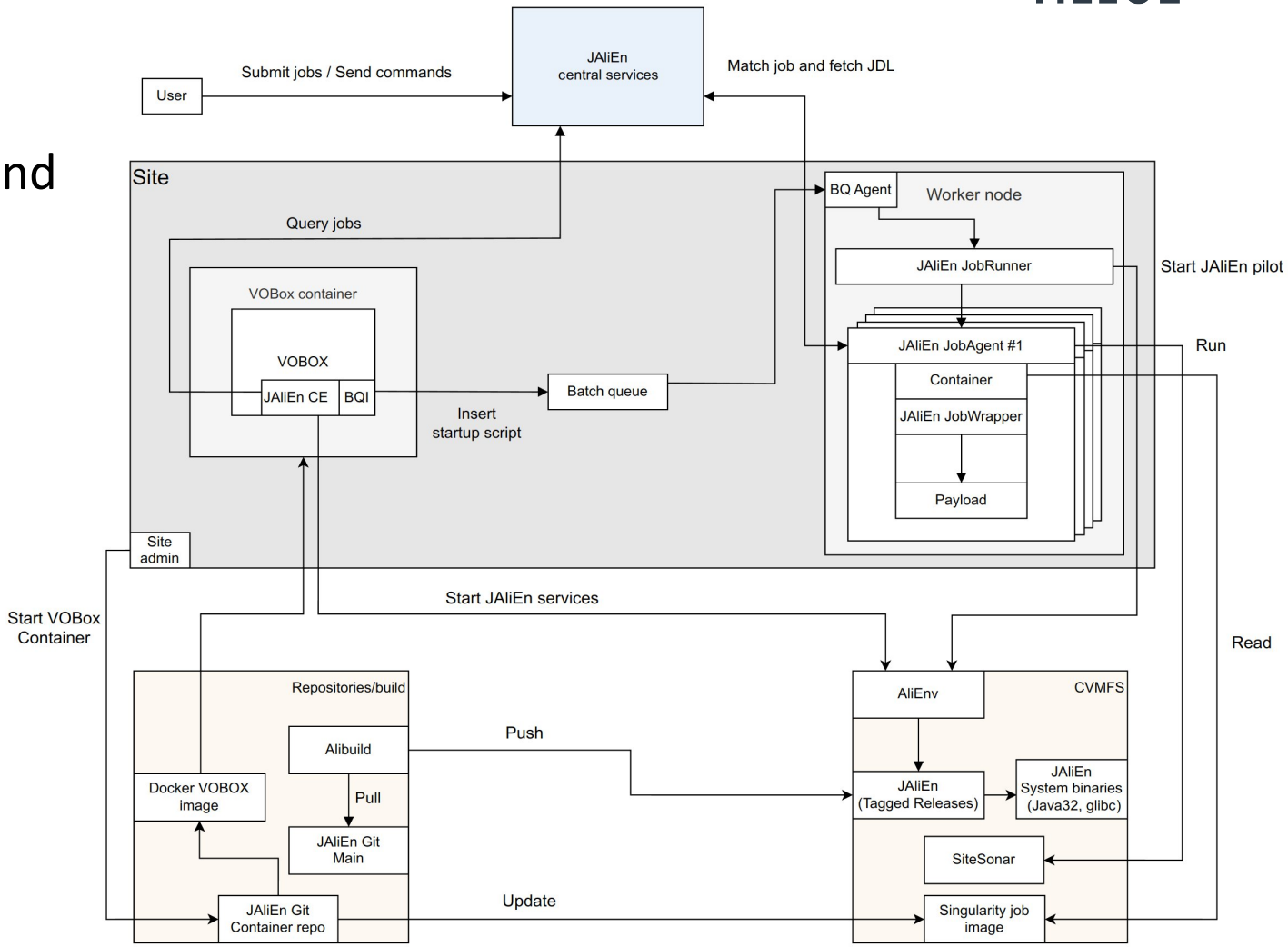
News from JAliEn implementation

and deployment

Maxim Storetvedt |US-ALICE Grid Operations Review | 16/05/2023

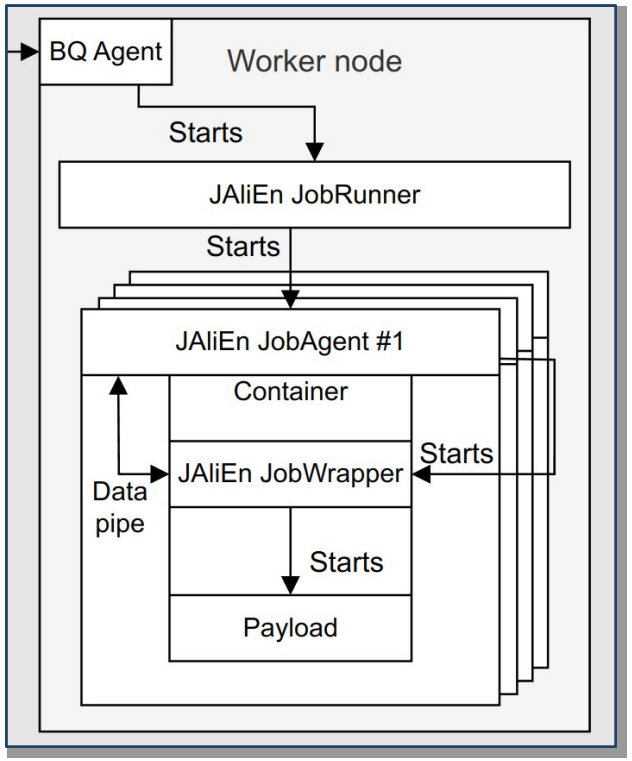
The ALICE & JAliEn Grid workflow

- **Containerised** core components
- Everything tied to a **central repository** and build system
  - Including the **payload environment**
- Versions and changes are **automatic**
  - Pushed to CVMFS as needed
- Essentials bootstrapped from **CVMFS**
  - Until we can get a container up and running



Job pilots and WNs

- Each startup script on WNs
  - Prepares environment
  - Loads pilot using libraries and Java from CVMFS
    - System agnostic
- Each JAliEn pilot consists of three components:
  - JAliEn **JobRunner**[1]: Resource/**multicore** handler
  - JAliEn **JobAgent**[2]: Job matcher/monitoring handler
  - JAliEn **JobWrapper**[2]: Payload executor
- The latter runs on a separate JVM for isolation
  - Automatically wrapped in a **container** by JobAgent
  - Handles payload that can be several cores per job slot



Payload environment

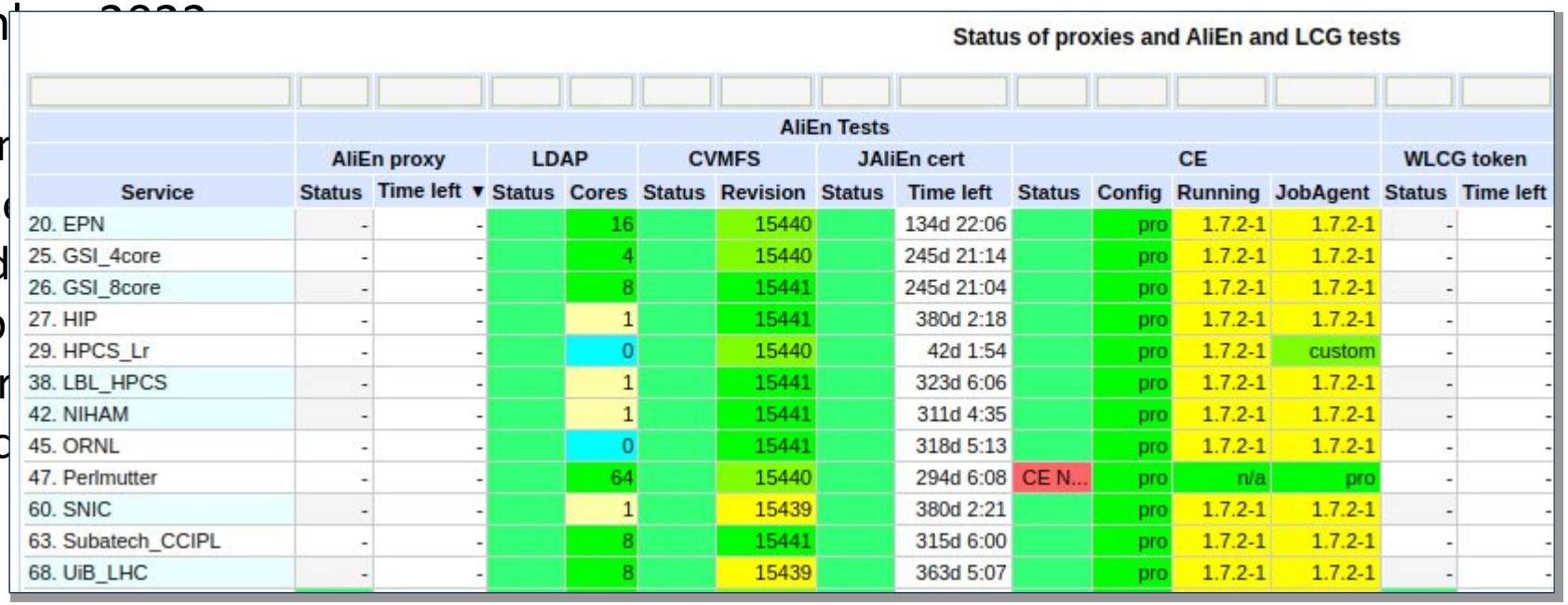
- By default, **all** Grid jobs are wrapped in a common **EL** container by JAliEn pilot
  - Provides a **tried-and-tested environment** on CentOS 7.9 across sites/nodes
  - Additional **isolation** from WN host
- Image as a sandbox directory located in CVMFS at
  - `/cvmfs/alice.cern.ch/containers/fs/singularity/centos-latest`
- Build recipe available on Gitlab
  - User PRs possible for package requests
- Two optional images can be set by **site**
  - **Alma 8.7**: For newer payloads (no ROOT5) and GPUs
  - **Alma 9.1:** Testing only (no production use)
- **GPUs are supported** through *Apptainer*
  - Compatibility check for supported container frameworks by JAliEn
  - GPUs auto detected, with flags/mounts added as needed

From last report

- JAliEn now on **all** Grid sites. No more (in)active AliEn instances
  - Final AliEn site (DCSC_KU) migrated 15/02/23
  - From November 2022:
    - **SiteSonar** integration (JobBroker matching)
    - Legacy container frameworks removed (SingularityCVMFS)
    - Emergence of EL8/EL9 sites across the ALICE Grid:
      - Already supported by JAliEn
      - Possibility of finer control of resources, without elevated privileges via **Cgroups v2**
      - Preparations for testing of Cgroups v2 compatibility for JAliEn
      - Subject to site availability

From last report

- JAliEn now on **all** Grid sites. No more (in)active AliEn instances
  - Final AliEn site (DCSC_KU) migrated 15/02/23
  - From Novem
    - **SiteSonar**
    - Legacy cor
    - Emergenc
      - Alread
      - Possib
      - Prepar
      - Subjec



Status of proxies and AliEn and LCG tests

	AliEn Tests													
	AliEn proxy		LDAP		CVMFS		JAliEn cert		CE				WLCG token	
Service	Status	Time left ▼	Status	Cores	Status	Revision	Status	Time left	Status	Config	Running	JobAgent	Status	Time left
20. EPN	-	-		16		15440		134d 22:06		pro	1.7.2-1	1.7.2-1	-	-
25. GSI_4core	-	-		4		15440		245d 21:14		pro	1.7.2-1	1.7.2-1	-	-
26. GSI_8core	-	-		8		15441		245d 21:04		pro	1.7.2-1	1.7.2-1	-	-
27. HIP	-	-		1		15441		380d 2:18		pro	1.7.2-1	1.7.2-1	-	-
29. HPCS_Lr	-	-		0		15440		42d 1:54		pro	1.7.2-1	custom	-	-
38. LBL_HPCS	-	-		1		15441		323d 6:06		pro	1.7.2-1	1.7.2-1	-	-
42. NIHAM	-	-		1		15441		311d 4:35		pro	1.7.2-1	1.7.2-1	-	-
45. ORNL	-	-		0		15441		318d 5:13		pro	1.7.2-1	1.7.2-1	-	-
47. Perlmutter	-	-		64		15440		294d 6:08	CE N...	pro	n/a	pro	-	-
60. SNIC	-	-		1		15439		380d 2:21		pro	1.7.2-1	1.7.2-1	-	-
63. Subatech_CCIPL	-	-		8		15441		315d 6:00		pro	1.7.2-1	1.7.2-1	-	-
68. UiB_LHC	-	-		8		15439		363d 5:07		pro	1.7.2-1	1.7.2-1	-	-

From last report

- JAliEn now on **all** Grid sites. No more (in)active AliEn instances
  - Final AliEn site (DCSC_KU) migrated 15/02/23
  - From November 2022:
    - **SiteSonar** integration (JobBroker matching)
    - Legacy container frameworks removed (SingularityCVMFS)
    - Emergence of EL8/EL9 sites across the ALICE Grid:
      - Already supported by JAliEn
      - Possibility of finer control of resources, without elevated privileges via **Cgroups v2**
      - Preparations for testing of Cgroups v2 compatibility for JAliEn
      - Subject to site availability

Key JAliEn/JA changes (~past 10 releases)

- **Fixes:**
  - Flag and work around potentially broken environments (no Alienv due to broken CVMFS Python3 on missing HEP_OSlibs)
  - Cleanup processes before file upload, to prevent files changes during upload
  - Fix for Slurm always reporting 1 job
  - Thread safety fixes: removal/rewrites of agent code relying on java.nio
  - Prevent jobs from failing on old (EL7) containers with newer GPU drivers
  - Prevent JAliEn cmd extras being read by payload
    - Also fix for exit code pollution caused by same decorating args
- **General improvements:**
  - Graceful shutdown across site components on signal
  - Core file checker will ignore Fluka core*.inp file
  - Max JVM heap size increased for JAs
  - Auto resubmit failed jobs if uncaught exceptions
  - Improved monitoring to prevent empty agents/runners
  - In-memory key stores to avoid logging keys
- **New features**
  - Retry delay added for failed jobs to prevent TQ “black holes”
  - Protection against uploading large (auto selected) logs
  - Allow specifying a custom containerizer and separate containerizer binaries
  - Enable optional support for using 64bit Java
  - Inclusion of Marta’s unused core allocation changes
  - Custom cmd insertion possible on JDL DebugTag

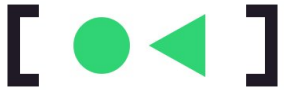
Full changelog at
http://alien.cern.ch

Using cgroups v2 in JAliEn – initial experiences

- Support for use of cgroups v2 added in JAliEn **1.6.3**
  - Used to control memory allowance of containers on supported sites
  - **EL8** with tweaked config needed, or **EL9**
  - Still **not a guarantee**, as multiple requirements must be satisfied
    - Cgroups v2 must be in “unified hierarchy mode”
    - Kernel version >= 4.15
    - Systemd version >= 224
    - “Systemd cgroups” enabled in Apptainer config
    - Systemd configured to delegate cgroups controllers to standard users
  - In other words, used by **very few** sites

Using cgroups v2 in JAliEn – initial experiences (2)

- No “smooth sailing” for where there should theoretically be support either
- Few available WNs that have support are mainly on EL8
  - Disabled by default
  - In some cases seen as mounted, but remains unavailable
  - Needs to be explicitly enabled by site admin
- Successfully enabling it on EL8 **still** not a guarantee it will work
  - Systemd version bundled with EL8 comes with a delegation bug
    - Affects use of unprivileged containers with cgroups v2
    - Requires a workaround to be put in place



Testing EL8 compatibility and cgroups v2

- Not enough to detect/verify compatibility through
  - Checking mounts
  - Checking config entries
  - Ensuring workaround present
- The same applies to using a simple test container before job start
  - May run normally at start, but crash after a small duration
    - Eventually responds with a cgroups v2 permissions error
- No **reliable** way of knowing if an **EL8 host** can run with cgroups v2!
  - Disable for EL8 altogether?

Testing EL8 compatibility and cgroups v2

- Not enough to detect/verify compatibility through
  - Checking mounts
  - Checking config entries
  - Ensuring workaround present
- The same applies to using a simple test container before job start
  - May run normally at start, but crash after a small duration
    - Eventually responds with a cgroups v2 permissions error
- No **reliable** way of knowing if an **EL8 host** can run with cgroups v2!
  - Disable for EL8 altogether?

Alternative: Use SiteSonar
– Run longer probe in BG

Transitioning to 64bit JDK

- JAliEn JobAgents originally on **32bit Java**
  - Workaround for Java's aggressive virtual memory allocation
    - Can be substantial on powerful WNs
  - No way to otherwise change or configure this behaviour
  - 32bit prevented the processes from being flagged for overusing resources
    - Allowed for quick and painless rollout of JAliEn across sites
    - Very small system footprint
- Not a "perfect" workaround
  - Requires library patching to ensure compatibility on newer hosts
  - Concerns regards to how long 32bit JDKs will remain maintained
    - New builds and security updates still produced by e.g. Azul (for now)
  - Increasing amounts of **new workarounds** needed to keep using it

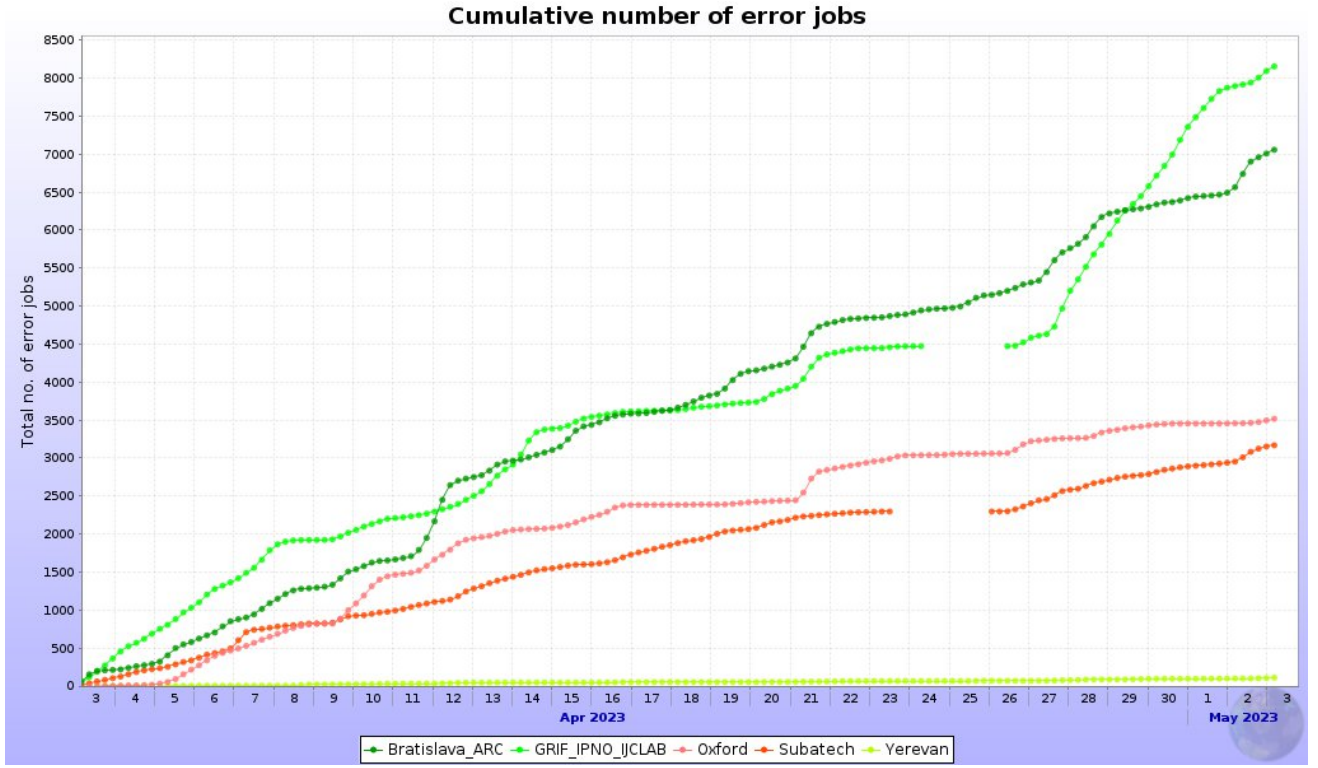
Transitioning to 64bit JDK (2)

- OpenJDK code examined for options / possible tweaks
  - Most virtual memory consumed by garbage collection
    - By default, **parallelised**, with each thread getting its own memory allocation
    - Can quickly spin out of hand on big WNs
  - Possible fix: switch to **serial** GC
    - Included in JDKs as an option
    - Much more restrained (virtual) memory allocation
  - Still *more* than the 32bit version
    - Host system dependent
  - But appears to be reported correctly by monitoring

OpenJDK

Transitioning to 64bit JDK (3)

- 64bit Java with serialised GC added as an option in **JAliEn 1.7.2**
  - Several sites switched around 26/04/23, with no visible issues so far*



Summary

- JAliEn is now used on **all sites**
  - Version status on proxies page
  - Changelog on http://alien.cern.ch
- Also running on **EL8/EL9** WNs when used at sites
  - **Cgroups v2** features used when possible
    - Limited support so far on **EL8**, with no reliable way to test
      - Block **EL8** / wait for **EL9**?
    - Could possibly be tested via **SiteSonar**
- Transition to fully **64bit** JAliEn
  - Avoids encountered limitations of 32bit
  - Moving forward with **serial** GC
  - Initial deployment promising
    - Will be **default** starting with **JAliEn 1.7.3**

Thank You

[Questions, comments?]

email: mstoretv@cern.ch