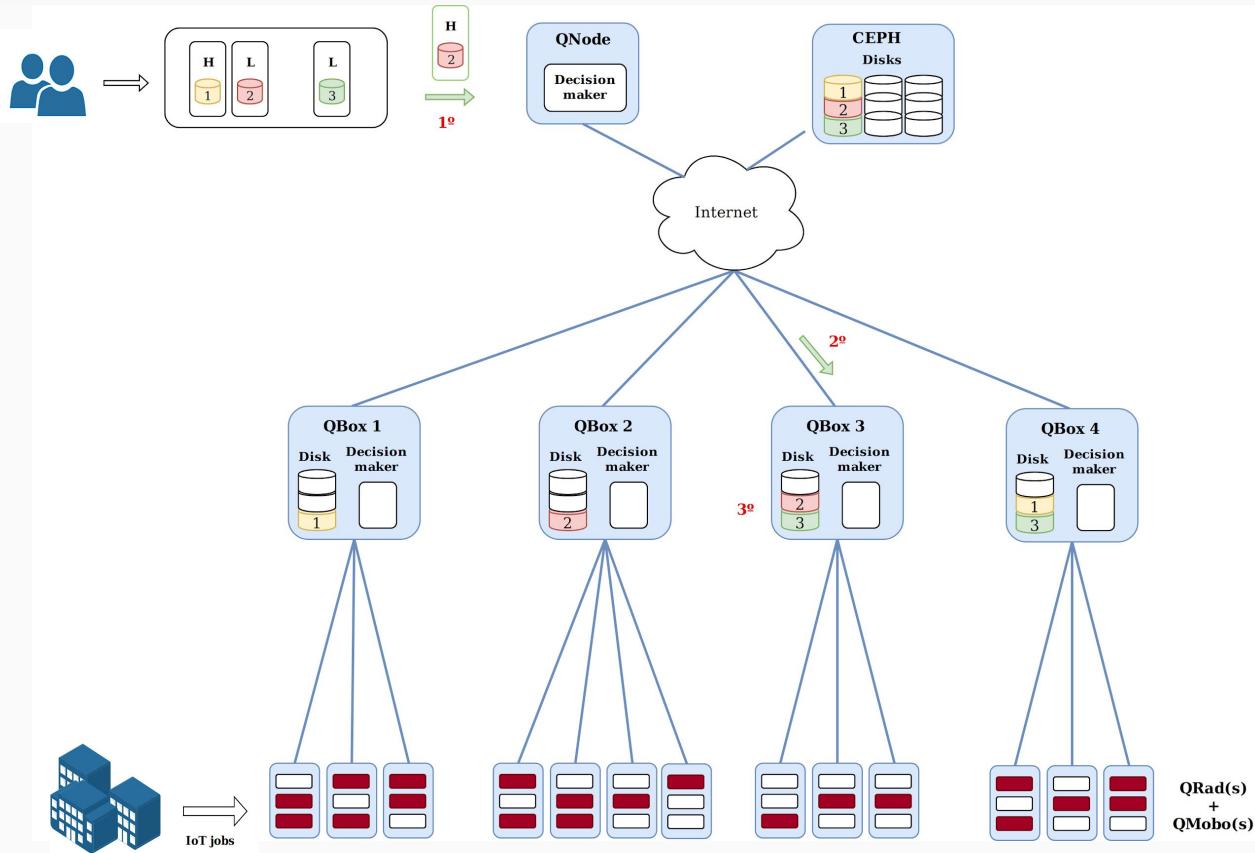
Job Allocation Policies - Standard



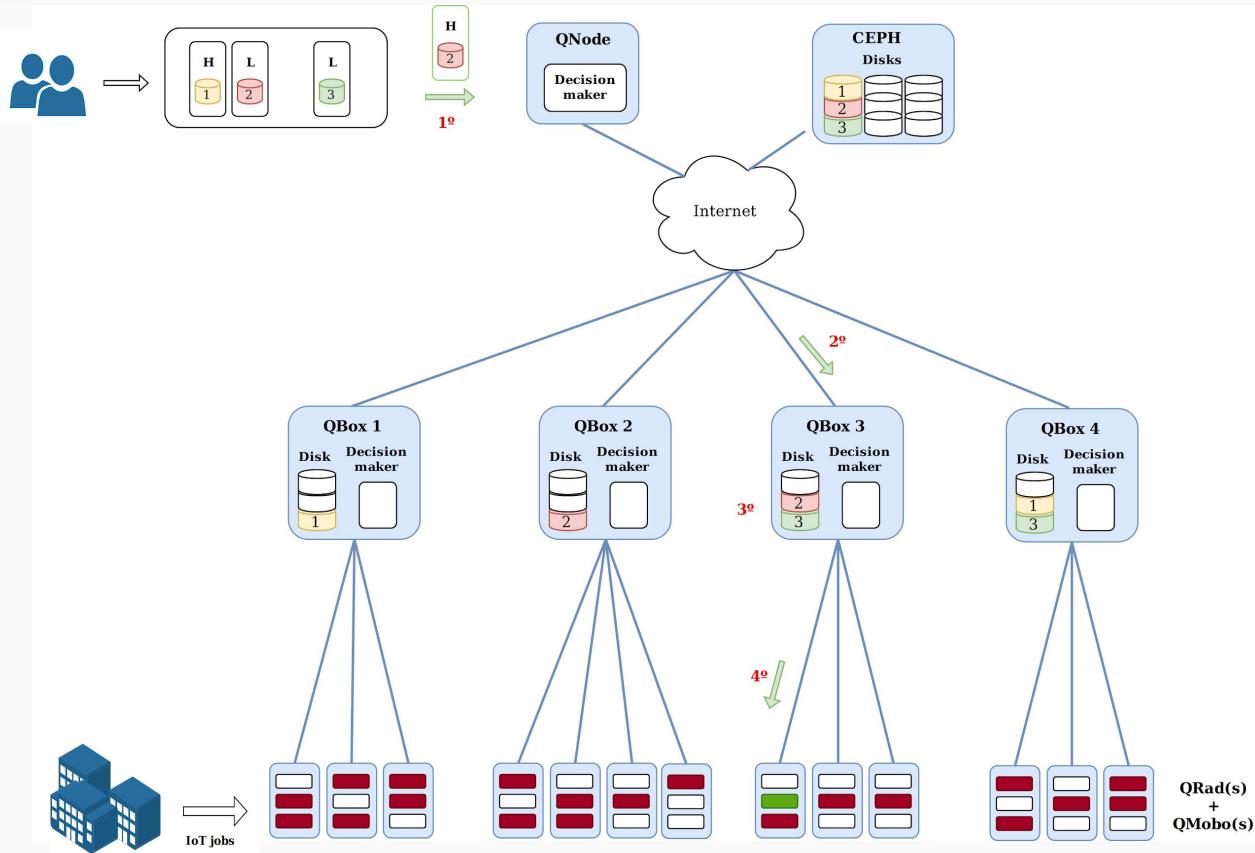
Job Allocation Policies - Standard



Job Allocation Policies - Standard



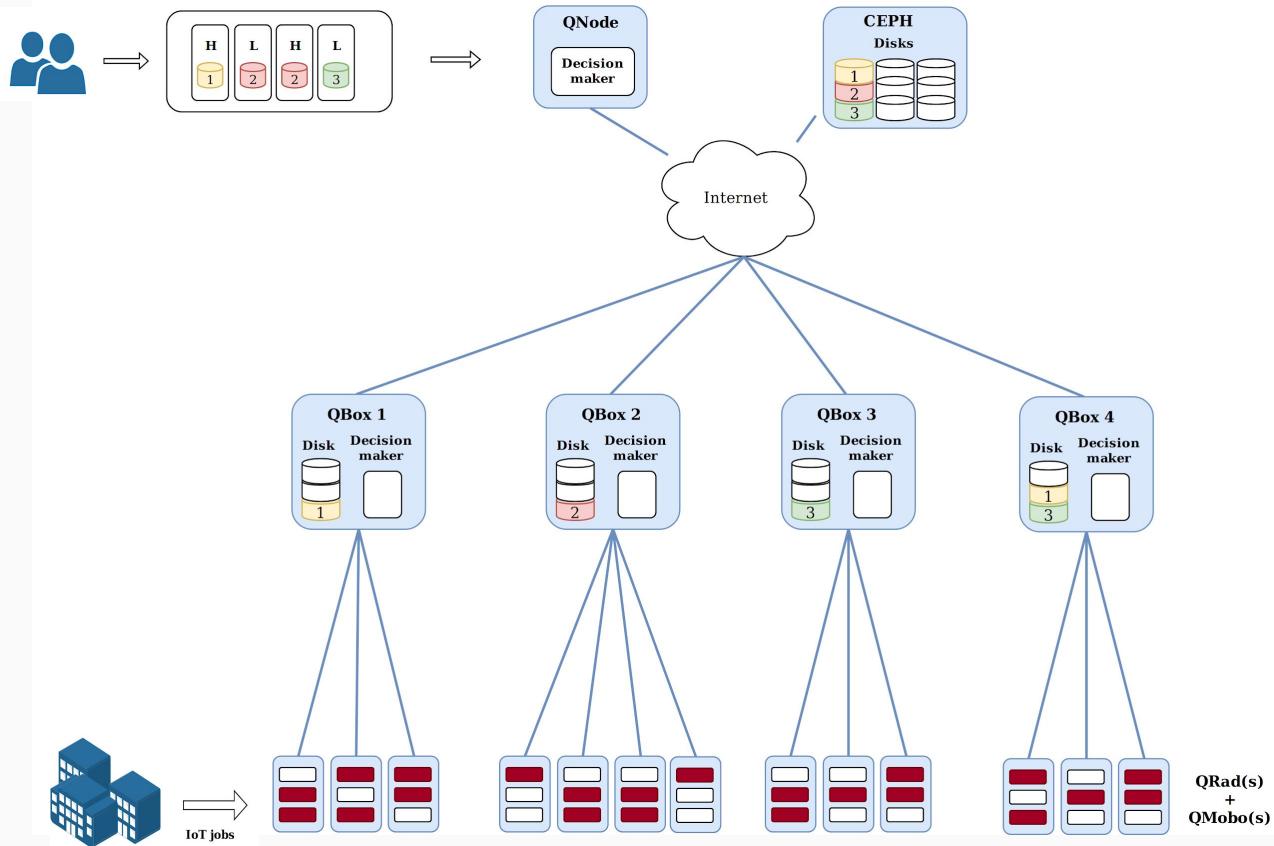
Job Allocation Policies - Standard



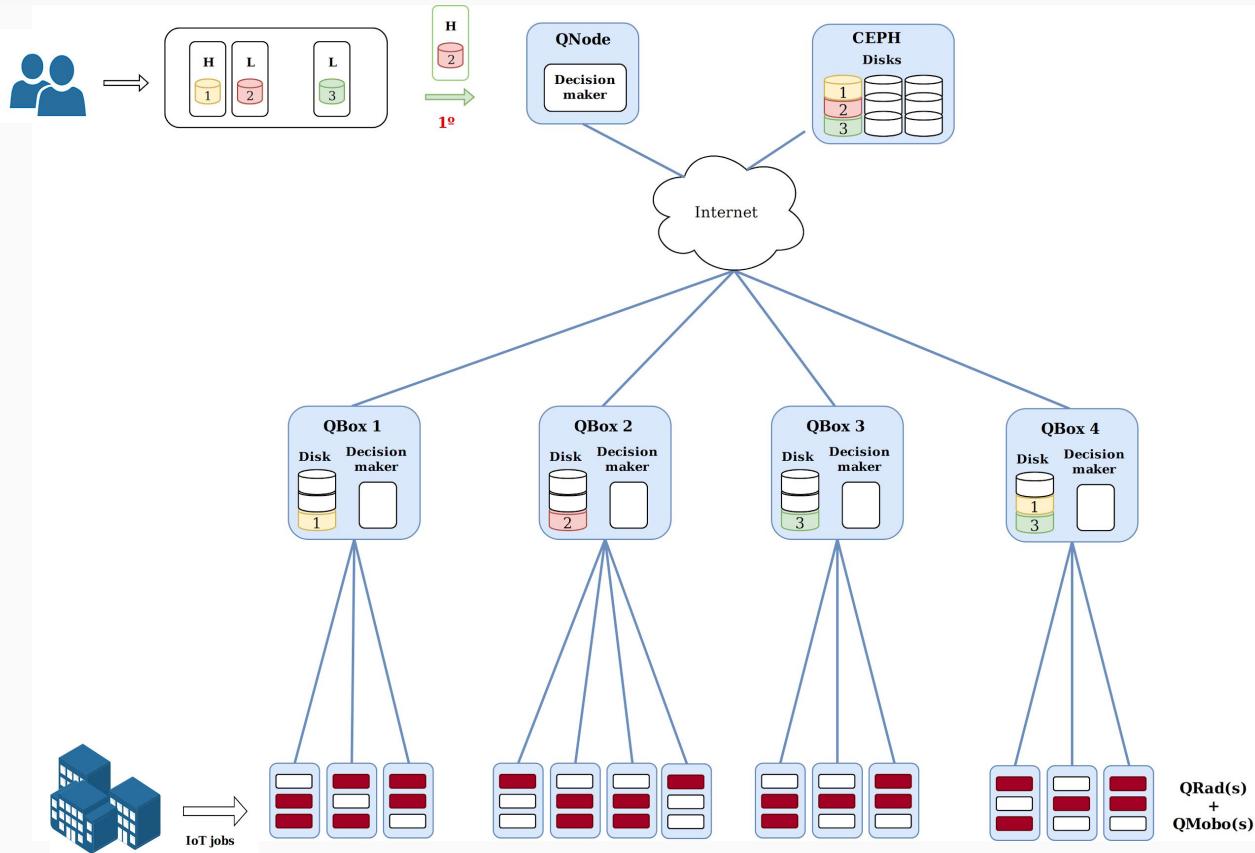
Locality Based

- Based on Standard policy
- It dispatches instances, ordered by their priorities, to the QRads that need more heating, **by prioritizing the ones that already have the required data set.**

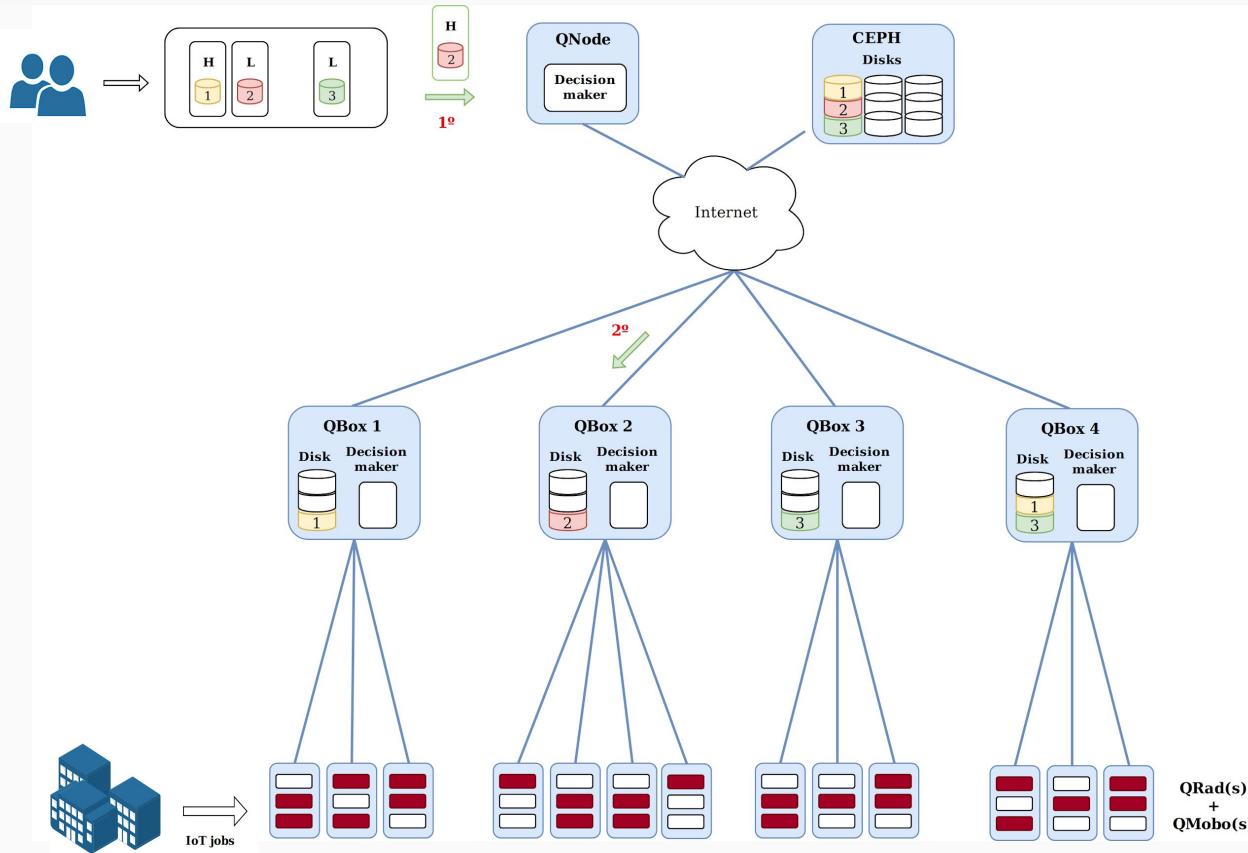
Job Allocation Policies - Locality Based



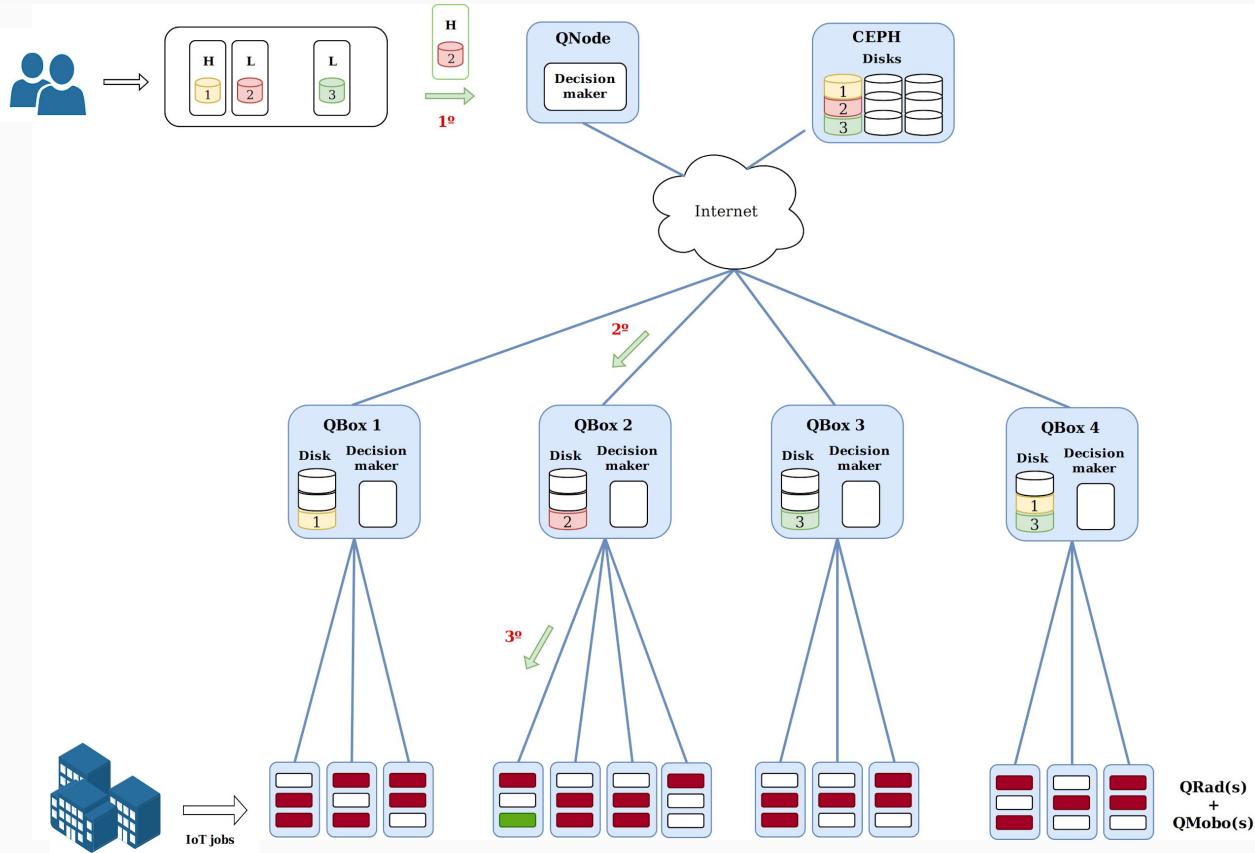
Job Allocation Policies - Locality Based



Job Allocation Policies - Locality Based



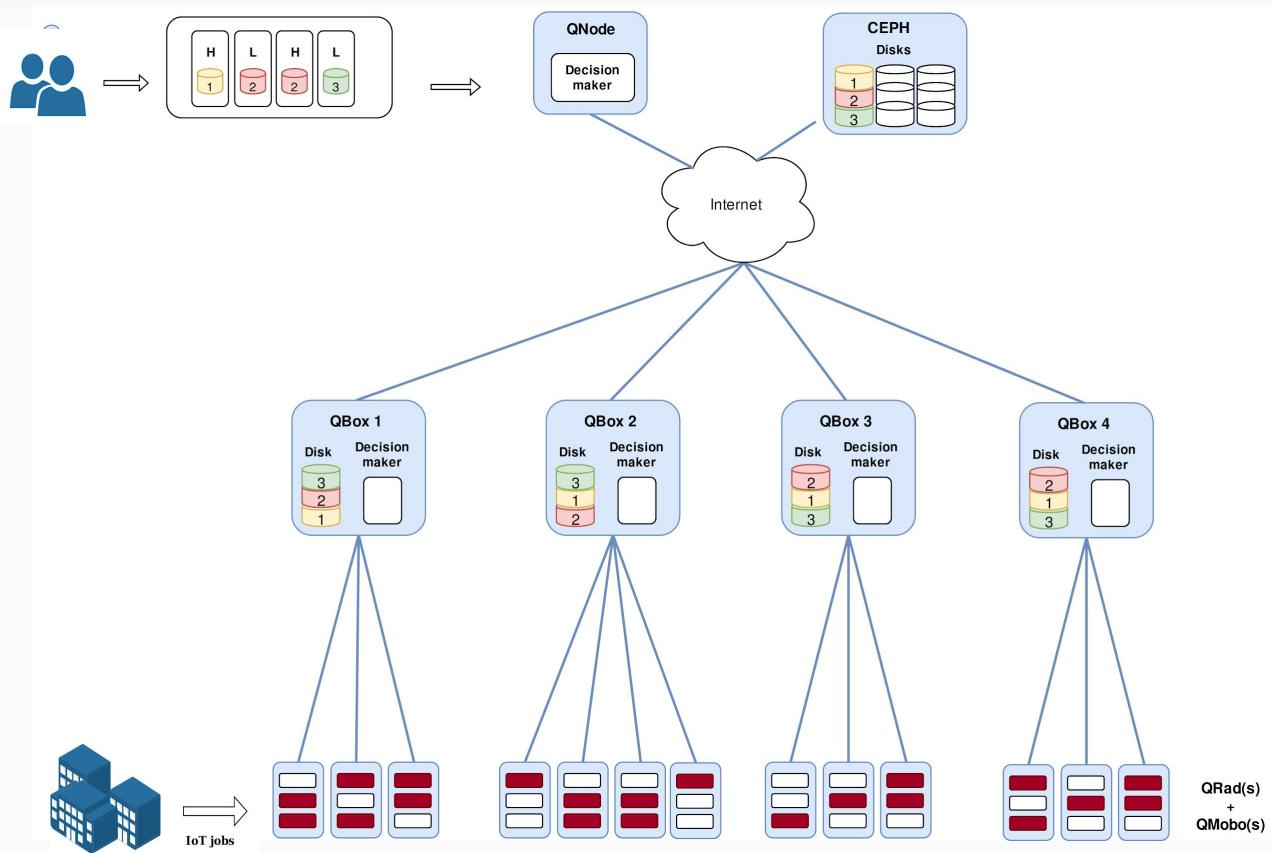
Job Allocation Policies - Locality Based



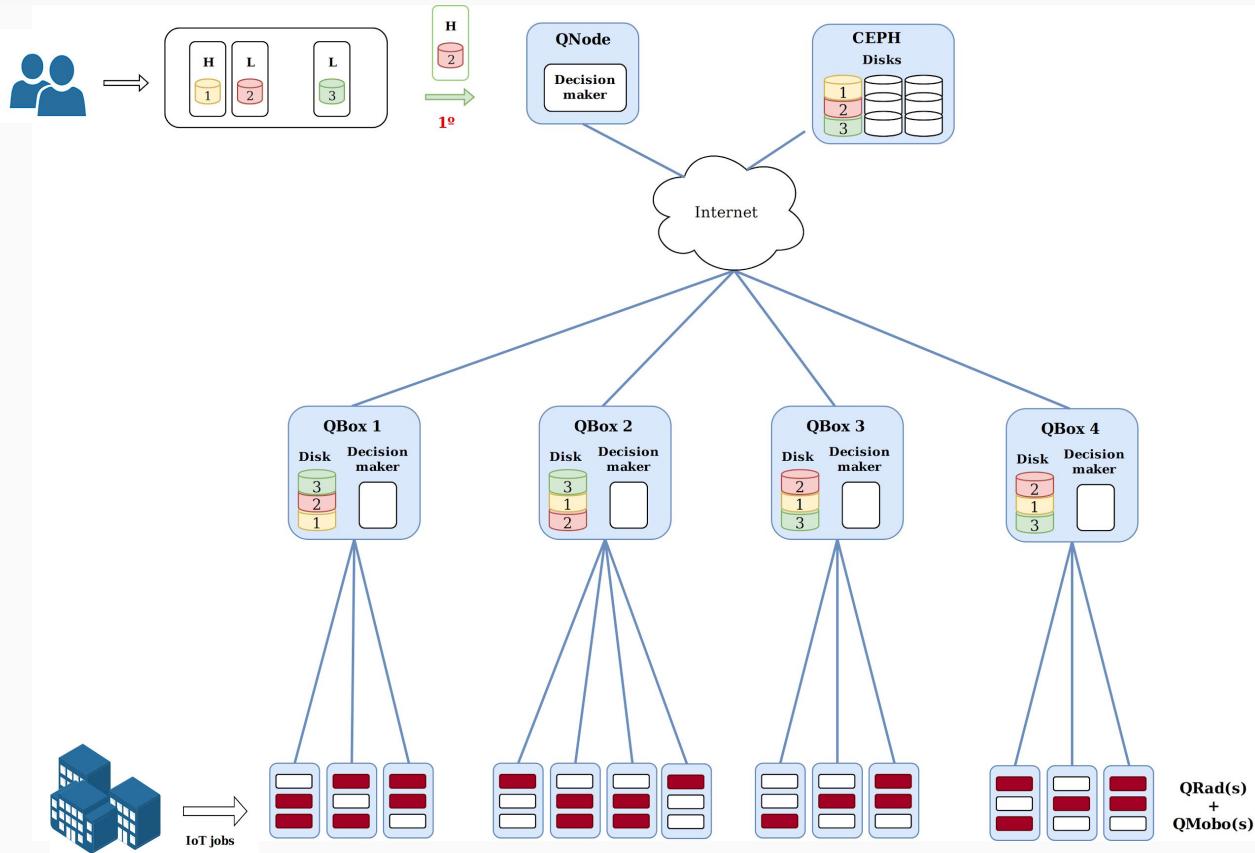
Full Replicate

- Based on Qarnot's policy
- It considers that all data sets are in all QRads upon an instance arrives.
- Dispatches instances, ordered by their priorities, to the QRads that need more heating.

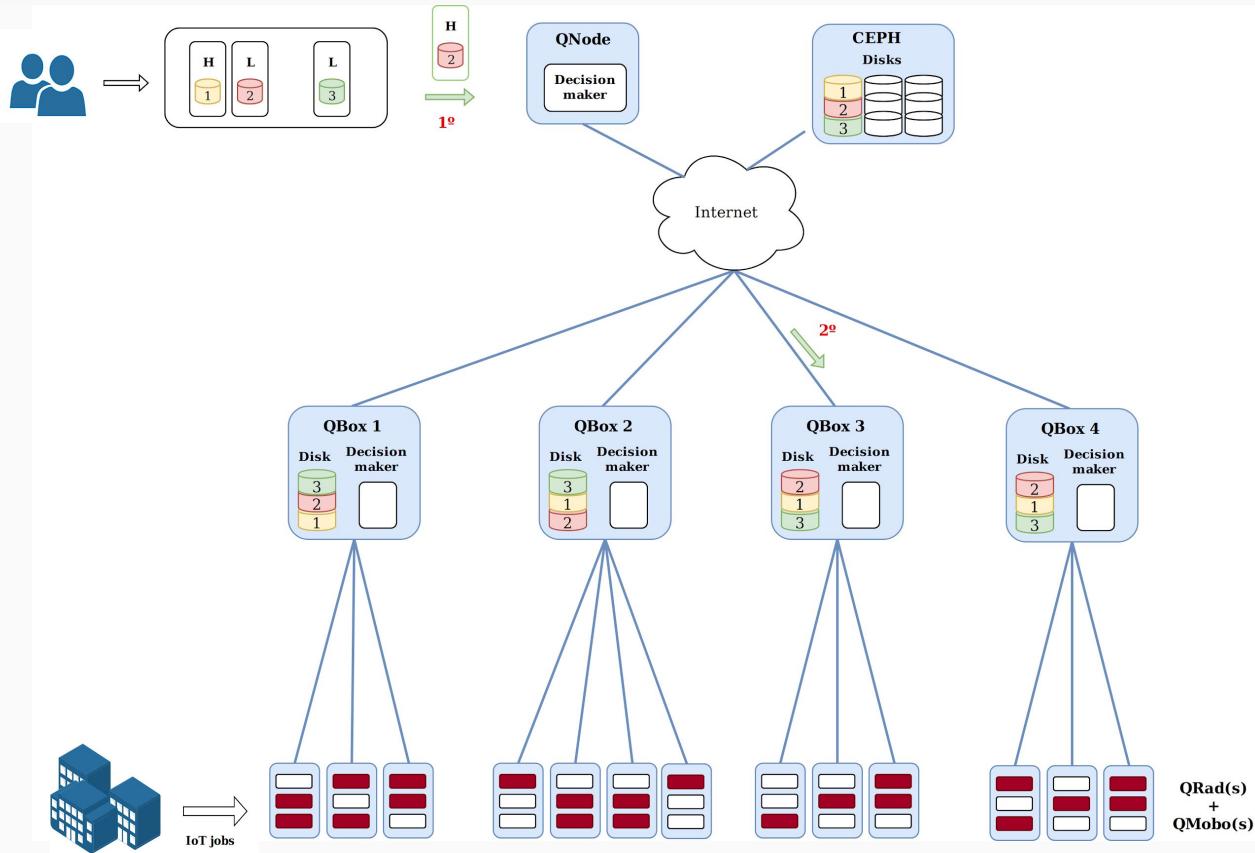
Job Allocation Policies - Full Replicate



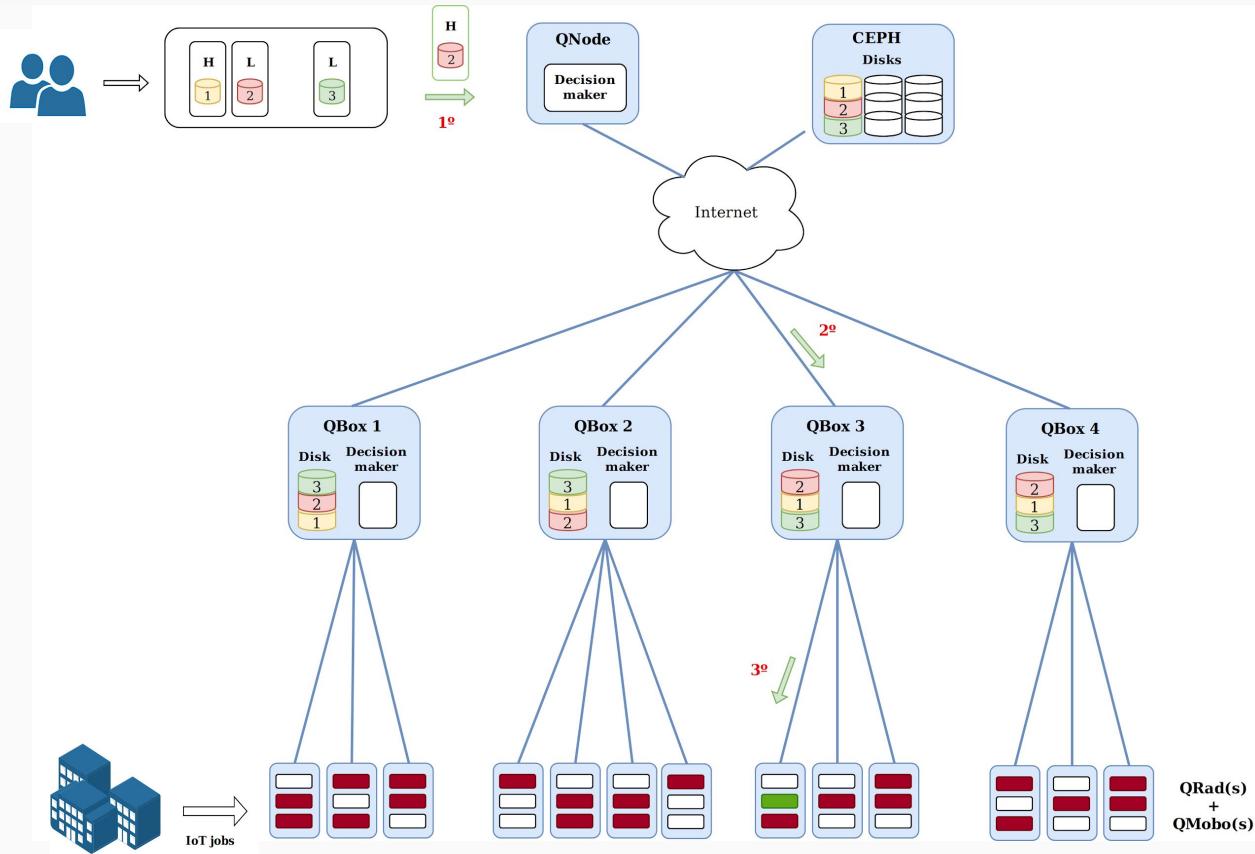
Job Allocation Policies - Full Replicate



Job Allocation Policies - Full Replicate



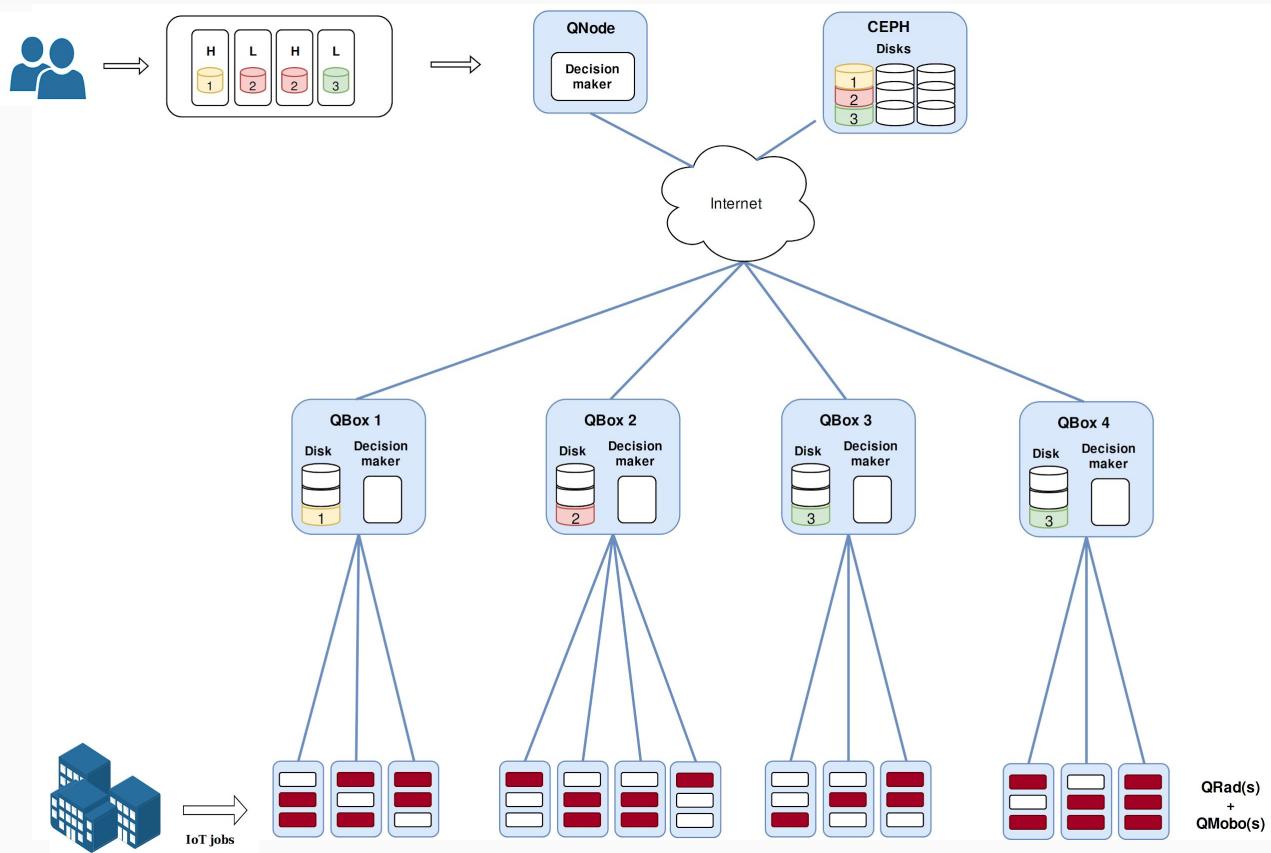
Job Allocation Policies - Full Replicate



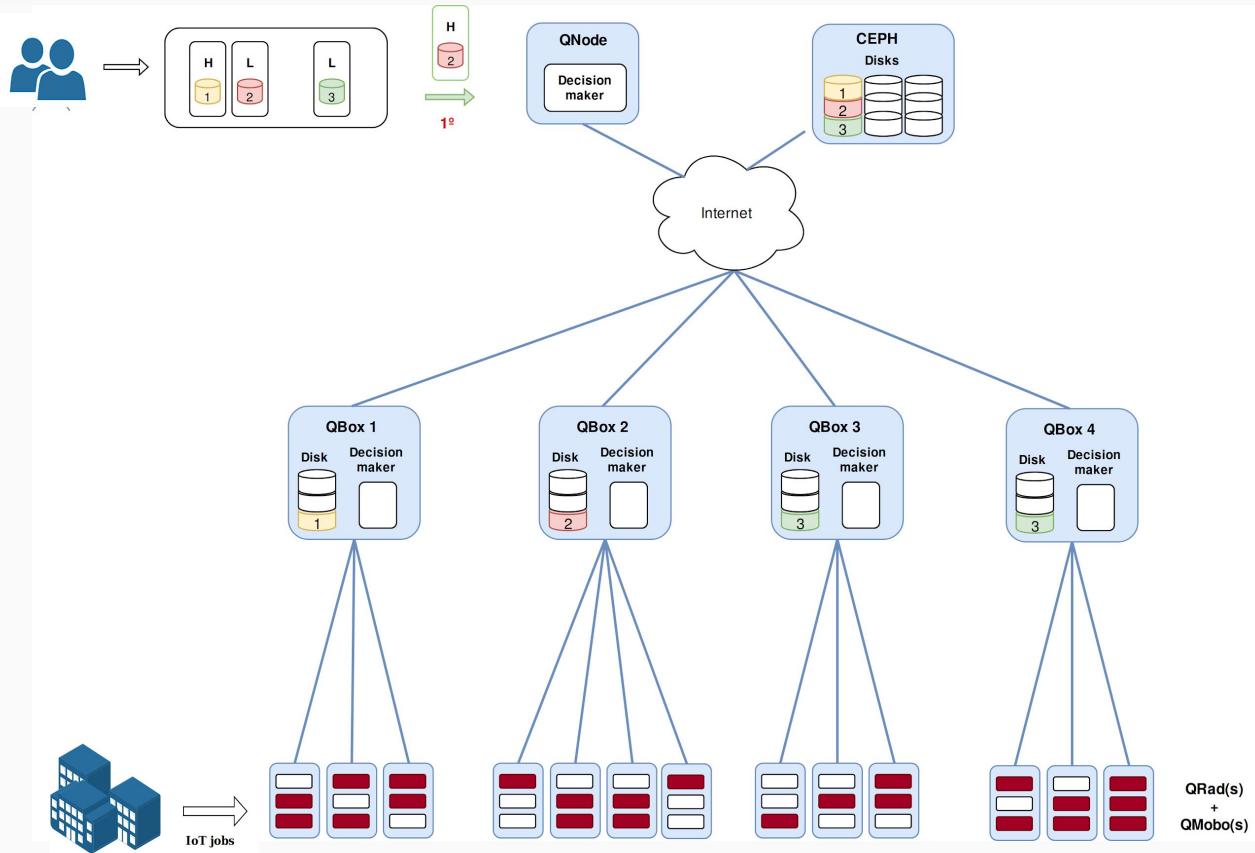
3-10 Replicate

- Based on the *LocalityBased* policy.
- Upon an instance arrives, its required data sets are transferred to the 3 or 10 QBoxes with least loaded disks.
- It dispatches instances, ordered by their priorities, to the QRads that need more heating, **by prioritizing the ones that already have the required data set.**

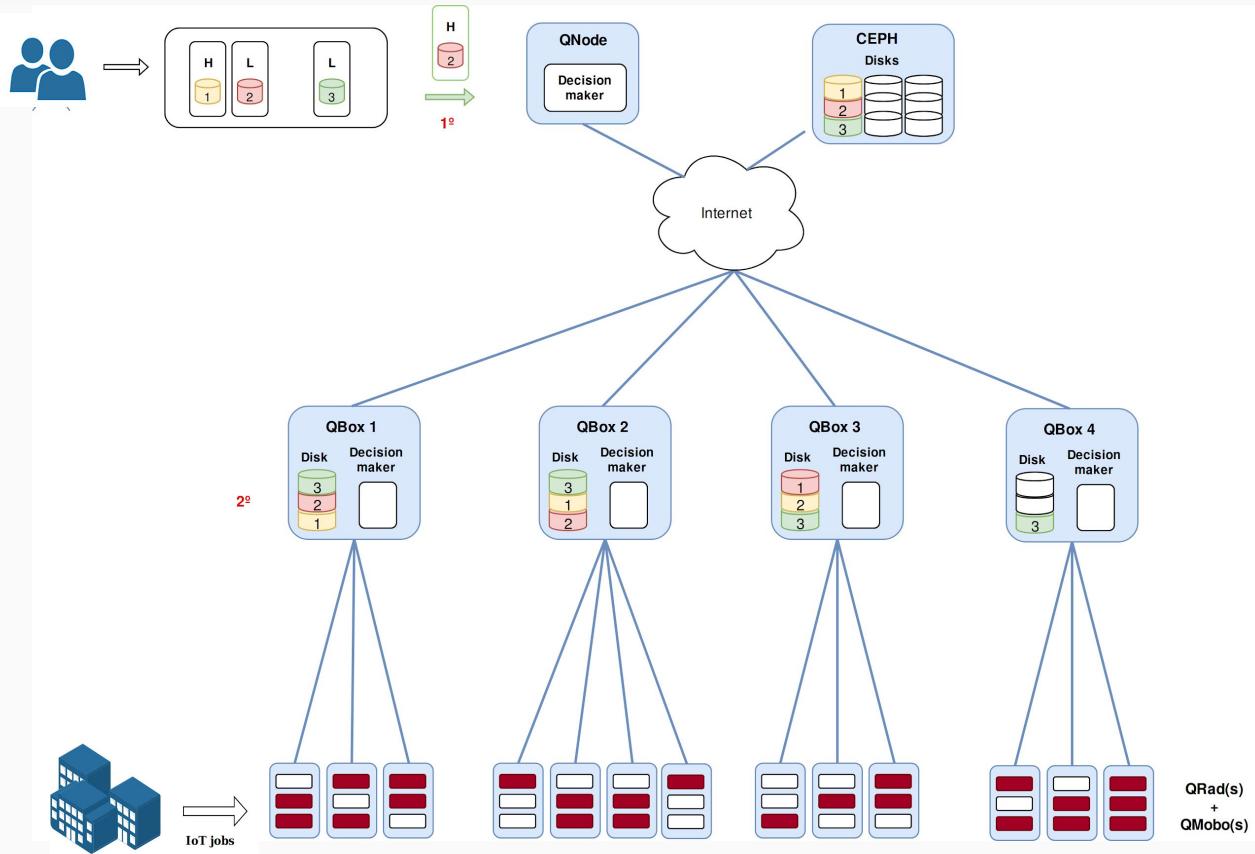
Job Allocation Policies - 3/10 Replicate



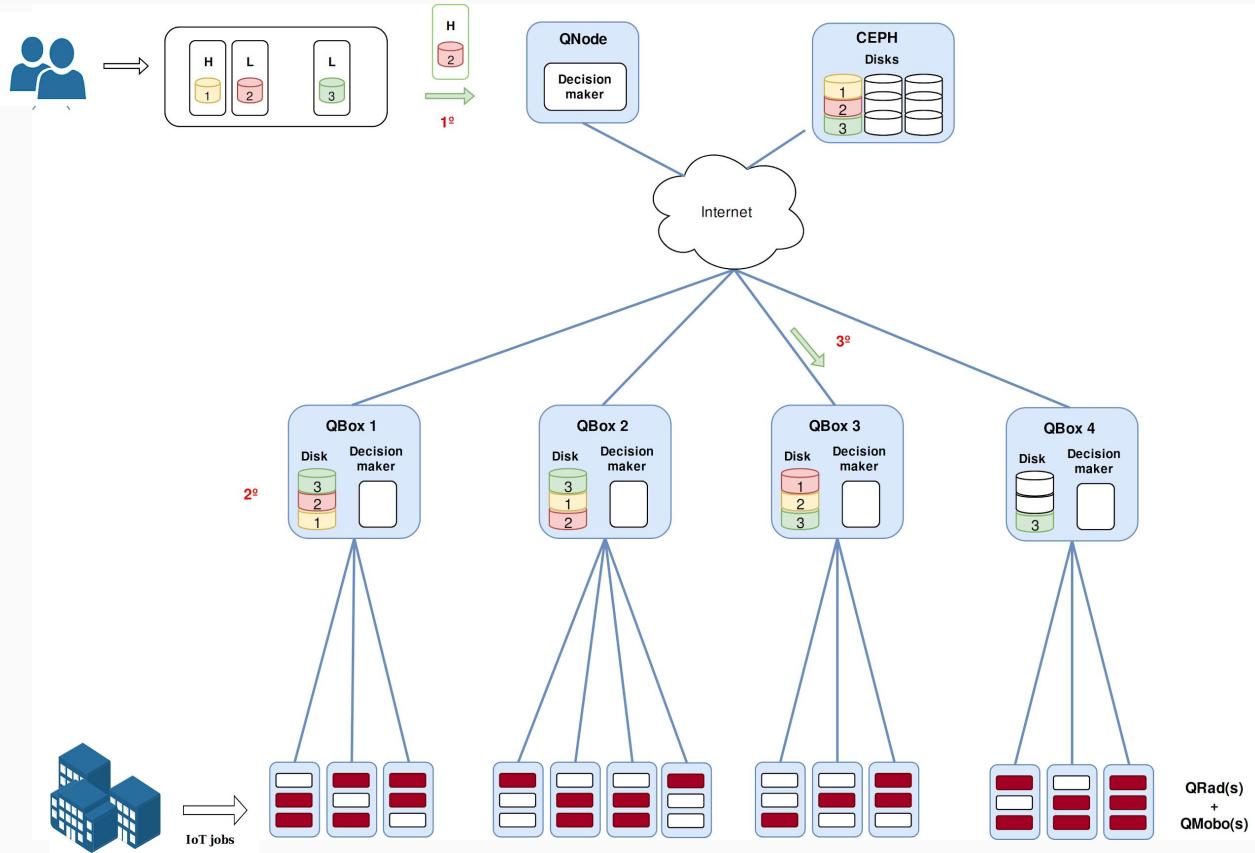
Job Allocation Policies - 3/10 Replicate



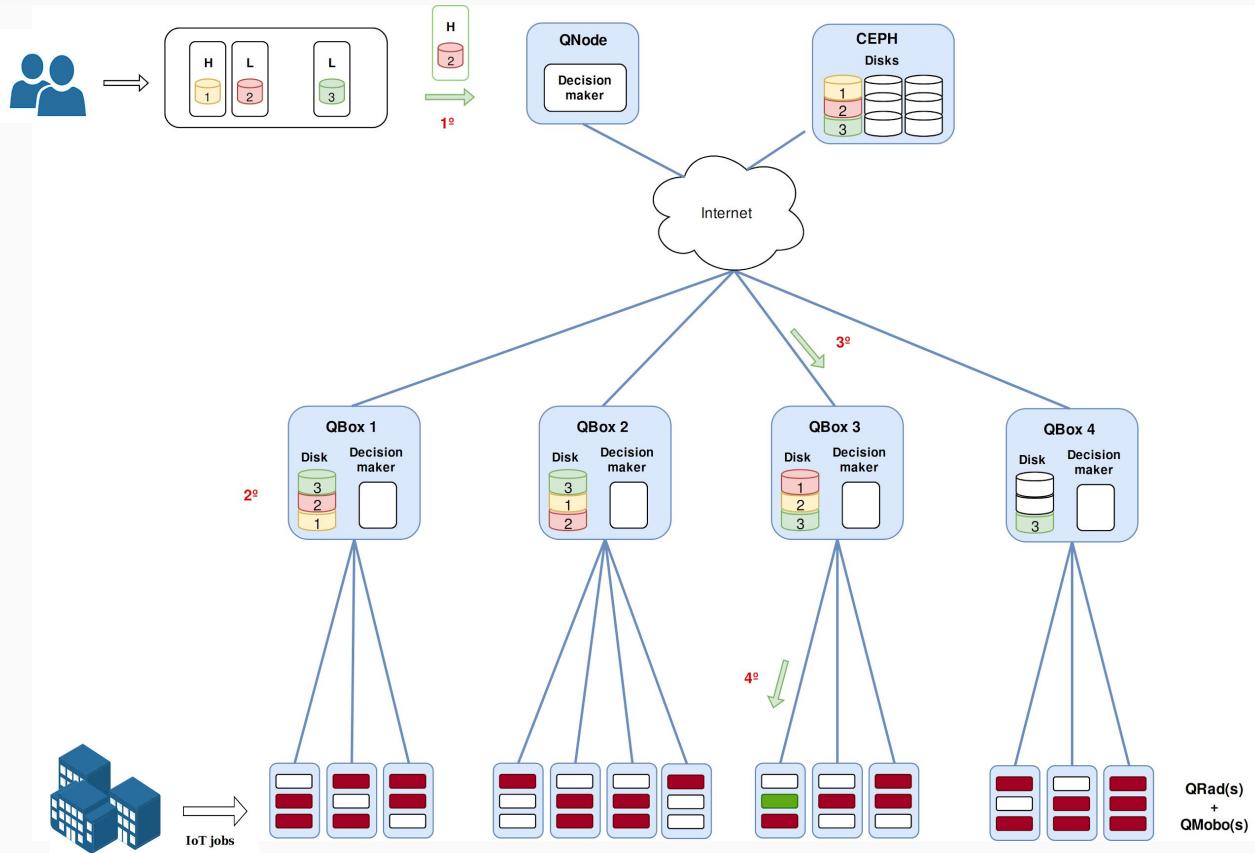
Job Allocation Policies - 3/10 Replicate



Job Allocation Policies - 3/10 Replicate

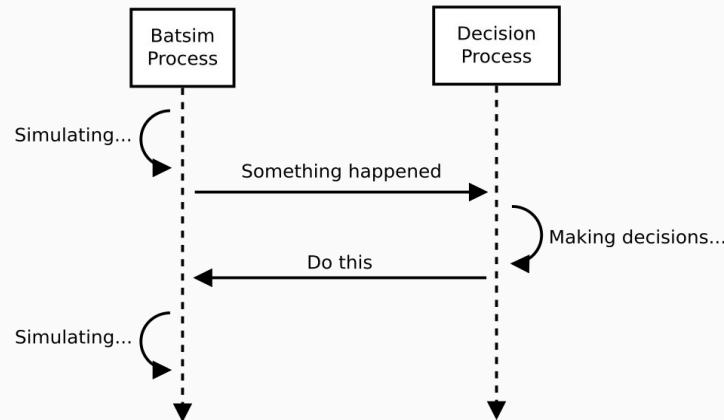


Job Allocation Policies - 3/10 Replicate



Platform Simulators

- **SimGrid (Platform)**: a scientific simulator to study the behavior of large scale distributed systems such as Grids, Clouds, HPC or P2P systems [9].
- **Batsim (Infrastructure)**: a dedicated simulator toolkit to help researchers investigate HPC scheduling strategies have been developed [10].
- **PyBatsim (Decision maker)**: Batsim API for development of scheduling policies.



Platform Simulators: An extension, the Qarnot extractor

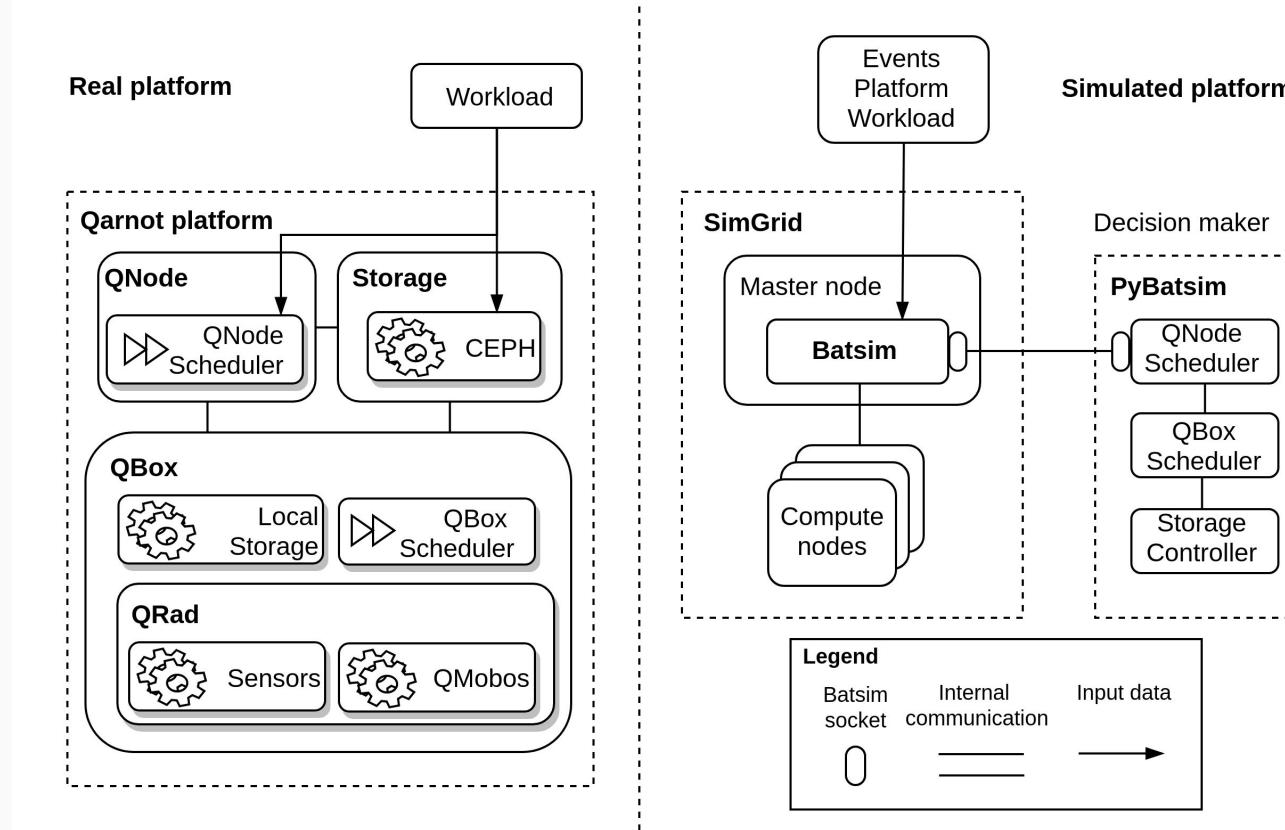
It generates from the Qarnot's logs a set of files given as input for a simulation:

- The platform description
- The list of instances
- The list of external events
- The list of datasets

It extracts other data to be compared with the simulation outputs:

- Logs of ambient temperature
- Logs of instances placements (not implemented yet)
- Times regarding the execution (start, submission and finish)

Platform Simulators: From the Qarnot's Platform to the Simulated One



Experiments

- **Deterministic simulations.**
- **Workloads-size :** 1 day, 3 days, 1 week and 3 weeks
- **Platform:** About ~3390 QMobos, from ~669 QRads, managed by ~20 QBoxes.

Analyses of results

Workloads :

- 4 workloads with size of 1 week:
 - 1 week starting from 03 May, denoted by 1w_03
 - 1 week starting from 10 May, denoted by 1w_10
 - 1 week starting from 17 May, denoted by 1w_17
 - 1 week starting from 24 May, denoted by 1w_24

Analyses :

- Jobs' processing time
- Data sets dependencies
- Job allocation metrics:
 - Number of data transfers
 - Total data transferred (GB)
 - Bounded Slowdown

Analyses of Results: Data Sets Dependencies

One can see that :

- The data set with ID 2 is required by about 6,700 instances, 91% of the total.
- The data set with ID 19, about 5,000 instances, 68% of the total number of instances.
- Other data set IDs reasonably required as 17, 34, 40 and 45.

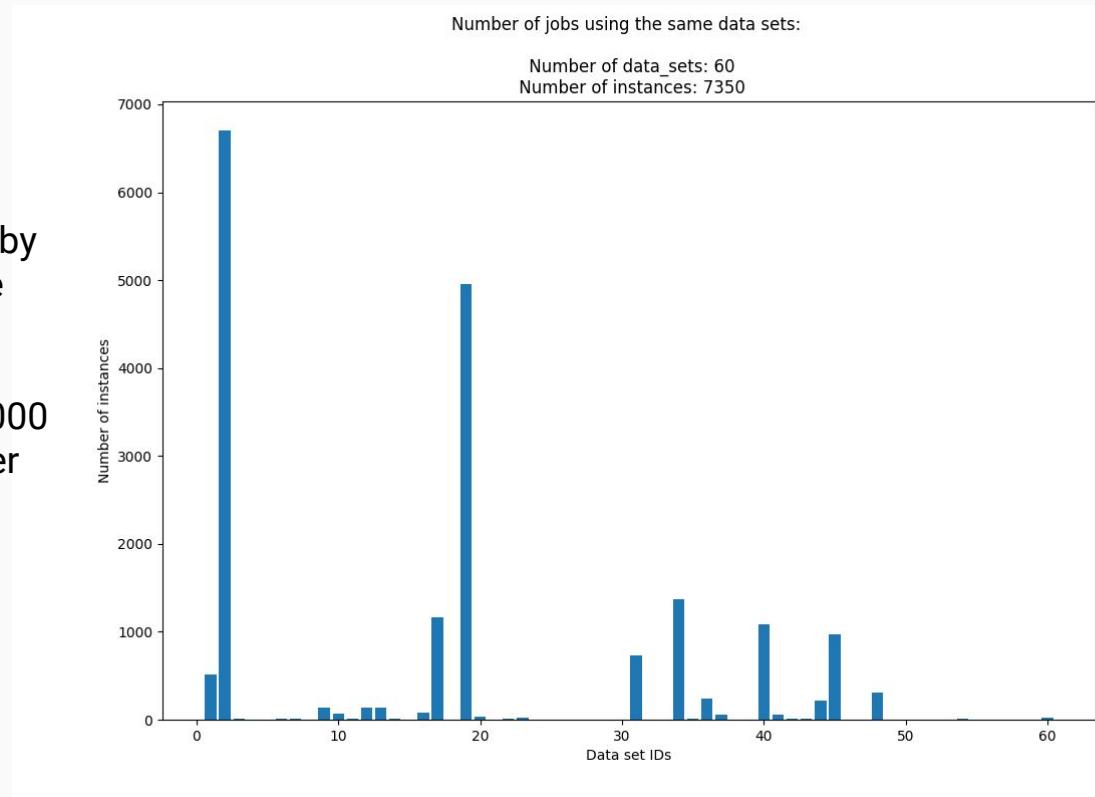


Figure: Data sets dependencies for workload 1w_03

Analyses of Results: Data Sets Dependencies

One can see that:

- The data set with ID 18 is required by about 2,900 48% of the total number of instances.
- The data sets with ID 34 and 16, about respectively 2,400 and 1,700 instances, 40% and 28%.
- Other data set IDs reasonably required as 1, 15, 24, 33 and 37.

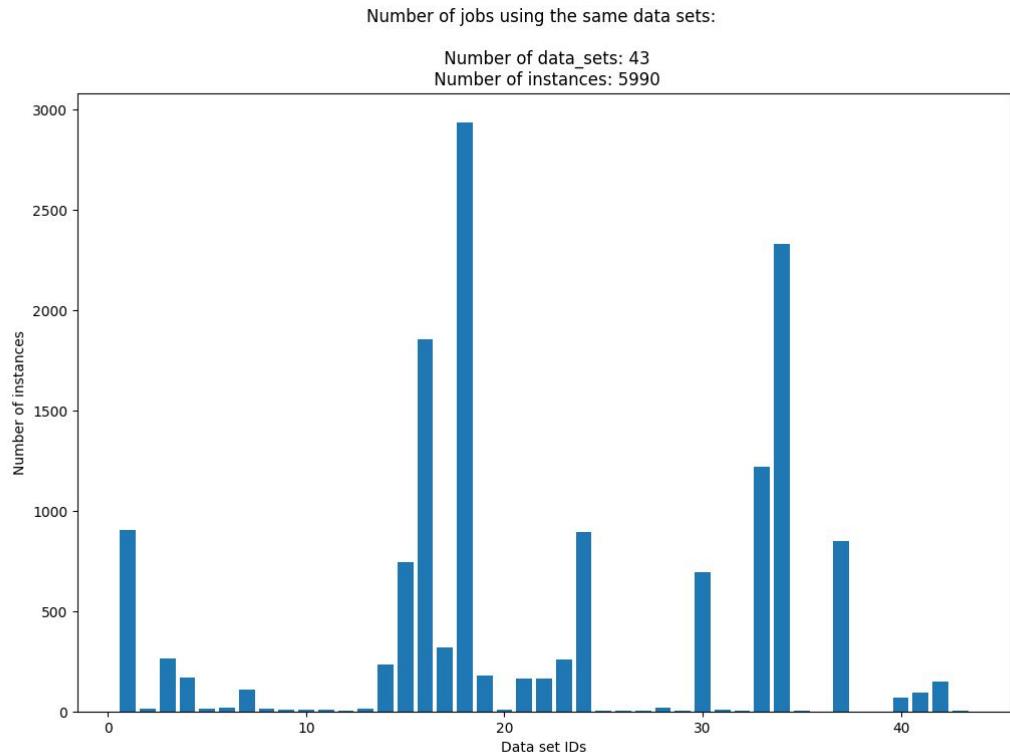


Figure: Data sets dependencies for workload 1w_10

Analyses of Results: Data Sets Dependencies

One can see that:

- The data set with ID 9 is required by about 3,700 instances, 67% of the total number of instances.
- The data sets with ID 25 about 1,800 instances, representing 33% of the total number of instances.
- Other data set IDs reasonably required as 1, 26 and 30

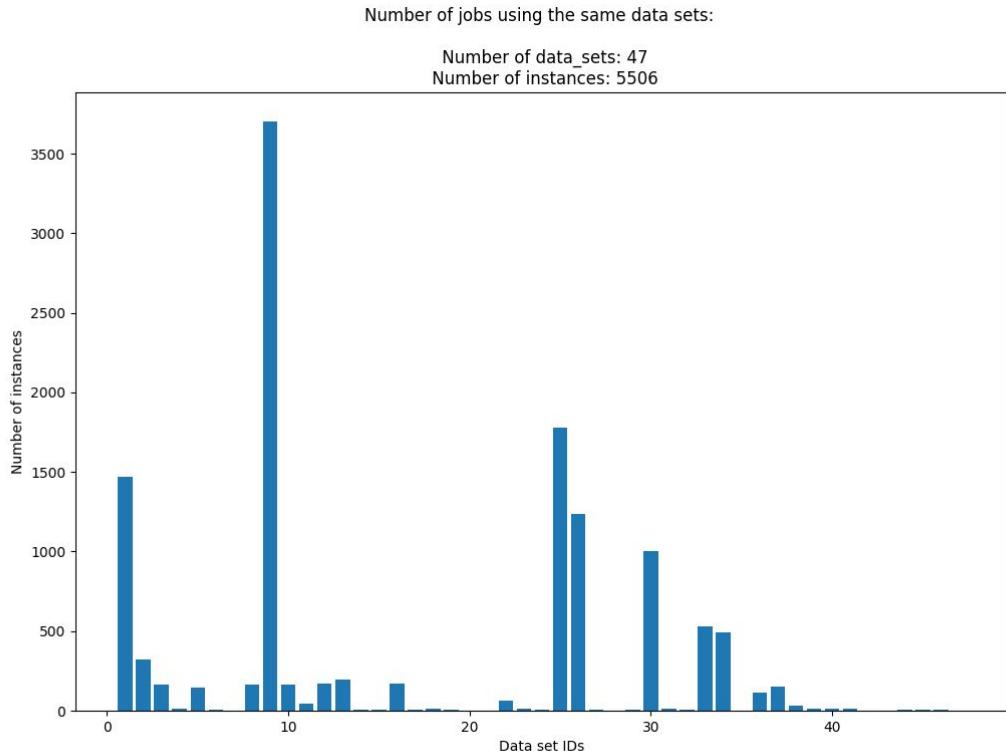


Figure: Data sets dependencies for workload 1w_17

Analyses of Results: Data Sets Dependencies

One can see that:

- The data set with ID 3 is required by about 8,500 instances, 96% of the total number of instances.
- It is followed by the data sets with ID 15 about 5,000 instances, 56% of the total number of instances.
- Other data set IDs reasonably required as 13 and 14

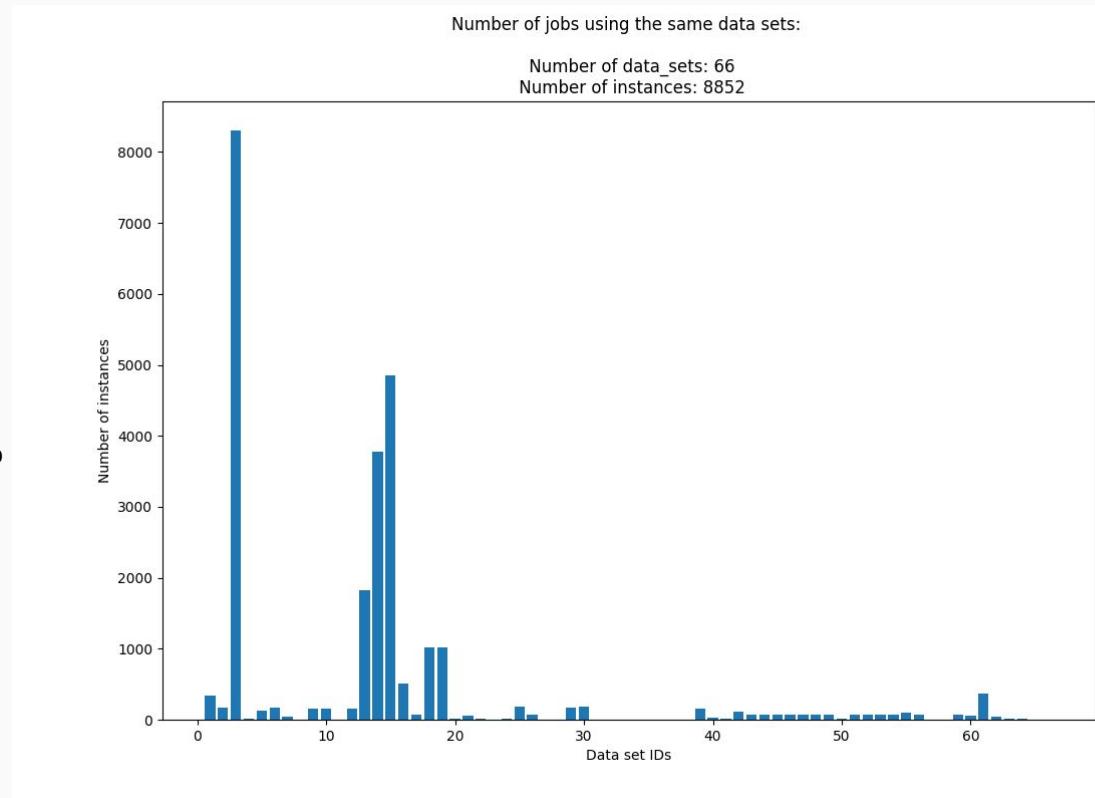


Figure: Data sets dependencies for workload 1w_24

Analyses of Results: Data Transfer

One can see that

- For both metrics the schedulers *FullReplicate*, *Replicate10* and *Replicate3* are the three with the highest values with the exception of the *Replicate3* for the workload 2 in the Number of data transfers.
 - It is totally expected since, respectively, they replicate data sets in all, 10 and 3 QBoxes.
- The *LocalityBased* got close or higher values in comparison with the *Standard* scheduler
 - It is explained by the data set dependencies.

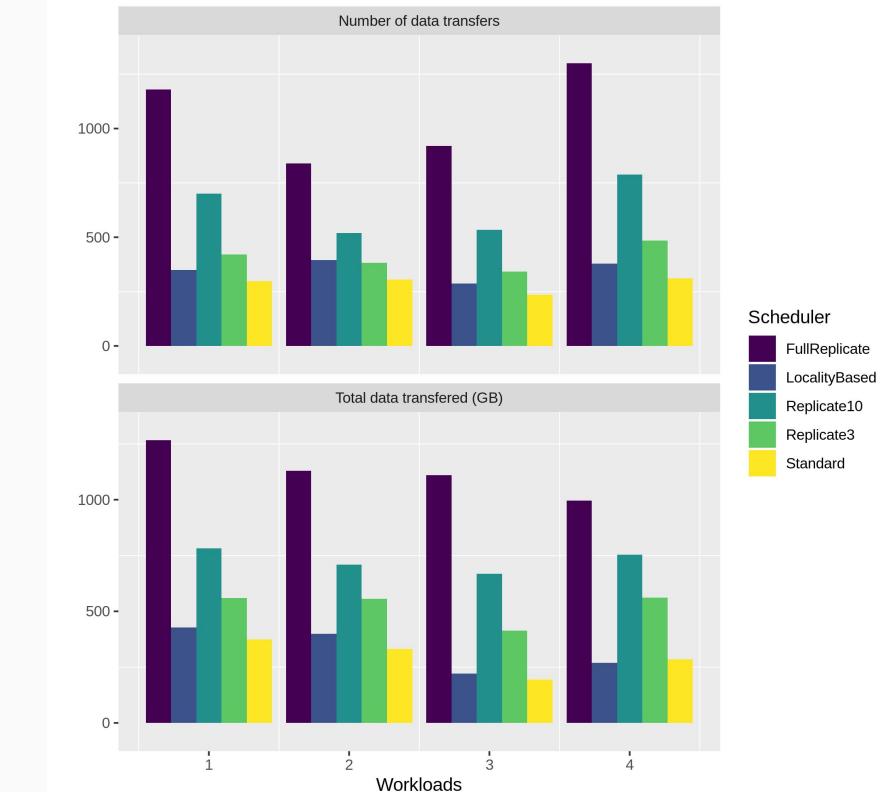


Figure: Metrics for data transfers.

Analyses of Results: **Bounded Slowdown**

- How does the waiting time behave?
- Does the waiting time is proportional with the job's size?

$$\text{bounded-slowdown} = \max \left\{ \frac{T_w + T_r}{\max\{T_r, \tau\}}, 1 \right\}$$

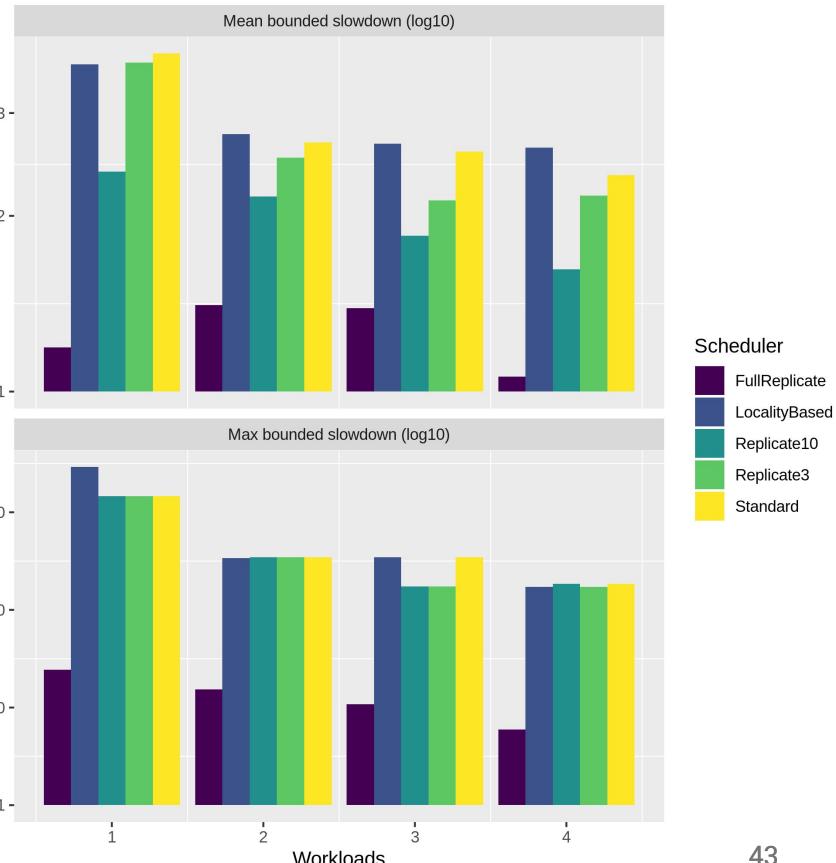
Where,

- T_w is the waiting time,
- T_r is the execution time,
- τ is a threshold.

Analyses of Results: Bounded Slowdown

One can see that

- The *FullReplicate* scheduler presents the lowest values for both metrics.
- For almost all the other cases, the *Replicate10* and *Replicate3* are the next lowest ones, with the exception of the Max bounded slowdown with the second and fourth workloads.
 - It is also totally expected since these schedulers replicate much more data sets than the *LocalityBased* and *Standard*.
- The *LocalityBased* presents close or higher values when compared with the *Standard* thanks the data sets dependencies.



Analyses of Results: Jobs' Processing Time

Table: Processing time distribution for all workloads

Statistics	1w_03	1w_10	1w_17	1w_24
Count	7350	5989	5497	8850
Mean (s)	465.96	582.25	480.21	403.93
Std (s)	817.18	2400.22	2268.20	1723.62
Min (s)	1.0	1.0	1.0	1.0
25% (s)	132.0	77.0	48.0	34.0
50% (s)	235.0	151.0	106.0	117.0
75% (s)	635.0	425.0	207.0	291.0
Max (s)	35372.0	27121.0	29700.0	28952.0

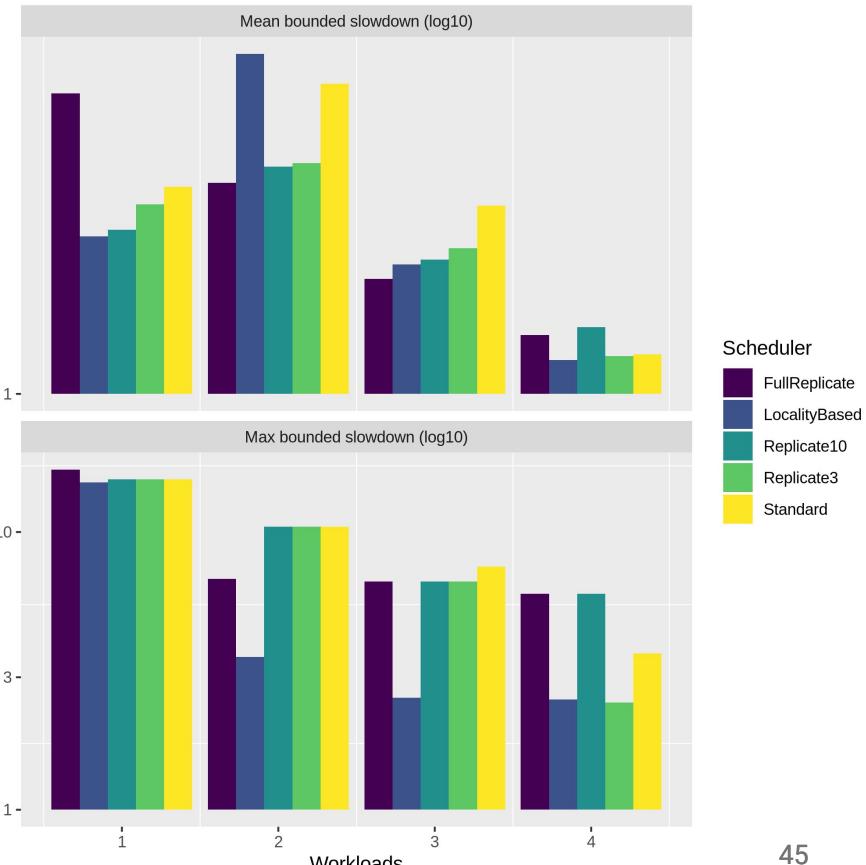
For all workloads, these distributions characterize the workloads as:

- 75% composed by short jobs,
- 25% composed by long jobs.

Analyses of Results: Bounded Slowdown for Long Jobs

One can see that:

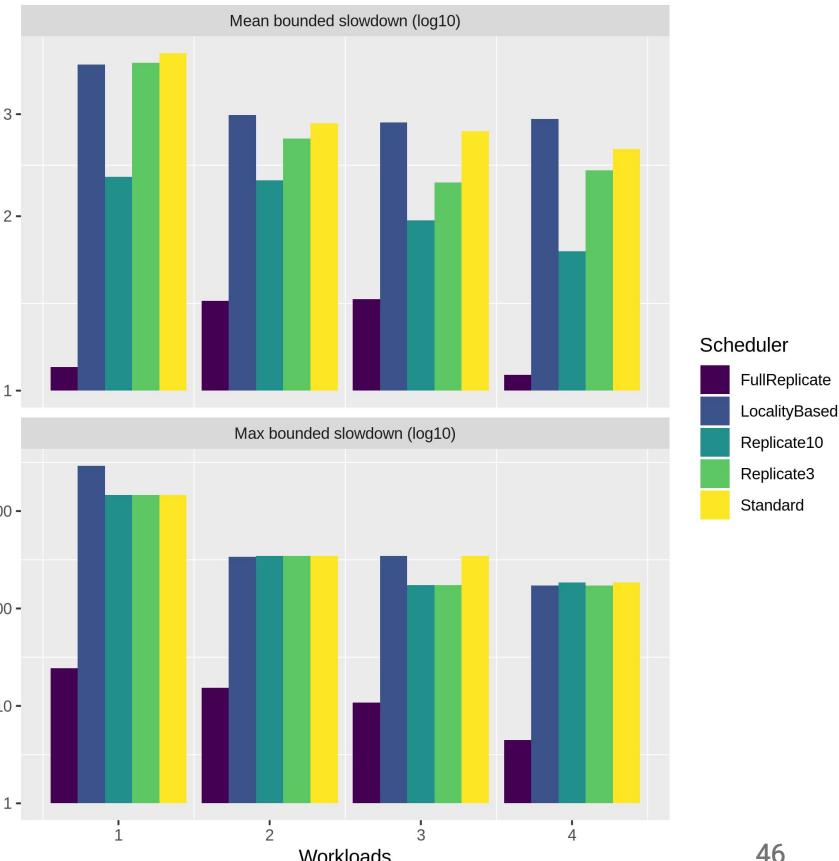
- The values for the *FullReplicate* now are among the highest ones.
 - We justify it by the *waiting_time* from the allocation decision process that, in general, takes more time to schedule long jobs than short jobs.



Analyses of Results: Bounded Slowdown for Short Jobs

As one can see that

- Presents the same behavior than the one with all jobs together, the highest are owned by the non replicated schedulers.
 - We attributed it to the jobs' *waiting_time*, since the *execution_time* is short.
- Finally, as the replicate based schedulers present low values when compared with the others.
 - We attributed that the *waiting_time* for the *LocalityBased* and Standard schedulers.



Analyses of Results: **Bounded Slowdown**

- Comparing the results filtered by short and long jobs, **the premise that short jobs are more sensitive to these metrics is true in our case**, because the first figure presents much more high values than the second.
- And considering that 75% of the jobs are being represented as the short jobs and 25% in as long jobs, we understood that **the general behavior of this metric is much more impacted by the short jobs** into these workloads.

Conclusions

- Edge Computing is a **computational paradigm that have been evolved from the Cloud Computing due to the growth of Mobile / IoT devices** that embedded enough computation power to avoid the data processment centralization on the Cloud.
- In order of the heterogeneity of such devices, **it is still difficult to extend the known solutions of Cloud to the Edge Computing**, then its importance of study.
- From the literature review **is not known a good Edge Platform simulator**, then this thesis aimed to **show an example of Simulated Platform**, that still on going, but is already possible to be applied in use cases as the Qarnot Computing.
- Using such platform, **we implemented different scheduling policies and realized experiments** following the jobs' processing time, data sets' dependencies, data transfers and bounded slowdown.
- Instead we do not have large experimental input, our analyses **characterized the Qarnot's workload** as composed by 75% of short jobs and 25% of long jobs, within high dependency on the same data sets, and **we were able to indicate the best scheduling policies** are those based on replication.
- In addition, these results are used in **a paper [] submitted to the IEEE MASCOTS 2019**.

Further Remarks

- The work developed during this thesis is **very useful to the Qarnot Computing** as:
 - The results showed that the **replication based scheduling policies could be better**.
 - An example of **easy implementation and modification of scheduling policies** to be studied.
 - A platform **to predict behavior** simulating more external events.
 - A platform **to simulate specific environment** such as an new office where QRads will be installed.
- From the literature, **several challenges have been emerged in the context of Cloud and Edge Computing**. We believe that this thesis **contributed with an example of implementation of scheduling policies in an Simulated Edge Platform**, hence it **was good step** to continue investigating such challenges, as the development of a **Digital Twin**, a goal of our work group.

Bibliography

- [1] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5):637–646, Oct 2016.
- [2] W. Shi and S. Dustdar. The promise of edge computing. *Computer*, 49(5):78–81, May 2016.
- [4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief. A survey on mobile edge computing: The communication perspective. *IEEE Communications Surveys Tutorials*, 19(4):2322–2358, Fourthquarter 2017.
- [5] Luiz Bittencourt, Roger Immich, Rizos Sakellariou, Nelson Fonseca, Edmundo Madeira, Marilia Curado, Leandro Villas, Luiz DaSilva, Craig Lee, and Omer Rana. The internet of things, fog and cloud continuum: Integration and challenges. *Internet of Things*, 3-4:134 – 155, 2018.
- [6] S. M. Parikh. A survey on cloud computing resource allocation techniques. In 2013 Nirma University International Conference on Engineering (NUiCONE), pages 1–5, Nov 2013.
- [7] Lu Huang, Hai-shan Chen, and Ting-ting Hu. Survey on resource allocation policy and job scheduling algorithms of cloud computing1. *Journal of Software*, 8, 02 2013.
- [8] Hameed Hussain, Saif Ur Rehman Malik, Abdul Hameed, Samee Ullah Khan, Gage Bickler, Nasro Min-Allah, Muhammad Bilal Qureshi, Limin Zhang, Wang Yongji, Nasir Ghani, Joanna Kolodziej, Albert Y. Zomaya, Cheng-Zhong Xu, Pavan Balaji, Abhinav Vishnu, Fredric Pinel, Johnatan E. Pecero, Dzmitry Kliazovich, Pascal Bouvry, Hongxi- ang Li, Lizhe Wang, Dan Chen, and Ammar Rayes. A survey on resource allocation in high performance distributed computing systems. *Parallel Computing*, 39(11):709 – 736, 2013.

Bibliography

- [9] Henri Casanova, Arnaud Giersch, Arnaud Legrand, Martin Quinson, and Frédéric Suter. Versatile, Scalable, and Accurate Simulation of Distributed Applications and Platforms. *Journal of Parallel and Distributed Computing*, 74(10):2899–2917, June 2014.
- [10] Pierre-François Dutot, Michael Mercier, Millian Poquet, and Olivier Richard. Batsim: a Realistic Language-Independent Resources and Jobs Management Systems Simulator. In 20th Workshop on Job Scheduling Strategies for Parallel Processing, Chicago, United States, May 2016.
- [11] Anderson da Silva, Clement Mommessin, Pierre Neyron, Denis Trystram, adwait bauskar, Adrien Lebre, Alexandre Van Kempen, Yanik Ngoko, and yoann ricordel. Investigating Placement Challenges in Edge Infrastructures through a Common Simulator. In Mascots 2019 - 27th IEEE International Symposium on the Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, pages 1–16, Rennes, France, October 2019.

Thank you for your attention!

Investigating Job Allocation Policies in Edge Computing Platforms

Anderson Andrei DA SILVA

Advisor: Prof. Denis TRYSTRAM

Jury:



**Master of Science in Informatics at Grenoble
Parallel, Distributed and Embedded Systems**

Université Grenoble Alpes



Platform Simulators: **Simgrid**

Platform simulator with realistic network and computation models.

Used to simulate the Qarnot platform:

- CPUs in QMobos
- QBox disks and CEPH
- Network links between CEPH and QBox disks

Plugin for temperature support (QRad and ambient air) w.r.t. power consumption.

Platform Simulators: **Batsim**

Infrastructure simulator for jobs and I/O scheduling, built on top of SimGrid.

Completes SimGrid's simulation with:

- The submission of tasks
- The submission of external events (e.g., target/outside temperature change)
- The communication with the decision making process

Platform Simulators: PyBatsim

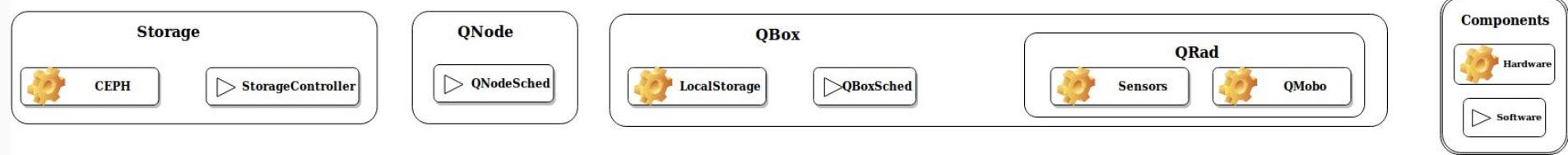
Decision making process, which talks with the Batsim process via a socket to manage:

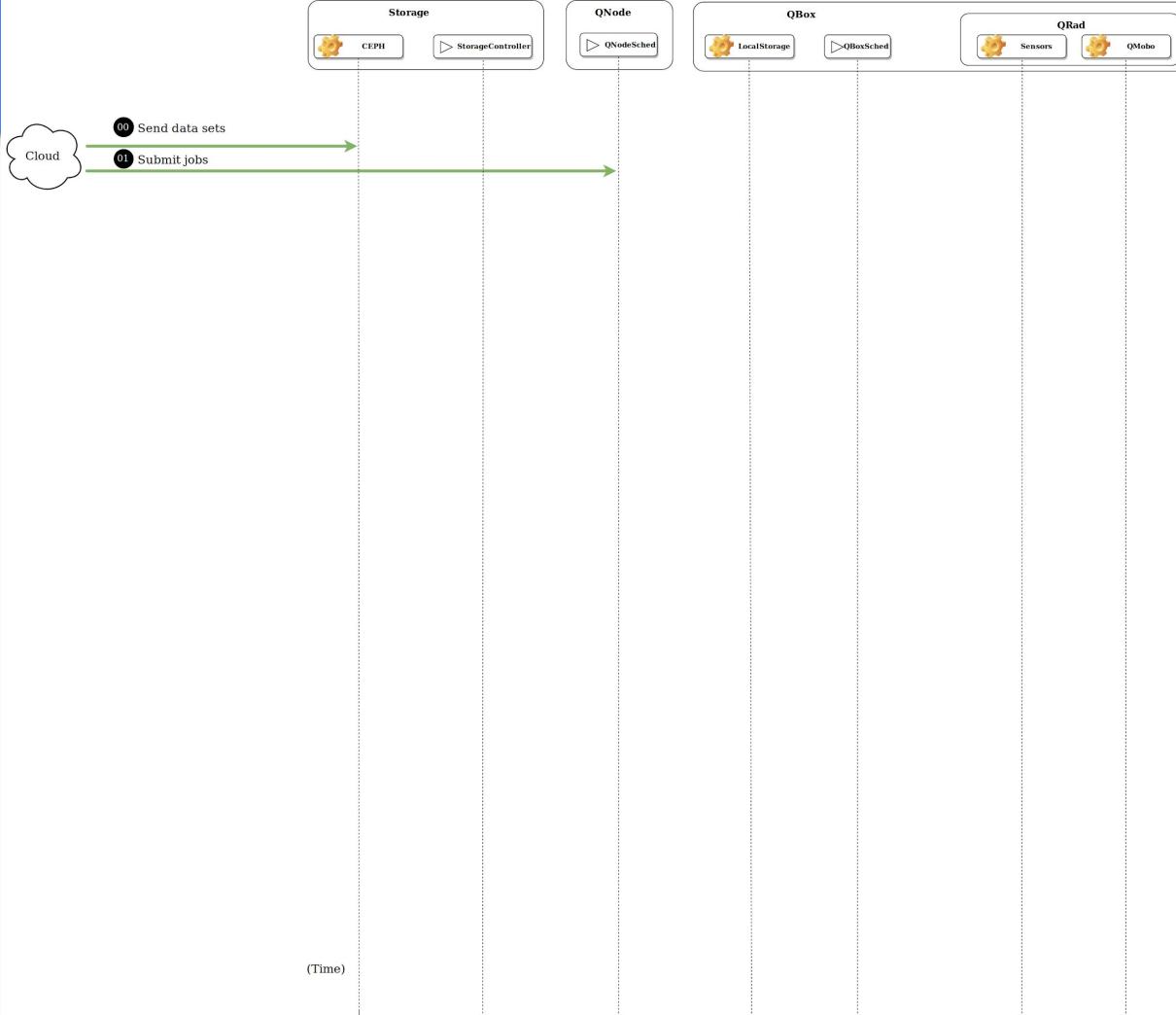
- The dispatch of tasks/instances (QNode scheduler)
- The placement of instances (QBox scheduler)
- The different storages and data movements (Storage Controller)
- The heating needs (Frequency Regulator)

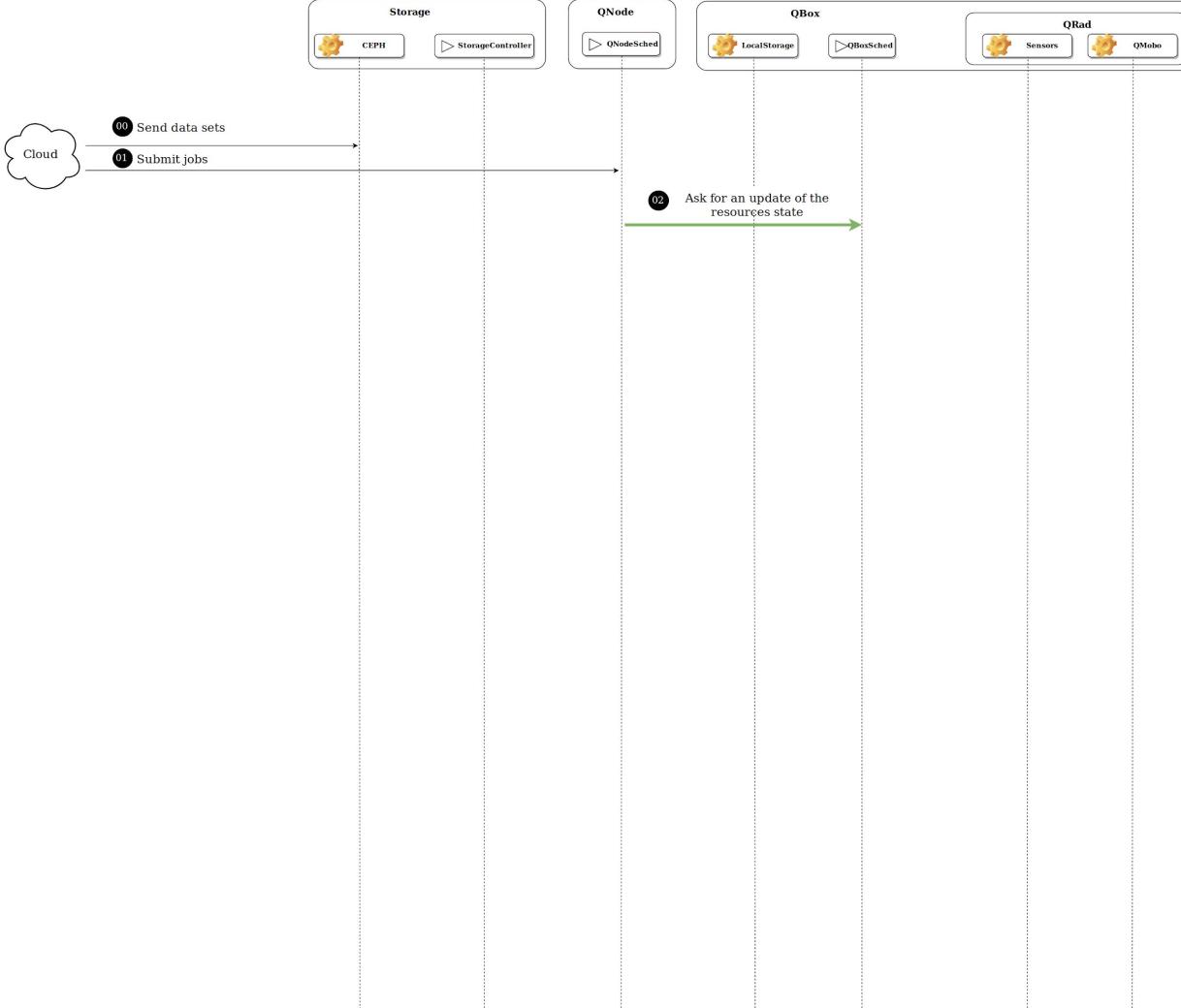
Platform Simulators: Limitations

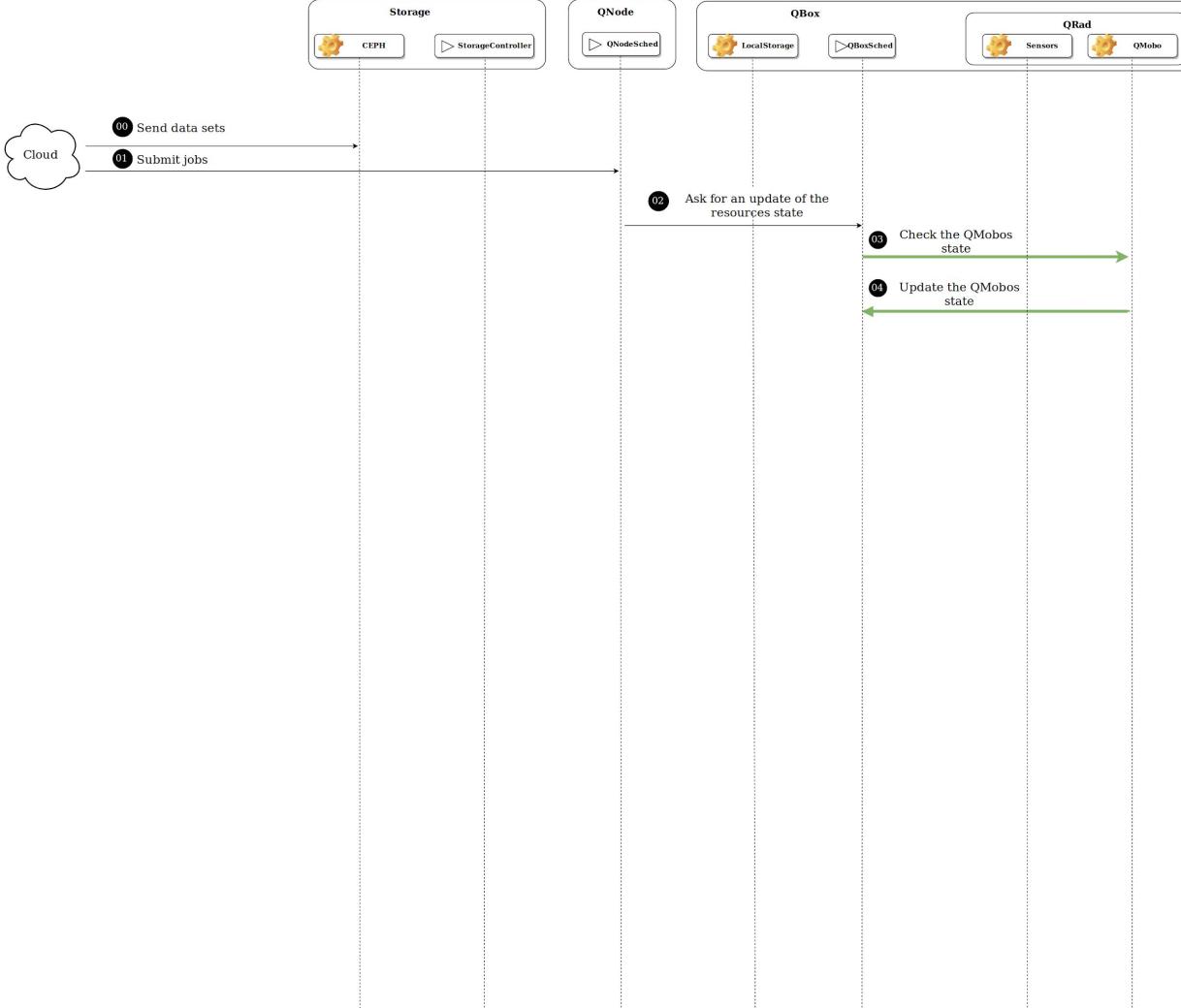
- No cluster tasks
- No booting time of QMobos before starting instances
- No real values of power/speed of CPUs
- Empty initial state of the platform
- No external event “QMobo X becomes (un)available” (do we want that?)

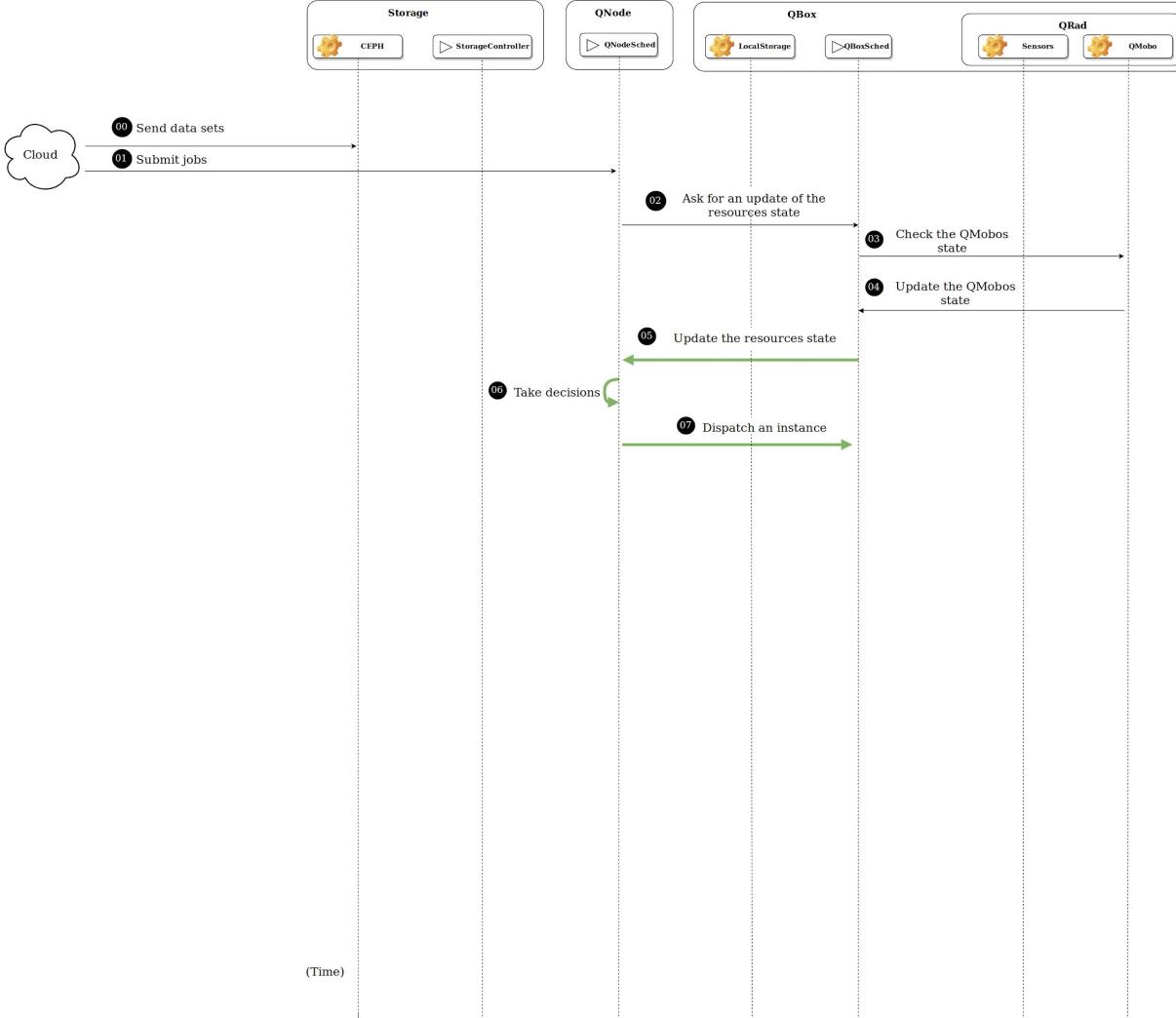
The real platform

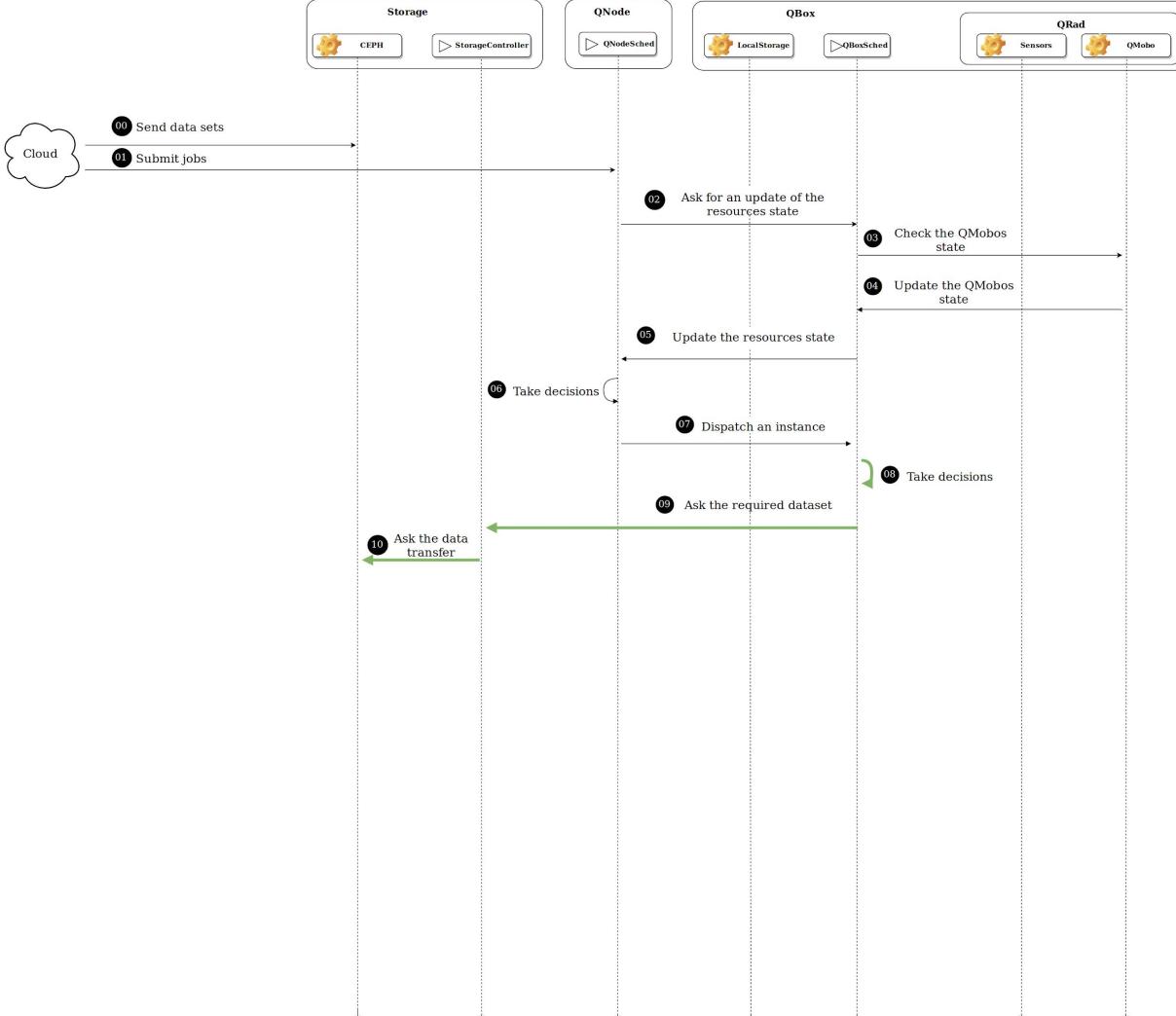


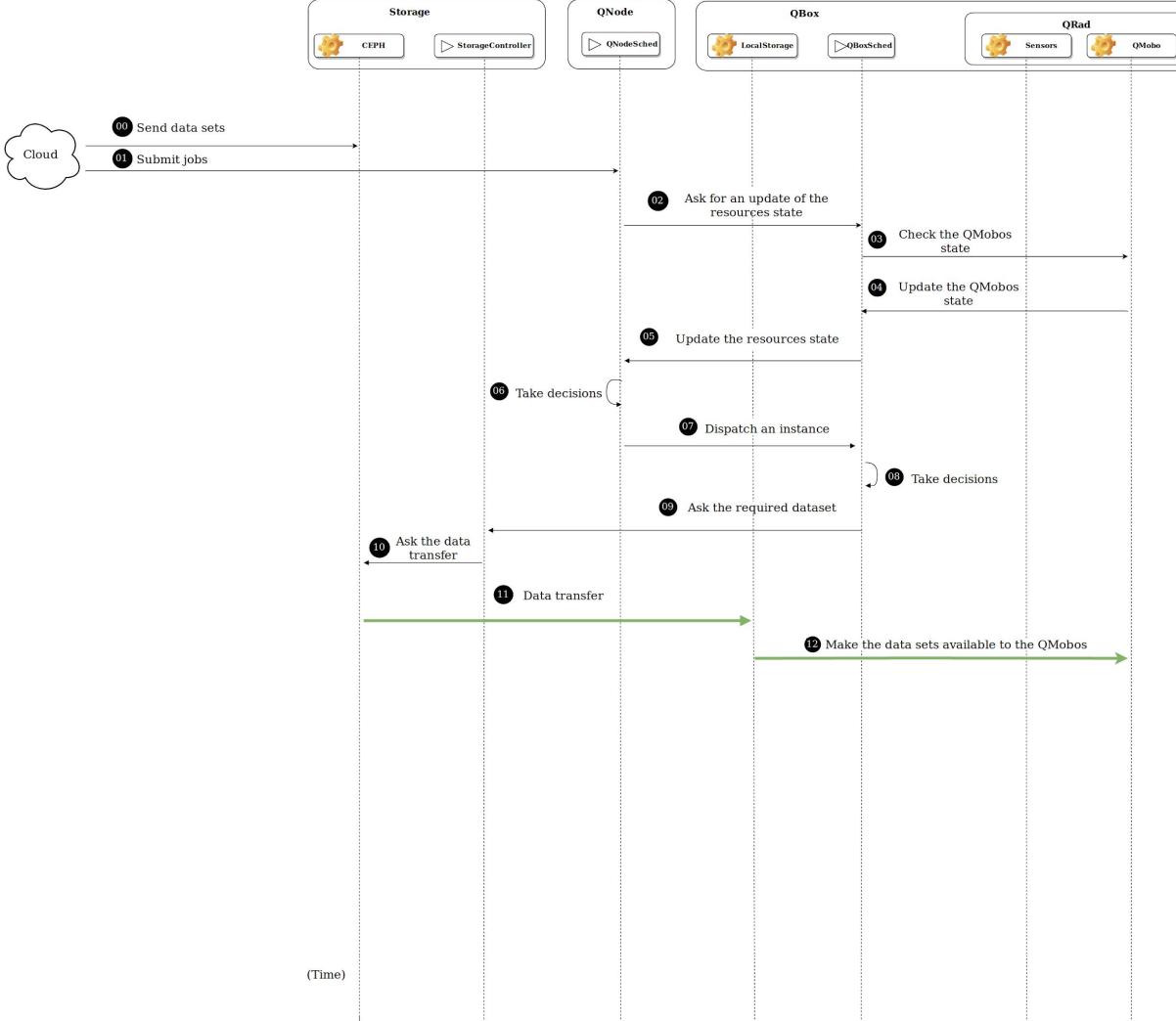


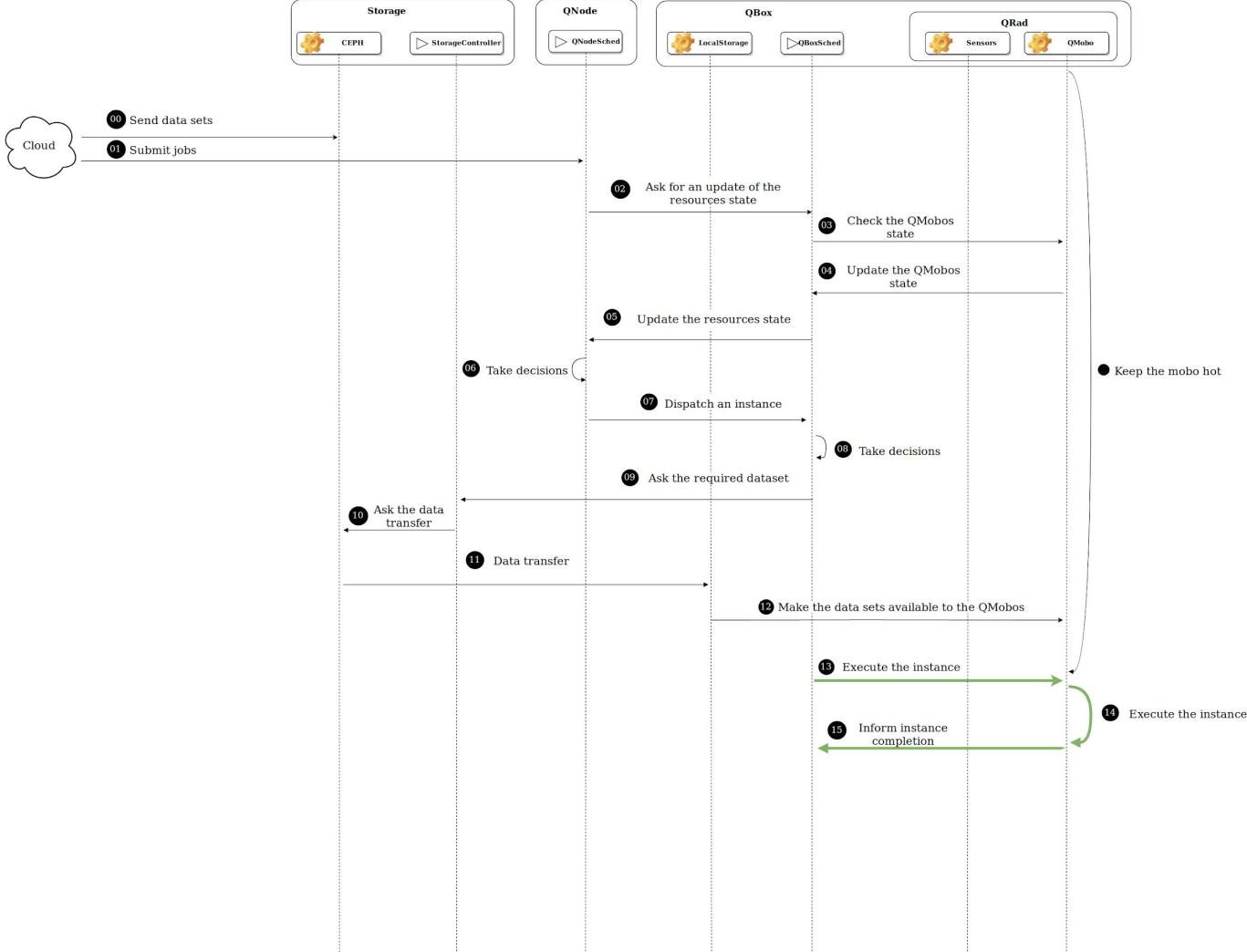


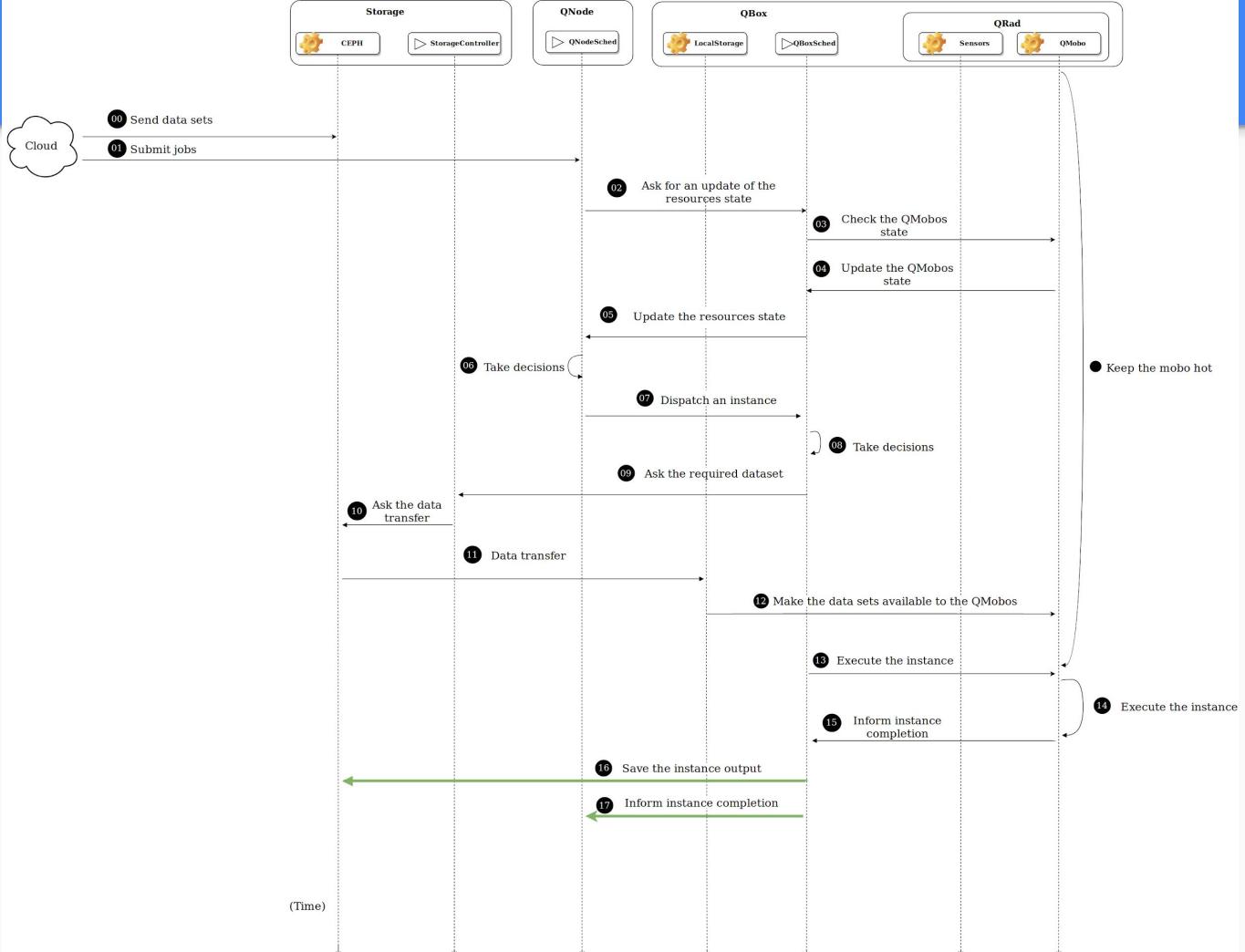


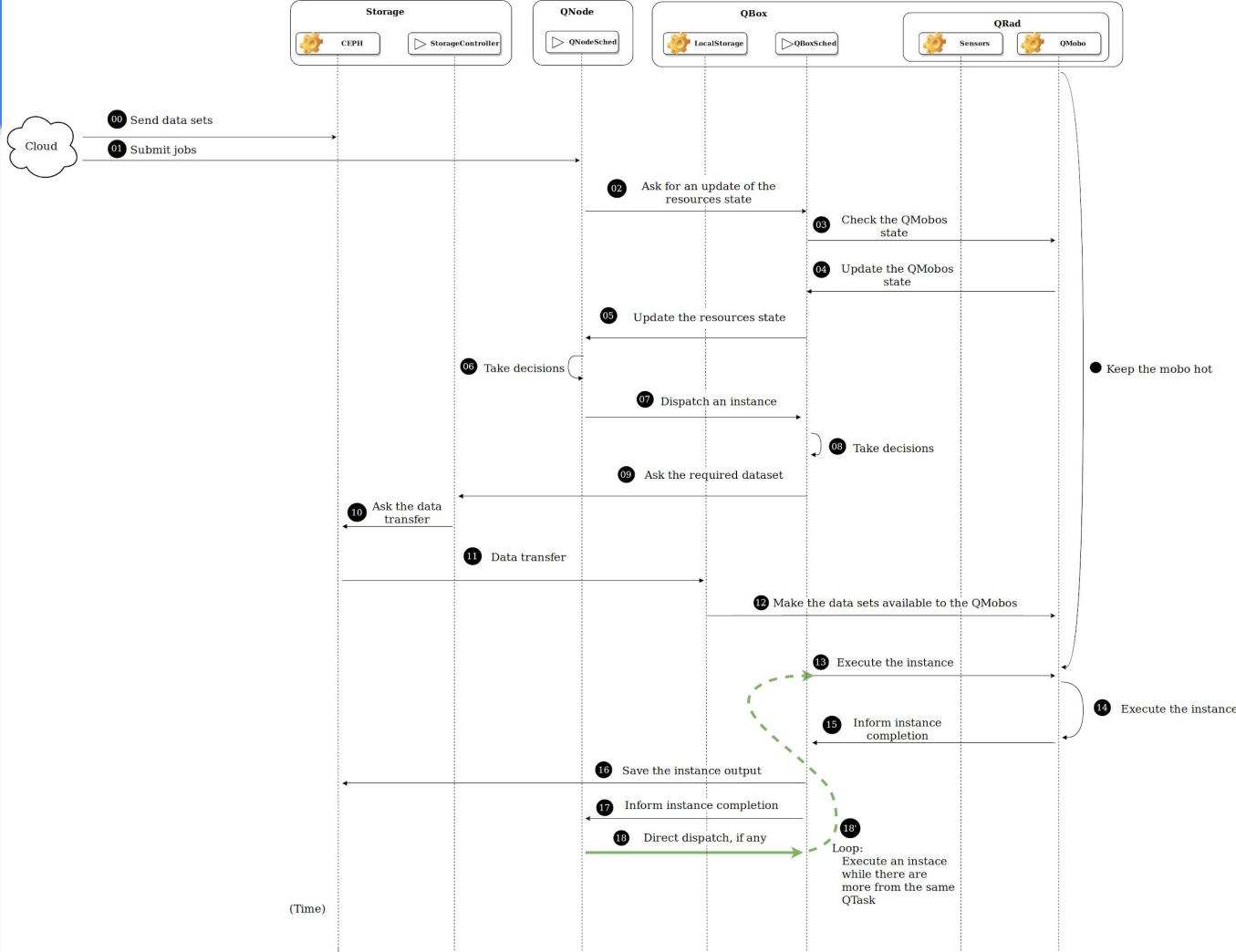


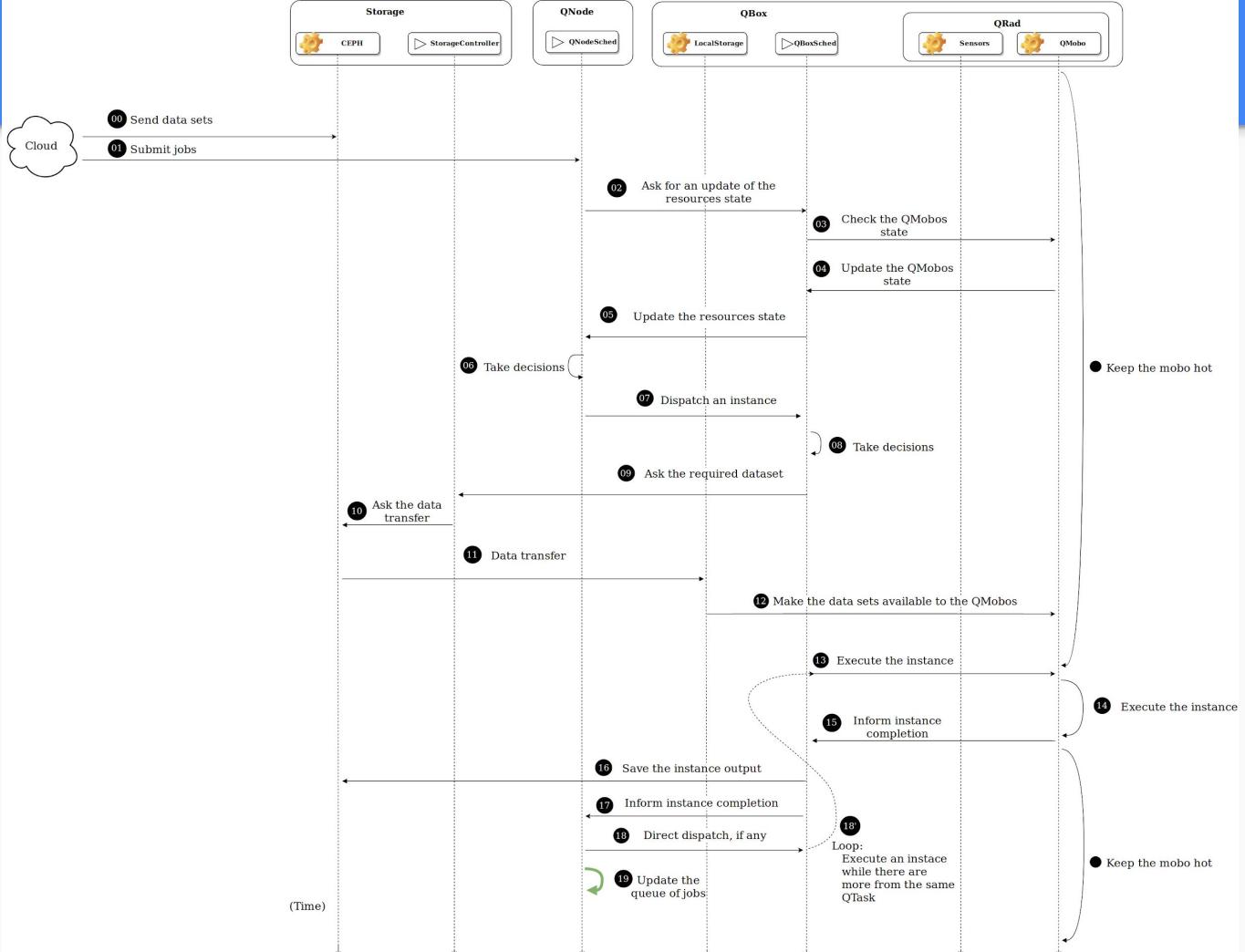






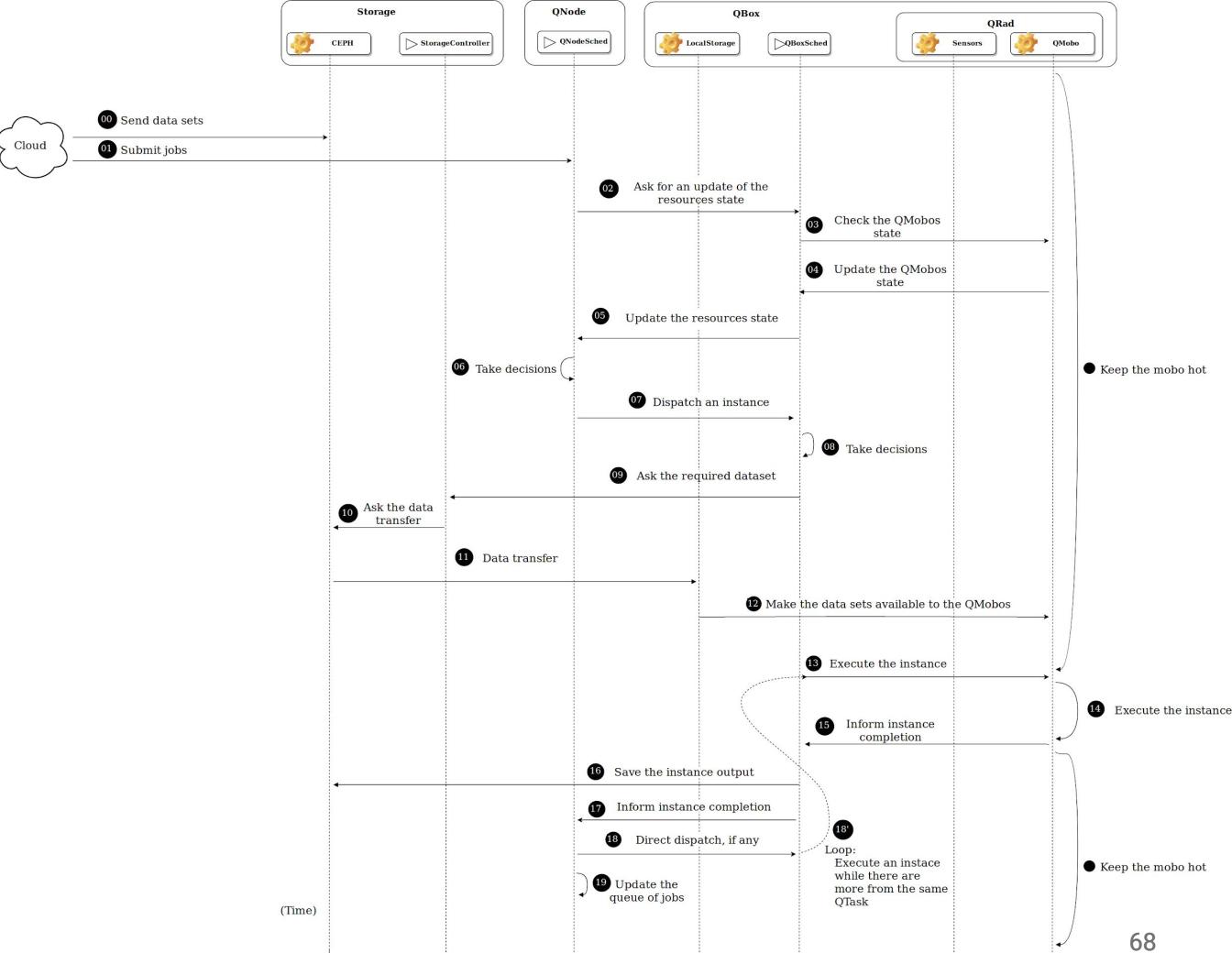






Overview

Based on this platform
we have developed
our simulation.



From the real platform to the simulated one

