



HAL
open science

Using Control Theory to Reduce Disk Congestion Caused by Unpredictable I/O in Cloud Computing

Thomas Collignon, Sophie Cerf, Alexandre van Kempen

► **To cite this version:**

Thomas Collignon, Sophie Cerf, Alexandre van Kempen. Using Control Theory to Reduce Disk Congestion Caused by Unpredictable I/O in Cloud Computing. Performance and Scalability of Storage Systems (Per3S), May 2024, Paris, France. hal-04679808v1

HAL Id: hal-04679808

<https://hal.science/hal-04679808v1>

Submitted on 28 Aug 2024 (v1), last revised 5 Sep 2024 (v2)

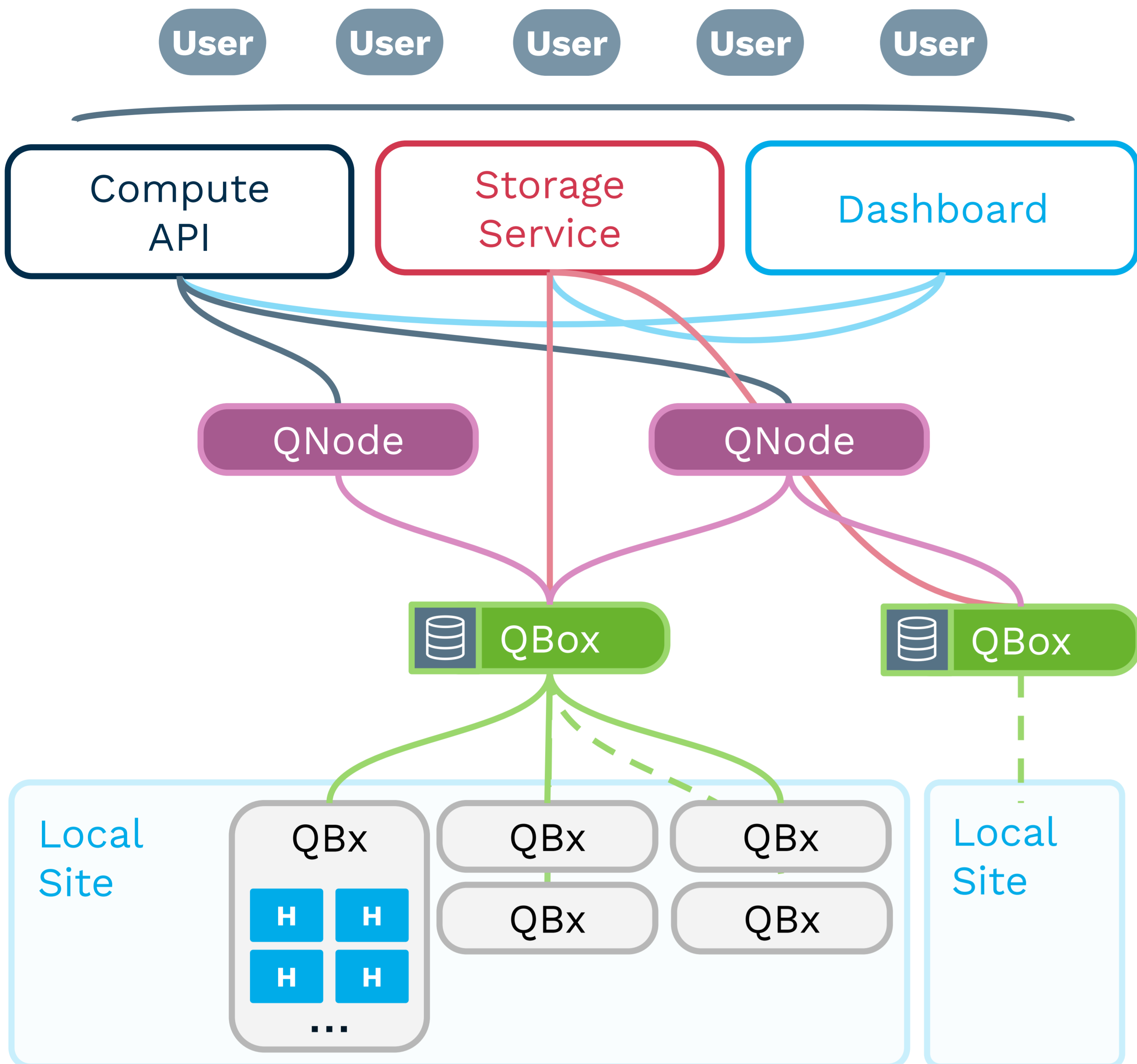
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Using Control Theory to Reduce Disk Congestion Caused by Unpredictable I/O in Cloud Computing

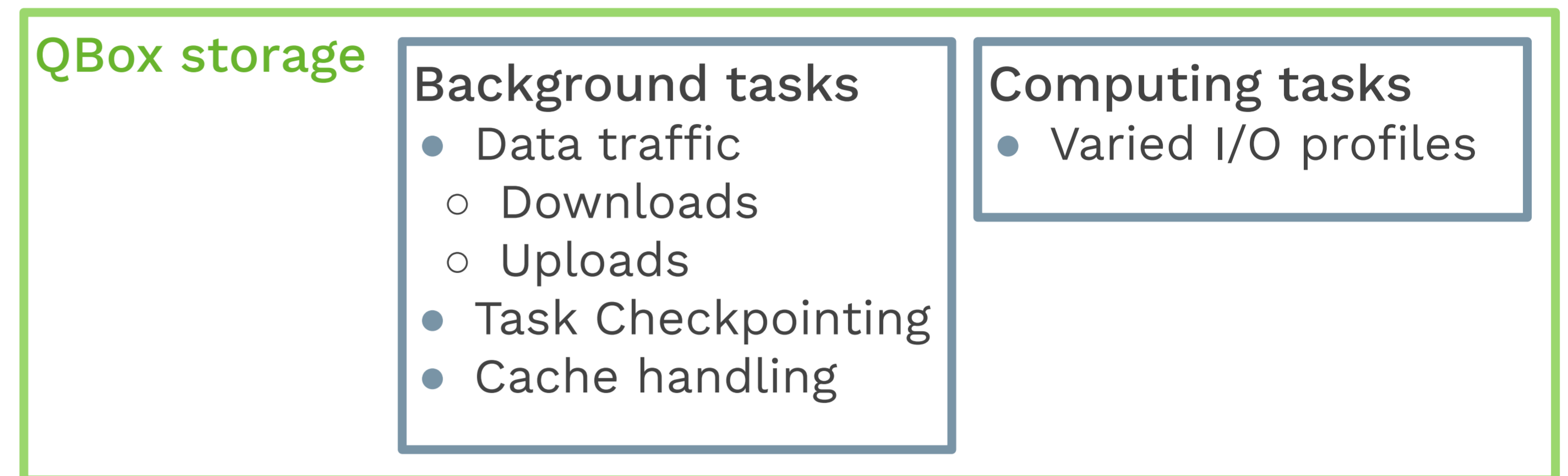
Thomas Collignon, Sophie Cerf, Alexandre Van Kempen

Qarnot infrastructure



Problem : I/O Contention on the QBox

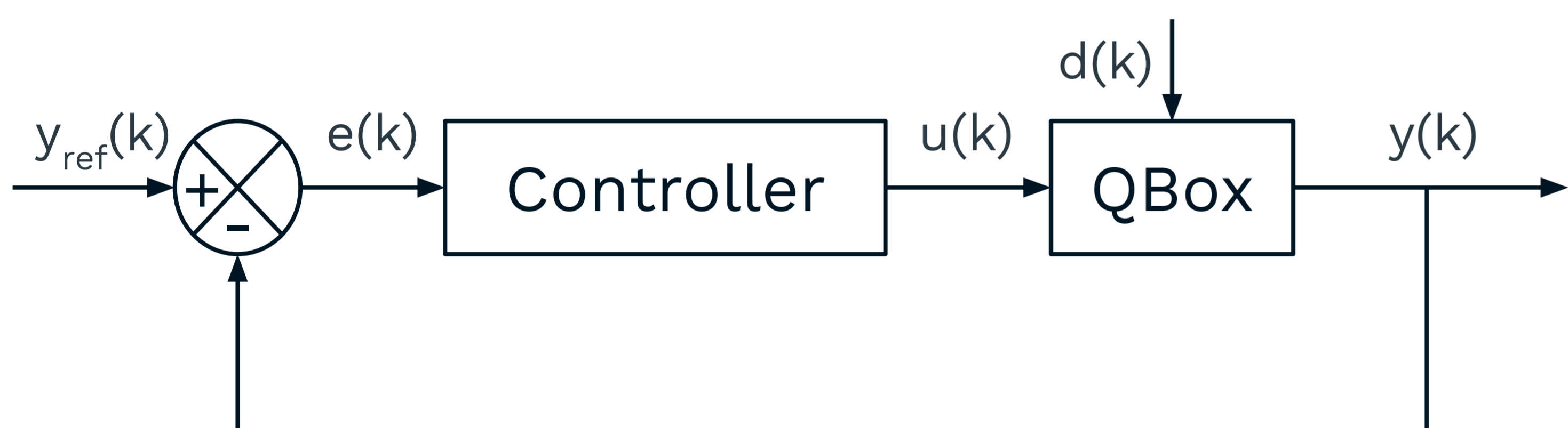
In a Distributed Cloud Computing system, computing nodes often share storage resources at the level of a cluster. Users can submit a wide variety of tasks which can be placed in the same cluster, sharing the same resources. Some computations may require strict I/O while other do not. It is then necessary to avoid any congestion or saturation of the file system. This can be done with an a priori knowledge of the I/O profiles of tasks and usual scheduling methods.



- I/O interferences between Background and Computing tasks
- Inherently hard to predict

Control Theory seems like an appropriate candidate to tackle this issue.

Approach : Control Theory



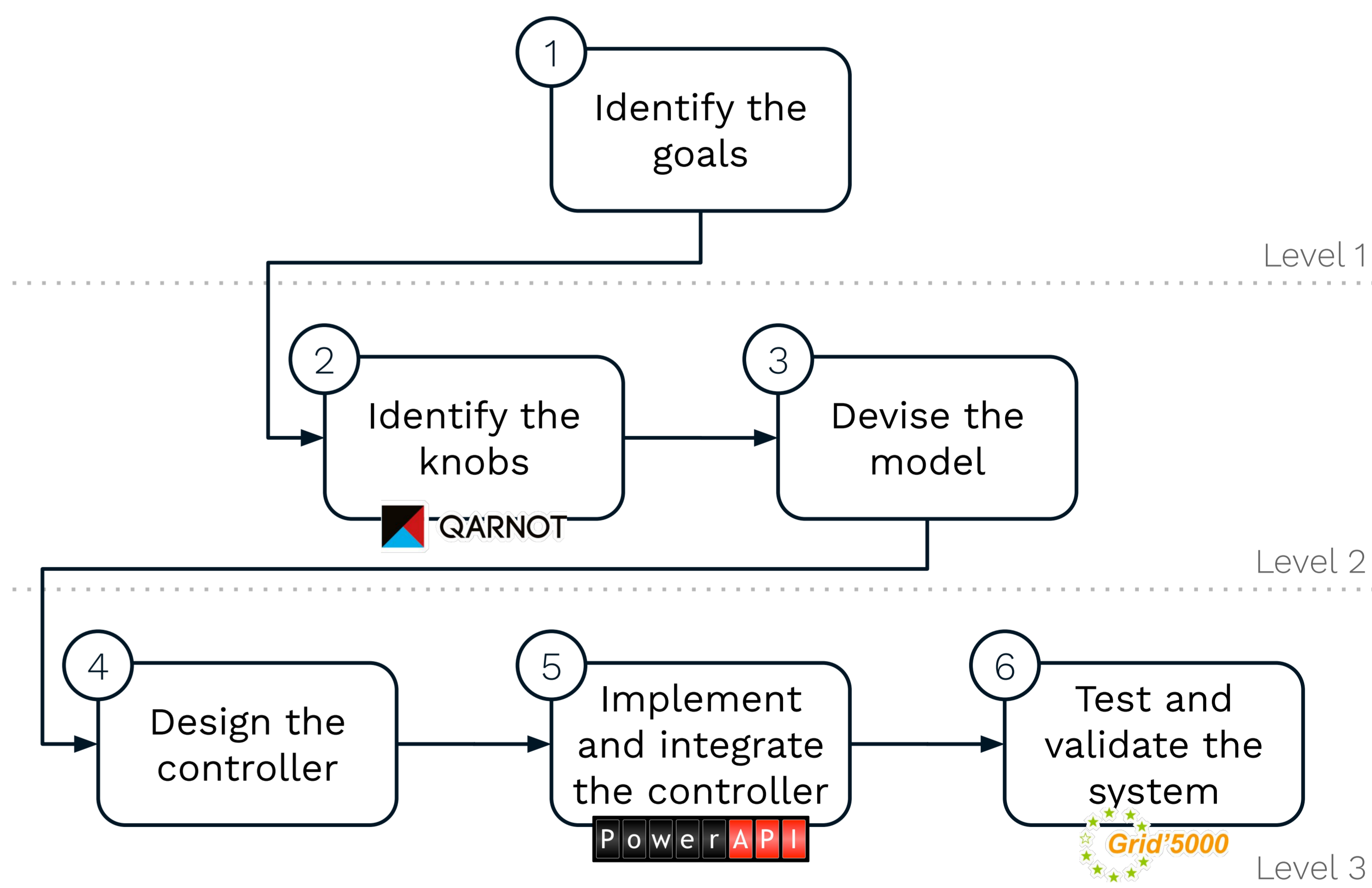
$$\text{Proportional : } u(k+1) = \mathbf{K}_p \times e(k) = K_p \times (y_{ref} - y(k))$$

$$\text{Proportional Integral : } u(k+1) = K_p \times e(k) + \mathbf{K}_i \times \sum_{j=0}^k e(j)$$

$$\text{Proportional Integral : Derivative } u(k+1) = K_p \times e(k) + K_i \times \sum_{j=0}^k e(j) + \mathbf{K}_d \times \frac{e(k) - e(k-1)}{\Delta t}$$

Control Theory consists in adapting dynamically the behavior of the selected system in order to reach the reference value. It requires instrumenting the infrastructure to implement sensors and actuators so that the controller can take decisions and interact with the system regarding the gap between the reference and the current state.

Developing a controller



Step by step methodology to develop and validate a controller.

Filieri, Antonio, et al. "Software engineering meets control theory." SEAMS, 2015.

Multiple metrics and actuators

Download / Upload of task Data

- Metrics
 - connection speed
 - data volume
- Actuators
 - allocated bandwidth
 - delays

Checkpointing

- Metrics
 - checkpointing duration
 - size of the task
- Actuators
 - delay

Cache

- Metrics
 - read/write frequency
 - cache misses
- Actuators
 - delay garbage collection
 - cache strategy

Every task

- Metrics
 - Filesystem metrics (Ceph, etc.)
- Actuators
 - cgroups v2 (CPU, memory, blockIO, etc.)

Lama, Palden, et al. "Performance isolation of data-intensive scale-out applications in a multi-tenant cloud." IPDPS, 2018.

