



# HPC System Lifetime Story: Workload Characterization and Evolutionary Analyses on NERSC Systems



**Gonzalo P. Rodrigo** - [gonzalo@cs.umu.se](mailto:gonzalo@cs.umu.se)

P-O Östberg - [p-o@cs.umu.se](mailto:p-o@cs.umu.se)

Erik Elmroth - [elmroth@cs.umu.se](mailto:elmroth@cs.umu.se)

Katie Antypas - [kantypas@lbl.gov](mailto:kantypas@lbl.gov)

Richard Gerber - [ragerber@lbl.gov](mailto:ragerber@lbl.gov)

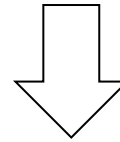
Lavanya Ramakrishnan - [iramakrishnan@lbl.gov](mailto:iramakrishnan@lbl.gov)

---

Distributed Systems Group – Umeå University, Sweden  
Data Science & Technology – Lawrence Berkeley National Lab

# Big picture: HPC evolution vs. batch schedulers

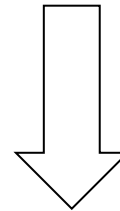
Infrastructure evolution: Exascale  
Application changes



Batch job schedulers were conceived to

- 1) order tightly coupled large parallel jobs.
- 2) achieve high utilization and low turn around time.

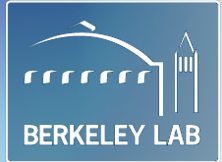
Are batch schedulers suitable for  
the current infrastructure and  
applications? Future?



**Understanding how workloads have  
evolved in the past**



# Systems analyzed



## Supercomputer

Hopper
Deployed January 2010
Cray XE6
Gemini Network
6,384 Nodes, 24 cores/node 154,216 cores
1.28 Pflops/s
Torque + Moab



## High performance Cluster

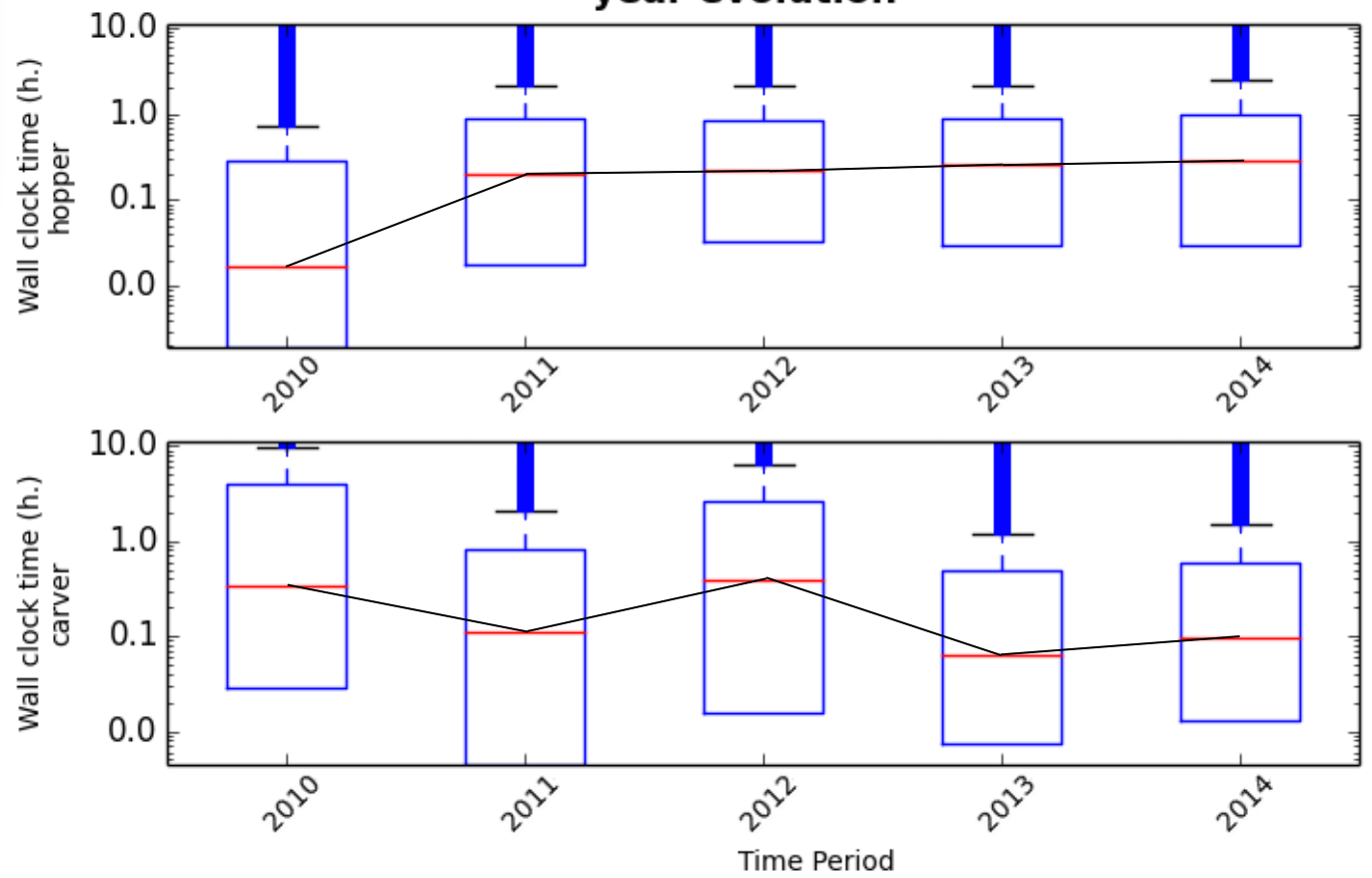
Carver
Deployed 2010
IBM iDataPlex
Infiniband (fat-tree)
1,120 Nodes, 8/12/32 cores/ node, 9,984 cores
106.5 Tflops
Torque + Moab





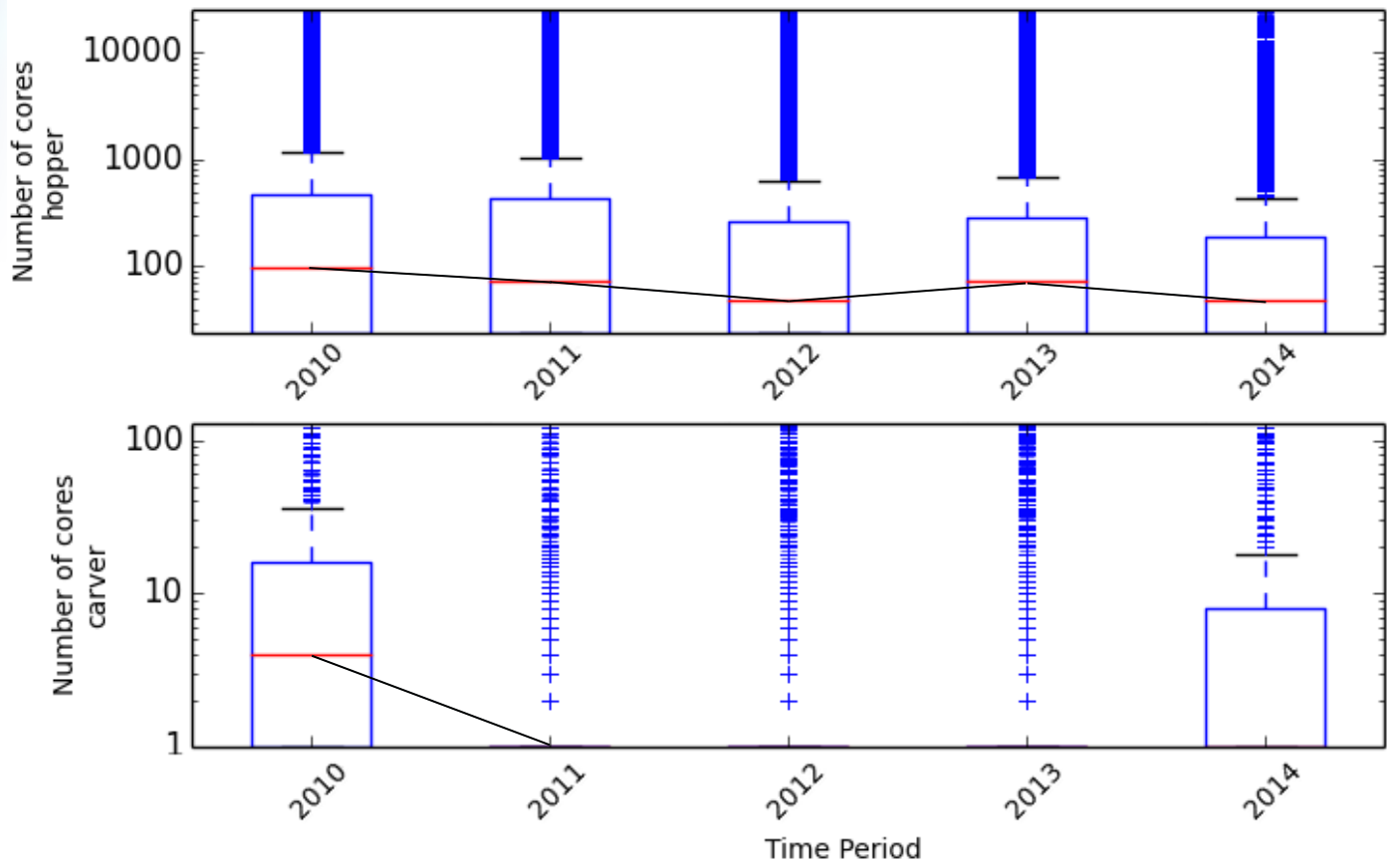
# Wall clock time

### Hopper Carver -Wall clock time per Job year evolution



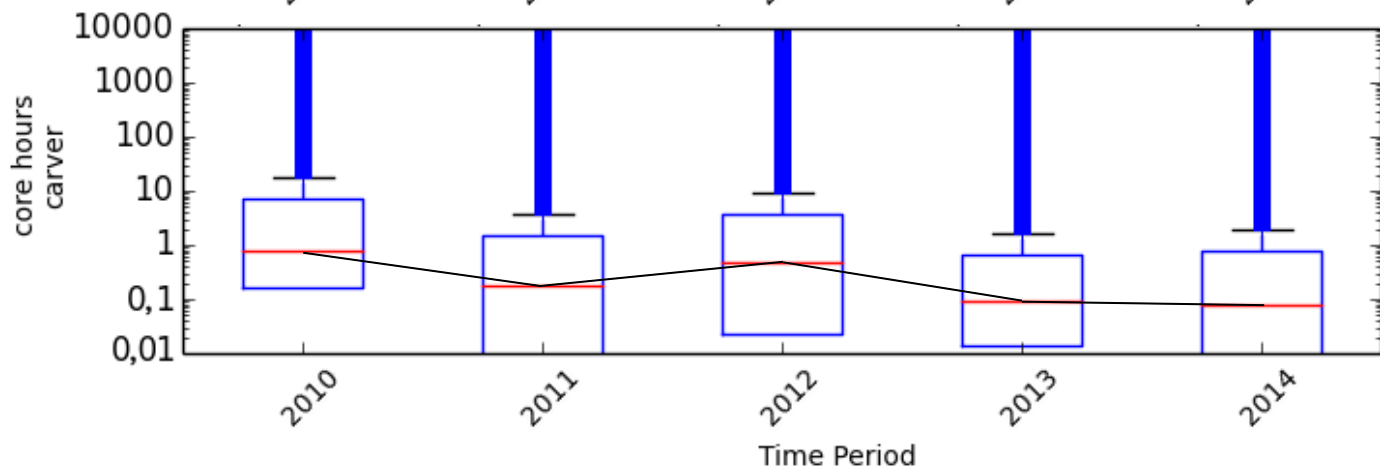
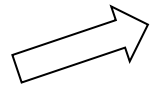
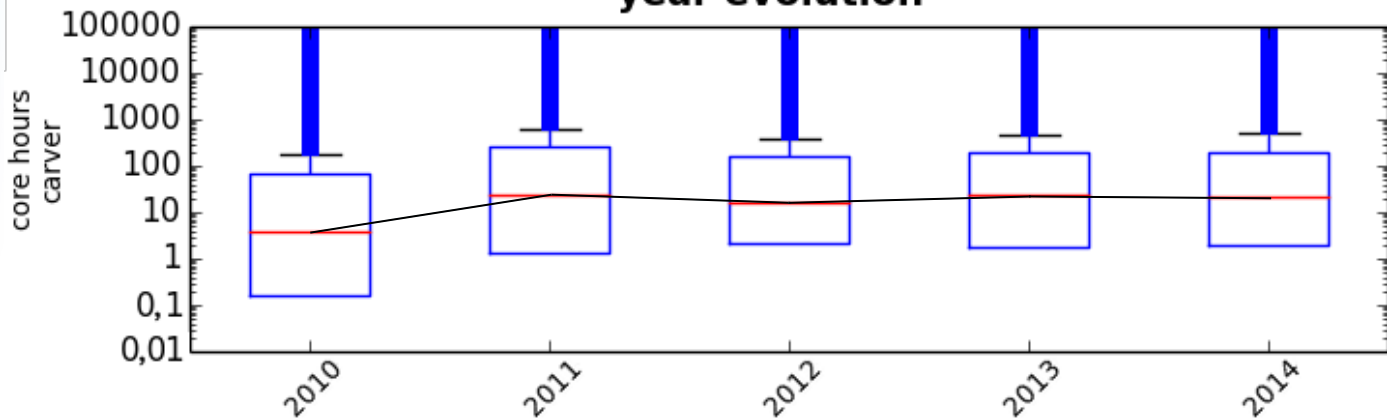
# Number of cores per job

### Hopper Carver - Number of cores per Job year evolution



# Core hours per job

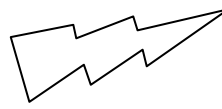
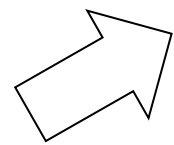
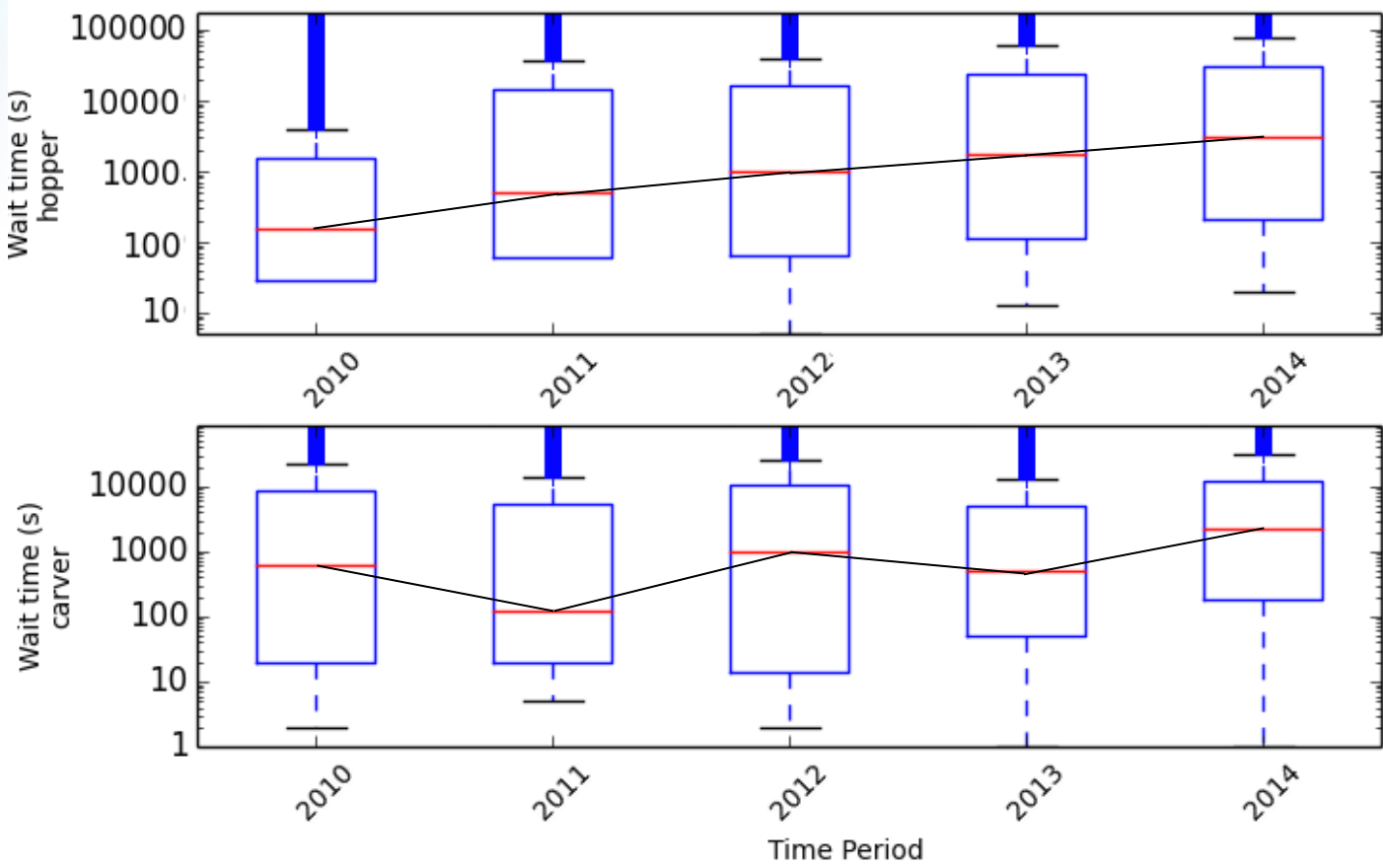
### Hopper Carver - Core hours per Job year evolution





# Wait time per job

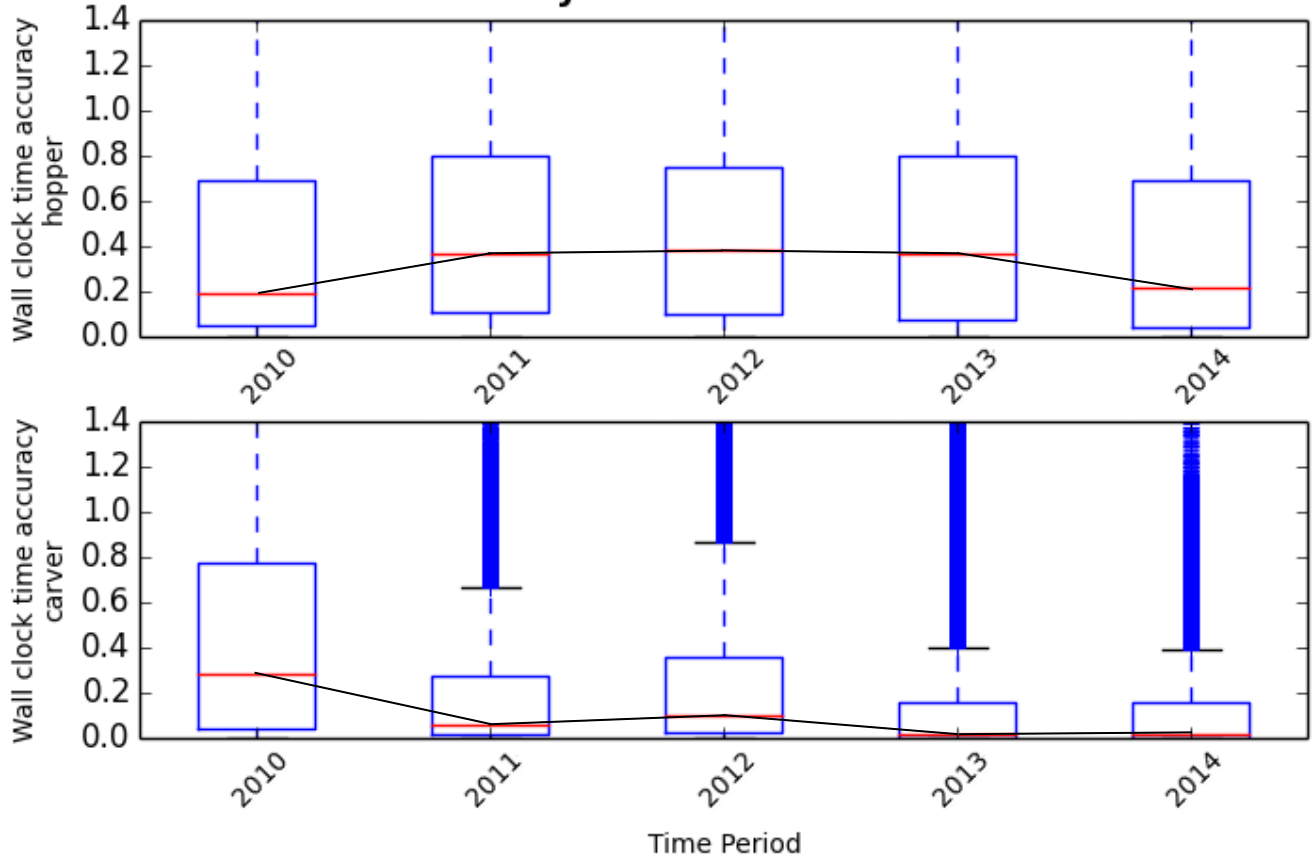
### Hopper Carver -Wait time per Job year evolution

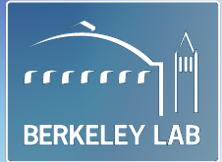




# Wall clock time accuracy

### Hopper Carver -Wall clock time accuracy per Job year evolution





## Summary

Two machines with very different starting workloads, become more similar towards the end.

Most jobs are not very long and very parallel

Systems get “more loaded” in time

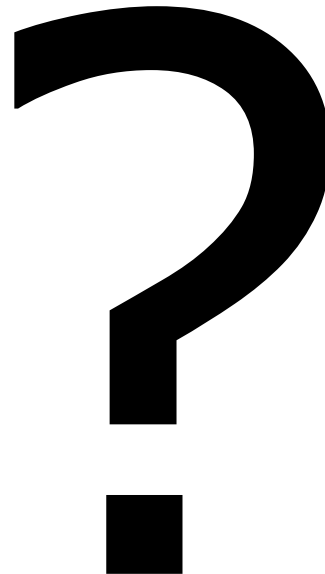
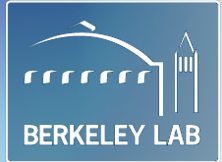
Users’ estimations are really inaccurate.

	2010		2014	
(medians)	Hopper	Carver	Hopper	Carver
<b>Wall Clock</b>	< 1 min	20 min	12 min	6 min
<b>Number of Cores</b>	100 cores	5 cores	30 cores	1 core
<b>Core Hours</b>	4 c.h.	0.9 c.h.	11 c.h.	0.09 c.h.
<b>Wait time</b>	100 s	10 min	20 min	20 min
<b>Wall clock accuracy</b>	0.2	0.25	0.21	< 0.1



Thank you very much!

---



**(If you want to know more: please, come  
to the poster presentation)**