

# The ALICE Tier-1 in Bergen 2020

Boris Wagner  
(personal opinion)

# Pledges

Compute	25.5 kHEPSPEC06
---------	-----------------

Disk	2.5 PB
------	--------

Tape	4 PB
------	------

# Middleware

- UiB uses
  - AliEn for computing resources
  - dCache for storage resources
  - IBM TSM for tape resources

# Norwegian Science Cloud

- Can be used by all scientist in Norway
- Hardware in two locations Oslo and Bergen
- Flexible assignment and distribution of VMs  
e.g. for temporary peak usage or high availability
- Scientists have full control over software stack  
by creating their own disk images
- Less support needed from sysadmins (only infrastructure)
- Makes scientific results reproducible by saving VM + data
- Easily extensible to other cloud providers

# Fully Virtualized Tier-1 @ UiB

- Tier-1 hardware is part of the Norwegian science cloud (NREC)
- Computing resources are a cluster of virtual machines
- Storage resources are virtual machines and virtual disk images on top of a distributed file system (Ceph)
- Tape resources are separate from the backup solution of UiB

**This should make the Tier-1 service  
independent of underlying hardware**

# Fully Virtualized Tier-1 @ UiB

- Tier-1 hardware is part of the Norwegian science cloud (NREC)
- Computing resources are a cluster of virtual machines
- Storage resources are virtual machines and virtual disk images on top of a distributed file system (Ceph)
- Tape resources are separate from the backup solution of UiB

**This should make the Tier-1 service  
independent of underlying hardware**

# Software Stack I

- Openstack  
Provisioning system for VMs on hardware.  
Define number of CPUs, RAM, disk space, network and OS per VM and also number of VMs
- Elasticcloud  
Create and modify a compute cluster on cloud infrastructure. Automatically creates frontend and worker nodes, batch queue system, monitoring, etc...  
Can shrink and extend number of worker nodes.

# Software Stack II

- Terraform - “Infrastructure as Code”  
Textual description of your complete computing infrastructure, like number of nodes, which servers, what network including routing (virtual switches)  
Makes whole setup reproducible and abstracts it away from real hardware
- Ansible  
Continuous configuration and change management. Initializes and keeps configuration of machines/VMs in a defined state. E.g. automatic configuration after update.  
Makes services and changes reproducible.



# People

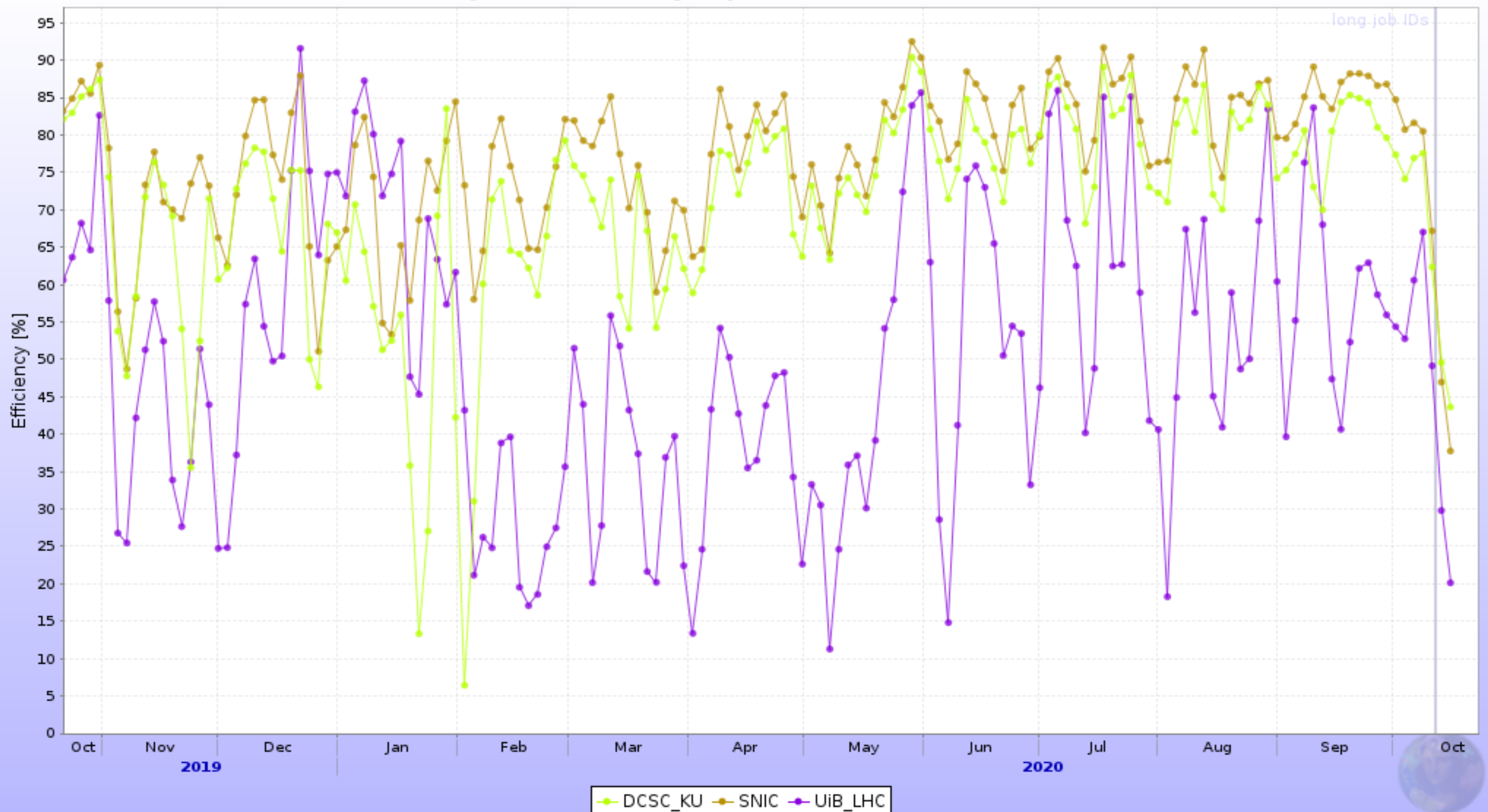
- (Raimund Kristensen)  
NREC – group leader UNIX
- (Tor Ladre)  
NREC
- Boris Wagner  
Cluster operation, Tier-1 operation

NREC is missing one person

# Problems

EFFICIENCY

Jobs efficiency (cpu time / wall time)

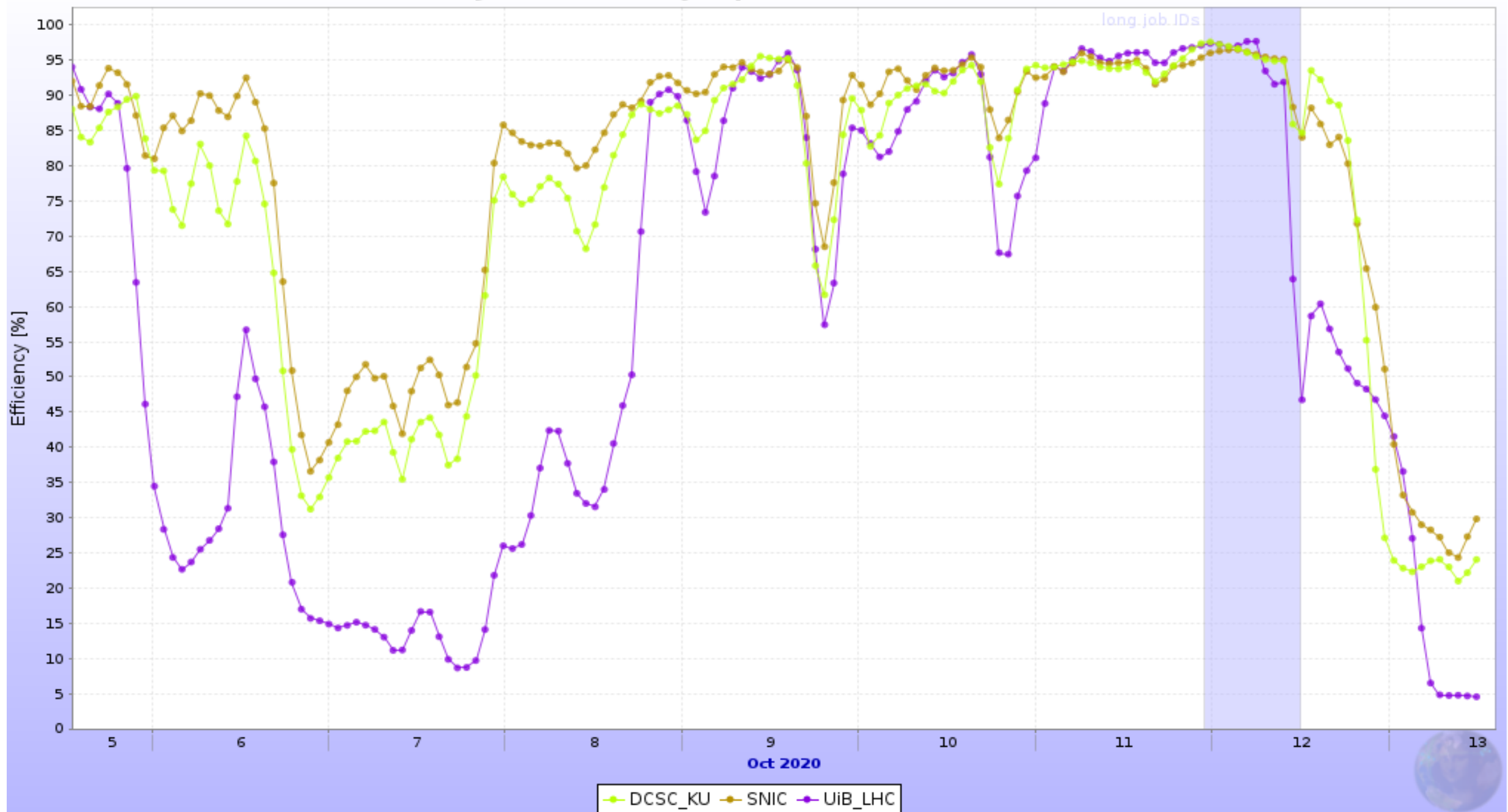


# Problems

EFFFICIIENCYYYYYY DEEETAAILLLL

[KSIZK conversion table](#)

Jobs efficiency (cpu time / wall time)



# Current Conclusion

- Advantages
  - Opportunistic resources can be added fast
  - Easily expandable
  - Independent of hardware or cloud provider
  - Reliable and reproducible infrastructure with development history

# Current Conclusion

- Advantages ?
  - Opportunistic resources can **maybe** added fast
  - **Not** Easily expandable – need special HW in VM
  - ~~Independent of hardware or cloud provider~~
  - Reliable and reproducible infrastructure with development history

# Current Conclusion 2

- Disadvantages
  - Complex interplay between complex software stacks makes the system brittle and hard to understand for new sysadmins (deep learning curve)
  - Flexibility not needed for Tier-1 operation, because all resources are used all of the time
  - Unnecessary loss of performance and capacity. Especially for storage and the dCache middleware.