Adaptive Computing: HPC as a Service, Privacy by Design

Intro on Moab 9.0.1, Viewpoint, & Moab Cloud  May 2016

Bernhard Schott
Adaptive Computing EMEA
Agenda

- Adaptive Computing update on Moab 9 – selected topics
  - Moab
    - Overview, Concepts
    - Advanced Scheduling
  - ViewPoint Portal
    - Ease of use, Architecture
    - Remote Viz
  - HPC on Demand
    - Elastic Computing
    - Moab Cloud Suite
- Questions and Answers
Adaptive Computing

- Largest provider of HPC job/workload management software

- Pioneer & innovator in scheduling and optimization
  - 80+ patents issued or pending applicable to world-class HPC, cloud and big data solutions

- Global partnerships include SGI, Intel, HP, Lenovo, Cray, & Microsoft

- Managing Leading Fortune 500 and Top 500 Systems
Moab Intelligence Engine

Core Moab® Technology
- 15+ years battle tested
- Patented (80+ Patents)
- Mimics real-world decision-making

Multi-dimensional Policies Optimize Across:
- Workload/application requirements
- Priorities and SLAs
- Time (real-time and future, predictive)
- Heterogeneous resources
Intelligent Workload Optimization

Moab® HPC Suite - Enterprise Edition

- CLI or Viewpoint End user Portal
- Viewpoint Admin Dashboard
- External System Report Interface

APIs (Web Services, CLI, etc.)

Moab Intelligence Engine

Decisions, policies, scheduling, allocation, & orchestration

Accounting

Resource Managers

Queue Mgr (Torque/Other)

Health Monitor
Provisioning
Storage
Network
Non-traditional

Other Resources:
- Licenses
- Disk
- Etc.

Workload

HW Resources
Moab HPC Suite - Enterprise Edition 9.0.0

This documentation grouping also contains add-on components and features. Refer to your license to determine the applicable documentation.

- **Release Notes**
  - HTML
  - PDF

- **Installation and Configuration Guide**
  - HTML
  - PDF

- **TORQUE Resource Manager Administrator Guide**
  - HTML
  - PDF

- **Moab Workload Manager Administrator Guide**
  - HTML
  - PDF

- **Moab Accounting Manager Administrator Guide**
  - HTML
  - PDF

- **Moab Web Services Reference Guide**
  - HTML
  - PDF

- **Moab Insight Administrator Guide**
  - HTML
  - PDF

- **Moab Viewpoint Administrator Guide**
  - HTML
  - PDF

- **Moab Docker Integration Reference Guide**
  - HTML
  - PDF
Moab structural overview: single logical cluster

The resource manager manages the nodes, launches and controls the workload. Torque can include resources from multiple physical “groups of resources”.

All compute nodes would belong to one “partition” and could be managed and used transparently as one single system image.
Moab structural overview: Multi-Cluster

Resources can be of different type:
Big SMP, clusters, new architectures … and adding Nitro to perform HTC
Moab structural overview: Grid

Two or more sites, peer or hierarchical configurations

Moab Intelligence Engine

Web Services API

Resource Mgr
TORQUE_A

Resource Mgr
TORQUE_B

Resources from all sites are available to all users, controlled by policies. Can be combined with the previously described configurations.

Large scale reference use case: www.hlrn.de

Der HLRN-Verbund
Jobs can be submitted to Viewpoint portal by **GUI**, To the scheduler by CLI command **msub**, To the resource manager by **qsub**

Execution for all jobs will be the same, based on Moab policies.
Advanced Scheduling
Optimized, Intelligent Scheduling

- Workload packing and **continuous** scheduling **optimization**
- Considers all job requests in queue in priority order
- Fully **uses all** available capacity to maximize throughput (backfill)
- Provides **up to 20%** utilization performance **improvement**
Resource Job Timeline

- Individual node usage
- Digestible Tetris-like graph
- Enables admins to monitor workload and resource utilization
- Identifies where system utilization can be improved
- Jobs presented in boxes with different sizes and shades to indicate
  - Number of cores
  - Size (darker the box the larger the job)
Advanced Scheduling: conquer fragmentation

- Moab is a planning reservation scheduler
  - Moab creates a scheduling plan into the future
  - Moab re-iterates the scheduling plan every scheduling cycle
    - Adjusts for changes and actual job completion times
- Control fragmentation: Moab resource allocation
  - Systematically allocates resources from “one side of the cluster”, e.g. from the highest compute node downwards.

- Virtually un-fragmentable cluster
  - Moab places new resource requirements as much into the future so there are sufficient resources to find.
Moab is a planning reservation scheduler
- Moab creates a scheduling plan into the future
- Every object in the scheduling plan is a reservation
- Reservations can be booked for periodical use, for example 4 reports/day
Advanced Scheduling: Reservations

- Moab is a planning reservation scheduler
  - Moab creates a scheduling plan into the future
  - Every object in the scheduling plan is a reservation
  - Reservations can be booked for periodical use, for example 4 reports/day
**Advanced Scheduling: day & night reservations**

- **Moab reservation use case**
  - Define reservation #17 for daytime resources for project A
    - Example: 100 procs (=job slots)
  - Define reservation #18 for nighttime resources for project A
    - Example: 400 procs (=job slots)

- **Submit jobs**
  - `<user project A>\$ qsub -l reservation 17,18 myjobscript`
  - Moab will verify that the user is actually member of project A
  - Moab will allocate resources from reservation #17 and/or #18
    - “xor” since here there is no overlap between #17 and #18
  - During daytime, only jobs using 100 procs in total are started
  - During nighttime, more jobs are started, using up to 400 procs
  - At the end of the nighttime reservation, the running jobs may be allowed to complete.
  - All new jobs comply to the daytime reservation
Quality of Service Policies
Privilege Management

- **Target Service Levels**
  - Priority/Fairshare
  - Application Response Time

- **Resource Access**
  - Quota Exceptions
  - Licenses
  - Specialized Resources (Phi, GPGPUs, etc.)

- **Service Access**
  - On Demand Power
  - Dynamic Provisioning (Resource or Network)
  - Deadline Guarantees
  - Topology Aware Scheduling

- **Escalation**
  - Preemption
  - Cloud-bursting

- **SLA-based Pricing**
ViewPoint Portal
Ease-of-use Driven Productivity

May 2016
Ease-of-use Driven Productivity

**User**

- Increase productivity of end users with **easier and faster portal-based submission** of tasks/jobs
- Expand HPC user base to **include non-IT skilled personnel**
- **Automate best practices** information into submission process in order to speed submission, reduce error and optimize processing speed
- **Easy to use Remote Visualization**

**Admin**

- Enable admins to **manage user requests with less time** due to best practices based templates, and user feedback which **helps users help themselves**
- Admins gain quick **visibility** into system and workload status and workload **troubleshooting**
Moab ViewPoint portal architecture

- Job submission webGUI
- Viewpoint uses Python and HTML5.
- Django (Web Framework)
- ac-python-api wraps calls to mws using python libraries
- Viewpoint uses Moab Insight database
Moab Web Services (MWS)

- Flexible RESTful interface using JSON data sets
- Easy way to interact with Moab objects and properties
- Easy to integrate with other Adaptive and 3rd party tools
- Allows for rapid custom user interfaces to be created
- Secure SSL/TLS connections
- Java/Python API classes and interfaces
- Fast and seamless integration with Moab and other Adaptive tools/clients
MWS – RESTful API to Moab

- Query MWS on Moab data & state
- Example: “Get Single Job”

GET \[http://localhost:8080/mws/rest/jobs/<name>\]?api-version=3

From command-line:
```
```

Note: in this example <name> => Moab.15

- Supported methods: GET, POST, PUT, DELETE
MWS – RESTful API to Moab

- **JSON Response example:**

```json
{
    "arrayIndex": null,
    "arrayMasterName": null,
    "attributes": [],
    "blocks": [
        {
            "category": "jobBlock",
            "message": null,
            "type": null
        }
    ],
    "bypassCount": 0,
    "cancelCount": 0,
    "commandFile": "/tmp/test.sh",
    "commandLineArgs": null,
    "completionCode": ...
}

<3 pages output, see page 167ff in MWSEnterprise-8.1.2.pdf>
End User Job Submission Portal
Ease-of-Use Driven Productivity

- Portal-based Submission
- Application Templates
- Script Builder
- File Management
End User Submission Portal

- **Job/Task Submission**
  - Submit workload without needing to learn complex CLI commands
  - Avoid errors with simple point and click rather than error-prone command line syntax
  - Utilize best-practices application templates to get results more efficiently

  - Point and Click
  - Intelligent Dropdown options
  - Customized to your applications
  - Submit standard jobs or arrays
Application Template Form Builder

- Application Templates
  - Reduce complexity for each application by hiding un-needed fields
  - Add application-specific fields to simplify script use
  - Implement best practices into default values
  - Share templates broadly or with specific users

- Select if Fields are Editable
- Select if Fields are Viewable
- Set Default Values
- Add Custom Fields
- Control Template Permissions
Error Reducing Script Builder

- **Script Builder**
  - Incorporate standard and custom per-application variables
  - Reduce errors with drag-and-drop variable insertion and usage counting
  - Application-specific fields reduce in script complexity
  - Share scripts with simple import/export functions

- Drag and Drop Variables
- Edit Script from Browser
Customize Script – Script builder (more detail)
Self-help Oriented Job Details

- **Job Details**
  - Self service to job state details and messages that explain what occurred and why
  - Users see the difference between what resources they asked for and what they used
  - Per task level tracking

- **Job State Details**
- **Analysis of Scheduler Decisions**
- **Resource Utilization**
Simple Integrated File Management

- **File Management**
  - Easily browse to input, output, and error files
  - Perform basic functions like moving, renaming, compressing and deleting files
  - Quickly validate output files with a browser-based preview functionality

- **Security**
  - Below FS granularity: configurable access to directories or links

- Quick Navigation
- Move / Rename
- Delete
- Compress / Uncompress
- File Preview
Viewpoint Remote Viz
Moab ViewPoint Remote Viz

- Select Remote Visualization Template
Moab ViewPoint Remote Viz

- **See Job Details**

  ![Moab ViewPoint Remote Viz screenshot]

  **Job Details**
  - **Job Id:** 177 (paraview)
  - **Submission Script:** N/A
  - **Reservation Name:** N/A
  - **Template:** Paraview

  **Status:** RUNNING

  **Remote Viz Session**
  - **Session Name:** paraview
  - **Session Command:** /home/hpotter/run...
  - **Session Geometry:** 1024x768

  **Job Messages**

  **Credentials**
  - **User:** hpotter
  - **Group:** hpotter
  - **Account:** engineering...
  - **Class:** batch
  - **Quality of Service:** None

  **CPU Statistics**

  **Time Frame**
  - **Start Time:** 2015-11-20 13:39:49
  - **Duration:** 00:12:00:00
  - **Completion Time:** None
  - **Actual Duration:** 00:00:00:36
Moab ViewPoint Remote Viz

- **Access applications remotely**
  - Applications
  - Interactive Terminal
  - KDE or Gnome Desktop

- **Adjust Image Quality:**
  - Adequate Frame Rate, Frame Windows according to network quality
Moab ViewPoint Remote Viz operation

1) User selects Remote-Viz template

3) Job details page provides link

4) Standard browser uses the Remote Viz session URL

2) Job gets started by Moab

Moab/Torque launches Remote Viz Server

Gateway-URL

Remote Viz Gateway

Moab Intelligence Engine

Resource Mgr
TORQUE

Web Services API
Admin Portal
Unify Administration & Reporting

- Administrator Reporting
- Workload Status Tracking
- Simplified Job Resource Viewing
Administrative Reporting

- **Dedicated Resources**
  - Visualize resource utilization
- **Node Summary**
  - Instant overview of node state
- **Workload Summary**
  - Color coded display of running, idle, removed and completed jobs
- **Search Workload List**
  - Quickly find specific jobs / tasks

- Dedicated System Resources
- Node Summary
- Workload Summary
- Searchable Workload List
Workload Status Tracking

- Comprehensive Job List
- Searching & Filtering
  - Search – based on job ID, submission ID
  - Filter – based on running, suspended, blocked, deferred, idle and other job states
- Per Job Detail View
  - Obtain key details like start or completion time, duration, priority, and other requirements
  - Cancel or modify jobs (priority, wall time, and other attributes)

- Review Job State
- Search / Filter Job List
- Per Job Details
- Cancel or Modify Jobs
Node Usage Tracking

- **Increases Node Visibility**
  - Review comprehensive job resources prior to admin actions
  - Respond to node failures
  - Reroute jobs to available nodes
  - Maximize system performance

- **Node Tab Includes:**
  - Current node list
  - Searching and filtering
  - Individual node details

- List Nodes with State
- Search / Filter Nodes
- Detail View of Nodes
Resource Job Timeline

- Tetris-like view of planned jobs per node
- Monitor resource utilization
- Jobs presented in boxes with different sizes and shades to indicate duration and number of cores requested
- Size (the darker the box, the larger the job)
- View packing efficiency
- Estimate node draining impact
- Identify ideal nodes to reserve

Adaptive Computing
Summary
Ease-of-use driven productivity

User
- Faster and easier submission
- Reduce knowledge requirement
- Reduce errors
- Automate best practices into submission
- Users improve efficiency with system feedback

Admin
- Manage users in less time
- Instant visibility into system and workload status
- Faster troubleshooting

Achieve more results from more productive users
HPC on Demand
Use cases & components
Elastic Computing – Add/Remove Local Shared Resources Use Case

- **Add/Remove resources to an existing cluster**
  - **Add** based on “Backlog Completion Time” threshold
    - Example: If backlog > 600 seconds (10 minutes) add a specified number of nodes for a specified duration (a.k.a. request geometry)
    - Option: Set request geometry to size & duration of highest priority job
  - **Remove** based on specified duration or “Node Idle Purge Time”
    - Example: If node is idle > 300 seconds (5 minutes) remove node

**Summary:**
With Moab’s elastic computing wait times reduce & service levels improve.

- HPC Users
- Workload Queue
- Administrators
- Moab
- OpenStack Provisioning & Virtualization

Requires: Externally supplied provisioning trigger setup.
Elastic Computing – Multi-tenant Cluster Use Case

- Rent access to self administered resources
- Dynamically expand/contract resource allocation
- Account for usage and SLAs
- Privacy by design

Coordinator Moab

http://www.hpc4health.ca/

Summary:
Set up, multiple independently managed tenant clusters with unified and per tenant accounting.
Docker Support

**Features:**
- Create, Cancel, Checkpoint and Requeue containers associated serial jobs
- Upcoming: container groups for parallel workload
- Manual or job template based launch
- Interactive and non-interactive job support
- Mount external volumes on a per image basis

**Benefits:**
- **Heterogeneous Per Job/App Environment:** Apply the specific OSes and libraries that achieve highest performance and comply with environment certification requirements on a per job/application basis
- **Checkpointing:** Enables basic checkpointing with container and file state being saved. Such jobs may be restarted on different nodes.
- **Mobility:** Containerized jobs are more easily migrated to other environments (grid, cloud, etc.)
- **Security:** Improves security and resource isolation
- **Easy to set up:** Add docker support to nodes and configure compute node to use a docker set up script
# Docker Job Lifecycle

<table>
<thead>
<tr>
<th>Queue</th>
<th>Job Container Setup</th>
<th>Run Job</th>
<th>Clean Up after Job Ends</th>
<th>Completion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job starts in Queued State</td>
<td>Launch Container (as Root) Add User ID</td>
<td>Run Job in Container</td>
<td>Stop Container</td>
<td>End</td>
</tr>
<tr>
<td>Restore from Checkpoint Container</td>
<td>Launch Copy of Container (as User)</td>
<td></td>
<td>Remove Container and Image</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Save Container to Registry</td>
<td></td>
</tr>
<tr>
<td>Checkpointed Job Run</td>
<td></td>
<td></td>
<td>Initiate checkpoint</td>
<td>Requeue</td>
</tr>
</tbody>
</table>

**Summary:**
Improve security, compliance w/ certified environments, OS flexibility, checkpointing & mobility w/ docker support.
Data Awareness / Scheduled Data Transfers

- **Data Workflows**
  - Improved Throughput/utilization
- **Data Caching (Burst Buffer / Data Warp)**
- **Data Proximity**
- **Topology Awareness**

**Summary:**
- **Tradition Scheduling:** Blocks node usage during data transfer
- **Data Workflows:** Optimizes start time and throughput

**The Way Others Do It**

<table>
<thead>
<tr>
<th>Step</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Data Transfer</td>
<td>Nodes Blocked</td>
</tr>
<tr>
<td>Processing Time</td>
<td>Nodes Blocked</td>
</tr>
<tr>
<td>Output Data Transfer</td>
<td>Nodes Blocked</td>
</tr>
</tbody>
</table>

**Our Way**

<table>
<thead>
<tr>
<th>Step</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Data Transfer</td>
<td>Nodes Not Blocked</td>
</tr>
<tr>
<td>Processing Time</td>
<td></td>
</tr>
<tr>
<td>Output Data Transfer</td>
<td>Nodes Not Blocked</td>
</tr>
</tbody>
</table>
Adaptive Computing Workflow Use Models

**Just Flow**
- Just run my Flow
- Example: file drag & drop triggers workflow execution
- MCM GUI & CLI

**Advanced Workflow**
- Template based workflow definition
- MCM GUI & CLI
- Flexibility in using prepared templates

**Workflow Unlimited**
- Full flexibility for power users
- Full scalability
- ASCII Form based workflow definition

**Workflow Portal**
- Viewpoint Web Portal
- Template based WF definition
- Zero client install
- ETA: 2017
Moab Usage Accounting

- Sets usage budget amounts for users, groups, accounts
- Resource allocations and scheduling checked against amounts
- Tracks and reports system usage by each user, group, account, etc. for showback or pay-for-use chargeback
- Allows for rich resource costing, accounting for Cloud
Nitro – High Throughput Scheduler

Extends Moab HPC Suite to High Throughput

- 10 – 100X faster job launch (Up to +200/second/core)
- Currently serial, single core workload only

**Summary:**
Seamlessly integrate high throughput into high capability systems
High Throughput Differences

- Ultra high-speed message queue
- Test results: 13500 tasks/s **executed** on 20 node AWS cluster. Submission rate 100’s of millions per second
- Combines many tasks into a Nitro session
- Schedules the session
  - Incur scheduling overhead only once
  - Not once per individual small task
- **Creates policies for the entire Nitro session**
- Limitations
  - Nitro sacrifices some granularity in management
    - i.e. individual tasks in a large session cannot be cancelled or pre-empted in isolation
    - The session is the unit of management and reporting
Moab Cloud Suite
HPC on Demand
Flexibility & Scale
Moab Cloud Suite (MCS)

Internal Systems/services

mcproxy server

mcproxy client

workload

moab

mcflex

mcdeploy

Cloud provider

accounting

© 2016 ADAPTIVE COMPUTING, INC.
Moab Cloud Services
Allocation, Deploy, Proxy

Allocate
- VM Creation
- Credential Management
- DNS Management

Deploy/configure
- Adaptive Services

Proxy
- SOA data management
- Session data management
- Licensing
Adaptive Cloud Scalability

- **Immensely Scalable**
  - Can connect to any number on private or public clouds
    - Per cloud resource pools can possess over 1M cores
  - Can manage billions of tasks
    - Distributed task scheduling
    - Architected to support unlimited backlog
    - > 1,000 tasks / core / instance
  - Can manage massive data
    - Pluggable data transport layer
  - Can be distributed and replicated
  - Can handle sub-millisecond latencies (< 0.2 ms)
User/Application Experience

- **Transparent**
  - No resource quantity specification
  - No on-premise vs off premise specification
  - No data transport specification
  - No workload management specification
  - Programmatic application and task management

- **SLA Centric**
  - User/app specifies application and SLA
  - System resources application so as to deliver SLA

- **Cost Centric**
  - System optimizes resource allocation so as to minimize costs
Cloud Services
Multiple Workload Types

**Moab supports preemption, sharing, load-balancing, SLA’s, and reservations, enabling management of multiple workload types**

- Most cloud allocations are static and dedicated
- Costing is fixed, but cloud server utilization is low
- Enabling support for multiple workload types with complimentary use cases can increase utilization and thus, net revenue
  - Type 1: Dedicated, persistent resources. No sharing allowed. Resource quantity is guaranteed
  - Type 2: Shared, persistent resources. ‘Transparent’ sharing is allowed but is sharing workload is preempted if/when priority applications can leverage resources
  - Type 3: SOA/Batch. Workload must accomplish computation within given timeframe, but can opportunistically grow/shrink as resources are available
  - Type 4: Opportunistic. Workload utilizes spot market pricing and spot market availability to utilize whatever resources are available.

- With multiple workload types, cloud providers can allocate and sell nearly 100% of resources to paying customers
Intelligent Workload Management

Executive Needs:
- Consolidate, Lower Costs
- Unify Accounting / Visibility
- Align Resource to Revenue Impact

Revenue Impact

Group A
- $$$
- SOA & HPC
- Service Guarantees
- Cloud Bursting

Group B
- $
- Big Data
- Low Costs
- Monthly Runs

Group C
- $
- HTC*
- Real-time
- Low Latency
- Elastic Computing

Group D
- $
- HPC
- Prioritization
- High MEM, GPUs

Use Case Needs

Executive Needs:
- Consolidate, Lower Costs
- Unify Accounting / Visibility
- Align Resource to Revenue Impact

Cloud Bursting
- Low Costs
- Monthly Runs

Executive Needs:
- Consolidate, Lower Costs
- Unify Accounting / Visibility
- Align Resource to Revenue Impact

Converged Computing:
- HPC / Big Data / High Throughput, SOA
- Integrated with Cloud

Summary
- Improve utilization, lower costs, gain SLA control, improve first to market agility.
Adaptive Computing: Moab Paves the Way for HPC & Beyond

- Moab provides HPC-
  - Scalability & Performance
  - Flexibility HPC ↔ Cloud
  - Beyond plain batch workload mgt: Services, SOA, HTC, ... Workflow
- Nitro enables HTC in HPC
- Moab Cloud Suite: workload convergence in flexible infrastructure

- Contact us, we are happy to discuss and find the best possible design for your HPC, HTC, or Cloud system!
  - bschott@adaptivecomputing.com
Questions?