Western Norway
University of
Applied Sciences

Grid services in a box

Container management in ALICE

Maxim Storetvedt

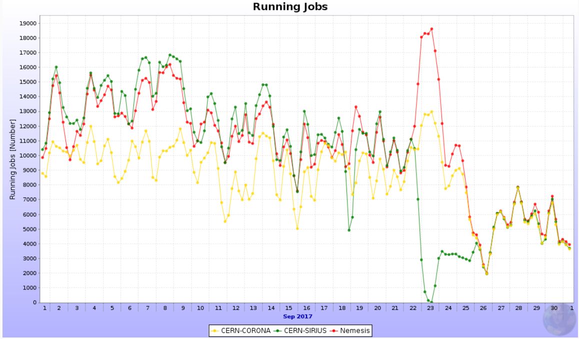
msto@hvl.no

July 12, 2018

ALICE

Using containers for site-services at ALICE

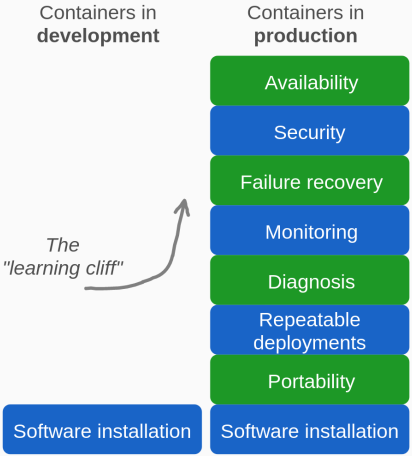
- This talk will focus on the initial experiences with managing containers for VOBOX use
  - Multiple deployed within ALICE as a pilot project
- Also planned for worker nodes
  - For more on this topic, see the talk by Miguel Martinez Pedreira on JAliEn



Production plot from MonALISA, Sep. 2017

Using containers for site-services at ALICE (2)

- Containers can provide several benefits over using virtual machines (VMs) for VOBOXes
  - Less overhead
  - Less use of storage
  - One-click deployment
- Container setup for VOBOXes is very different from VMs – especially for production purposes
- The next slides are dedicated to examining
  - Configuration
  - Downtime prevention
  - Performance

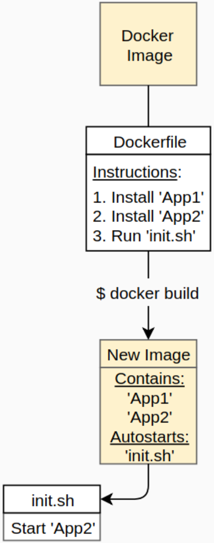


Selected VOBOX Container platform

- **Docker** used within ALICE for site-service containers
- Other container platforms available
  - **Singularity** quickly gaining ground within HPC
- Site-services, like VOBOXes, need a full networking stack
  - Not currently available in Singularity
  - Available in platforms like Docker and Rkt

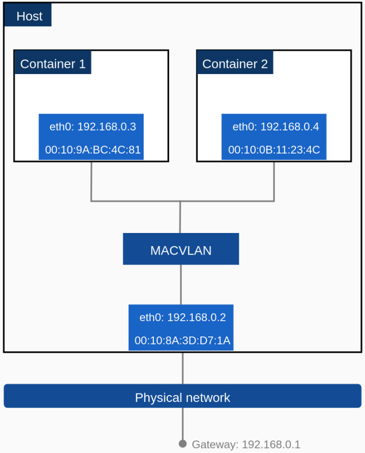
ALICE VOBOX image configuration

- We need automatic startup of VOBOX services at container launch
- Dockerfiles
  - Scripts composed of various commands to perform on a base image
- An image must be rebuilt to reflect changes in a Dockerfile
  - Since this is a pilot project, changes are frequent → frequent downtime
- Solved by pointing to a script within the container – e.g. /etc/init.sh



ALICE VOBOX Network Configuration

- MACVLAN – A reverse VLAN
  - A VLAN maps an OS side of a networking interface to multiple virtual networks on its network side
  - A MACVLAN maps a network side of an interface to multiple virtual interfaces, each with their own MAC address
  - Traffic sent from the virtual interfaces is sent directly to the underlying network, and identified by the assigned MAC address.
- VOBOX containers networked using MACVLAN
  - Allows containers to appear as normal machines on the network



MACVLAN architecture

ALICE VOBOX host configuration

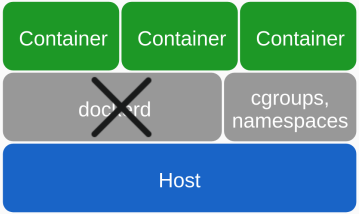
- VOBOXes need many files open simultaneously
  - Will quickly reach default system limit for maximum open files when more than two VOBOX containers run on a single host
  - Causes services to freeze or terminate
  - System limit must be increased to avoid these issues
- Autofs disabled on all hosts
  - Otherwise known to cause problems for CVMFS in containers

ALICE VOBOX host configuration (2)

- Host connectivity
  - The host and its containers can not reach/ping each other
    - Specific to how MACVLAN works
  - Separate Docker bridge created to obtain this connectivity
- Kernel access privileges
  - Containers have limited access privileges by default
    - Several tools and services may fail to launch
    - Most networking tools are affected
  - Full privileges granted for VOBOXes
    - Limited risk for this purpose, as VOBOXes are handled by sysadmins

Preventing containerised VOBOX downtime

- The ALICE containerised VOBOXes use the Live Restore feature
  - Allows containers to run without the Docker service
  - Useful for system updates → avoid downtime
  - Containers must still reconnect with Docker sometime
    - Will otherwise eventually fail due to log-buffer overflow
- Container management tools can handle automatic restarts for terminated containers
  - Swarm is bundled with Docker, but dying (gradually replaced by Kubernetes)
  - Not used for VOBOXes (not efficient for few containers)

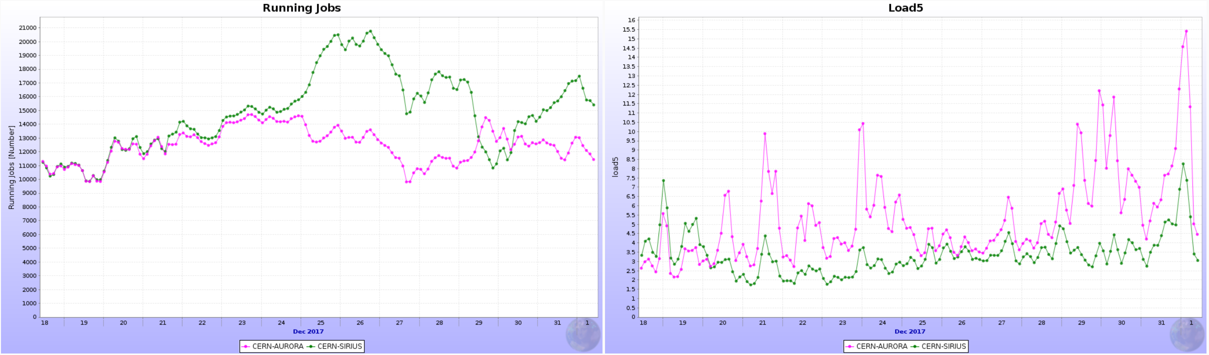


By default, terminating **dockerd** kills all containers

Performance

- Performance monitored over longer periods
    - Tested with both the AUFS and Overlay2 storage drivers
    - Performance and system load shown to be similar to VMs
    - Faster VOBOX restart after updates/failures compared to VMs
        - Less overhead
    - Smaller storage footprint

Performance (2)

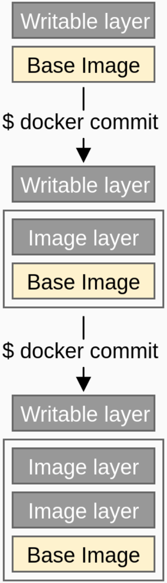


Left: Container running production jobs compared to a VM
Right: Container load compared with the same VM, for the same interval

The container can run more jobs, with less load, compared the VM

Performance – flattened images

- Performance decreases when the number of storage layers increases
    - Common for copy-on-write filesystems
    - All changes to a container are stored on a separate storage layer
        - New layer added for each commit
    - Flattened images used during testing
        - All additional layers merged into one



Conclusion

- ALICE is ready for moving site-services to containers
  - Well tested in production
    - Stability
  - Positive results in terms of load/performance
- More VOBOX containers to be deployed
- Also relevant for worker nodes – see the talk by Miguel Martinez Pedreira on JAliEn.

End

Thank you

Questions or comments?

E-mail: msto@hvl.no