IML Roadmap and Community

Joe Grund
IML Team Lead
jgrund@whamcloud.com
Agenda

► IML Background / Overview
► Current Work
► Potential Future Work
► Where to find project / communicate with team
Background

► Integrated Manager for Lustre (IML) is an open source suite of tools for deploying, managing, and monitoring Lustre filesystems
► IML simplifies Lustre administration with intuitive interfaces and near real-time feedback
► Works with new and existing Lustre installations
► Monitors performance and system health
► Proven in production at hundreds of sites
► Used successfully in environments with over 100 OSTs.
Background - Deployment

- Deploy Lustre filesystems from one centralized location
- Near-realtime feedback
- Bring filesystem online from first principles or deploy monitoring for an existing filesystem
- Deploy specialized assets, HSM
- Add more assets over time
Configure / change state of Lustre and related components

- Uses state-machine to reach end state from different starting points
  - Starting LNet, state machine ensures packages are installed + kernel modules loaded before bringing LNet up

Handle recovery situations fencing, failover

- Automatic configuration of High Availability through Corosync, Pacemaker, and PDU / IPMI integration
Background - Monitoring

► Holistic system metrics
  • Rich visualizations
  • Drill into filesystem, target, server
  • Find and monitor top jobs
► Aggregate logs across cluster
► HSM Copytool activity monitoring
► Alerts to cluster issues
  • GUI / Email / API
► Searchable command / event / alert log / history
Background - Development History

**Chroma 1.0 – 2012**
- Bare Metal install
- Start / Stop / Add / Remove FS and targets
- GUI, REST, CLI interfaces
- HA over Lustre targets

**IEEL 1.0 – 2013**
- Chroma renamed to IML

**IML 2.1 – 2014**
- Heatmap visualization.
- Job stats collection and display
- Hierarchical Storage Management (HSM) support and display
- Custom profile Support
- Near realtime support

**IML 2.2 – 2015**
- ZFS Monitored mode support
- Enhanced parallel server deployment
- Enhanced command display / drilldown

**IML 3.0 – 2016**
- Pacemaker / Corosync config
- NID config
- Queryable status
- Architectural improvements

**IML 3.1 - 2016**
- ZFS support for managed mode.
- Near realtime jobstats monitoring
- Tree-view

**IML 4.0 - 2017**
- First open source release
- Tracks Lustre LTS
- appliance -> services

**IML 4.0.7 - 2018**
- IML renamed to Integrated manager for Lustre
Background - Upgradability

- Support upgrades from closed-source IEEL versions to open-source Whamcloud versions
  - Documents describe how to upgrade from
Current Work

- Increase scalability of device detection
- Increase modularity of components
- Start moving towards separate management / monitoring code paths
- Add standard deployment with Docker
- Updated metrics
Current Work

► **libzfs integration**
  - [https://github.com/whamcloud/integrated-manager-for-lustre/issues/535](https://github.com/whamcloud/integrated-manager-for-lustre/issues/535)

► **ZED integration**
  - [https://github.com/whamcloud/integrated-manager-for-lustre/issues/536](https://github.com/whamcloud/integrated-manager-for-lustre/issues/536)

► **Reactive Architecture**
  - [https://github.com/whamcloud/integrated-manager-for-lustre/issues/533](https://github.com/whamcloud/integrated-manager-for-lustre/issues/533)

► **Full Modularity**
  - [https://github.com/whamcloud/integrated-manager-for-lustre/issues/534](https://github.com/whamcloud/integrated-manager-for-lustre/issues/534)

► **Docker support**
  - [https://github.com/whamcloud/integrated-manager-for-lustre/issues/705](https://github.com/whamcloud/integrated-manager-for-lustre/issues/705)

► **Re-worked metrics**
  - [https://github.com/whamcloud/Lustre-monitor/issues/1](https://github.com/whamcloud/Lustre-monitor/issues/1)
Current Work - libzfs integration

- IML looking to utilize libzfs for multiple purposes
  - Lower level interface over invoking commands / parsing CLI output
  - Fine grained collection of pools / datasets / props / VDEV tree
    - easy to collect more data later
  - Useful for monitoring + management enhancements
Current Work - ZED Integration

- IML looking to use ZED in for multiple purposes
  - Discovery of pool / dataset / property / VDEV changes
    - This is currently a manual scanning process
    - Will allow for closer to real-time changes to propagate
    - Better scaling (very fast for very large sites)

- Surfacing alerts in the IML GUI / API
  - Alerting on critical events across a cluster
  - Searchable history of all events across a cluster
Current Work - Reactive Architecture

► IML looking to flip device discovery from push to pull
  • Adds further scalability
  • Has lower resource usage
  • More responsive
  • Current iteration uses polling + serial introspection of devices
Current Work - Modularity

► IML looking to deliver itself completely via RPM
  • No more tarball
  • Ship everything via Fedora Copr

► Benefits
  • Semver minor updates via yum update
  • Components evolve independently

► Approach
  • Two repos, devel + non-devel
  • devel gets continuous updates
  • non-devel gets production ready packages.

► Those not wanting stream of updates can disable upgrades / perform offline install / upgrade
Current Work - Docker Support

- Adding support for docker on the manager
  - Can run IML manager on any OS that supports latest Docker
- Images built / available on docker cloud
  - [https://cloud.docker.com/swarm/imlteam/repositories/list](https://cloud.docker.com/swarm/imlteam/repositories/list)
- Provide an (optional) installer that fully configures IML upon first install.
- Future possibility of HA for IML manager + collocation on storage servers
Current Work - Re-worked Metrics

► Increasing range of Lustre versions supported
► Solution: utilize `lctl get_param + lnetctl export`
  - Create a standalone binary that only aggregates stats and outputs in JSON / YAML (other formats to follow)
  - Reusable tool, zero runtime dependencies

► Solution: utilize timeseries database + Grafana dashboard
  - Allow for stat storage and display unassociated with IML
  - Allow for operators to create custom charts as needed
  - Embed charts back into IML dashboard
Potential Future Work

► Enhanced Deployment
  • IML should make it even easier to setup Lustre
    o Deploy large scale clusters with minimal operator intervention
      – Describe ideal cluster state
      – Expose variants as scalable UI widgets
      – Deploy installation in parallel with a single click

► Lustre Snapshot Management
  • IML should be able to manage Lustre snapshots via GUI
    o Schedule snapshots for filesystems at some regular interval
    o Ad-hoc snapshot on filesystem(s)
    o View / delete previous snapshots
    o Rollback to a given snapshot
    o Rename a snapshot
    o Take snapshot at key points (i.e. Lustre upgrade)
Potential Future Work

► Full ZFS Management
  • IML should provide full ZFS management
    o Show all pools and datasets across a cluster
    o Provide drill-down navigation to elicit more detail on a selected target
    o Show the status of pools and datasets
      – Where imported, mounted, error conditions, configuration
    o Management
      – Create zpools / datasets
      – Support creation of various pool configurations: RAID-Z, Mirrored...
      – JBOD enclosure GUI

► I18n Support
  • IML text currently English, but IML is used all over the world
  • Modify/contribute *.po files consumed by services
Where to find IML

► 4.0.x Releases: https://github.com/whamcloud/integrated-manager-for-lustre/releases
► RPMs: https://copr.fedorainfracloud.org/coprs/managerforlustre/
► Help docs: https://whamcloud.github.io/Online-Help/
► Issues: https://github.com/whamcloud/integrated-manager-for-lustre/issues
► Direct line of communication via: https://gitter.im/whamcloud/integrated-manager-for-lustre
► Demo sandbox through vagrant: https://github.com/whamcloud/Vagrantfiles/
► Email: iml@whamcloud.com
Closing

► IML is a project with a long history and continues advancing
  • In production at hundreds of sites worldwide
  • IML 4.0 was first open source release in Oct 2017
  • Eight maintenance updates to 4.0 release
  • Next release IML 4.1 brings even more enhancements
  • Continue iterating / evolving

► Possible to upgrade from IEEL to IML 4
  • Upgrade docs for both 2.4.x and 3.1.x lines