Containers, VOBoxes and Singularity

Maxim Storetvedt

Department of Computing, Mathematics, and Physics

November 9, 2017

Western Norway
University of
Applied Sciences

Background

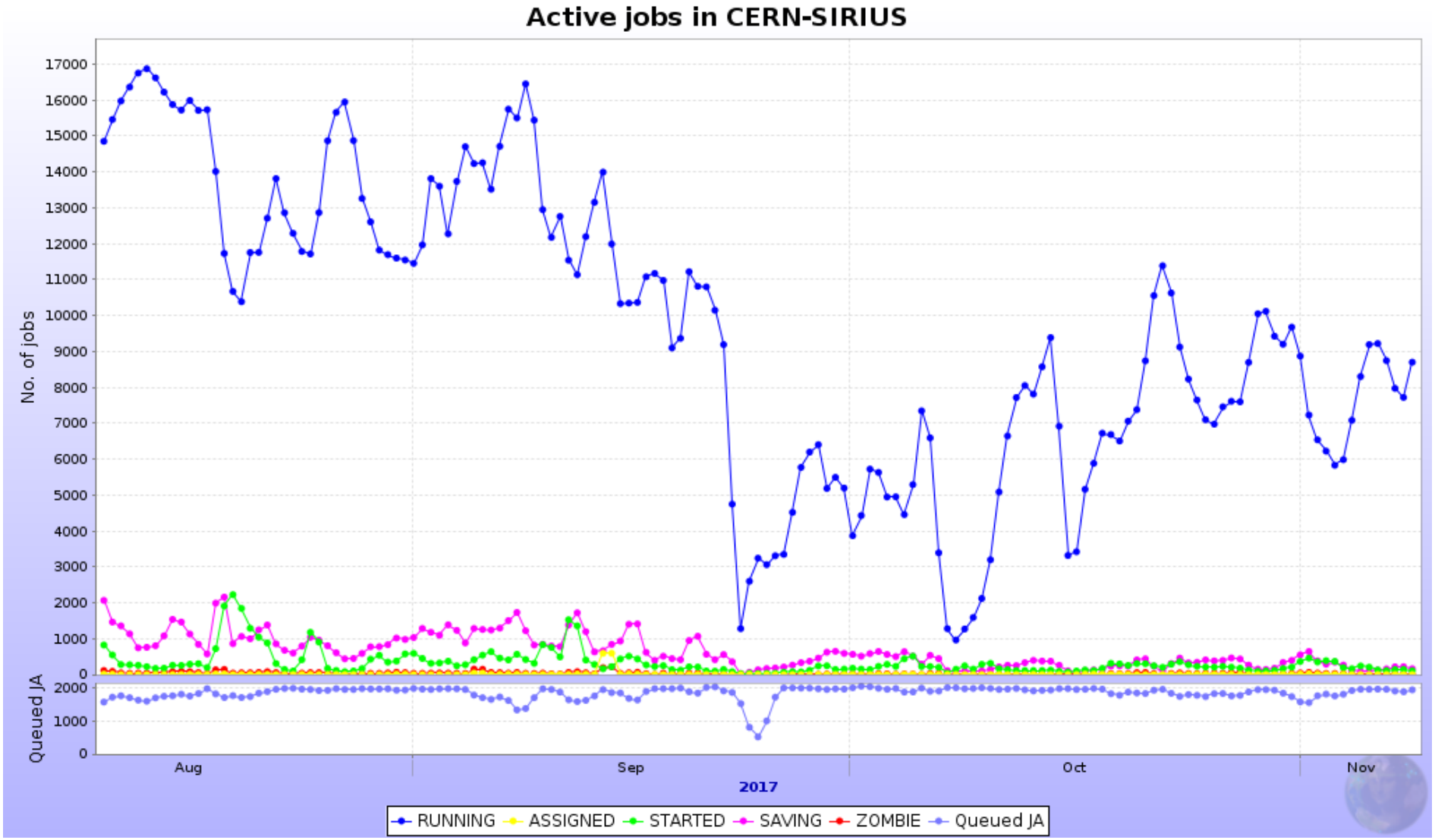
- Containers
  - “OS-level virtualisation”
  - Useful for
    - Deployment of pre-packaged services
    - Lightweight isolation

Background

- Previous issues with using VMs for Vo-Boxes
  - Persistent network freezes
    - Manual restart required
  - Source of problem not identified
- Workaround using containers
  - Singularity
    - WLCG adoption increasing
  - **Docker**
    - More suitable as a "VM replacement"

Background

- One VO-Box container site initially launched
  - Test-site: "Nemesis"
  - Promising results!
    - Reliability
    - Performance
- CERN-Sirius later converted
  - Originally a VM site


Active jobs in CERN-SIRIUS
No. of jobs
Queued JA
Aug
Sep
Oct
Nov
2017
RUNNING
ASSIGNED
STARTED
SAVING
ZOMBIE
Queued JA

Docker terminated after a system update

- $\rightarrow$ Site containers terminated
- Restarted, but
  - Services must be started manually
    - No init.d
  - CVMFS not working
    - "Too many symbolic links"
  - DNS settings reset

What to do?(!)

Lessons learned

- Some form of init system needed
  - Dockerfiles
    - Build instructions for images
    - Allow specifying startup commands
- An image must be rebuilt to apply changes in a Dockerfile
- Workaround: point to a custom init script within the container
  - e.g /etc/init.sh

Lessons learned

- CVMFS
  - Mounted within the container from the host
    - No additional access privileges required
  - Must have been accessed on the host in order to work
    - “Too many symbolic links”
    - If not, container must be restarted
- Not ideal behaviour in a production environment!
  - Access privileges elevated to allow CVMFS within the actual containers

Lessons learned

- DNS
  - Will be reset when a container is restarted by Docker
  - Can instead be specified within Docker
    - Will apply to all containers

Outlook

- Tailored VO-Box image for containers
- “Production ready”
  - Generic
  - Accompanying Dockerfile & init.sh script
  - CVMFS built-in
  - No need for manual interventions(!)
- To be used on three new sites
  - Workflow separation for HelixNebula Science Cloud providers

Singularity

Singularity

- Lightweight container platform
- Tailor made for the HPC use-case
  - Overlapping requirements with WLCG
- Useful as a mechanism for isolation
  - File isolation
  - Process isolation

Singularity

- Goal: Let Job Agents use Singularity to separate workloads from other processes
- But, why not just stick with Docker?
- Singularity is
  - Lightweight
  - Has no background services
  - Works without root
  - No installation/configuration required

Singularity

- Questions in need of answers:
  - How to set up / configure Singularity for this use-case?
  - How to distribute / maintain image
- Moving towards:
  - Sites not required to provide Singularity...
  - . . . but should support it
  - CVMFS
  - As used in OpenScienceGrid

Singularity

- Requirements:
  - CVMFS
  - Linux kernel 3.10.0-693 or above (EL 7.4)

Running jobs in Singularity

- A read-only Singularity for job execution:
  - Setup comparable to OSG Singularity in CVMFS
  - Slight change in configurations
    - Overlay and bind-control enabled
    - Setuid disabled(!)
  - CVM3 from CVMFS

Initial Findings

- Isolation
  - File isolation
  - Process isolation
  - ~~Kernel isolation~~
- Job execution
  - Possible
  - Nonprivileged users

Outlook

- Needs “real world” testing
  - Bugs
  - Errors
  - Config adjustments
- Better image for running ALICE payloads(?)

Thanks!

Questions?