Batsim Returns?

Millian Poquet
Univ. Toulouse III, IRIT, Sepia team

2024-05-16
New Challenges in Scheduling Theory, Aussois 2024

Acknowledgments

- GIS neOCampus of Université Toulouse III Paul Sabatier
- This work was carried out within the MaaS action of the VILAGIL project, a project co-financed by Toulouse Métropole and France 2030 as part of the Territoires d'innovation program operated by the Banque des territoires

Study of distributed systems and applications

Various platforms: **HPC**, clouds, decentralized clouds...

Complex issues: **energy**, resilience, scalability, interferences, heterogeneity...

Theoretical approaches limitations

- Most used models ignore complex issues
- Approx ratio of $\frac{3}{2} - 10^{-36}$ is cool... but on usual inputs,
  is it better than the simple/stupid algorithm used in production?

Study of distributed systems and applications

Various platforms: **HPC**, clouds, decentralized clouds...

Complex issues: **energy**, resilience, scalability, interferences, heterogeneity...

Theoretical approaches limitations

- Most used models ignore complex issues
- Approx ratio of $\frac{3}{2} - 10^{-36}$ is cool... but on usual inputs,
  is it better than the simple/stupid algorithm used in production?

Experimental methodological approaches

- Direct experimentation: real apps on real platform
- **Simulation**: app models on platform models
- Something in between: emulation, partial simulation...

Building simulators from scratch is risky

How useful is a simulator whose results cannot be trusted?

- Models validated?
- Implementation tested?
- Model instantiation evaluated?

Doing it thoroughly may take (dozens of) years!

Building simulators from scratch is risky

How useful is a simulator whose results cannot be trusted?

- Models validated?
- Implementation tested?
- Model instantiation evaluated?

Doing it thoroughly may take (dozens of) years!

Using SimGrid (or any already well-evaluated simulation framework) helps a lot

- Sane model validation methodology for > 20 years
- Thoroughly tested Implementation
- Model instantiation responsibility is still on you

Building simulators from scratch is risky

How useful is a simulator whose results cannot be trusted?

- Models validated?
- Implementation tested?
- Model instantiation evaluated?

Doing it thoroughly may take (dozens of) years!

Using SimGrid (or any already well-evaluated simulation framework) helps a lot

- Sane model validation methodology for > 20 years
- Thoroughly tested Implementation
- Model instantiation responsibility is still on you

How to make SimGrid easily usable for experimentation on scheduling?

Outline

1. Batsim overview
2. Batsim 5
3. Batsim future directions

Batsim overview

Design rationale

"Simulator in *lang A* 🤮 ⟶ I'll reimplement it in *lang B*!"

- Enable decision making code **in any language**
- Decouple decision making code from simulator

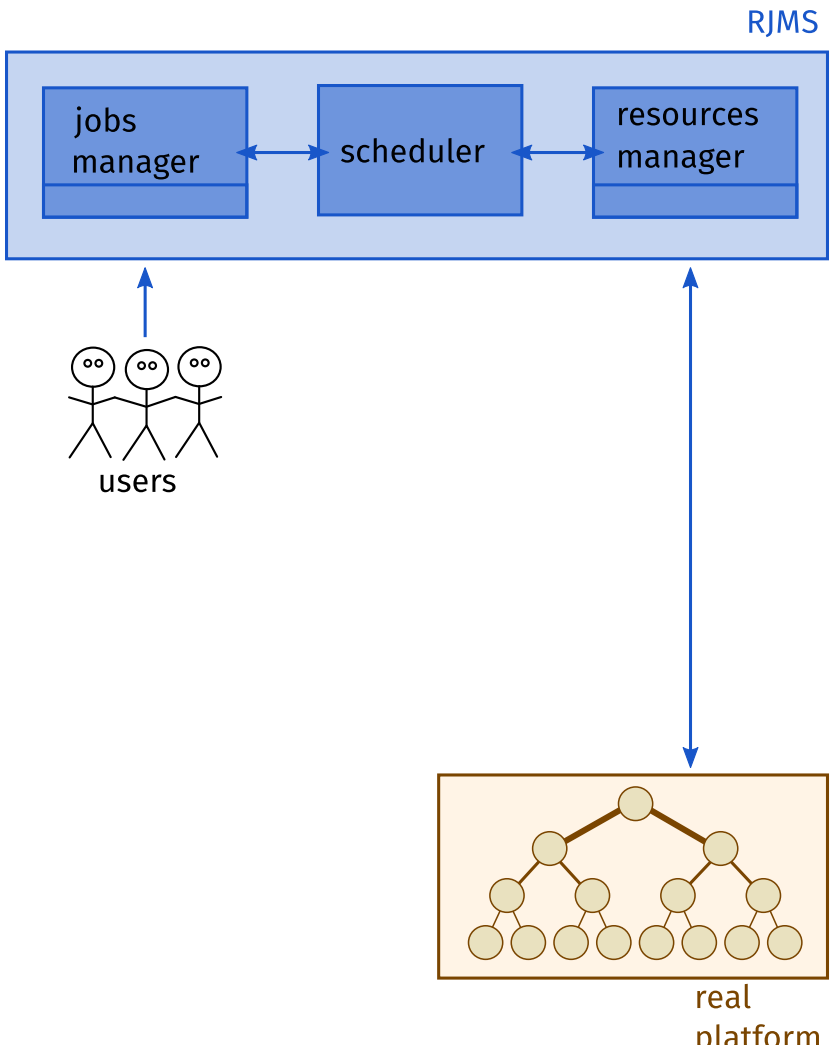
Study research prototypes or production codes

- **Strongly decouple** decision making code from simulator

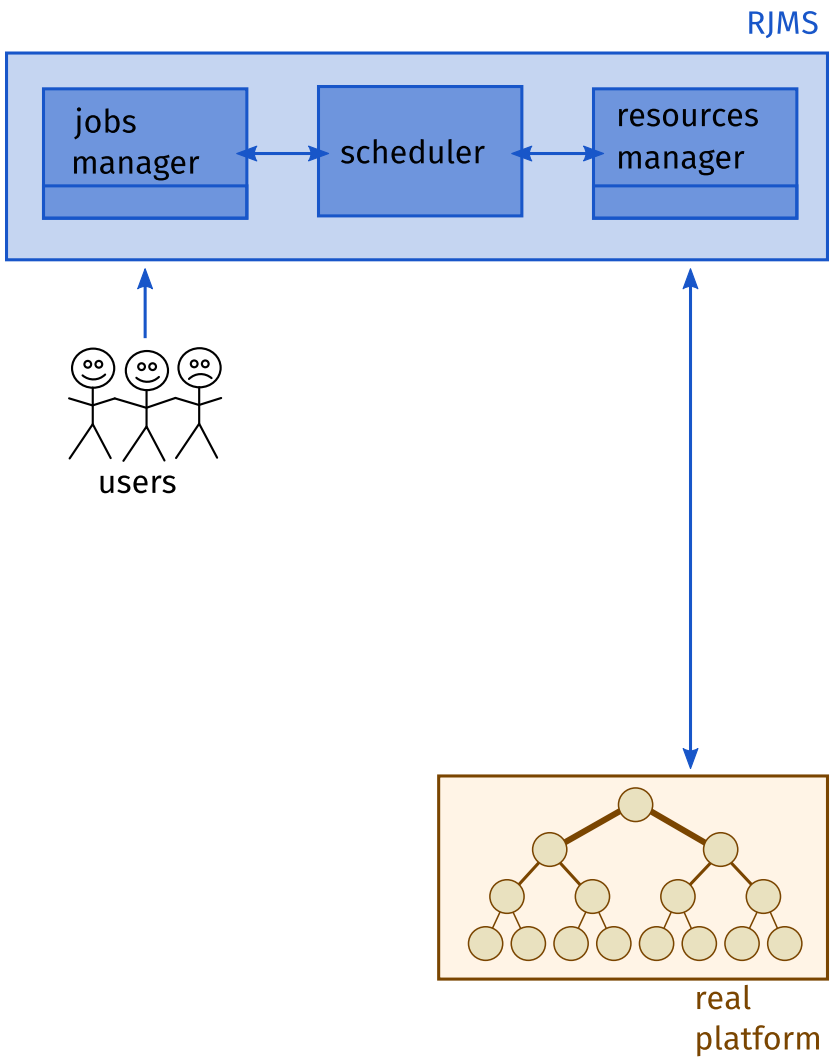
Reproducibility & experimental workflows

- Avoid user interaction during the simulation
- Read input data
- Generate output **data** (not plots)

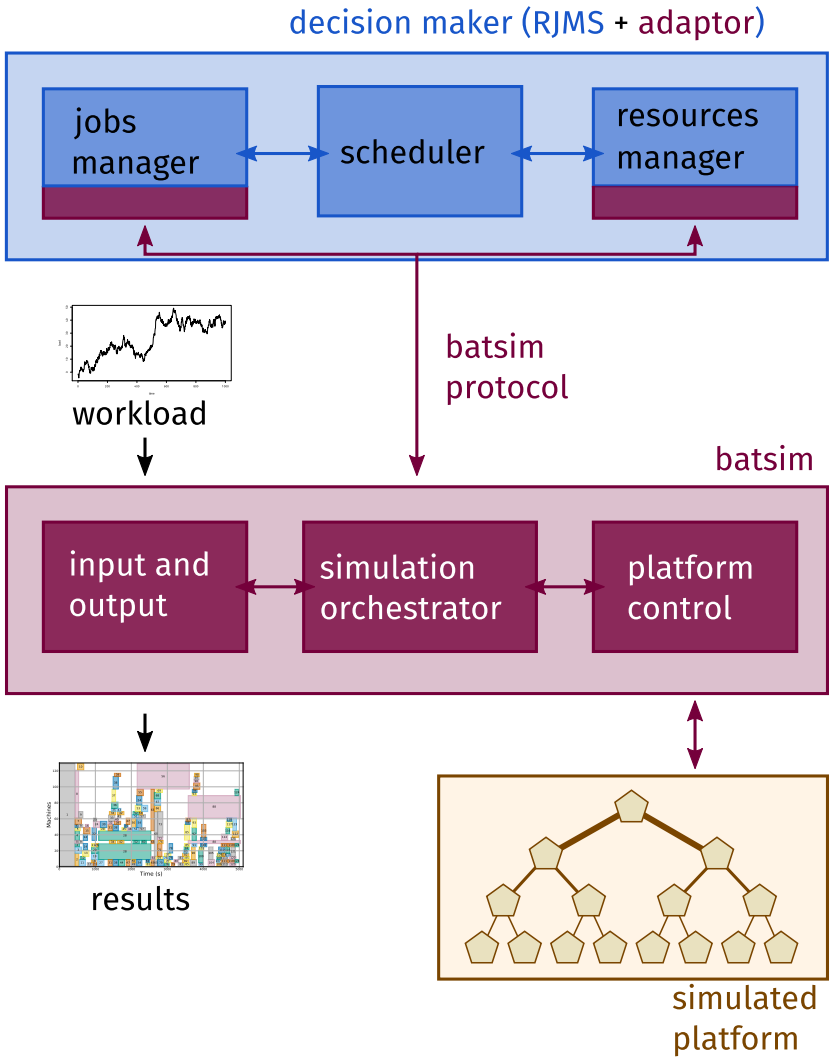
Real


RJMS
jobs manager
scheduler
resources manager
users
real platform

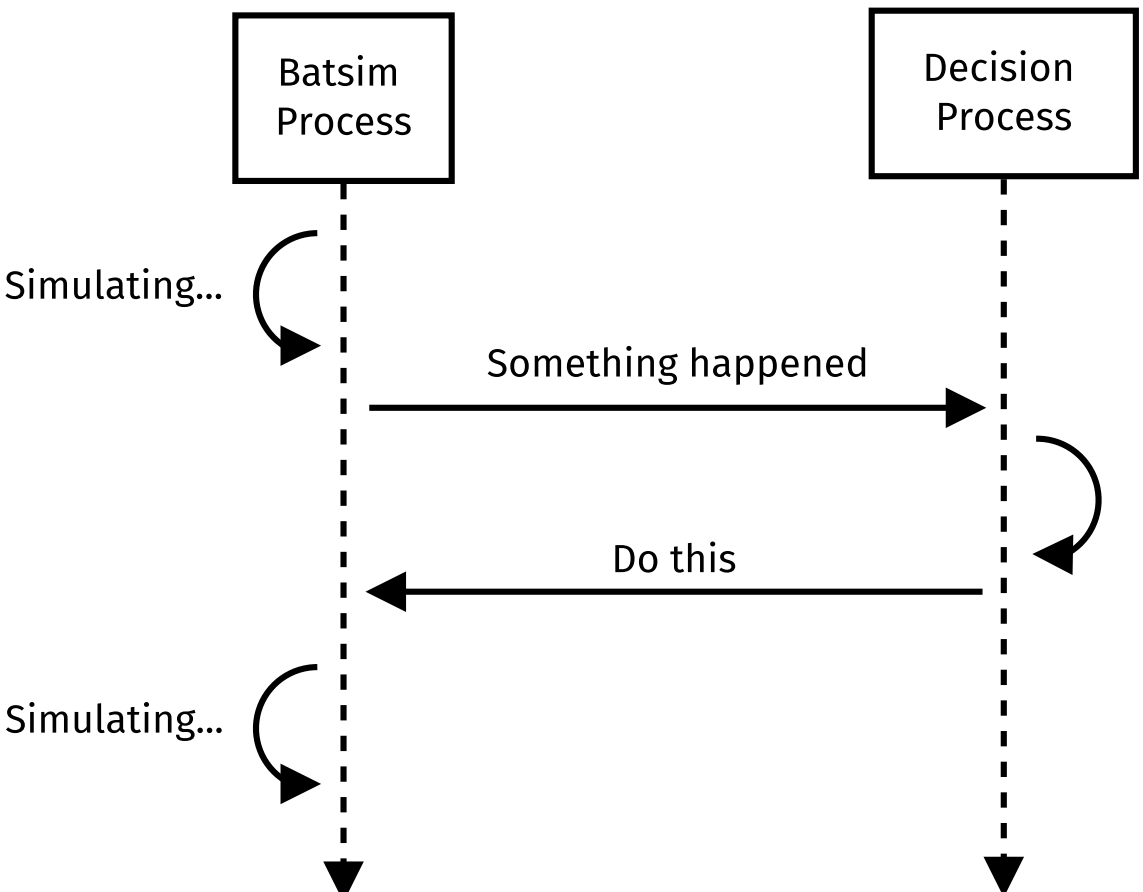
Real



Batsim simulation



Protocol



Classical scheduling events

- Job submitted
- Job finished

Resource management decisions

- Execute job j on $M = \{1, 2\}$
- Shutdown $M = \{3, ..., 5\}$

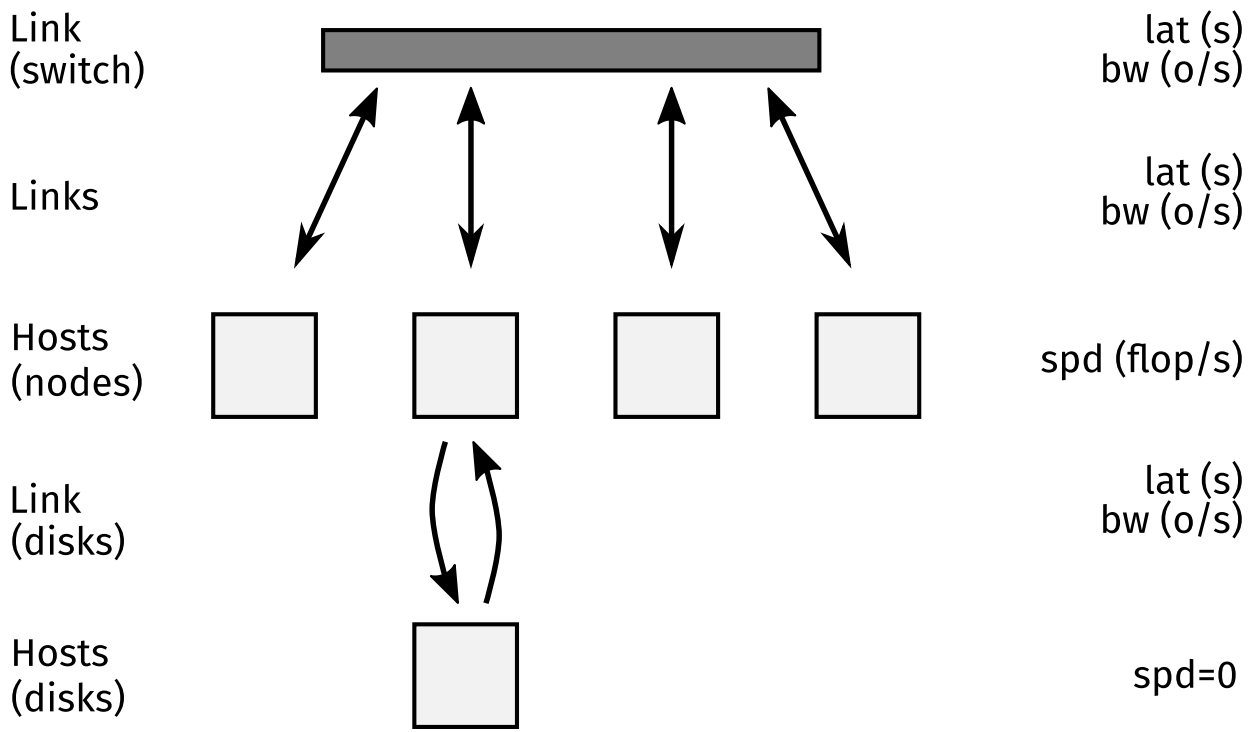
Simulation/monitoring control

- Call scheduler at $t = 120$
- How much energy used?

Platform: 2 resource types

Computation **host**
- speed (flop/s)
- power consumption
  linear to usage for each pstate

Network **link**
- bandwidth (o/s)
- latency (s)
- power consumption



Workload: jobs and profiles

Jobs: scheduler view

- User resource request
- (Walltime)
- Simulation profile

Profiles: simulator view

- How to simulate the app?

Profile types

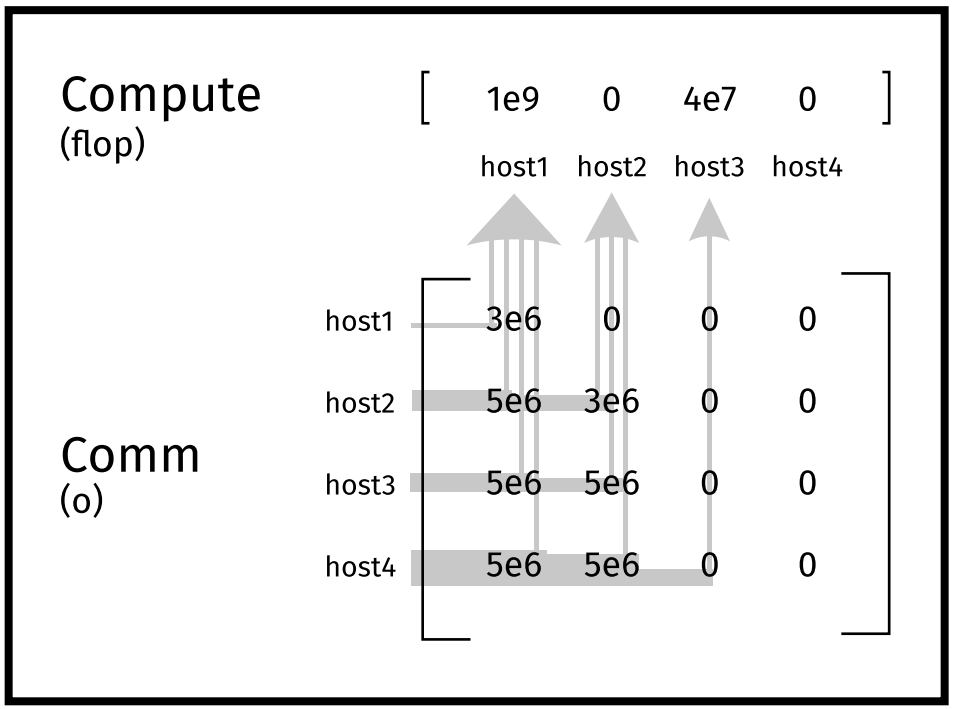
- Fixed length
- Trace replay (MPI, usage…)

Workload: jobs and profiles

Jobs: scheduler view
- User resource request
- (Walltime)
- Simulation profile

Profiles: simulator view
- How to simulate the app?

Profile types
- Fixed length
- Trace replay (MPI, usage...)
- Parallel task

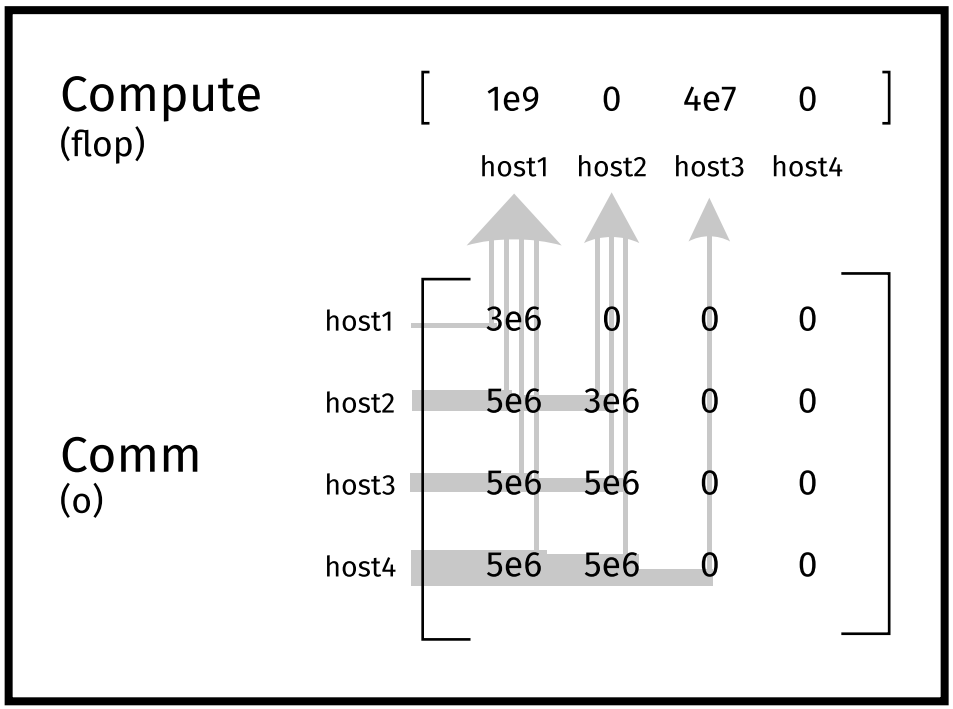


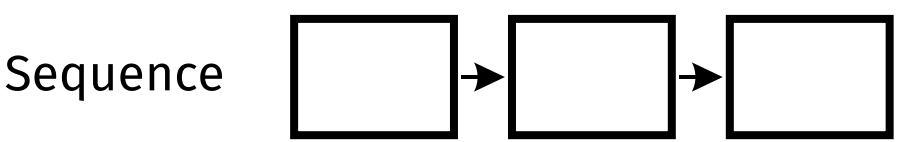
Workload: jobs and profiles

Jobs: scheduler view
- User resource request
- (Walltime)
- Simulation profile

Profiles: simulator view
- How to simulate the app?

Profile types
- Fixed length
- Trace replay (MPI, usage...)
- Parallel task
- Composition
  - Sequence

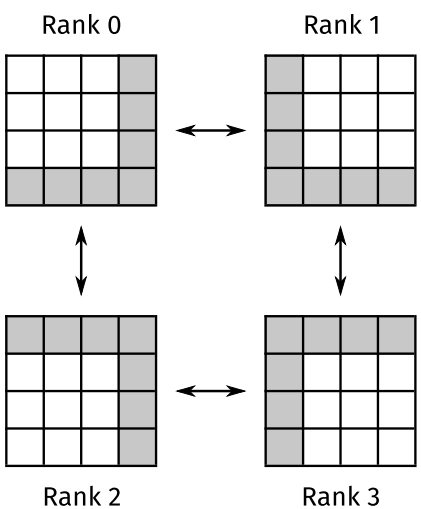




Application modeling example: 2D Stencil with checkpoints

Application behavior

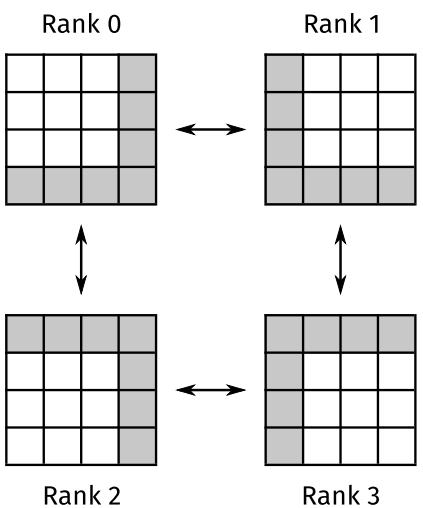
- Begin: load data from PFS
- Do 1000 iterations:
  1. local computations
  2. exchange data with neighbors
- Every 100 iterations do checkpoint on PFS



Application modeling example: 2D Stencil with checkpoints

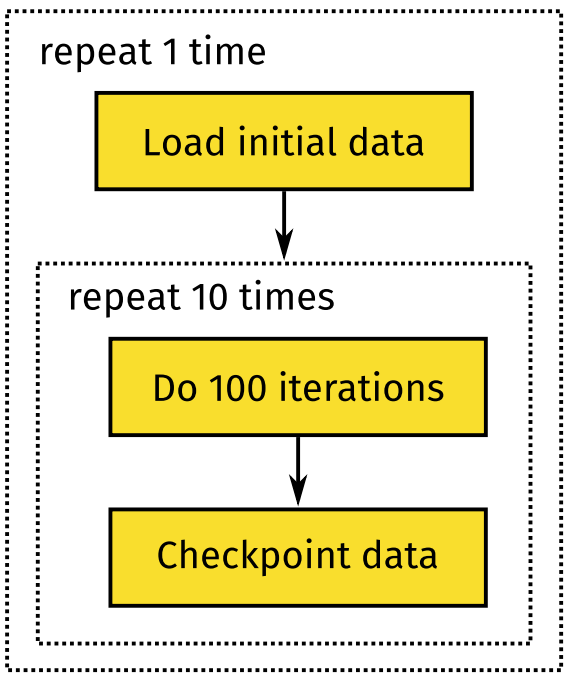
Application behavior

- Begin: load data from PFS
- Do 1000 iterations:
  1. local computations
  2. exchange data with neighbors
- Every 100 iterations do checkpoint on PFS



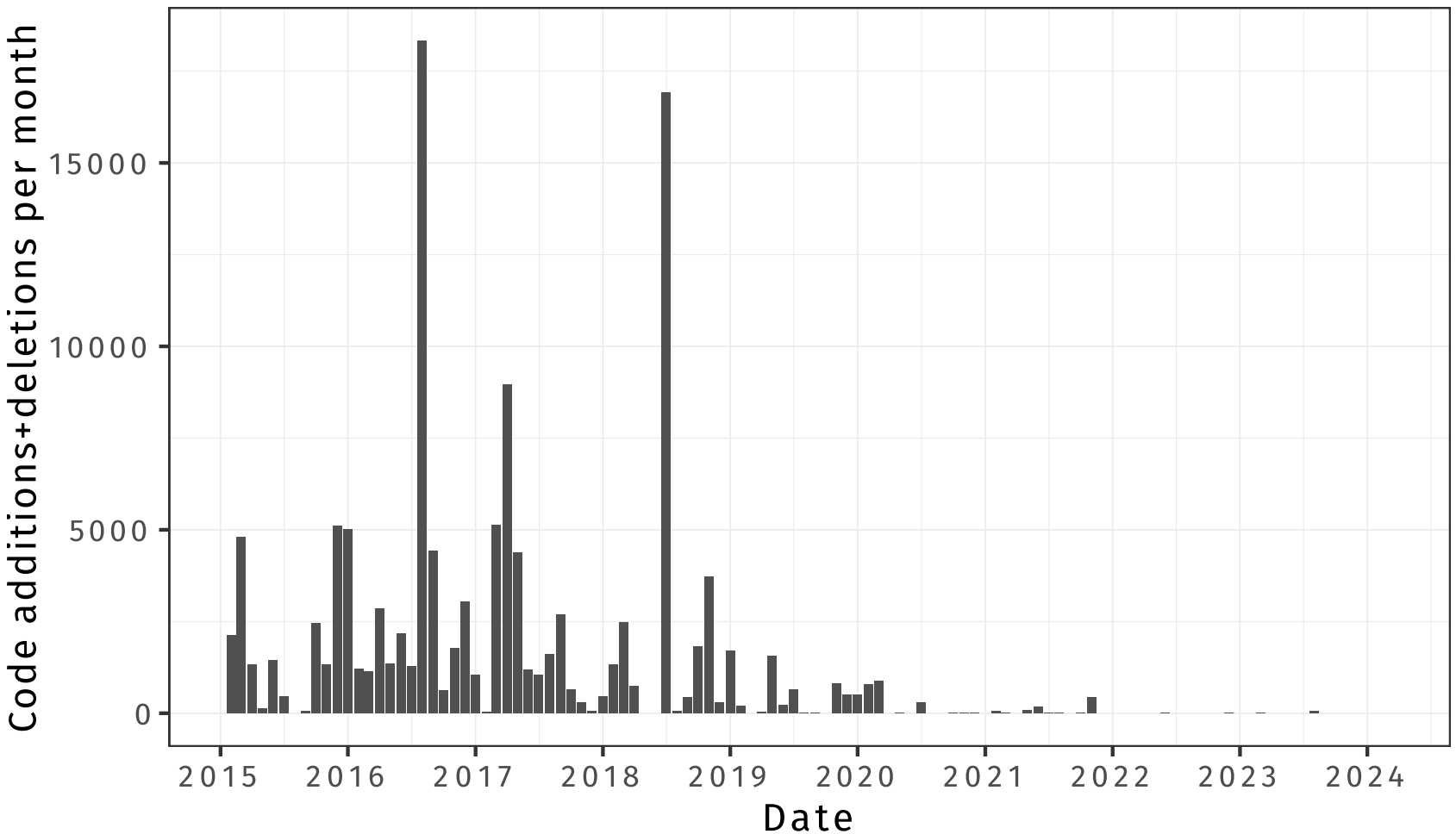
Profile example using

- 3 parallel tasks
- 2 sequences

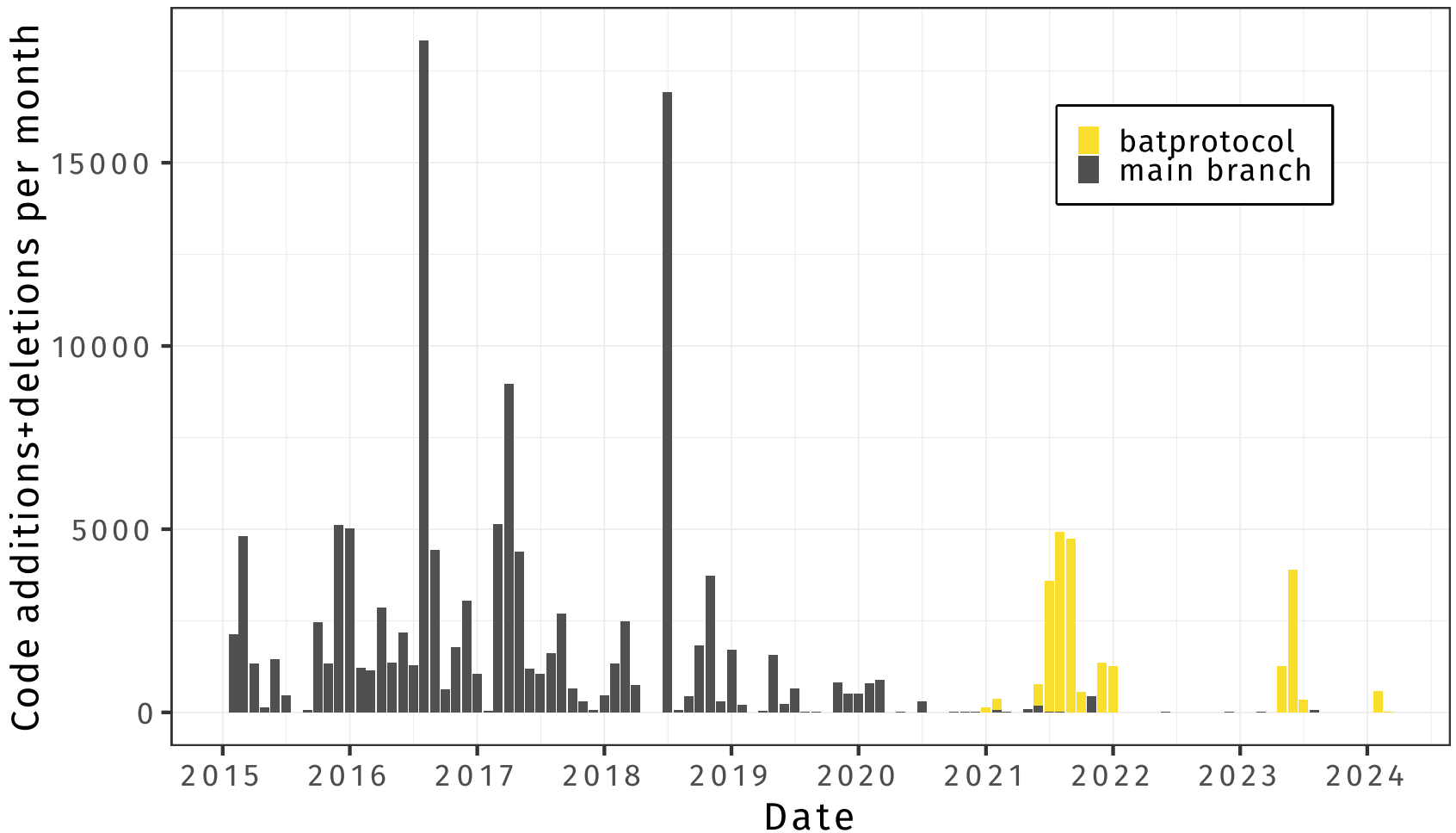


Batsim 5

Development timeline



Development timeline



Motivations for big changes

Maintainability

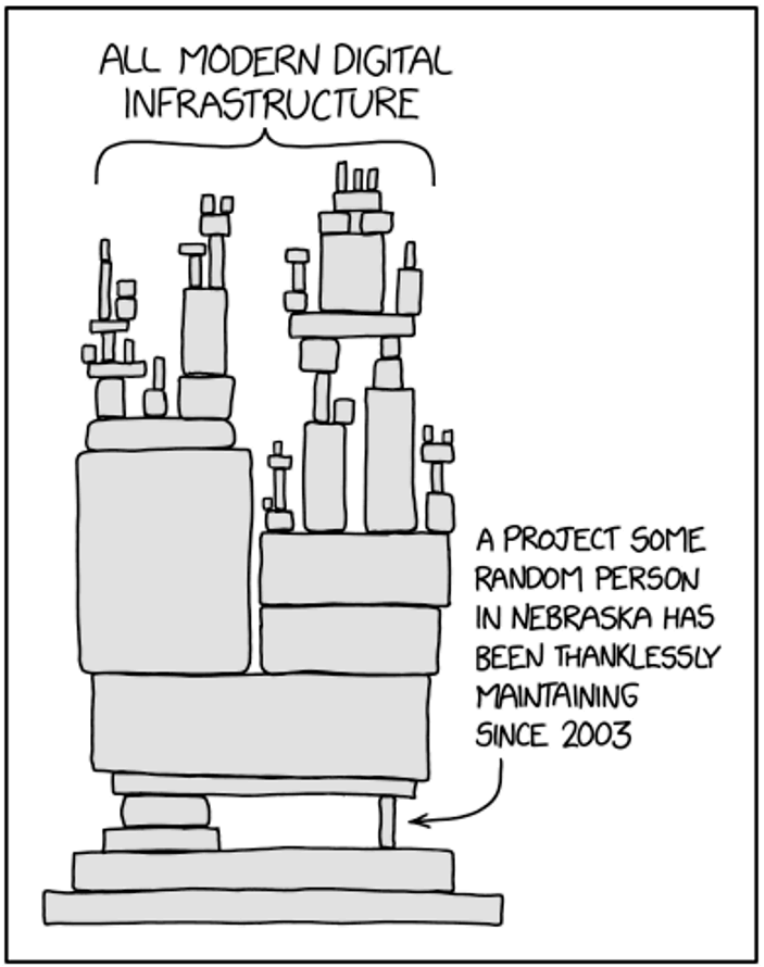
- Bloat: protocol copypasta, not so KISS…
- Distributed projects sync
- Test dependency cycles

Performance

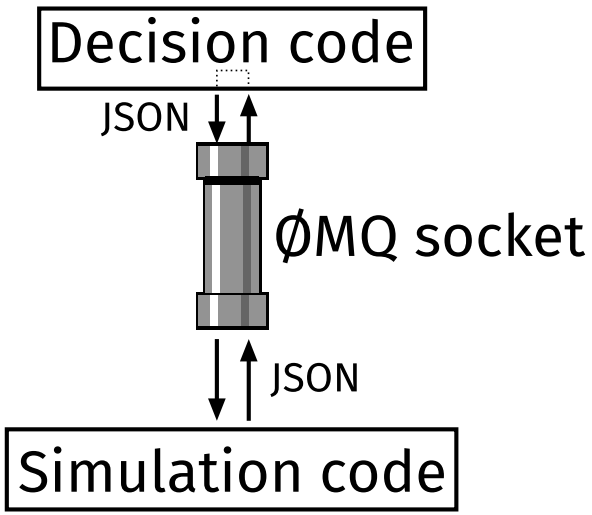
- De/serialization overhead
- Syscall party

Usability

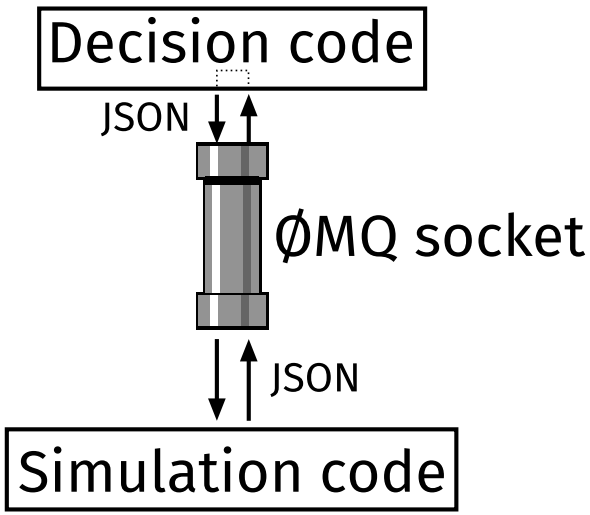
- Shared ports/files cause bugs
- Packaging and experimental traceability
- Debugging

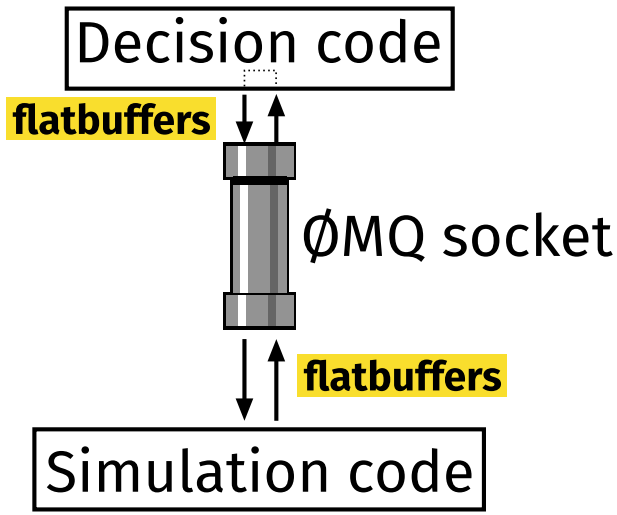


Simu/decision communication overhaul

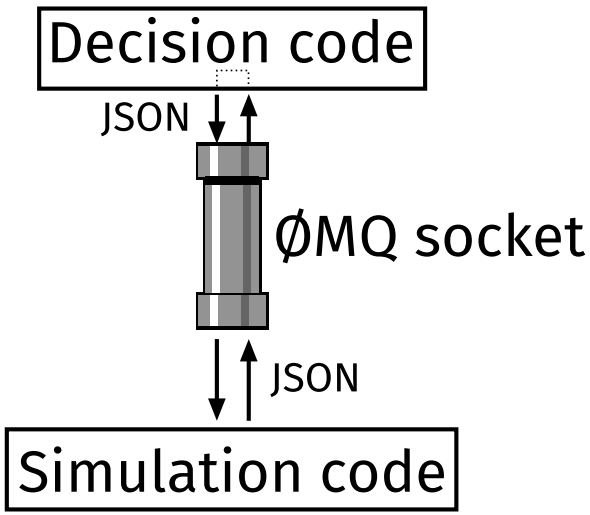


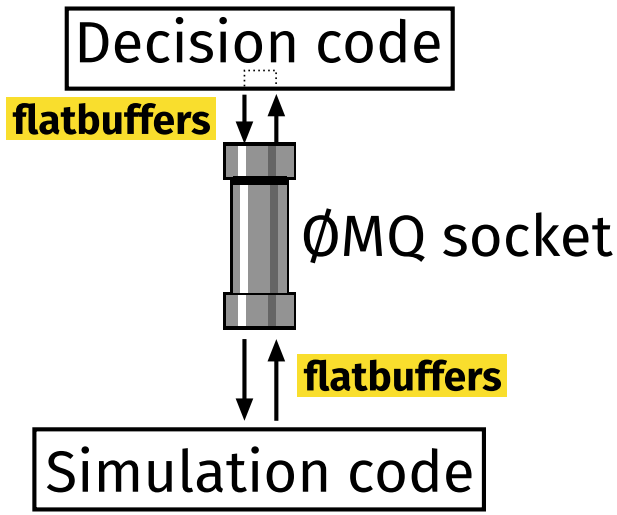
Simu/decision communication overhaul

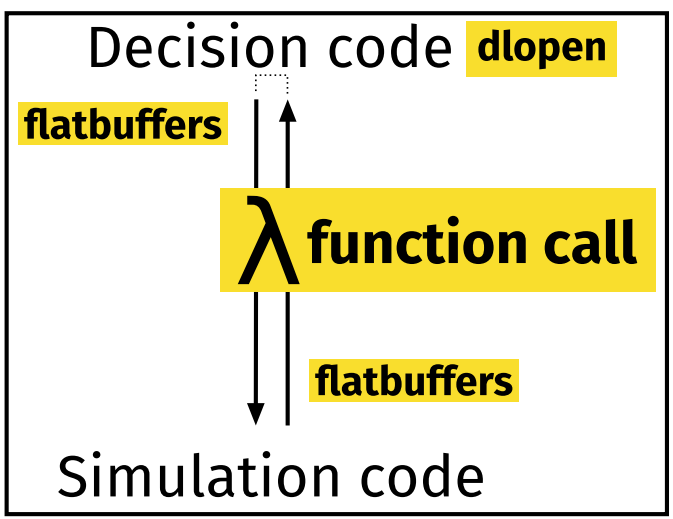




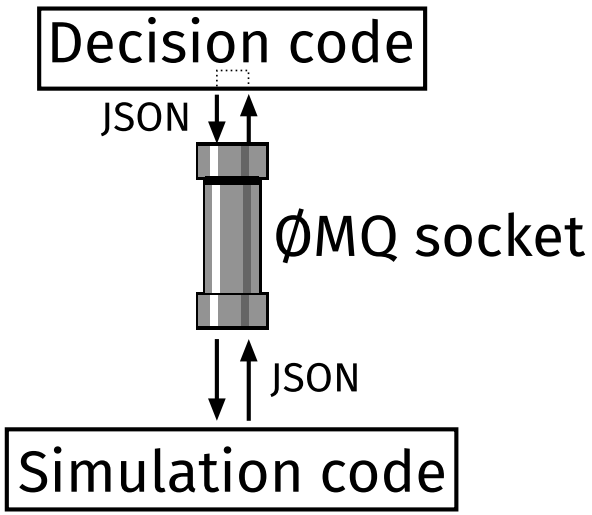
Simu/decision communication overhaul

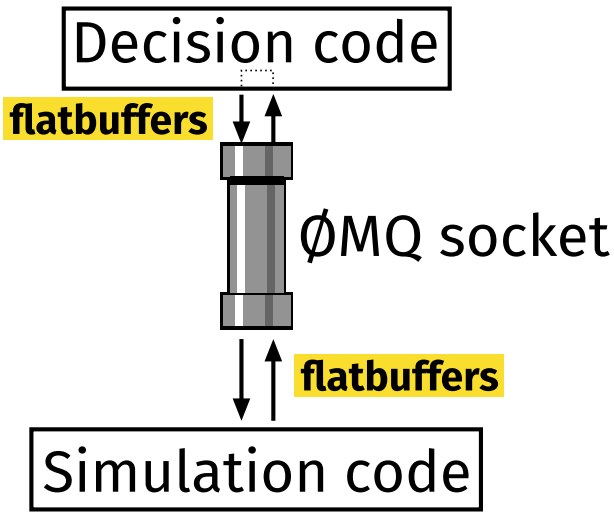


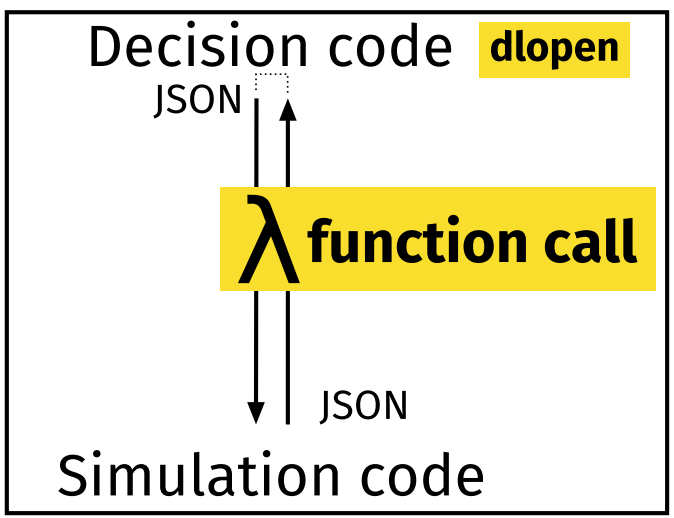


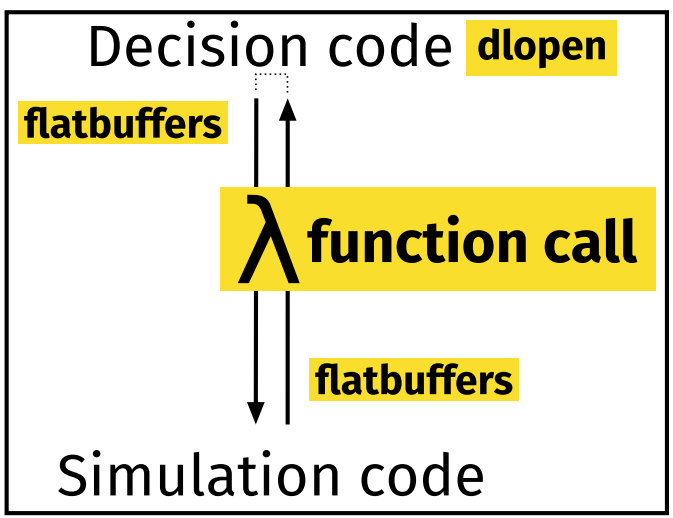


Simu/decision communication overhaul









Application modeling and profile composition

Batsim 4

- Sequential
- 1 app ptask + 1 IO ptask

Batsim 5

- Sequential
- Parallel (union of N ptasks)
- Parallel (fork join of profiles)

Reusable components

Batmen[1] implemented as a fork from Batsched (C++ sched algos for Batsim).
$\longrightarrow$ impossible to reuse directly unless your sched is in C++

Multiple EDCs possible (for *simple* cases such as this one) in Batsim 5

[1] *Replay with Feedback: How does the performance of HPC system impact user submission behavior?* Maël Madon, Georges Da Costa, Jean-Marc Pierson. FGCS 2024.
https://hal.science/hal-04432711v1
https://gitlab.irit.fr/sepia-pub/mael/batmen

Probes

Batsim 4

- 1 metrics: energy consumption of the whole platform from 0 to now
- tedious to use: `CALL_ME_LATER(10s)` $\longrightarrow$ `GIMME_ENERGY(now)`

Batsim 5

- Support for platform instantaneous metrics
  - Power of hosts and links
  - Load of hosts and links
- Probe types
  - One shot
  - Automatic period=T for n iterations or ∞
- Resource selection
- Data aggregation (spatial. temporal?)
- Event filtering (*e.g.*, only call sched when total power > 1 MW)

Where are we now?

Implementation status

- Serialization library **done**
- Schedulers as libraries **done**
- Probes: protocol **done** but code **todo**
- Profiles.
  - delay, ptask, replay **done** & much better tested
  - Ptask variants **todo**
  - New compositions **todo**

Already usable[2] for some use cases! Release Soon™

---

[2] *Light-weight prediction for improving energy consumption in HPC platforms.*
Danilo Carastan-Santos, Georges Da Costa, Millian Poquet, Patricia Stolf, Denis Trystram. Euro-Par 2024.
https://hal.science/hal-04566184v1
https://zenodo.org/doi/10.5281/zenodo.11173631

Batsim future directions

Long term need: Evaluation of scalable models

Core models (SimGrid)

- **`smpi`** replay model validated[3] but does not scale for platform-level scheduling
- **`ptask_L07`** bad sharing of network links[4] when mixing **huge & tiny** ptasks
- **`ptask_bmf`** (B. Donassolo, A. Legrand, 2022) seems to fix this issue

  but does not always terminate… even when applying noise to values 😢

High level application/phase models (Batsim)

- PDGEMM seems to behave correctly on **`ptask_bmf`**[4]
- Other applications not evaluated…

---

[3] *Toward More Scalable Off-Line Simulations of MPI Applications.*
Henri Casanova, Anshul Gupta, Frédéric Suter. Parallel Processing Letters, 2015.

[4] *Advanced Simulation for Resource Management, Chapter 5.*
Adrien Faure. PhD thesis, Univ Grenoble Alpes, 2018.

Next years: Batsim-related Sepia funded workforce

- PEPR Tase
  - 1 PhD on placement of decentralized services (IoT, distributed ML)
- VILAGIL project
  - 1 postdoc on resource management (smart city, renewable energy)
- ANR Delight
  - 1 PhD on blackbox power opti. & exp. repro. on FL

- PEPR Cloud
  - 1 PhD on simulation of decentralized services
- PEPR NumPEx
  - 1 PhD on monitoring and control at scale & app modeling
  - 1 PhD on resource management

Next years: Expected improvements

Application (power) modeling & evaluation
- Services
- Federated learning
- HPC benchmarks. proxy/mini apps?

Usability for specific use cases
- Services
- Mobility
- Decision making with real-time monitoring (control theory...)

Thanks!
- Tutos & docs on https://batsim.rtfd.io
- Contact us on https://framateam.org/batsim
- Contact me on millian.poquet@irit.fr