

Job Simulation for large-scale PBS based Clusters with the Maui Scheduler

Georg Zitzlsberger, Branislav Janský, Jan Martinovič (IT4Innovations, VŠB – Technical University of Ostrava)

IT4Innovations
national
supercomputing
center



Abstract

For large-scale High Performance Computing centers with a wide range of different projects and heterogeneous infrastructures, efficiency is an important consideration. Understanding how compute jobs are scheduled is necessary for improving the job scheduling strategies in order to optimize cluster utilization and job wait times. This increases the importance of a reliable simulation capability, which in turn requires accuracy and comparability with historic workloads from the cluster. Not all job schedulers have a simulation capability, including the Portable Batch System (PBS) resource manager. Hence, PBS based centers have no direct way to simulate changes and optimizations before they are applied to the production system. We propose and discuss how to run job simulations for large-scale PBS based clusters with the Maui Scheduler. For validation purposes, we use historic workloads collected at the IT4Innovations supercomputing center, and demonstrate the viability of our approach.

Objectives

Investigated:

- Is the Maui Scheduler capable of simulating large-scale systems (>1k nodes)?
- Can the Maui Scheduler be used to simulate PBS workloads?
 - Does an equivalent configuration exist?
 - Is the already existing (historic) workload information sufficient for a simulation?
- How accurate is the simulation versus reality?

Not investigated in current work:

- Can we find a better configuration in the simulation for the real system?
 - Does the current PBS configuration allow efficient cluster resource usage?
 - Can the average wait times for the jobs be further reduced?
- Analyze effects of limiting simulation to weekly windows

Related Work

Simulation toolboxes:

- SimGrid: For large-scale distributed systems such as Grids, Clouds, HPC or P2P.
- GridSim: In contrary to SimGrid, supports multiple competing users, applications, and schedulers, each with their own policies.
- SimMatrix: Better resource usage than GridSim and SimGrid and capable with Exascale level systems, but only validated against a 4,096 core system.
- Alea: Job scheduler, building on GridSim and validated on a system with approx. 5,200 cores (102,657 jobs over 4 months).
- CQSIm: Python based with awareness of dynamic electricity pricing, 164k cores



Problems with Toolboxes:

- Toolboxes require manual implementation of job scheduling algorithms
- No in situ replacement for HPC job management systems

Simulators in HPC schedulers:

- Slurm Simulator: Slurm environment; validated on TACC (6.400 nodes; 12 weeks)
- LSF: Only prediction of job start times (further plans announced)
- Maui Scheduler: Can replace PBS/TORQUE job scheduler; simulation capability already used for smaller clusters (e.g. 13 nodes; 3 months)
- Cobalt: Comes with Qsim, validated at ANL (Mira cluster)



Advantage of the Maui Scheduler:

- Mature & flexible configuration with simulation capability
- Can be used as a meta-scheduler for the PBS/TORQUE resource manager

Acknowledgements

This work was supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center – LM2015070", by the ERDF in the IT4Innovations national supercomputing center – path to exascale project (CZ.02.1.01/0.0/0.0/16_013/0001791) within the OPRDE, and ANTAREX, a project supported by the EU H2020 FET-HPC program under grant agreement No. 671623.

Simulated PBS Environment

Infrastructure (Salomon cluster):

- PBS Professional 13.1 based
- Intel Xeon E5-2680v3 (12 cores), 2 sockets
- Total 1,008 nodes:
 - 432 with 2 Intel Xeon Phi coprocessors
 - 576 without coprocessors
- Total cores: 24,192 + 51,840 (coprocessors)



- Rpeak: 2,011 TFLOP/s

TOP 500 rankings:

- #39 (June 2015)
- ...
- #139 (June 2018)

PBS configuration:

- Job priority p :

$$p(q, f, w) = 1,000 * q + \frac{f}{1,000} + \frac{w}{864,000} \quad \text{with} \quad f = 10^6 * \left(1 - \frac{u_p}{u_t}\right)$$

- Project based fairshare f as ratio of usage for project u_p and total usage u_t
- Conservative backfilling and fairshare with 168h half-life
- Wait time w of job in seconds avoids starvation of older job requests

- No node sharing (scheduling on node granularity)
- Multiple queues with different priorities and limitations:

Queue name	Number of nodes	Priority q	Maximum wall time	Maui Scheduler configuration
qexp	1,006 + 2 dedicated (max. 8 per user, max. 32 total)	150	1h	CLASSCFG[dexp] PRIORITY=150 MAXNODE=32 MAXNODEPERUSER=8
qprod	1,006	0	48h	CLASSCFG[qprod] PRIORITY=0 MAXNODE=256
qlong	576 (max. 256 total)	0	144h	CLASSCFG[qlong] PRIORITY=0 MAXNODE=256
qmp	1,006	0	4h	CLASSCFG[qmp] PRIORITY=0 MAXNODE=752
qfree	1,006 (max. 752 total)	-1,024	12h	CLASSCFG[qfree] PRIORITY=-1024 MAXNODE=752
qmic	2 * 432	0	24h	CLASSCFG[qmic] PRIORITY=0
SERVICE	1,008	512	N/A	CLASSCFG[SERVICE] PRIORITY=512

Applying the Maui Scheduler's Simulator

1. Required changes to the Maui Scheduler:

- Increase **MAX_MTASK** (cores) and **MMAX_JOB** (jobs) to 32,768 and 80,000
- Increase **MAX_PRIO_VAL** from 10^9 to 10^{12}
- Optional: Use absolute job dates (Maui Scheduler's simulator uses relative dates)

2. Simulate unexpected node downtimes with user reservations:

- for** $step := 1$ to $last_step$ **do begin**
- $\forall$ nodes switching offline between $step - 1$ and $step$:
 - canceljob**: delete node's job, and
 - setres**: place unlimited reservation on node
- $\forall$ nodes switching back online between $step - 1$ and $step$:
releasers: remove reservation from node
- schedctl**: advance one time step
- end**

3. Simulate system maintenance using system reservations:

```
setres -s 08:00:00_01/29/2017 -d 20000 ALL
```

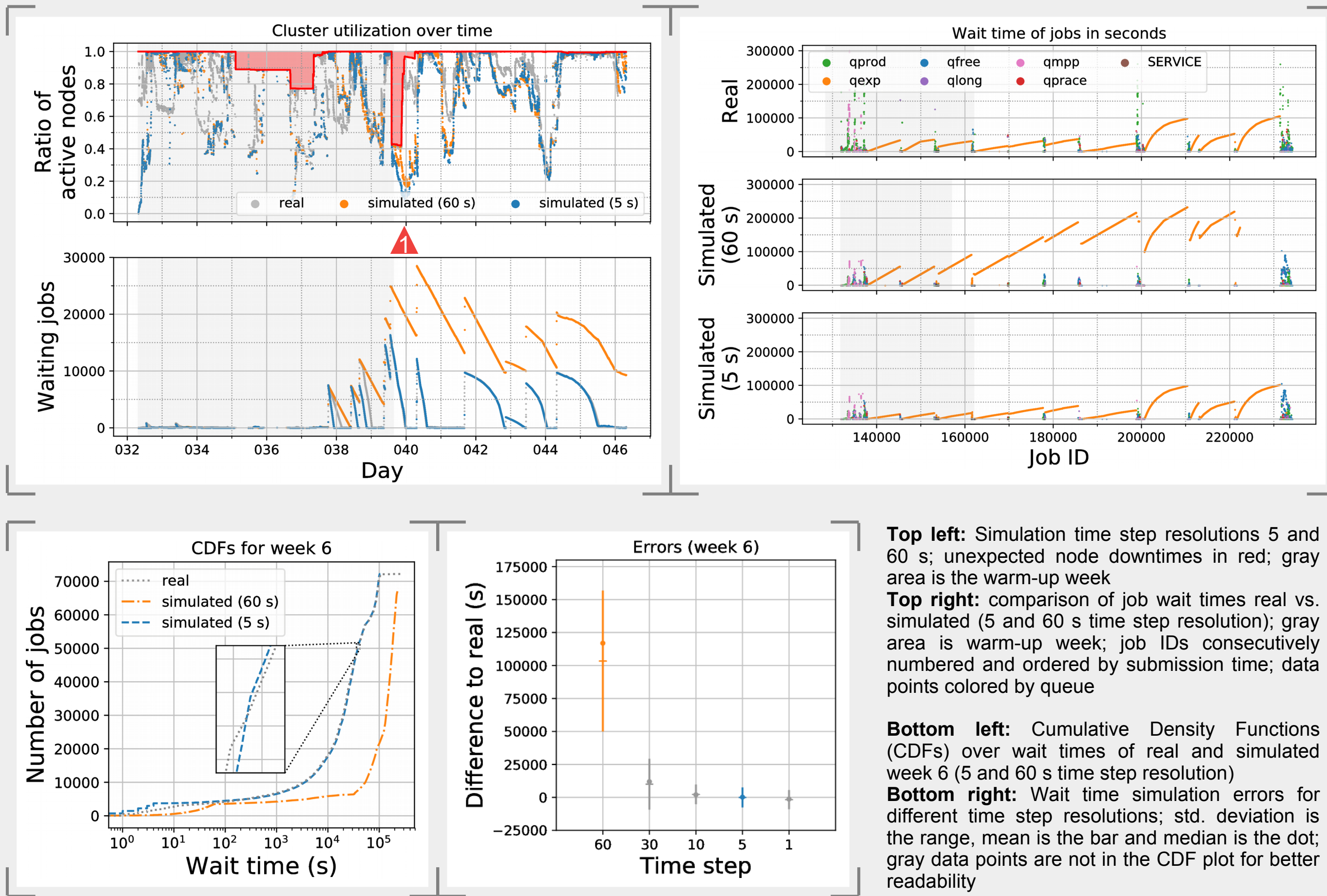
4. Cluster configuration and historic workloads (trace files):

- 50 week simulation in 2017
- Data retrieved from PBS' SQL storage (removed <1% of inconsistent job records)
- Includes submitted jobs, unexpected node downtimes & system maintenance
- Simulation in weekly windows (one week for warm-up)

Results: Accuracy & Time Step Resolutions

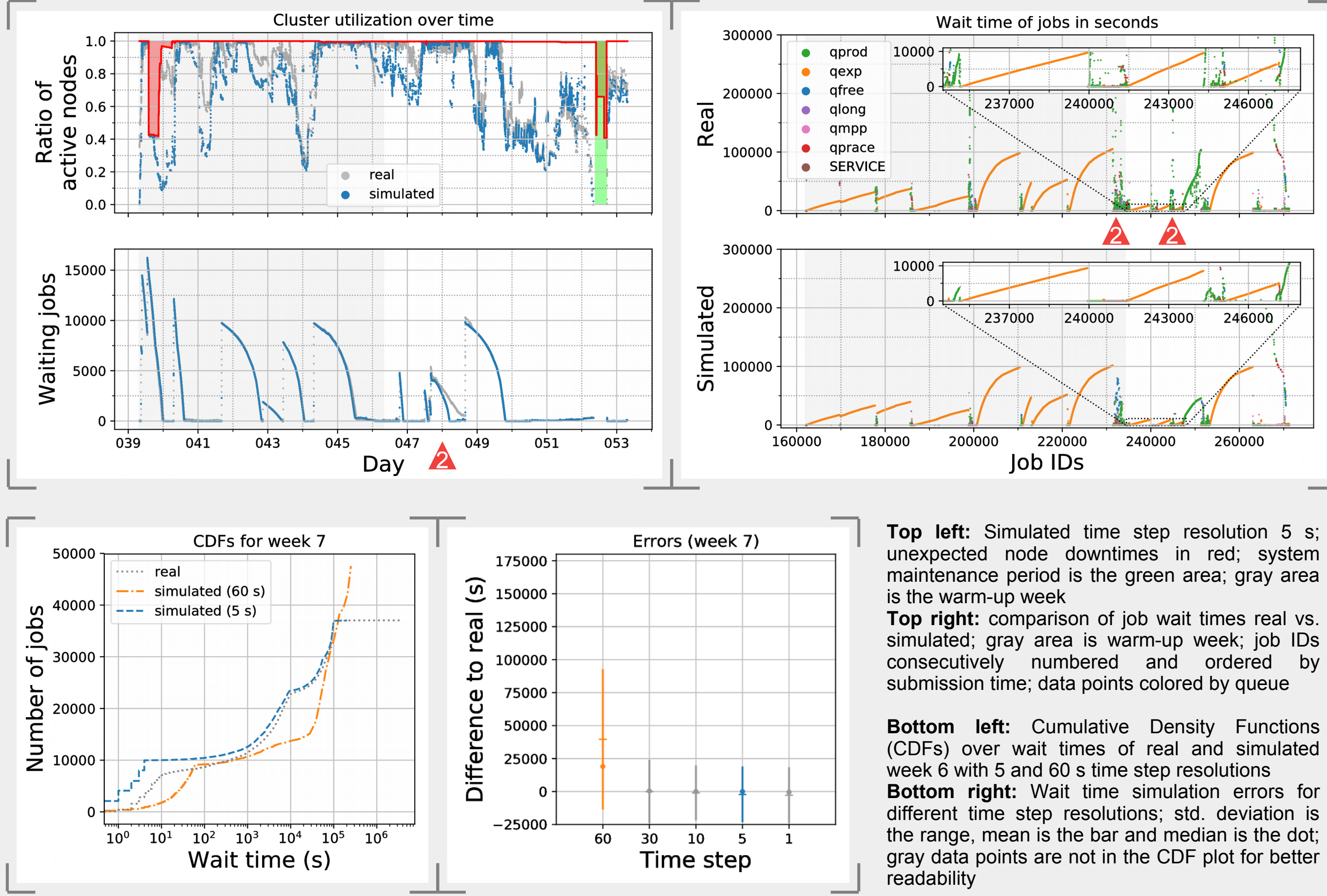
Simulation of week 6:

⚠: Termination of entire job with failed nodes



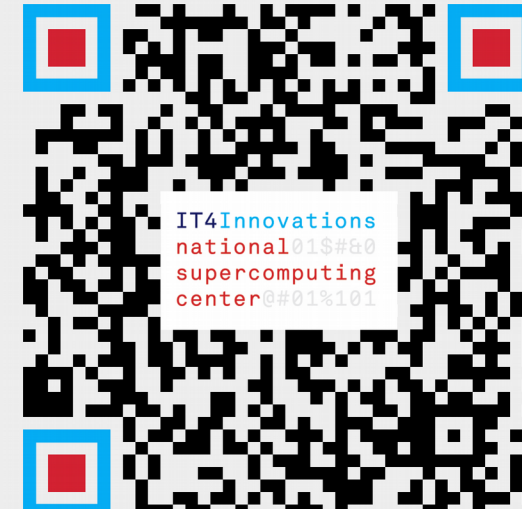
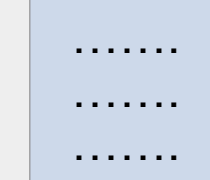
Simulation of week 7:

⚠: No job dependency information/simulation



Rich Collateral Available

The **Dockerfile & environment, configuration files, interactive plots, the original paper**, and more are available on GitHub:
<https://github.com/It4innovations/Maui-Simulation>



Contact



georg.zitzlsberger@vsb.cz

