UNIVERSITY OF CAMBRIDGE

Job Accounting with XDMoD

Stuart Rankin

sjr20@cam.ac.uk

Research Computing Services (http://www.hpc.cam.ac.uk/)
University Information Services (http://www.uis.cam.ac.uk/)

4th April 2019 / IRIS F2F Meeting

Job accounting on the CSD3/Cumulus HPC facility

A complex service with a complex user base:

- Multiple, heterogeneous clusters (with different allocation units).
- Multiple service levels (paying & non-paying).
- Multiple user classes (internal UCAM, EPSRC Tier2, STFC DiRAC, IRIS, industrial).
- Resource allocations are controlled through SLURM.

Job accounting on the CSD3/Cumulus HPC facility

A complex service with a complex user base:

- Multiple, heterogeneous clusters (with different allocation units).
- Multiple service levels (paying & non-paying).
- Multiple user classes (internal UCAM, EPSRC Tier2, STFC DiRAC, IRIS, industrial).
- Resource allocations are controlled through SLURM.

Job accounting on the CSD3/Cumulus HPC facility

A complex service with a complex user base:

- Multiple, heterogeneous clusters (with different allocation units).
- Multiple service levels (paying & non-paying).
- Multiple user classes (internal UCAM, EPSRC Tier2, STFC DiRAC, IRIS, industrial).
- Resource allocations are controlled through SLURM.

Job accounting on the CSD3/Cumulus HPC facility

A complex service with a complex user base:

- Multiple, heterogeneous clusters (with different allocation units).
- Multiple service levels (paying & non-paying).
- Multiple user classes (internal UCAM, EPSRC Tier2, STFC DiRAC, IRIS, industrial).
- Resource allocations are controlled through SLURM.

Job accounting on the CSD3/Cumulus HPC facility

A complex service with a complex user base:

- Multiple, heterogeneous clusters (with different allocation units).
- Multiple service levels (paying & non-paying).
- Multiple user classes (internal UCAM, EPSRC Tier2, STFC DiRAC, IRIS, industrial).
- Resource allocations are controlled through SLURM.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

Open XDMoD

http://open.xdmod.org/7.5 (most recent is 8.0)

- Open source version of the XDMoD developed for XSEDE.
- PHP web application (running on an RHEL7 OpenStack VM).
- Data warehouse of HPC job records.
- Supports typical HPC job queries against a hierarchy of users and defined resources.
- Application kernel and SUPReMM modules not yet tested.

How XDMoD ingests data

- Raw job data and association information extracted from the batch scheduler using native tools.
- A hierarchy of groups (PIs) is defined and imported.
- Job data is shredded (loaded into the database).
- Data is ingested (processed, prepared and optimised for querying).

How XDMoD ingests data

- Raw job data and association information extracted from the batch scheduler using native tools.
- A hierarchy of groups (PIs) is defined and imported.
- Job data is shredded (loaded into the database).
- Data is ingested (processed, prepared and optimised for querying).

How XDMoD ingests data

- Raw job data and association information extracted from the batch scheduler using native tools.
- A hierarchy of groups (PIs) is defined and imported.
- Job data is shredded (loaded into the database).
- Data is ingested (processed, prepared and optimised for querying).

How XDMoD ingests data

- Raw job data and association information extracted from the batch scheduler using native tools.
- A hierarchy of groups (PIs) is defined and imported.
- Job data is shredded (loaded into the database).
- Data is ingested (processed, prepared and optimised for querying).

Customised ingestion

Custom slurm helper script extracting association and job information from SLURM DB:

- Split out each sub-cluster (set of partitions) into paid and unpaid virtual clusters.
- Normalise the PI/group field from the SLURM account:
  - Use SLURM hierarchical associations.
  - UCAM groups correspond to individual PIs (parents of SLURM accounts).
  - CORE, DiRAC and Tier2 groups correspond to project-specific SLURM accounts.
- Rationalize and resolve special cases.
- Run nightly by cron.

Customised ingestion

Custom slurm helper script extracting association and job information from SLURM DB:

- Split out each sub-cluster (set of partitions) into paid and unpaid virtual clusters.
- Normalise the PI/group field from the SLURM account:
  - Use SLURM hierarchical associations.
  - UCAM groups correspond to individual PIs (parents of SLURM accounts).
  - CORE, DiRAC and Tier2 groups correspond to project-specific SLURM accounts.
- Rationalize and resolve special cases.
- Run nightly by cron.

Customised ingestion

Custom slurm helper script extracting association and job information from SLURM DB:

- Split out each sub-cluster (set of partitions) into paid and unpaid virtual clusters.
- Normalise the PI/group field from the SLURM account:
  - Use SLURM hierarchical associations.
  - UCAM groups correspond to individual PIs (parents of SLURM accounts).
  - CORE, DiRAC and Tier2 groups correspond to project-specific SLURM accounts.
- Rationalize and resolve special cases.
- Run nightly by cron.

Customised ingestion

Custom slurm helper script extracting association and job information from SLURM DB:

- Split out each sub-cluster (set of partitions) into paid and unpaid virtual clusters.
- Normalise the PI/group field from the SLURM account:
  - Use SLURM hierarchical associations.
  - UCAM groups correspond to individual PIs (parents of SLURM accounts).
  - CORE, DiRAC and Tier2 groups correspond to project-specific SLURM accounts.
- Rationalize and resolve special cases.
- Run nightly by cron.

Customised ingestion

Custom slurm helper script extracting association and job information from SLURM DB:

- Split out each sub-cluster (set of partitions) into paid and unpaid virtual clusters.
- Normalise the PI/group field from the SLURM account:
  - Use SLURM hierarchical associations.
  - UCAM groups correspond to individual PIs (parents of SLURM accounts).
  - CORE, DiRAC and Tier2 groups correspond to project-specific SLURM accounts.
- Rationalize and resolve special cases.
- Run nightly by cron.

Hierarchy

Custom script building the hierarchy:

- XDMoD supports a 3 level hierarchy into which PI/groups are inserted.
- Preserve SLURM DB hierarchical structure with PI, Dept and School for internal UCAM users.
- For other SLURM accounts seed the XDMoD hierarchy using the SLURM Organization field.

Hierarchy

Custom script building the hierarchy:

- XDMoD supports a 3 level hierarchy into which PI/groups are inserted.
- Preserve SLURM DB hierarchical structure with PI, Dept and School for internal UCAM users.
- For other SLURM accounts seed the XDMoD hierarchy using the SLURM Organization field.

Hierarchy

Custom script building the hierarchy:

- XDMoD supports a 3 level hierarchy into which PI/groups are inserted.
- Preserve SLURM DB hierarchical structure with PI, Dept and School for internal UCAM users.
- For other SLURM accounts seed the XDMoD hierarchy using the SLURM Organization field.

Hierarchy

Custom script building the hierarchy:

- XDMoD supports a 3 level hierarchy into which PI/groups are inserted.
- Preserve SLURM DB hierarchical structure with PI, Dept and School for internal UCAM users.
- For other SLURM accounts seed the XDMoD hierarchy using the SLURM Organization field.

The end result

What does it look like?

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).

Issues

- XDMoD supports only a 3 level hierarchy.
- Role-based access is poorly developed.
- LDAP authentication works, but authorisation required hacking.
- Only understand CPU hours as a unit.
- Hierarchical structure and cluster dimensions cannot vary with time.
- Pen testing revealed some horrors (reported).