



# ProFaaSinate: Delaying Serverless Function Calls to Optimize Platform Performance

Trever Schirmer, Valentin Carl, Tobias Pfandzelter, David Bermbach  
TU Berlin | WoSC 2023 | 11. December 23

# Serverless



Just the Code



Serverless Platform



(Infinite) Scalability + Elasticity



Hidden Abstractions

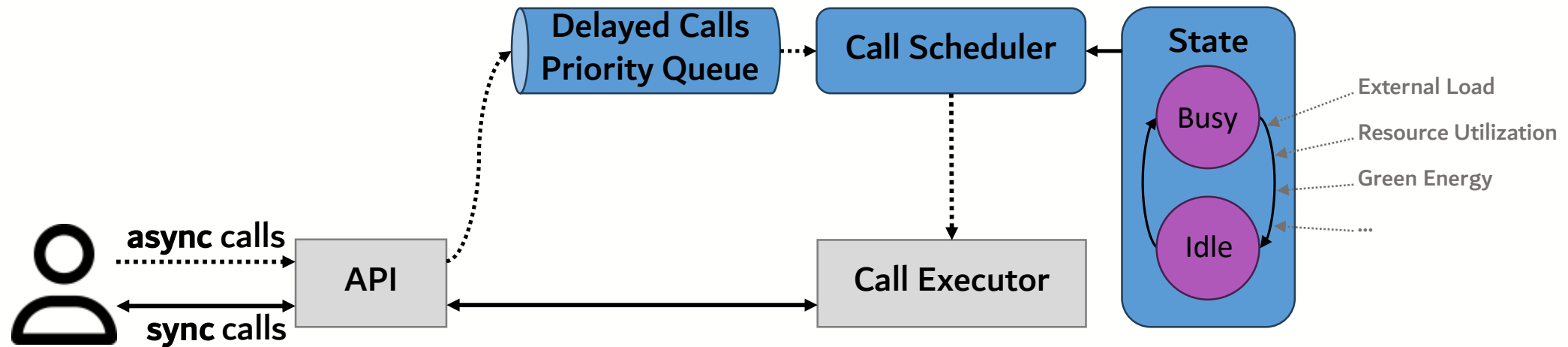


Unreliable Performance

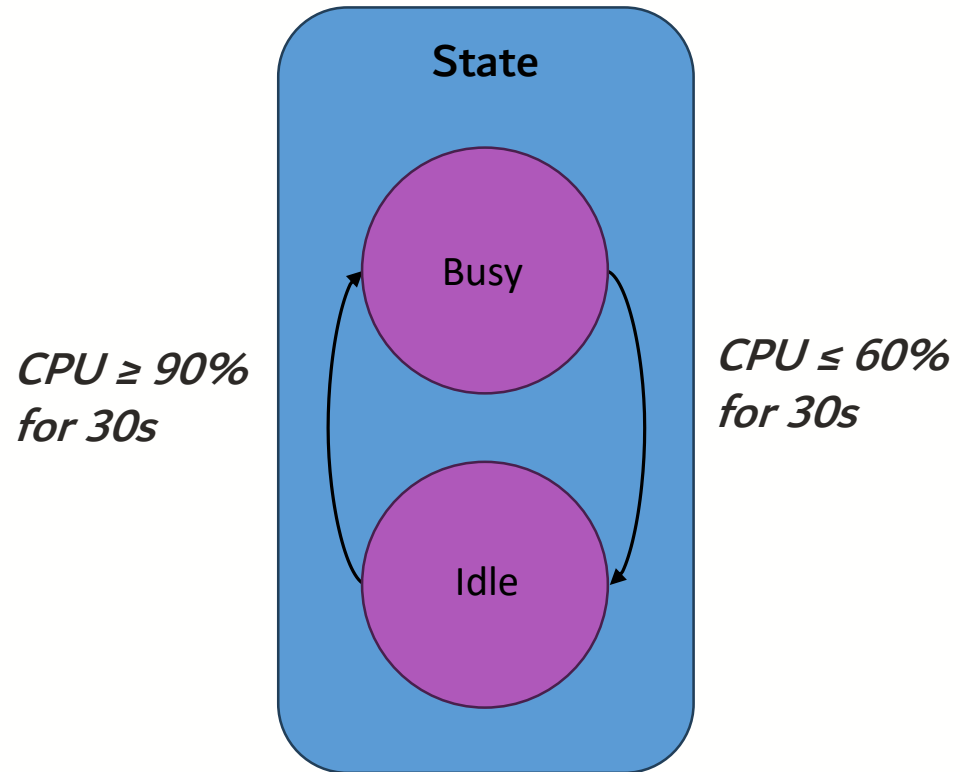


Expensive?

# ProFaaSinate



# Scheduler

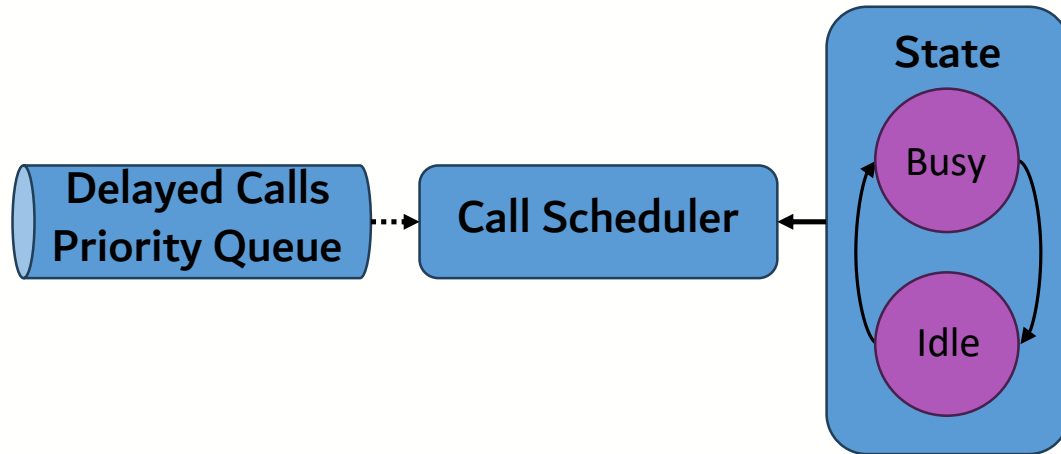


## Feedback Control System

Platform-specific Resource Bottlenecks

Watch out for *hunting oscillation!*

# Queue



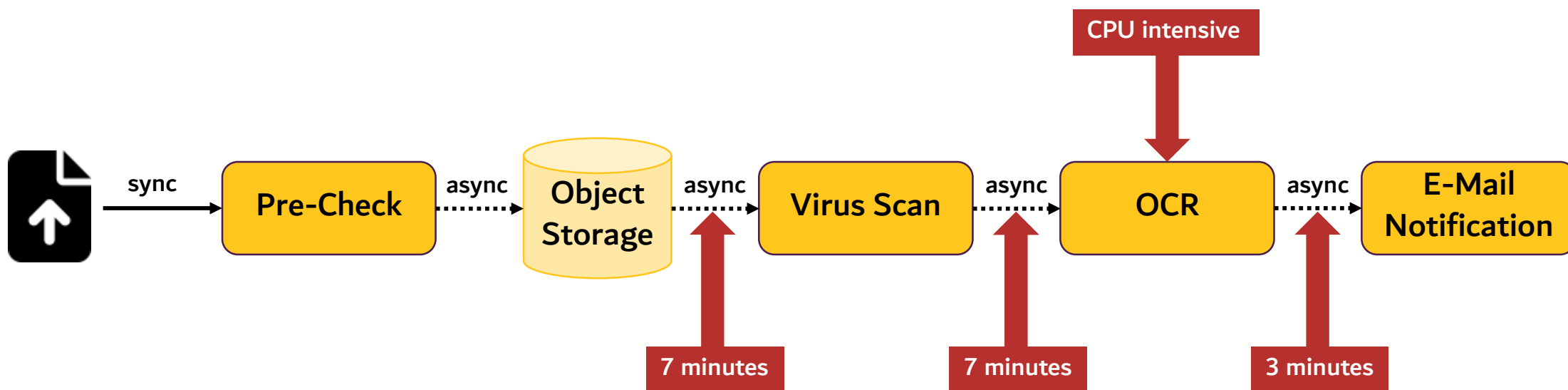
Scheduler takes calls from Queue.

*Busy State*: Only calls that expire soon

*Idle State*: Calls that expire soon, but always at least N

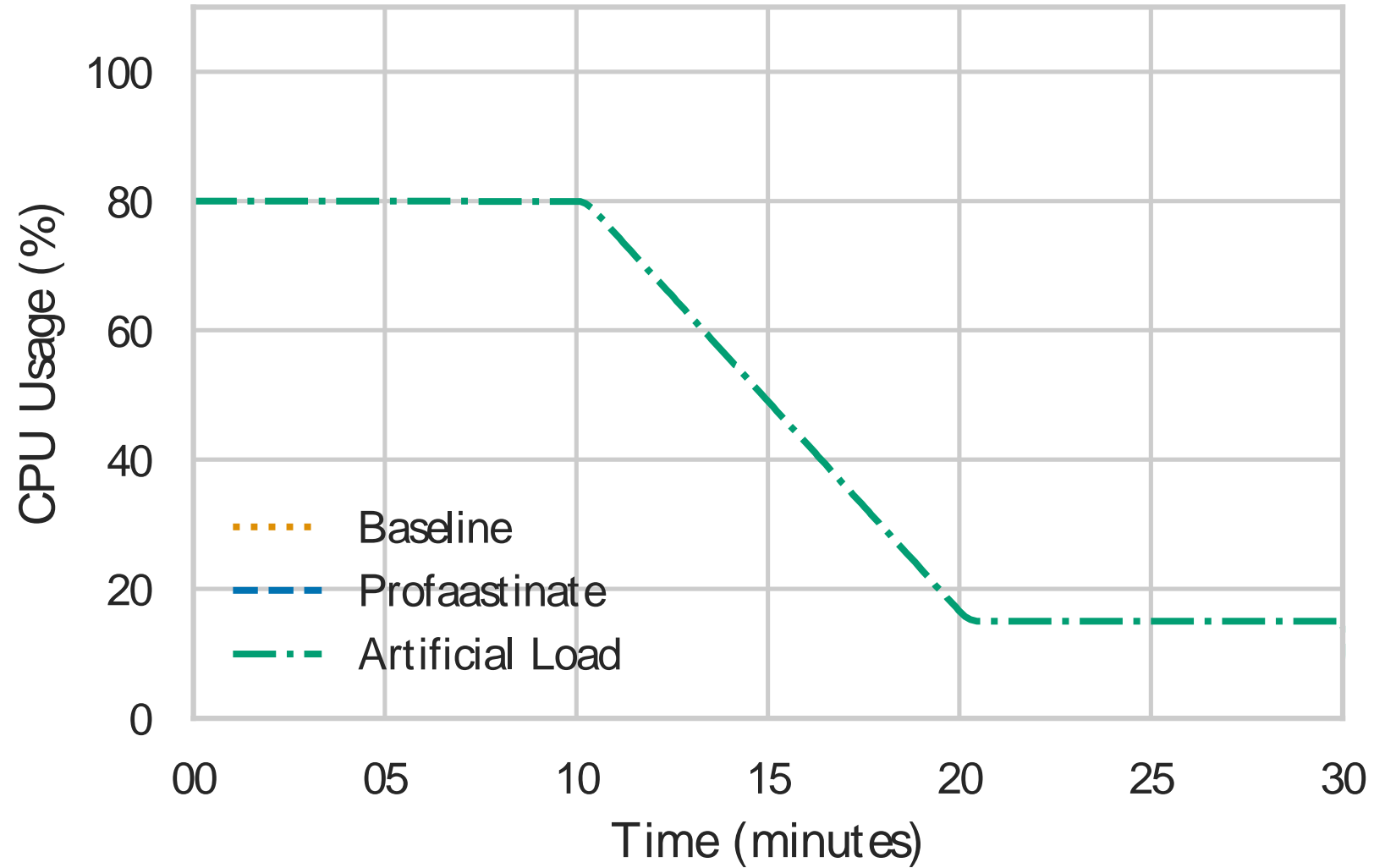
→ Tune based on normal load on platform

# Evaluation: Use Case

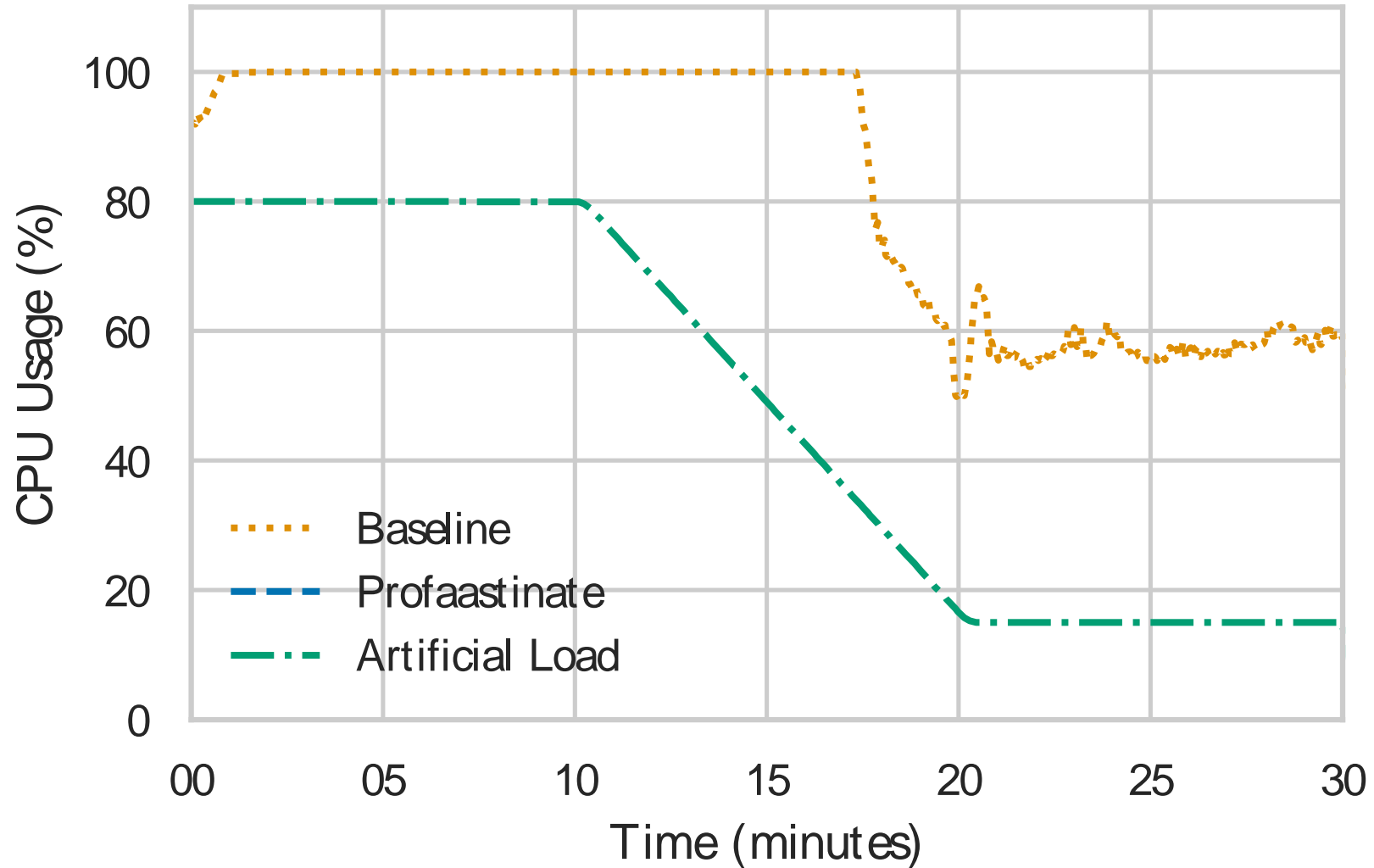


=> 1 req/s for 30m

# Evaluation: CPU Usage

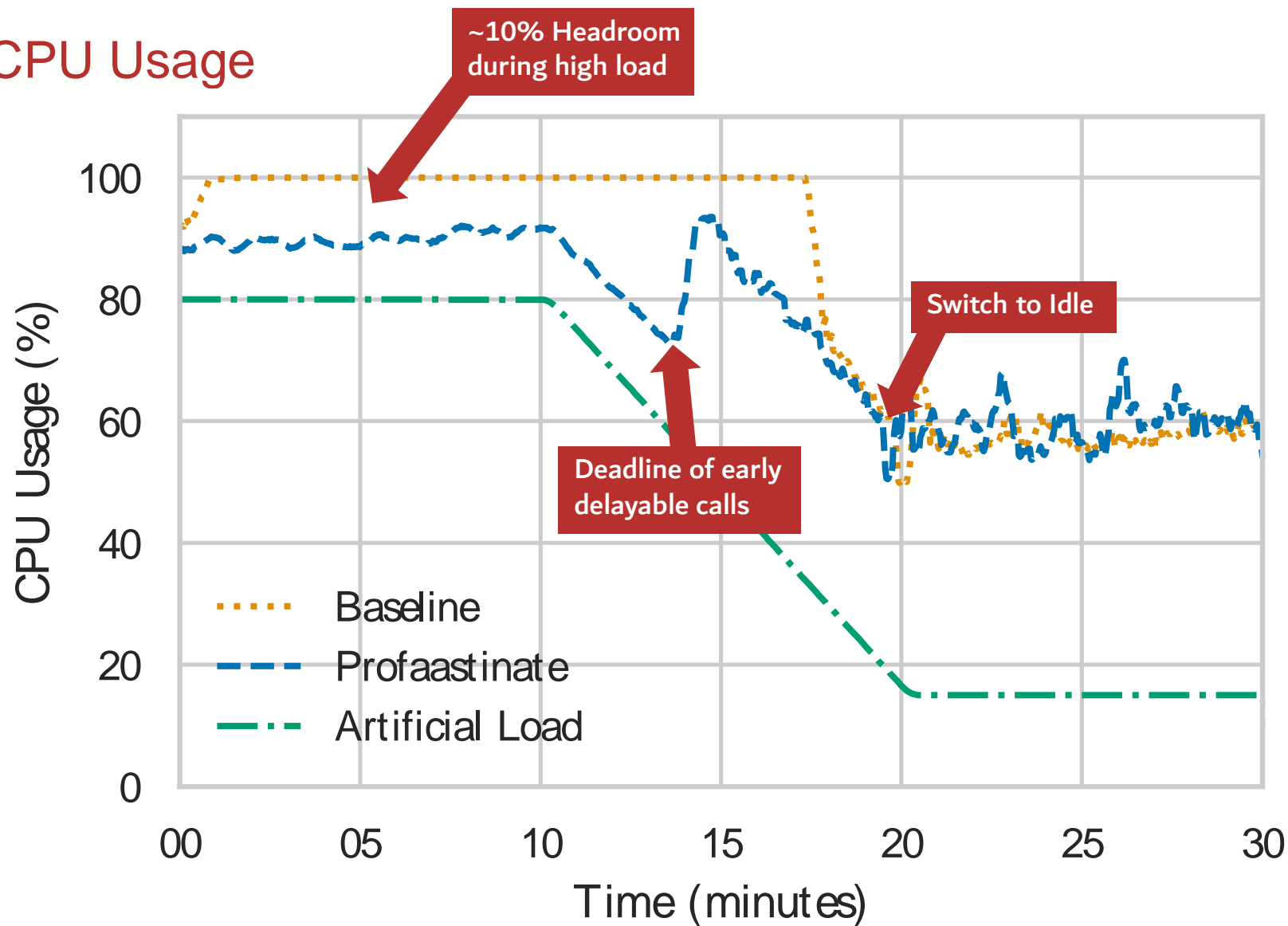


# Evaluation: CPU Usage

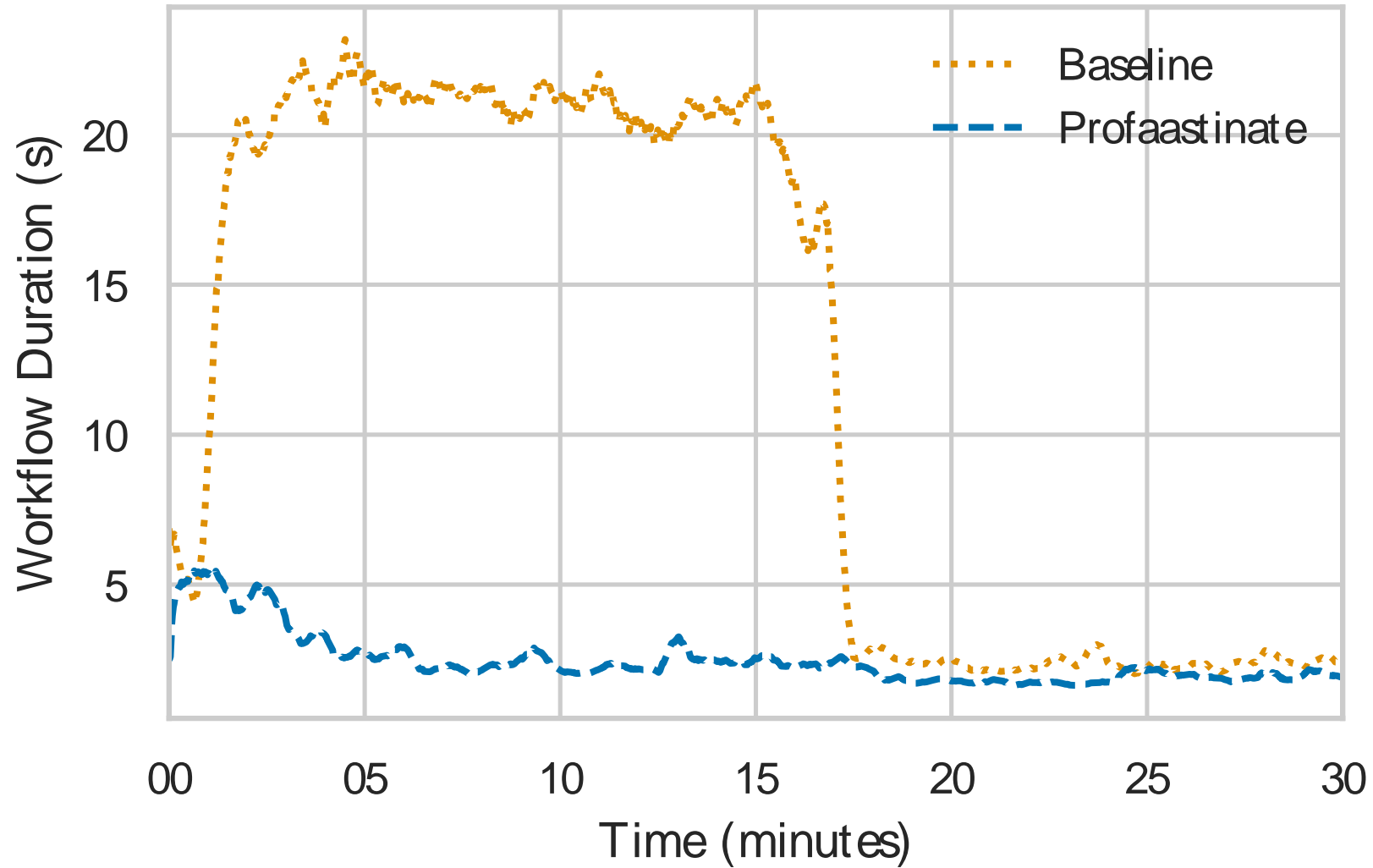




# Evaluation: CPU Usage



# Evaluation: Workflow Duration



# XFaaS: Hybrid Serverless

Alireza Sahraei<sup>+</sup>,  
Neeraj Patel<sup>+</sup>,  
Pradeep Venkat<sup>+</sup>

Shigna Nagaraja<sup>+</sup>,  
Andrii Golovei<sup>+</sup>,  
Ravinder Thind<sup>+</sup>,

<sup>+</sup> Meta Platforms, <sup>+</sup> IBM Research

<sup>+</sup> Carnegie Mellon University

## Abstract

Function-as-a-Service (FaaS) has become a key component of Serverless Computing. The ability to provision and manage functions on-demand shifts the focus from hardware provisioning to software provisioning. This report on how FaaS provides a challenge and the level of hardware utilization.

Andrii Golovei<sup>+</sup>, Pradeep Venkat<sup>+</sup>,  
Skarlatos Katsaros<sup>+</sup>, Vipul Patel<sup>+</sup>, Ravinder  
Thind<sup>+</sup>, An Jin<sup>+</sup>, and Chunqiang Tang<sup>+</sup>. 2023.  
Cost Serverless Functions at Meta .  
*Journal of the ACM on Operating Systems Principles*  
18, Koblenz, Germany. ACM, New York,  
[doi.org/10.1145/3600006.3613155](https://doi.org/10.1145/3600006.3613155)

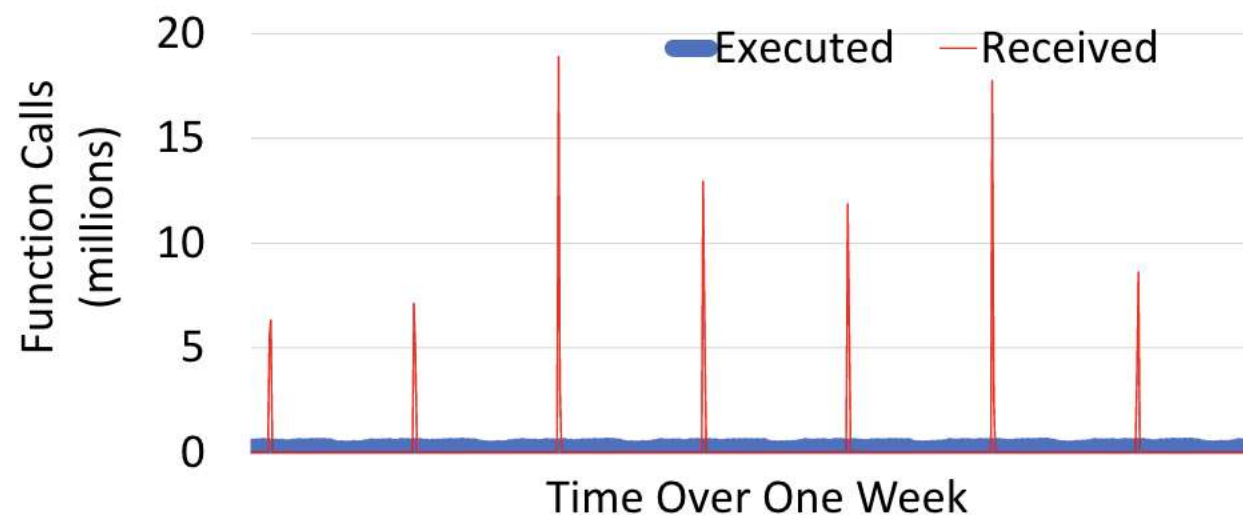
## XFaaS: Time-Shift

**Solution for high variance of load.** To reduce hardware costs, we intentionally provision XFaaS with hardware that is insufficient for its peak demand and then leverage several techniques to manage overload situations. First, it employs time-shift computing to postpone the execution of certain functions. Each function has a criticality and a deadline, with the deadline ranging from seconds to 24 hours. When XFaaS reaches its capacity limit, delay-tolerant functions are deferred to off-peak hours for execution. If the capacity is still

## XFaaS: Evaluation

At Meta, we operate a hyperscale private cloud that includes a FaaS platform called *XFaaS*. XFaaS processes **trillions of function calls per day on more than 100,000 servers spread across tens of datacenter regions.** Due to the hyperscale of

## XFaaS: Spiky Functions



**Figure 4.** The **load of a spiky function**. This function allows its function calls to be executed with a 24-hour SLO. XFaaS leverages this property to spread out its function execution.

## Conclusion

### ProFaaSinate

Some serverless function calls can be delayed.

This can be used to “shave-off” load peaks.

Our scheduler uses two states: Idle and Busy.

*Future Work:* Scheduling Complexity ( $\simeq$  Queue Complexity)

### Contact

 [ts@mcc.tu-berlin.de](mailto:ts@mcc.tu-berlin.de)

 [tu.berlin/mcc](https://tu.berlin/mcc)

## Sources

Title Photo by [Paulo Resende](#) on [Unsplash](#)

Ottifant: <https://www.tagblatt.ch/kultur/otto-waalkes-staaark-mein-ottifant-steht-jetzt-im-duden-schon-wieder-ein-karriere-highlight-fuer-otto-waalkes-ld.2350449>