Moab Accounting Manager

Administrator Guide 9.1.3

September 2018

Adaptive Computing Enterprises, Inc.
704 Goodlette Road North
Naples, FL 34102
+1 (239) 330-6083
www.adaptivecomputing.com

Scan to open online help

Contents

Welcome

Welcome to *Moab Accounting Manager 9.1.3 Administrator Guide*.

This guide is intended as a reference for system administrators.

Chapter 1: Overview

Moab Accounting Manager is an accounting management system that allows for usage tracking, charge accounting, and allocation enforcements for resource usage in technical computing environments. It acts somewhat like a bank in which credits are deposited into funds with constraints designating which entities may access the funds. As resources or services are utilized, funds are charged and usage recorded. It supports familiar operations such as deposits, withdrawals, transfers, and refunds. It provides balance and usage feedback to users, managers, and system administrators.

Since the accounting and billing models vary widely from organization to organization, Moab Accounting Manager has been designed to be extremely flexible, featuring customizable usage and fund configurations, and supporting a variety of tracking, charging and allocation models. Attention has been given to scalability, security, and fault tolerance.

1.1 Background

Moab Accounting Manager was originally developed as open source software called the Gold Allocation Manager at Pacific Northwest National Laboratory (PNNL) under the Department of Energy (DOE) Scalable Systems Software (SSS) SciDAC project. It has been extended and enhanced by Adaptive Computing Enterprises, Inc. (formerly Cluster Resources, Inc.) and is in production use at many commercial, government and educational sites.

1.2 Conceptual Overview

Moab Accounting Manager was designed to be used in technical computing environments for usage tracking, charge accounting and allocation enforcement. Usage tracking involves resource usage in customizable usage records. Charge accounting involves calculating and recording charges for usage for invoicing or cost tracking. Allocation enforcement involves establishing limits on the use of system resources by defining separate funds having limited debit or credit balances.

In this overview, we will assume that you want to track or charge for workload resource usage. The use of resources by a job or reservation may result in a usage record. The usage record tracks the resources that were used, whom they were used by, and (optionally) how much the usage cost.

With Moab Accounting Manager, it is possible to allocate resource credits to various parties. This is done by associating a cost for the usage by deciding on a currency unit (generically referred to as credits), whether based on a real currency such as dollars, or a reference currency such as billing units or processor seconds. Next you will define charge rates in this currency for the components of your usage (consumable resource costs, multipliers, fees, etc.). You may create pools of funds called allocations via deposits that can be debit- or credit-based, finite or infinite, and limited to a time frame in which they can be used. These allocations are deposited into logical containers called funds which have constraints that distinguish the conditions under which the funds can be used.

Moab Workload Manager interacts with Moab Accounting Manager to ensure sufficient funds and to track and charge for usage. A typical usage pattern might be as follows. When a job is submitted, a quote is obtained to see how much it will cost and to verify that you have sufficient funds. When it is time for the job to start, a lien (or hold) is placed against your funds for the amount of the requested resources. When the job ends, the appropriate fund is debited and the lien is removed. A usage record is updated with the charge amount and job usage details. The actual composition of the interactions is very flexible and will be defined by the accounting mode and interaction methods.

1.3 Features

Feature	Description
Dynamic Charging	Rather than post-processing resource usage records on a periodic basis to rectify fund balances, charging can occur incrementally throughout usage or at usage completion.
Liens	A hold (called a lien) is placed against the funds for the estimated amount of credits before the usage begins, followed by appropriate charges during and/or at the end of the usage, thereby preventing accounts from using more resources or services than were allocated to them.
Customizable Usage Records	Usage record fields can be configured by the site to track custom usage properties.
Flexible Fund Allocation	A uniquely flexible design allows resource or service credits to be allocated to arbitrary entities and purposes.
Expiring Allocations	Credits may be restricted for use within a designated time period allowing sites to implement a use-it-or-lose-it policy to prevent year-end resource exhaustion and establishing an allocation cycle.
Flexible Charging	The billing system can track and charge for composite time-based or non-time-based resource or service usage, and apply flexible charge multipliers and fees.
Guaranteed Quotes	Users and resource brokers can determine ahead of time the cost of using resources or services.
Credit and Debit Allocations	Allocations feature an optional credit limit allowing support for both debit and credit models. This feature can also be used to enable overdraft protection for specific funds.
Infinite Allocations	Deposits can be made with infinite amounts or infinite credit limits when used with a supporting database.
Powerful Querying	A powerful querying and update mechanism (based on SQL queries) that facilitates flexible reporting and streamlines administrative tasks.
Nonintrusiveness	Object-level, attribute-level and correlated defaults may be established for arbitrary objects such as users, accounts and organizations. Additionally, these objects may be configured to be automatically created the first time they are seen by the resource management system. These features allow the accounting system to be used with less impact and involvement from users and administrators.

Feature	Description
Consistency	Moab Accounting Manager has been engineered for robustness, consistency and resiliency. Complex operations are atomic and are automatically rolled back on failure.
Security	Multiple security mechanisms for strong authentication and encryption.
Role-Based Authorization	Fine-grained (instance-level) Role Based Access Controls are provided for all operations which allows users to view and manipulate only those objects permitted to them.
Dynamic Customization	Sites can create or modify record types on the fly enabling them to meet their custom accounting needs. Dynamic object creation allows sites to customize the types of accounting data they collect without modifying the code. This capability turns this system into a generalized information service. This capability is extremely powerful and can be used to manage all varieties of custom configuration data, or to function as a persistence interface for other components.
Web Interface	A powerful dynamic web-based GUI is provided for easy remote access for users, managers and administrators which displays only the actions allowed by their role.
Journaling	A journaling mechanism preserves the indefinite historical state of all objects and records. This powerful mechanism allows historical bank statements to be generated, provides an undo/redo capability and allows commands to be run as if it were any arbitrary time in the past.
Event Scheduler	An event engine can be used to schedule arbitrary Moab Accounting Manager commands to run periodically or at a designated time in the future.

1.4 Interfaces

Moab Accounting Manager provides a variety of means of interaction, including command-line interfaces, graphical user interfaces, application programming interfaces, and communication protocols.

In this topic:

1.4.1 Command-Line Clients

The command-line clients provided feature rich argument sets and built-in documentation. These commands allow scripting and are the preferred way to interact with Moab Accounting Manager for basic usage and administration. Use the `--help` option for usage information or the `--man` option for a manual page on any command.

Example 1-1: Listing Users Using a Command-Line Client

```
mam-list-users
```

1.4.2 Interactive Control Program

The `mam-shell` command uses a control language to issue object-oriented requests to the server and display the results. The commands may be included directly as command-line arguments or read from stdin. Use the `ShowUsage:=True` option after a valid Object Action combination for usage information on the command.

Example 1-2: Listing Users Using the mam-shell Control Program

```
mam-shell User Query
```

> ⚠ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. Do not use this command unless you understand the syntax and the potential for unintended results.

1.4.3 Web-based Graphical User Interface

A powerful and easy-to-use web-based GUI permits browser access by users, managers and administrators according to their role definitions.

Example 1-3: Listing Users via the Web GUI

Click on `Manage Users` -> `List Users`

1.4.4 Perl API

You can access the full functionality via the Perl API. Use perldoc to obtain usage information for the Moab Accounting Manager Perl MAM modules.

Example 1-4: Listing Users Using the Perl API

```
use MAM;
my $request = new MAM::Request(object => "User", action => "Query");
my $response = $request->getResponse();
foreach my $datum ($response->getData())
{
 print $datum->toString(), "\n";
}
```

1.4.5 Java API (no longer supported)

> ℹ️ Beginning with Moab Accounting Manager 9.0, Java API is no longer supported, use Perl API instead.

1.4.6 SSSRMAP Wire Protocol

It is also possible to interact with Moab Accounting Manager by directly using the SSSRMAP Wire Protocol and Message Format over the network. Documentation for these protocols can be found at SSS Resource Management and Accounting Documentation.

Example 1-5: Listing Users via the SSSRMAP Wire Protocol

```
POST /SSSRMAP HTTP/1.1
Content-Type: text/xml; charset="utf-8"
Transfer-Encoding: chunked
190
<?xml version="1.0" encoding="UTF-8"?>
<Envelope>
 <Body actor="scottmo" chunking="True">
  <Request  action="Query" object="User"></Request>
 </Body>
 <Signature>
  <DigestValue>azu4obZswzBt89OgATukBeLyt6Y=</DigestValue>
  <SignatureValue>YXE/C08XX3RX4PMU1bWju+5/E5M=</SignatureValue>
```

```
  <SecurityToken type="Symmetric" name="scottmo"></SecurityToken>
 </Signature>
</Envelope>
0
```

1.5 Documentation

The documentation for Moab Accounting Manager includes this Administrator Guide, release notes, built-in man pages, module documentation and online documentation.

- *Moab Accounting Manager Administrator Guide* — The *Moab Accounting Manager Administrator Guide* is a comprehensive manual for users and administrators of Moab Accounting Manager and includes information about features, interfaces, installation, getting started, usage, configuration and customization. The Administrator Guide can be found under the `$PREFIX/doc` directory in `.pdf` and `.html` formats. These documents are also available online.

- *Release Notes* — The Release Notes describe the primary features and fixes included in the release, along with notes to aid in migration from previous versions and can be found under the doc directory in the distribution tarball.

- *Command Line built-in Man Pages and Usage Synopsis* — All command-line clients support a `--man` option that provides full documentation of the command options and a --help option that provides a brief usage synopsis.

- *Module Perl Pod Documentation* — Documentation for Moab Accounting Manager Perl modules can be viewed by changing directory to the `$PREFIX/lib` directory and running `perldoc <modulename>`, e.g. `perldoc MAM::Request`.

- *Online Documentation* — The *Moab Accounting Manager Administrator Guide* can be found online at http://www.adaptivecomputing.com/support/documentation-index/. The Gold project web page at http://www.adaptivecomputing.com/products/open-source/gold/ and includes the original Gold project documentation.

1.6 License

The Moab Accounting Manager software and associated documentation, data and information include parts which are copyrighted by Adaptive Computing Enterprises, Inc., and parts which are copyrighted by Battelle Memorial Institute. The terms and conditions for the use and redistribution of these parts are governed by the Moab Accounting Manager License and the BSD License respectively. Refer to the LICENSE file for details.

1.6.1 Moab Accounting Manager License

Copyright (C) 2006 - 2018 Adaptive Computing Enterprises, Inc. All rights reserved.

The Moab Accounting Manager License specifies the terms and conditions for use and redistribution.

The Moab Accounting Manager License applies to the Moab Accounting Manager software offered by Adaptive Computing Enterprises, Inc. By installing or using this software, Licensee accepts a non-exclusive license from Adaptive Computing Enterprises, Inc. and is bound to accept acknowledgement of and abide by the notices and conditions of the Moab Accounting Manager License.

1.6.2 BSD License

Copyright (C) 2003 - 2005 Pacific Northwest National Laboratory, Battelle Memorial Institute. All rights reserved.

The BSD license specifies the terms and conditions for use and redistribution.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Battelle nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Chapter 2: Initial Setup

After installation, you will need to perform certain steps to prepare Moab Accounting Manager to fulfill its desired role in your environment.

Moab Accounting Manager can be configured for a myriad of use cases. It can be used in different accounting modes such as for usage tracking, notional charging, or allocation enforcement. This chapter will walk you through the steps you need to take to integrate and initialize the accounting manager.

Related Topics

- Select an Appropriate Accounting Mode

- Integrate Moab Accounting Manager With Your Resource Manager

- Follow the Setup Guide for Your Selected Accounting Mode

2.1 Select an Appropriate Accounting Mode

Moab Accounting Manager can be configured to be used in a variety of different accounting modes. Some sites may wish to create and enforce resource usage limits through allocations. Others may want to impute a charge amount to their workload, but never deny workload based on availability of funds. Still others may not wish to calculate a charge at all, but simply record the usage details of the workload. Select the accounting mode from the following options that best matches your requirements.

- **strict-allocation** - Use this mode if you wish to strictly enforce allocation limits. Under this mode, you can prevent workload from running if the end-users do not have sufficient funds. Funds, allocations, quotes, liens, charge rates, and usage records support this mode. Before a job runs, MAM places a lien (or hold) against the user's funds to prevent overcommitment of their allocation. When a job completes, MAM removes the lien, debits the user's allocation, and records the workload usage details and charge in a usage record. This is the normal default.

- **fast-allocation** - Use this mode if you wish to debit allocations but need higher throughput by eliminating the lien and quote of strict-allocation mode. If you implement it properly through scripts, you can replace the lien and quote of strict-allocation mode with an asynchronous balance check, causing MAM to disable the accounts from further use after the first job that causes the fund to become negative. Funds, allocations, balance checks, charge rates and usage records support this mode.

- **notional-charging** - Use this mode if you wish to calculate and record charges for workload usage but not keep track of fund balances or allocation limits. Charge rates and usage records support this mode. The workload usage details and charge are recorded in a usage record.

- **usage-tracking** - Use this mode if you wish to simply record workload usage details but not to calculate a charge nor keep track of fund balances or allocation limits. Usage records support this mode.

Refer to the corresponding setup guide to prepare Moab Accounting Manager and Moab Workload Manager for your selected accounting mode. See Follow the Setup Guide for Your Selected Accounting Mode.

2.2 Integrate Moab Accounting Manager With Your Resource Manager

If you have not already done so, you will need to integrate with your resource management system. See Integration.

2.3 Follow the Setup Guide for Your Selected Accounting Mode

Refer to the appropriate setup guide to prepare Moab Accounting Manager and your resource manager for your selected accounting mode.

- strict-allocation - See Strict Allocation Setup Guide
- fast-allocation - See Fast Allocation Setup Guide
- notional-charging - See Notional Charging Setup Guide
- usage-tracking - See Usage Tracking Setup Guide

Chapter 3: Strict Allocation Setup Guide

This chapter will walk you through the typical steps needed to set up Moab Workload Manager and Moab Accounting Manager to use the strict allocation accounting mode.

With the strict allocation accounting mode, you can establish rigorous limits on the use of compute resources by your various parties. This is done by associating a cost for the usage by deciding on a currency unit, generically referred to as credits, whether based on a real currency such as dollars, or a reference currency such as billing units or processor-seconds, and then creating charge rates based on this currency. Funds are established to contain credit allocations attributed to specific accounts. Users are designated as members of the accounts. Deposits are made into funds associated with the accounts creating allocations. An allocation cycle may be established whereby allocations are considered for renewal on a regular periodic basis (such as yearly, quarterly or monthly).

Before a job is started, Moab Workload Manager will verify that the user has sufficient credits to run the job by attempting to place a hold against their funds (referred to as a lien). When a job completes, the user's funds will be debited via a charge, usage information will be recorded for the job and the lien will be removed. Users or managers can query the status of their allocations or details of their job charges and resource utilization.

> ⚠️ You will need to be a Moab Accounting Manager System Administrator to perform many of the tasks in this chapter. It is assumed that you have already installed Moab Workload Manager and installed, bootstrapped, and started Moab Accounting Manager before performing the steps discussed in this chapter.

> ℹ️ For testing or demo purposes, an initialization script is available that provides a similar affect to running the example commands in this chapter to minimally set up MAM for the strict-allocation accounting mode with a small amount of dummy sample data. It will not perform the Moab configuration steps described in this chapter. It can be cleaned up by running the hpc-cleanup.sh script.
>
> ```
> $ ./hpc-strict-allocation.sh
> ```

3.1 Set the Strict Allocation Accounting Mode

Set the AMCFG[mam] MODE parameter to strict-allocation in `moab.cfg` and set the accounting.mode parameter to strict-allocation in both the `mam-server.conf` and `mam-client.conf` files. Since strict allocation is the default accounting mode in both Moab Workload Manager and Moab Accounting Manager, it may not be necessary to do anything here unless you were previously using a different accounting mode.

Example 3-1: Setting the Accounting Mode to strict-allocation

AMCFG[] MODE parameter must be set in the Moab server configuration file (moab.cfg). After editing the moab.cfg file, you will need to restart moab.

```
# vi /opt/moab/etc/moab.cfg
AMCFG[mam] MODE=strict-allocation

# mschedctl -R
```

The accounting.mode parameter must be set in the server and client configuration files (mam-server.conf and mam-client.conf). After editing the mam-server.conf file, you will need to restart mam-server.

```
$ vi /opt/mam/etc/mam-server.conf
accounting.mode = strict-allocation

$ vi /opt/mam/etc/mam-client.conf
accounting.mode = strict-allocation

$ mam-server -r
```

3.2 Decide on a Currency and Set the Currency Precision

Since we will be calculating charges, we will need to decide which currency unit a MAM credit represents and sets the currency precision accordingly. For this example we will define a currency in which one credit represents the value of using one processor core for one hour. We will assume for simplicity that a processor-hour on one machine will have the same value as a processor-hour on another machine. Charge rates will be specified relative to this currency unit. Monetary transactions such as deposits and charges will be specified in terms of this currency. Since we want to be able to track and account for short jobs, we will specify a currency precision of two so that our currency credits will be represented as a floating point number with two decimal places. If instead we were to have chosen to use processor-seconds as the currency base, we would want to set the currency.precision value to zero since processor seconds can easily be represented as an integer with no decimal places. If we were to have chosen to use dollars as the currency base, we would have set the currency.precision value to two.

Example 3-2: Setting the Currency Precision to Two

The currency precision value must be set in the server and client configuration files (`mam-server.conf` and `mam-client.conf`). It must also be set in the GUI configuration file (`mam-gui.-conf`) if you will be using the web GUI. If you make changes in `mam-server.conf`, you must restart *mam-server*.

```
$ vi /opt/mam/etc/mam-server.conf

currency.precision = 2

$ vi /opt/mam/etc/mam-client.conf

currency.precision = 2

$ mam-server -r
```

3.3 Customize the Usage Record

The usage properties that you can track are limited by the properties sent by your resource manager to MAM. You can customize the usage record by adding additional properties for which you would like to track.

See 14.9 Customizing the Usage Record Object - page 151 for information on customizing the properties tracked in the usage record. If you are using Moab Workload Manager, see Accounting Properties Reported to Moab Accounting Manager in the *Moab Workload Manager Administrator Guide* for the list of usage record properties included with the accounting calls to MAM.

3.4 Define Charge Rates

Since we are charging, we must establish the charge rates for the usage. In our example, we will define a charge rate that charges 1 credit for each processor-hour utilized by the job. See Managing Charge Rates for more detailed information on setting up charge rates.

Example 3-3: Define a Charge Rate for Processors

```
$ mam-create-chargerate -n Processors -z 1/h -d "1 credit per processor-hour"
Successfully created 1 charge rate

$ mam-list-chargerates

Name        Value Amount Description
----------- ----- ------ -------------------------
Processors       1/h    1 credit per processor-hour
```

3.5 Define Accounts

Next we will define some accounts and assign users to the accounts. We will also associated each account with an organization so that usage reports can be generated for the organization level as well as the account and user level. We will create accounts for biology, chemistry, and film and assign them some users. The biology and chemistry account will be associated with the sciences organization while the film account will be associated with the arts organization. See Managing Accounts for more information on setting up accounts.

Example 3-4: Define the Biology, Chemistry, and Film Accounts

```
$ mam-create-account -a biology -o sciences -u amy,bob -d "Biology Department"
Successfully created 1 account

$ mam-create-account -a chemistry -o sciences -u amy,dave -d "Chemistry Department"
Successfully created 1 account

$ mam-create-account -a film -o arts -u bob,dave -d "Film Department"
Successfully created 1 account

$ mam-list-accounts

Name       Active Users       Organization Description
-------    ------ ---------   ------------ -----------------
biology    True   amy, bob    sciences     Biology Department
chemistry  True   amy, dave   sciences     Chemistry Department
film       True   bob, dave   arts         Film Department
```

3.6 Create Funds

The next task will be to create the funds which will hold the allocated credits. A fund is much like a numbered bank account, where credits can be deposited and are defined by constraints that distinguish who or what can use the contained credits and for what purposes. In this example, we will create a fund for each of the three accounts. The reason that funds are defined separately from accounts is that it is possible to create multiple funds for the same account. For example, you might have a fund that can be used for the chemistry account only when running the red cluster, and another fund that is used for the chemistry account when using a certain quality of service. See Chapter 10: Managing Funds - page 95 for more detailed information on setting up funds.

In this example, we will assume that we want to establish a periodic allocation cycle with predesignated allocation amounts being deposited on a quarterly schedule. In order to facilitate this, we will associate a default deposit amount with the science funds. For the biology fund, we will configure it to make a resetting deposit for 5000 credits for each period. The chemistry fund is going to be disabled at the end of the allocation period. The film account will remain unaffected by allocation renewals. See Chapter 11: Managing Allocations - page 115 for more information on periodic allocations.

Example 3-5: Create a Fund for Each of the Three Accounts

```
$ mam-create-fund -a biology -n "biology" --default-deposit 5000
Successfully created 1 fund with id 1 and 1 constraint

$ mam-create-fund -a chemistry -n "chemistry" --default-deposit -1
Successfully created 1 fund with id 2 and 1 constraint

$ mam-create-fund -a film -n "film"
Successfully created 1 fund with id 3 and 1 constraint

$ mam-list-funds

ID  Name      Constraints       Allocated Balance DefaultDeposit Description
--  --------- ----------------- --------- ------- -------------- -----------
1   biology   Account=biology        0.00    0.00        5000.00
2   chemistry Account=chemistry      0.00    0.00          -1.00
3   film      Account=film           0.00    0.00
```

3.7 Make Deposits

Now we need to allocate credits to these funds by making deposits to them. An allocation has a start and end time associated with it declaring the time frame in which it can be used (defaulting to a start time of the present and an end time of infinity). It can also have a credit limit which defines the extent to which the allocation is allowed to go negative. Allocations can be reset on a periodic basis or future allocations with different time frames can be pre-created within a fund to establish an allocation cycle and set expectations for credit expenditure. See Managing Allocations and Making Deposits for additional information.

In this example, we will allocate 5000 and 3000 credits to the biology and chemistry accounts respectively. The film account will be given a credit limit of 2000 credits which allows them to charge up to 2000 credits before rectifying their fund. When making a deposit we must specify the fund we are depositing into unless the fund can be unambiguously determined by its constraint references (i.e. there is only a single fund associated with the account biology). In the next example, we will utilize the fund's deposit amount in the first deposit, specify the amount explicitly in the second deposit, and establish a credit allocation in the third deposit.

Example 3-6: Making Deposits

```
$ mam-deposit -a biology

Successfully deposited 5000.00 credits into fund 1
Successfully created 1 allocation

$ mam-deposit -z 3000 -a chemistry

Successfully deposited 3000.00 credits into fund 2
Successfully created 1 allocation

$ mam-deposit -L 2000 -a film

No credits were deposited into fund 3
Successfully created 1 allocation
```

Let's examine the allocations we just created and its effect on the funds.

```
$ mam-list-allocations

Id Fund StartTime           EndTime   InitialDeposit Allocated CreditLimit Remaining PercentUsed
-- ---- ------------------- --------  -------------- --------- ----------- --------- ---------
1  1    2018-08-09 18:18:56 Infinity        5000.00   5000.00        0.00   5000.00      0.00
2  2    2018-08-09 18:18:56 Infinity        3000.00   3000.00        0.00   3000.00      0.00
3  3    2018-08-09 18:18:57 Infinity           0.00      0.00     2000.00      0.00      0.00

$ mam-list-funds

Id Name      Constraints      Allocated Balance DefaultDeposit Description
-- --------- ---------------- --------- ------- -------------- -----------
1  biology   Account=biology   5000.00 5000.00        5000.00
```

```
 2   chemistry  Account=chemistry   3000.00 3000.00              0.00
 3   film       Account=film           0.00    0.00
```

3.8 Check the Balance

We can verify the resulting balance (see Querying the Balance).

Example 3-7: Let's Look at Amy's Balance

```
$ mam-balance -u amy

Id Name      Balance CreditLimit Available
-- --------- ------- ----------- ---------
1  biology   5000.00        0.00   5000.00
1  chemistry 3000.00        0.00   3000.00
```

3.9 Automate Allocation Renewal

To facilitate the automatic renewal of our allocations, we will create a repeating event that resets all funds (see Creating Events) at the beginning of each new quarter.

Example 3-8: Create an Automatic Allocation Renewal Event

```
# vi /opt/mam/etc/mam-server.conf

event.scheduler = true

$ mam-server -r

$ mam-create-event --fire-command "Fund Reset" -s "2019-01-01" --rearm-period "3
months^"
Successfully created 1 event

$ mam-list-events

Id FireCommand FireTime   ArmTime             RearmPeriod EndTime Notify
RearmOnFailure FailureCommand CatchUp CreationTime       Description
-- ----------- ---------- ------------------- ----------- ------- ------ -------------
-- -------------- ------- ------------------- -----------
1  Fund Reset  2019-01-01 2018-08-09 18:21:28 3 months^                  False
               True    2018-08-09 18:21:28
```

3.10 Run a Job

Let's submit a job and examine the effects on the accounting system.

Example 3-9: Submit a Job

```
$ echo sleep 300 | msub -A chemistry -l procs=12,walltime=600
```

3.11 The Usage Lien

When a job starts, Moab Workload Manager typically creates a lien (or hold) against the appropriate allocations based on the estimated duration of the job. We will examine the effect of a running job on the accounting system (see Managing Liens).

Example 3-10: Examine the Effect of a Running Job on the Accounting System

```
$ mam-list-liens

Id Instance Amount StartTime           EndTime             Duration UsageRecord Funds
Description
-- -------- ------ ------------------- ------------------- -------- ----------- -----
-----------
1  74         2.00 2018-08-09 18:22:42 2018-08-09 18:22:42 600      1           2
```

This lien will decrease our available balance by the amount reserved.

```
$ mam-balance -u amy -a chemistry

Id Name      Balance Reserved Effective CreditLimit Available
-- --------- ------- -------- --------- ----------- ---------
2  chemistry 3000.00    2.00  2998.00        0.00   2998.00
```

The actual allocation has not changed.

```
$ mam-list-allocations -a chemistry

Id  Fund    Active StartTime           EndTime  Amount  CreditLimit Deposited
Description
--- ------- ------ ------------------- -------- ------- ----------- --------- -------
----
2   2       True   2018-08-09 18:18:56 Infinity 3000.00        0.00   3000.00
```

Note that the lien resulted in the initial creation of a usage record for the job with Stage Start.

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime EndTime
Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- --------- ------- -----------
1  Job  74         0.00 Start amy  faculty chemistry sciences     batch normal
   colony  1     12                              0
```

3.12 The Usage Charge

After a job completes, any associated liens are removed and a charge is issued against the appropriate allocations based on the resources and actual wallclock time used by the job. An allocation is debited and the usage record is modified with the charge and usage information.

Example 3-11: Examine the Effect of a Completed Job on the Accounting System

Your allocation will now have gone down by the amount of the charge.

```
$ mam-list-allocations -u amy -a chemistry

Id Fund StartTime          EndTime   InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------------------- -------- -------------- --------- ----------- --------- --
---------
2  2    2018-08-09 18:18:56 Infinity       3000.00   3000.00        0.00   2999.00
     0.03
```

However, your balance actually goes up (because the lien that was removed was larger than the actual charge).

```
$ mam-balance -u amy -a chemistry

Id Name      Balance Reserved Effective CreditLimit Available
-- --------- ------- -------- --------- ----------- ---------
2  chemistry 2999.00     0.00   2999.00        0.00   2999.00
```

A usage record for the job was updated as a side-effect of the charge. See 14.2 Querying Usage Records - page 141.

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime
EndTime          Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- ------------------ --------------
----- -----------
1  Job  74         1.00 End   amy  faculty chemistry sciences     batch normal
   colony  1     12                          300     2018-08-09 18:22:42 2018-08-09
18:27:42
```

3.13 Usage Refund

Now, we will illustrate the effect of issuing a refund for the user's job. For more information, see Issuing Usage Refunds.

Example 3-12: Refund the Job

```
$ mam-refund -J 74

Successfully refunded 1.00 credits to usage record 1 for instance 74
```

Our balance is back as it was before the job ran.

```
$ mam-balance -u amy -a chemistry

Id Name      Balance Reserved Effective CreditLimit Available
-- --------- ------- -------- --------- ----------- ---------
2  chemistry 3000.00    0.00   3000.00        0.00   3000.00
```

The allocation, of course, is likewise restored.

```
$ mam-list-allocations -u amy -a chemistry

Id Fund StartTime           EndTime   InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------------------- -------- -------------- --------- ----------- --------- --
---------
2  2    2018-08-09 18:18:56 Infinity       3000.00   3000.00        0.00   3000.00
     0.00
```

Notice that the usage charge is now zero because the job has been fully refunded.

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User
Group   Account   Organization Class  QualityOfService Machine Nodes Processors CPUTim
e Memory Duration StartTime           EndTime             Description
-- ---- -------- ------ ----- ----- ------- --------- ------------ ------ ------------
---  ------- ----- ---------- ------- ------ -------- ------------------- ------------
------- -----------
1  Job  74         0.00 End   amy   faculty chemistry sciences     batch  normal
     colony  1     12                       300      2018-08-09 18:22:42 2018-08-09
18:27:42
```

3.14 List Transactions

Let's list the transactions relating to this job (see Querying Transactions).

Example 3-13: Listing Transaction Details for This Job

```
$ mam-list-transactions -J 74 --full

Id   Object      Action  Actor Key Child Instance Count Amount Delta Balance Remaining
User Account    Machine Fund Allocation UsageRecord Duration Description Details


                                                      CreationTime
ModificationTime    Deleted RequestId TransactionId
---- ----------- ------- ----- --- ----- -------- ----- ------ ----- ------- ---------
---- --------- ------- ---- --------- ----------- -------- ----------- --------------
-----------------------------------------------------------------------------------
-----------------------------------------------------------------------------------
----------------------------------------------------- ------------------
-------------------- ------- --------- -------------
6481 UsageRecord Create  root  1         74       1
                                         1          0
Charge=0,Deleted=False,Stage=Start,Type=Job

                                                          2018-08-
09 18:22:42 2018-08-09 18:22:42 False   8615      6481
6484 UsageRecord Reserve root  1         74       1      2.00
amy   chemistry colony             1          600
Class=batch,Group=research,ItemizedCharges:=12 [Processors] * 0.000277777777777778
[ChargeRate{Processors}] * 600 [Duration] =
2,Nodes=1,Organization=sciences,Processors=12,QualityOfService=normal
                                     2018-08-09 18:22:42 2018-08-09 18:22:42 False
8615      6484
6489 UsageRecord Charge  root  1  74    74        1      1.00 -1.00 2999.00   2999.00
amy   chemistry colony  2   2          1         300
CPUTime=1800,Class=batch,EndTime:=1413997758,Group=research,ItemizedCharges:=12
[Processors] * 0.000277777777777778 [ChargeRate{Processors}] * 300 [Duration] =
1,Nodes=1,Organization=sciences,Processors=12,QualityOfService=normal,StartTime:=14139
97458 2018-08-09 18:27:42 2018-08-09 18:27:42 False   8627      6489
6495 UsageRecord Refund  root  1         74       1      1.00  1.00 3000.00   3000.00
                      2   2          1
                                                          2018-08-09 18:28:58
2018-08-09 18:28:58 False   8636      6495
```

3.15 Examine Fund Statement

Finally, you can examine the fund statement for the activities (see Obtaining a Fund Statement).

Example 3-14: You Can Request an Itemized Fund Statement Over All Time for Use `amy` *and the* `chemistry` *Account (fund 2)*

```
$ mam-statement -u amy -a chemistry

###########################################################################
#
# Includes fund 2 (chemistry)
# Generated on Tue Aug 9 18:29:53 2018.
#
# Reporting fund activity from -infinity to now.
#
###########################################################################
Beginning Balance:             0.00
-------------------------- -----------------------------
Total Credits:              3001.00
Total Debits:                 -1.00
-------------------------- -----------------------------
Ending Balance:            30000.00

############################## Credit Detail ##############################
Object      Action  Instance Amount  Balance Time
----------- ------- -------- ------- ------- -------------------
Fund        Deposit          3000.00 3000.00 2018-08-09 18:18:56
UsageRecord Refund  74          1.00 3000.00 2018-08-09 18:28:58

############################## Debit Detail ###############################
Object      Action Instance Account   User Machine Amount Balance Time
----------- ------ -------- --------- ---- ------- ------ ------- -------------------
UsageRecord Charge 74       chemistry amy  colony   -1.00 2999.00 2018-08-09 18:27:42
############################## End of Report ##############################
```

Chapter 4: Fast Allocation Setup Guide

This chapter will walk you through the typical steps needed to set up Moab Workload Manager and Moab Accounting Manager to use the fast allocation accounting mode.

If you want to enforce allocations by debiting funds in Moab Accounting Manager, but need higher throughput by eliminating the lien and quote operations of the strict allocation accounting mode, you can use the fast allocation accounting mode. With the fast allocation accounting mode, Moab Workload Manager checks a cached account balance, and jobs or reservations may be prevented from starting or continuing after the balance has become zero or negative. As with the strict allocation accounting mode, you establish limits on the use of compute resources by your various parties. This is done by associating a cost for the usage by deciding on a currency unit, generically referred to as credits, whether based on a real currency such as dollars, or a reference currency such as billing units or processor-seconds, and then creating charge rates based on this currency. Funds are established to contain credit allocations attributed to specific accounts. Users are designated as members of the accounts. Deposits are made into funds associated with the accounts creating allocations. An allocation cycle may be established whereby allocations are considered for renewal on a regular periodic basis (such as yearly, quarterly or monthly).

Before a job is started, Moab Workload Manager will check its internal cache to verify that the user has a positive account balance. When a job completes, the user's funds will be debited via a charge, usage information will be recorded for the job and Moab's account balance cache is updated. Since Moab Accounting Manager is not contacted at job submission or start time in order to verify account membership, additional configuration is needed in Moab to synchronize account information with Moab Accounting Manager. Additionally, since the cache in Moab Workload Manager supports only account based funds, when using the fast allocation accounting mode, funds having no constraints or having non-account constraints should not be used.

> ⚠ You will need to be a Moab Accounting Manager System Administrator to perform many of the tasks in this chapter. It is assumed that you have already installed Moab Workload Manager and installed, bootstrapped, and started Moab Accounting Manager before performing the steps discussed in this chapter.

> ℹ For testing or demo purposes, an initialization script is available that provides a similar affect to running the example commands in this chapter to minimally set up MAM for the fast-allocation accounting mode with a small amount of dummy sample data. It will not perform the Moab configuration steps described in this chapter. It can be cleaned up by running the hpc-cleanup.sh script.
>
> ```
> $ ./hpc-fast-allocation.sh
> ```

In this chapter:

4.1 Set the Fast Allocation Accounting Mode

Set the AMCFG[mam] MODE parameter to fast-allocation in `moab.cfg` and set the accounting.mode parameter to fast-allocation in both the `mam-server.conf` and `mam-client.conf` files.

Example 4-1: Setting the Accounting Mode to fast-allocation

AMCFG[] MODE parameter must be set in the Moab server configuration file (moab.cfg). After editing the moab.cfg file, you will need to restart moab.

```
# vi /opt/moab/etc/moab.cfg
AMCFG[mam] MODE=fast-allocation

# mschedctl -R
```

The accounting.mode parameter must be set in the server and client configuration files (mam-server.conf and mam-client.conf). After editing the mam-server.conf file, you will need to restart mam-server.

```
$ vi /opt/mam/etc/mam-server.conf
accounting.mode = fast-allocation

$ vi /opt/mam/etc/mam-client.conf
accounting.mode = fast-allocation

$ mam-server -r
```

4.2 Additional Moab Configuration

Since Moab will be checking an internal account balance cache when starting jobs and reservations instead of contacting Moab Accounting Manager, we need to periodically update Moab Workload Manager with account information from Moab Accounting Manager so that Moab can correctly apply default accounts and enforce account memberships. Additionally, it is beneficial to poll the account balances periodically so that external actions such as new deposits, transfers, etc. will be reflected properly in Moab's account balance cache.

Example 4-2: Configuring Moab to Synchronize Account Information

We will set AMCFG[] CREATECRED=TRUE in order to enable Moab to query accounts, users, user membership in accounts, and users' default accounts from Moab Accounting Manager and define them in Moab. We will set the AMCFG[] REFRESHPERIOD parameter to the interval that we want Moab to update these credential updates as well as its account balance cache. We will also set the ENFORCEACCOUNTACCESS parameter to TRUE in order to tell Moab to restrict users to use only accounts to which they belong.

```
# vi /opt/moab/etc/moab.cfg

AMCFG[mam]  CREATECRED=TRUE
AMCFG[mam]  REFRESHPERIOD=2:00:00
ENFORCEACCOUNTACCESS  TRUE

# mschedctl -R
```

4.3 Decide on a Currency and Set the Currency Precision

Since we will be calculating charges, we will need to decide what currency unit a MAM credit represents and set the currency precision accordingly. For this example we will define a currency in which one credit represents the value of using one processor core for one hour. We will assume for simplicity that a processor-hour on one machine will have the same value as a processor-hour on another machine. Charge rates will be specified relative to this currency unit. Monetary transactions such as deposits and charges will be specified in terms of this currency. Since we want to be able to track and account for short jobs, we will specify a currency precision of two so that our currency credits will be repesented as a floating point number with two decimal places. If instead we were to have chosen to use processor-seconds as the currency base, we would want to set the currency.precision value to zero since processor seconds can easily be represented as an integer with no decimal places. If we were to have chosen to use dollars as the currency base, we would have set the currency.precision value to two.

Example 4-3: Setting the Currency Precision to Two

The currency precision value must be set in the server and client configuration files (`mam-server.conf` and `mam-client.conf`). It must also be set in the GUI configuration file (`mam-gui.conf`) if you will be using the web GUI. If you make changes in `mam-server.conf`, you must restart *mam-server*.

```
$ vi /opt/mam/etc/mam-server.conf

currency.precision = 2

$ vi /opt/mam/etc/mam-client.conf

currency.precision = 2

$ mam-server -r
```

4.4 Customize the Usage Record

The usage properties that you can track are limited by the properties sent by your resource manager to MAM. You can customize the usage record by adding additional properties for which you would like to track.

See 14.9 Customizing the Usage Record Object - page 151 for information on customizing the properties tracked in the usage record. If you are using Moab Workload Manager, see Accounting Properties Reported to Moab Accounting Manager in the *Moab Workload Manager Administrator Guide* for the list of usage record properties included with the accounting calls to MAM.

4.5 Define Charge Rates

Since we are charging, we must establish the charge rates for the usage. In our example, we will define a charge rate that charges 1 credit for each processor-hour utilized by the job. See Managing Charge Rates for more detailed information on setting up charge rates.

Example 4-4: Define a Charge Rate for Processors

```
$ mam-create-chargerate -n Processors -z 1/h -d "1 credit per processor-hour"
Successfully created 1 charge rate

$ mam-list-chargerates

Name         Value Amount Description
----------- ----- ------ -------------------------
Processors        1/h    1 credit per processor-hour
```

4.6 Define Accounts

Next we will define some accounts and assign users to the accounts. We will also associate each account with an organization so that usage reports can be generated for the organization level as well as the account and user level. We will create accounts for biology, chemistry and film and assign them some users. The biology and chemistry account will be associated with the sciences organization while the film account will be associated with the arts organization. See Managing Accounts for more information on setting up accounts.

Example 4-5: Define the Biology, Chemistry, and Film Accounts

```
$ mam-create-account -a biology -o sciences -u amy,bob -d "Biology Department"
Successfully created 1 account

$ mam-create-account -a chemistry -o sciences -u amy,dave -d "Chemistry Department"
Successfully created 1 account

$ mam-create-account -a film -o arts -u bob,dave -d "Film Department"
Successfully created 1 account

$ mam-list-accounts

Name      Active Users     Organization Description
--------- ------ -------- ------------ --------------------
biology   True   amy,bob  sciences     Biology Department
chemistry True   amy,dave sciences     Chemistry Department
film      True   bob,dave arts         Film Department
```

4.7 Create Funds

The next task will be to create the funds which will hold the allocated credits. A fund is much like a numbered bank account, where credits can be deposited and are defined by constraints that distinguish who or what can use the contained credits and for what purposes. In this example, we will create a fund for each of the three accounts. The reason that funds are defined separately from accounts is that it is possible to create multiple funds for the same account. For example, you might have a fund that can be used for the chemistry account only when running the red cluster, and another fund that is used for the chemistry account when using a certain quality of service. See Managing Funds for more detailed information on setting up funds.

In this example, we will assume that we want to establish a periodic allocation cycle with pre-designated allocation amounts being deposited on a quarterly schedule. In order to facilitate this, we will associate a default deposit amount with the science funds. For the biology fund, we will configure it to make a resetting deposit of 5000 credits for each period. The chemistry fund is going to be disabled at the end of the allocation period. The film account will remain unaffected by allocation renewals. See Managing Allocations for more information on periodic allocations.

Example 4-6: Create a Fund for Each of the Three Accounts

```
$ mam-create-fund -a biology -n "biology" --default-deposit 5000
Successfully created 1 fund with id 1 and 1 constraint

$ mam-create-fund -a chemistry -n "chemistry" --default-deposit -1Successfully created
1 fund with id 2 and 1 constraint

$ mam-create-fund -a film -n "film
Successfully created 1 fund with id 3 and 1 constraint

$ mam-list-funds

Id Name      Constraints        Allocated Balance   DefaultDeposit Description
-- --------- ------------------ --------- --------- -------------- -----------
1  biology   Account=biology         0.00      0.00        5000.00
2  chemistry Account=chemistry       0.00      0.00          -1.00
3  film      Account=film            0.00      0.00
```

4.8 Make Deposits

Now we need to allocate credits to these funds by making deposits to them. An allocation has a start and end time associated with it declaring the time frame in which it can be used (defaulting to a start time of the present and an end time of infinity). It can also have a credit limit which defines the extent to which the allocation is allowed to go negative. Allocations can be reset on a periodic basis or future allocations with different time frames can be precreated within a fund to establish an allocation cycle and set expectations for credit expenditure. See Managing Allocations and Making Deposits for additional information.

In this example, we will allocate 5000 and 3000 credits to the biology and chemistry accounts respectively. The film account will be given a credit limit of 2000 credits which allows them to charge up to 2000 credits before settling their fund. When making a deposit we must specify the fund we are depositing into unless the fund can be unambiguously determined by its constraint references (i.e. there is only a single fund associated with the account biology). In the next example, we will utilize the fund's default deposit amount in the first deposit, specify the amount explicitly in the second deposit and establish a credit allocation in the third deposit.

Example 4-7: Making Deposits

```
$ mam-deposit -a biology

Successfully deposited 5000.00 credits into fund 1
Successfully created 1 allocation

$ mam-deposit -z 3000 -a chemistry

Successfully deposited 3000.00 credits into fund 2
Successfully created 1 allocation

$ mam-deposit -L 2000 -a film

No credits were deposited into fund 3
Successfully created 1 allocation
```

Let's examine the allocations we just created and its effect on the funds.

```
$ mam-list-allocations

Id Fund StartTime           EndTime   InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------------------- -------- -------------- --------- ----------- --------- --
---------
1  1    2018-08-09 18:18:56 Infinity        5000.00   5000.00        0.00   5000.00
    0.00
2  2    2018-08-09 18:18:56 Infinity        3000.00   3000.00        0.00   3000.00
    0.00
3  3    2018-08-09 18:18:57 Infinity           0.00      0.00     2000.00      0.00
    0.00

$ mam-list-funds

Id Name      Constraints       Allocated Balance DefaultDeposit Description
-- --------- ----------------- --------- ------- -------------- -----------
1  biology   Account=biology    5000.00 5000.00        5000.00
```

```
  2   chemistry  Account=chemistry    3000.00 3000.00              0.00
  3   film       Account=film            0.00    0.00
```

4.9 Check the Balance

We can verify the resulting balance (see Querying the Balance).

Example 4-8: Let's Look at Amy's Balance

```
$ mam-balance -u amy

Id Name      Balance CreditLimit Available
-- --------- ------- ----------- ---------
1  biology   5000.00        0.00   5000.00
1  chemistry 3000.00        0.00   3000.00
```

4.10 Automate Allocation Renewal

To facilitate the automatic renewal of our allocations, we will create a repeating event that resets all funds (see Creating Events) at the beginning of each new quarter.

Example 4-9: Create an Automatic Allocation Renewal Event

```
# vi /opt/mam/etc/mam-server.conf

event.scheduler = true

$ mam-server -r

$ mam-create-event --fire-command "Fund Reset" -s "2019-01-01" --rearm-period "3
months^"
Successfully created 1 event

$ mam-list-events

Id FireCommand FireTime    ArmTime             RearmPeriod EndTime Notify
RearmOnFailure FailureCommand CatchUp CreationTime       Description
-- ----------- ---------- ------------------- ----------- ------- ------ ------------
- -------------- ------- ------------------- -----------
1  Fund Reset  2019-01-01 2018-08-09 18:21:28 3 months^                  False
               True    2018-08-09 18:21:28
```

4.11 Run a Job

Now, let's submit a job and examine the effects on the accounting system.

Example 4-10: Submit a Job

```
$ echo sleep 300 | msub -A chemistry -l procs=12,walltime=600
```

4.12 The Usage Charge

After a job completes, a charge is issued against the appropriate allocations based on the resources and actual wallclock time used by the job. An allocation is debited and the usage record is modified with the charge and usage information.

Example 4-11: Examine the Effect of a Completed Job on the Accounting System

Your allocation and balance will have gone down by the amount of the charge.

```
$ mam-list-allocations -u amy -a chemistry

Id Fund StartTime           EndTime   InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------------------- --------- -------------- --------- ----------- --------- --
---------
2  2    2018-08-09 18:18:56 Infinity         3000.00   3000.00        0.00   2999.00
     0.03

$ mam-balance -u amy -a chemistry

Id Name      Balance CreditLimit Available
-- --------- ------- ----------- ---------
2  chemistry 2999.00        0.00   2999.00
```

The usage record for the job was updated as a side-effect of the charge (see Querying Usage Records).

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime
EndTime            Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- ------------------ --------------
----- -----------
1  Job  74         1.00 End   amy  faculty chemistry sciences     batch normal
   colony  1     12                        300      2018-08-09 18:22:42 2018-08-09
18:27:42
```

4.13 Usage Refund

Now, we will illustrate the effect of issuing a refund for the user's job (see Issuing Usage Refunds).

Example 4-12: Refund the Job

```
$ mam-refund -J 74

Successfully refunded 1.00 credits to usage record 1 for instance 74
```

Our balance is back as it was before the job ran.

```
$ mam-balance -u amy -a chemistry

Id Name      Balance CreditLimit Available
-- --------- ------- ----------- ---------
2  chemistry 3000.00        0.00   3000.00
```

The allocation, of course, is likewise restored.

```
$ mam-list-allocations -u amy -a chemistry

Id Fund StartTime          EndTime  InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------------------ -------- -------------- --------- ----------- --------- --
---------
2  2    2018-08-09 18:18:56 Infinity        3000.00   3000.00        0.00   3000.00
     0.00
```

Notice that the usage charge is now zero because the job has been fully refunded.

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime
EndTime          Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- ------------------- ---------------
----- -----------
1  Job  74         0.00 End   amy  faculty chemistry sciences     batch normal
   colony  1     12                        300      2018-08-09 18:22:42 2018-08-09
18:27:42
```

4.14 List Transactions

Let's list the transactions relating to this job (see Querying Transactions).

Example 4-13: Listing Transaction Details for This Job

```
$ mam-list-transactions -J 74 --full

Id   Object     Action  Actor   Key Child Instance Count Amount Delta Balance
Remaining User Account   Machine Fund Allocation UsageRecord Duration Description
Details

CreationTime       ModificationTime    Deleted RequestId TransactionId
---- ----------- ------- ------- --- ----- -------- ----- ------ ----- ------- -------
-- ---- --------- ------- ---- --------- ---------- -------- ---------- -----------
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
----------------------------------------------------------- ----------------
-- ------------------- ------- --------- -------------
6489 UsageRecord Charge  root    1   74    74       1     1.00 -1.00 2999.00
2999.00 amy   chemistry colony  2   2            1         300
CPUTime=1800,Class=batch,EndTime:=1413997758,Group=research,ItemizedCharges:=12
[Processors] * 0.000277777777777778 [ChargeRate{Processors}] * 300 [Duration] =
1,Nodes=1,Organization=sciences,Processors=12,QualityOfService=normal,StartTime:=14139
97458 2018-08-09 18:27:42 2018-08-09 18:27:42 False   8627     6489
6495 UsageRecord Refund  root    1         74       1     1.00  1.00 3000.00
3000.00                   2   2            1


                                                             2018-08-09
18:28:58 2018-08-09 18:28:58 False   8636     6495
```

4.15 Examine Fund Statement

Finally, you can examine the fund statement for our activities (see Obtaining a Fund Statement).

Example 4-14: We Can Request an Itemized Fund Statement Over All Time for Use `amy` *and the* `chemistry` *Account (fund 2)*

```
$ mam-statement -u amy -a chemistry

###############################################################################
#
# Includes fund 2 (chemistry)
# Generated on Tue Aug 9 18:29:53 2018.
# Reporting fund activity from -Infinity to Now.
#
###############################################################################

Beginning Balance:                  0.00
------------------ -------------------
Total Credits:                   3001.00
Total Debits:                      -1.00
------------------ -------------------
Ending Balance:                  3000.00

############################## Credit Detail ##############################

Object       Action   Instance Amount  Balance Time
----------- ------- -------- ------- ------- -------------------
Fund         Deposit           3000.00 3000.00 2018-08-09 18:18:56

UsageRecord Refund  74         1.00 3000.00 2018-08-09 18:28:58

############################## Debit Detail ##############################

Object       Action Instance Account   User Machine Amount Balance Time
----------- ------ -------- --------- ---- ------- ------ ------- -------------------
UsageRecord Charge 74       chemistry amy  colony   -1.00 2999.00 2018-08-09 18:27:42

############################## End of Report ##############################
```

Chapter 5: Notional Charging Setup Guide

This chapter will walk you through the typical steps needed to set up Moab Workload Manager and Moab Accounting Manager to use the notional charging accounting mode.

Some sites may want to use Moab Accounting Manager to calculate and record charges, but not to impose allocation limits or prevent any workload from running. With notional charging, charge rates will be used to calculate a cost for using resources, but there is no need to make deposits, debit funds or keep track of allocation limits. Although it would be possible to set up accounts and assign users to specific accounts, this chapter will assume that account membership is not going to be enforced. If you would prefer to enforce account membership, you can continue to use the notional charging accounting setup as described in this chapter, but you will need to additionally define accounts and account memberships as well as configure Moab to synchronize account information from Moab Accounting Manager as described in Chapter 4: Fast Allocation Setup Guide - page 37. Liens, balance queries and quotes are not needed. Our main task is to define charge rates.

At the end of a job, Moab Workload Manager will send usage information to the accounting manager. Moab Accounting Manager will calculate a charge and store this with the job usage record.

> ⚠️ You will need to be a Moab Accounting Manager System Administrator to perform many of the tasks in this chapter. It is assumed that you have already installed Moab Workload Manager and installed, bootstrapped, and started Moab Accounting Manager before performing the steps discussed in this chapter.

> ℹ️ For testing or demo purposes, an initialization script is available that provides a similar affect to running the example commands in this chapter to minimally set up MAM for the notional-charging accounting mode with a small amount of dummy sample data. It will not perform the Moab configuration steps described in this chapter. It can be cleaned up by running the hpc-cleanup.sh script.
>
> ```
> $ ./hpc-notional-charging.sh
> ```

In this chapter:

5.1 Set the Notional Charging Accounting Mode

Set the AMCFG[mam] MODE parameter to notional-charging in `moab.cfg` and set the accounting.mode parameter to notional-charging in both the `mam-server.conf` and `mam-client.conf` files.

Example 5-1: Setting the Accounting Mode to notional-charging

AMCFG[] MODE parameter must be set in the Moab server configuration file (moab.cfg). After editing the moab.cfg file, you will need to restart moab.

```
# vi /opt/moab/etc/moab.cfg
AMCFG[mam] MODE=notional-charging

# mschedctl -R
```

The accounting.mode parameter must be set in the server and client configuration files (mam-server.conf and mam-client.conf). After editing the mam-server.conf file, you will need to restart mam-server.

```
$ vi /opt/mam/etc/mam-server.conf
accounting.mode = notional-charging

$ vi /opt/mam/etc/mam-client.conf
accounting.mode = notional-charging

$ mam-server -r
```

5.2 Decide on a Currency and Set the Currency Precision

Since we will be calculating charges, we will need to decide what currency unit a MAM credit represents and set the currency precision accordingly. For this example we will define a currency in which one credit represents the value of using one processor core for one hour. We will assume for simplicity that a processor-hour on one machine will have the same value as a processor-hour on another machine. Charge rates will be specified relative to this currency unit. Monetary transactions such as deposits and charges will be specified in terms of this currency. Since we want to be able to track and account for short jobs, we will specify a currency precision of two so that our currency credits will be represented as a floating point number with two decimal places. If instead we were to have chosen to use processor-seconds as the currency base, we would want to set the currency.precision value to zero since processor seconds can easily be represented as an integer with no decimal places. If we were to have chosen to use dollars as the currency base, we would have set the currency.precision value to two.

Example 5-2: Setting the Currency Precision to Two

The currency precision value must be set in the server and client configuration files (`mam-server.conf` and `mam-client.conf`). It must also be set in the GUI configuration file (`mam-gui.conf`) if you will be using the web GUI. If you make changes in `mam-server.conf`, you must restart *mam-server*.

```
$ vi /opt/mam/etc/mam-server.conf

currency.precision = 2

$ vi /opt/mam/etc/mam-client.conf

currency.precision = 2

$ mam-server -r
```

5.3 Customize the Usage Record

The usage properties that you can track are limited by the properties sent by your resource manager to MAM. You can customize the usage record by adding additional properties for which you would like to track.

See 14.9 Customizing the Usage Record Object - page 151 for information on customizing the properties tracked in the usage record. If you are using Moab Workload Manager, see Accounting Properties Reported to Moab Accounting Manager in the *Moab Workload Manager Administrator Guide* for the list of usage record properties included with the accounting calls to MAM.

5.4 Define Charge Rates

Since we are charging, we must establish the charge rates for the usage. In our example, we will define a charge rate that charges 1 credit for each processor-hour utilized by the job. See Managing Charge Rates for more detailed information on setting up charge rates.

Example 5-3: Define a Charge Rate for Processors

```
$ mam-create-chargerate -n Processors -z 1/h -d "1 credit per processor-hour"
Successfully created 1 charge rate

$ mam-list-chargerates

Name         Value Amount Description
-----------  ----- ------ --------------------------
Processors         1/h    1 credit per processor-hour
```

5.5 Run a Job

Now, let's submit a job and examine the effects on the accounting system.

Example 5-4: Submit a Job

```
$ echo sleep 300 | msub -A chemistry -l procs=12,walltime=600
```

5.6 The Usage Charge

After a job completes, a usage record is generated with the charge and resource usage information.

Example 5-5: List the Usage and Charge for Our Job

```
$ mam-list-usagerecord

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime
EndTime          Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- ------------------ --------------
----- -----------
1  Job  74        1.00 End   amy   faculty chemistry sciences     batch normal
  colony  1    12                       300       2018-08-09 18:22:42 2018-08-09
18:27:42
```

5.7 Usage Refund

Now, we will illustrate the effect of issuing a refund for the user's job (see Issuing Usage Refunds).

Example 5-6: Refund the Job

```
$ mam-refund -J 74

Successfully refunded 1.00 credits to usage record 1 for instance 74
```

Notice that the usage charge is now zero because the job has been fully refunded.

```
$ mam-list-usagerecords

Id Type Instance Charge Stage User Group   Account   Organization Class
QualityOfService Machine Nodes Processors CPUTime Memory Duration StartTime
EndTime           Description
-- ---- -------- ------ ----- ---- ------- --------- ------------ ----- --------------
-- ------- ----- ---------- ------- ------ -------- ------------------ --------------
----- -----------
1  Job  74         0.00 End   amy  faculty chemistry sciences    batch normal
   colony  1    12                             300 2018-08-09 18:22:42 2018-08-09
18:27:42
```

5.8 List Transactions

Let's list the transactions relating to this job. See 17.1 .

Example 5-7: Listing Transaction Details for This Job

```
$ mam-list-transactions -J 74 --full

Id   Object      Action  Actor   Key Child Instance Count Amount User Account
Machine Fund Allocation UsageRecord Duration Description Details


                                         CreationTime       ModificationTime
 Deleted RequestId TransactionId
---- ----------- ------- ------- --- ----- -------- ----- ------ ---- --------- ------
- ---- ---------- ----------- -------- ---------- ------------------------------------
--------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------
----------------------------------------------- ------------------ ------------------ -
------- --------- -------------
6489 UsageRecord Charge  root    1   74    74       1     1.00 amy  chemistry colony
                1           300
CPUTime=1800,Class=batch,EndTime:=1413997758,Group=research,ItemizedCharges:=12
[Processors] * 0.000277777777777778 [ChargeRate{Processors}] * 300 [Duration] =
1,Nodes=1,Organization=sciences,Processors=12,QualityOfService=normal,StartTime:=14139
97458 2018-08-09 18:27:42 2018-08-09 18:27:42 False  8627     6489
6495 UsageRecord Refund  root    1         74       1     1.00
                1


                                         2018-08-09 18:28:58 2018-08-09 18:28:58 False
  8636     6495
```

Chapter 6: Usage Tracking Setup Guide

This chapter will walk you through the typical steps needed to set up Moab Workload Manager and Moab Accounting Manager to use the usage tracking accounting mode.

When used solely for usage tracking, Moab Accounting Manager logs resource usage in usage records. This usage can be queried to report what resources were used when and by whom. In this case, there is no need for charge rates, funds, allocations, liens or quotes. There is no need to define account membership.

At the end of a job, Moab Workload Manager will send usage information to the accounting manager. Moab Accounting Manager will store this information in a job usage record.

⚠️ You will need to be a Moab Accounting Manager System Administrator to perform many of the tasks in this chapter. It is assumed that you have already installed Moab Workload Manager and installed, bootstrapped, and started Moab Accounting Manager before performing the steps discussed in this chapter.

ℹ️ For testing or demo purposes, an initialization script is available that provides a similar affect to running the example commands in this chapter to minimally set up MAM for the usage-tracking accounting mode with a small amount of dummy sample data. It will not perform the Moab configuration steps described in this chapter. It can be cleaned up by running the hpc-cleanup.sh script.

```
$ ./hpc-usage-tracking.sh
```

In this chapter:

6.1 Set the Usage Tracking Accounting Mode

Set the `AMCFG[mam] MODE` parameter to `usage-tracking` in `moab.cfg` and set the `accounting.mode` parameter to `usage-tracking` in both the `mam-server.conf` and `mam-client.conf` files.

Example 6-1: Setting the Accounting Mode to usage-tracking.

`AMCFG[] MODE` parameter must be set in the Moab server configuration file (`moab.cfg`). After editing the `moab.cfg` file, you will need to restart Moab.

```
# vi /opt/moab/etc/moab.cfg
AMCFG[mam] MODE=usage-tracking

# mschedctl -R
```

The `accounting.mode` parameter must be set in the server configuration file (`mam-server.conf`). After editing the `mam-server.conf` file, you will need to restart `mam-server`.

```
$ vi /opt/mam/etc/mam-server.conf
accounting.mode = usage-tracking

$ mam-server -r
```

The `accounting.mode` parameter should also be set in the client configuration file (`mam-client.conf`).

```
$ vi /opt/mam/etc/mam-client.conf
accounting.mode = usage-tracking
```

6.2 Customize the Usage Record

The usage properties that you can track are limited by the properties sent by your resource manager to MAM. You can customize the usage record by adding additional properties for which you would like to track.

See 14.9 Customizing the Usage Record Object - page 151 for information on customizing the properties tracked in the usage record. If you are using Moab Workload Manager, see Accounting Properties Reported to Moab Accounting Manager in the *Moab Workload Manager Administrator Guide* for the list of usage record properties included with the accounting calls to MAM.

6.3 Run a Job

Now, let's submit a job and examine the effects on the accounting system.

Example 6-2: Submit a Job

```
$ echo sleep 300 | msub -A chemistry -l procs=12,walltime=600
```

6.4 Query Job Usage Information

After a job completes, usage information is recorded. Let's examine the usage record that was created (see Querying Usage Records).

Example 6-3: List Usage Records

```
$ mam-list-usagerecords

Id Type Instance Stage User Group   Account   Organization Class QualityOfService
Machine Nodes Processors CPUTime Memory Duration StartTime           EndTime
  Description
-- ---- -------- ----- ---- ------- --------- ------------ ----- ---------------- ----
--- ----- ---------- ------- ------ -------- ------------------ ------------------ -
---------
1  Job  74       End   amy  faculty chemistry sciences    batch normal
colony 1     12                      300     2018-09-17 15:42:43 2018-09-17
15:47:22
```

Chapter 7: Managing Users

A user is a person authorized to use a resource or service. Default user properties include the common name, phone number, email address, default account, and description for that person. A user can be created, queried, modified, and deleted. By default, a standard user may only query his or her own user record.

User queries allow the specification of filter options which narrow down the users that will be returned to those belonging to the specified account.

In this chapter:

7.1 Creating Users

To create a new user, use the command `mam-create-user`.

```
mam-create-user {[-u] <user_name>} [-A | -I] [-n <common_name>] [--
phone <phone_number>] [--email <email_address>] [-a <default_
account>] [-d <description>] [-X, --extension <property>=<value>]...
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose]
[--version] [--about]
```

Example 7-1: Creating a user

```
$ mam-create-user -n "Smith, Robert F." --email "bob@bank.com" --phone "(509) 555-
1234" bob

Successfully created 1 user
```

Related Topics

- mam-create-user

7.2 Querying Users

To display user information, use the command *mam-list-users*.

mam-list-users [[-u] <user_pattern>] [-A | -I] [-X, --extension
<property>=<value>]... [-a <account_name>] [--full] [—
show <attribute_name>,...] [--long] [--wide] [--format
csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--
quiet] [--version] [--about]

Example 7-2: Listing standard info about active users

```
$ mam-list-users -A


Name    Active    CommonName                PhoneNumber          EmailAddress
   DefaultAccount         Description
------ --------- ------------------------ -------------------- ----------------------
-- -------------------- ----------------
amy    True      Wilkes, Amy               (509) 555-8765       amy@bank.com
bob    True      Smith, Robert F.          (509) 555-1234       bob@bank.com
```

Example 7-3: Displaying bob's phone number

```
$ mam-list-users —show PhoneNumber bob —quiet

(509) 555-1234
```

Example 7-4: Listing amy's accounts

```
$ mam-list-users —show Accounts amy -l -q

-------------
chemistry
biology
```

Example 7-5: Listing all users belonging to the chemistry account

```
$ mam-list-users —show Name -a chemistry -q

-------------
amy
dave
```

Related Topics

- mam-list-users

7.3 Modifying Users

To modify a user, use the command `mam-modify-user`.

```
mam-modify-user {[-u] <user_name>} [-A | -I] [-n <common_name>] [--
phone <phone_number>] [--email <email_address>] [-a <default_
account>] [-d <description>] [-X, --extension <property>=<value>]...
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose]
[--version] [--about]
```

Example 7-6: Deactivating a user

```
$ mam-modify-user -I bob

Successfully modified 1 user
```

> ℹ In order for user validity enforcement to occur, the Values property for the UsageRecord User attribute must be set to '`@User`'.
>
> ```
> $ mam-shell Attribute Modify Object==UsageRecord Name==User Values=@User
> ```

Example 7-7: Setting a user's default account

```
$ mam-modify-user -a chemistry amy

Successfully modified 1 user
```

Example 7-8: Changing a user's email address

```
$ mam-modify-user --email "rsmith@cs.univ.edu" bob

Successfully modified 1 user
```

Related Topics

mam-modify-user

7.4 Deleting Users

To delete a user, use the command *mam-delete-user*.

mam-delete-user {[-u] <user_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 7-9: Deleting a user

```
$ mam-delete-user bob

Successfully deleted 1 user
```

Related Topics

- mam-delete-user

7.5 User Auto-Generation

If user auto-generation is enabled (this is the default), users are automatically created when first added as a member to an account or role. It is also possible to automatically create users when first encountered in a usage function (charge, reserve or quote). In order for user auto-generation to occur, the `AutoGen` property for the User object must be set to `True` (this is the default). Additionally, for user auto-generation to occur when a user is added as a member of another object (such as Account) via an association table (e.g. AccountUser), the `Values` property for the user attribute of the Association (e.g. Name) must be set to `@User`, indicating that that value should be constrained to be a valid instance of the User object. For user auto-generation to occur when initially encountered in a usage function, the `Values` property of the user attribute of the UsageRecord object must be similarly set to `@User`. The auto-creation of users can be completely disabled by setting the `AutoGen` property for the User object to `False`.

Example 7-10: Enable auto-generation of users when initially seen in a charge

```
$ mam-shell Attribute Modify Object==UsageRecord Name==User Values=@User

Successfully modified 1 attribute
```

Example 7-11: Disable all auto-generation of users

```
$ mam-shell Object Modify Name==User AutoGen=False

Successfully modified 1 object
```

See Object Auto-Generation for more information about the auto-generation of objects.

Related Topics

7.6 Default User

It is possible to set a global default user to which usage would be ascribed in quotes, liens, or charges where no user is specified. This can be accomplished by setting the `DefaultValue` property for the User object to the desired user.

It is also possible to set a custom user default or a specific object, which will result in usage being ascribed to the specified user when the object is attributed to the usage. This is done by creating a default usage override modifier. For example, to specify that `acmeuser` be the default user for usage associated with the acme organization, you might first create an attribute called `DefaultUser` for the `Organization` Object with the `Values` property of `@?=User`. Then you would populate the new `DefaultUser` property for the acme organization with the value of `acmeuser`. See Customizing Objects for more information on default and other usage override modifiers.

Example 7-12: Assign a global default user

```
$ mam-shell Object Modify Name==User DefaultValue=anonymous

Successfully modified 1 object
```

Chapter 8: Managing Accounts

An account represents a work entity requiring the use of resources or services for a common purpose. Users may be designated as members of an account and may be allowed to share its allocations. If the special 'ANY' user is added to an account, then any user may use funds allocated to the account. The user members may be designated as active or inactive, and as an administrator for the account. Default account properties include the description, the organization it is part of, and whether or not it is active. An account can be created, queried, modified and deleted. An account's user membership can also be adjusted. By default, a standard user may only query accounts they belong to.

Account queries allow the specification of filter options which narrow down the accounts that will be returned to those having the specified users in them.

8.1 Creating Accounts

To create a new account, use the command *mam-create-account*.

mam-create-account {[-a] <account_name>} [-A | -I] [-o <organization_name>] [-d <description>] [-X, --extension <property>=<value>]... [-u [^|!][+|-]<user_name>,...]... [--create-fund True|False] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

When defining users, the optional caret or exclamation symbol indicates whether the user should be created as an admin (^) or not (!) for the account. The optional plus or minus sign can precede each member to indicate whether the member should be created in the active (+) or inactive (−) state. By default, a user will be created in the active state but not an admin. Multiple users may be passed to the -u option in a comma-delimited list. Alternatively, multiple -u options may be specified.

> **ⓘ** If the Fund object's AutoGen property is set to true (see Fund Auto-generation), a fund will be automatically created for the account (unless overridden with the --createFund option). The auto-generated fund will be associated with the new account.

Example 8-1: Creating an account

```
$ mam-create-account -d "Chemistry Department" chemistry

Successfully created 1 account
```

Example 8-2: Creating an account that can be used by any user

```
$ mam-create-account -d "Common Account" -u ANY common

Successfully created 1 account.
```

Example 8-3: Creating an account and specifying user members at the same time

In this example, we make amy the account admin and associate the account with the sciences organization.

```
$ mam-create-account -d "Chemistry Department" -u ^amy,bob,dave chemistry -o sciences

Successfully created 1 account
```

Related Topics

- mam-create-account

8.2 Querying Accounts

To display account information, use the command *mam-list-accounts*.

mam-list-accounts [[-a] <account_pattern>] [-A | -I] [-o <organization_name>] [-X, --extension <property>=<value>]... [-u <user_name>] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 8-4: Listing all info about all accounts

```
$ mam-list-accounts
Name           Active     Users         Organization   Description
-------------- ---------- ------------- -------------- ------------------------
biology        True       amy,^bob      sciences       Biology Department
chemistry      True       ^amy,^dave    sciences       Chemistry Department
film           True       amy,^dave     arts           Film Department
```

Example 8-5: Displaying the name and user members of an account in long format

```
$ mam-list-accounts -—show Name,Users -long chemistry

Name           Users
-------------- -------
chemistry      ^amy
dave
```

Example 8-6: Listing all account names

```
$ mam-list-accounts --show Name --quiet

biology
chemistry
film
```

Example 8-7: Listing all accounts that have dave *as a member*

```
$ mam-list-accounts --show Name -u dave --quiet

chemistry
film
```

Related Topics

- mam-list-accounts

8.3 Modifying Accounts

To modify an account, use the command *mam-modify-account*.

```
mam-modify-account {[-a] <account_name>} [-A | -I] [-o <organization_
name>] [-d <description>] [-X, --extension <property>=<value>]... [--
add-user(s) [^|!][+|-]<user_name>,...]... [--del-user(s) <user_
name>,...] [--mod-user(s) [^|!][+|-]<user_name>,...]... [--debug] [--
site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version]
[--about]
```

User members may be added, removed or modified in an account. When adding user members to an account, the optional caret or exclamation symbol indicates whether the user should be created as an admin (^) or not (!) for the account. The optional plus or minus signs can precede each member to indicate whether the member should be created in the active (+) or inactive (−) state. When modifying user members of an account, the caret symbol or exclamation symbol indicates the user should be changed to become an admin (^) or not (!) for the account. The plus or minus signs indicate whether the user should be changed to become active (+) or inactive (−). If an active or admin modifier is not specified, that aspect of the user member will remain unchanged. If the user.firstaccountdefault server parameter is set to true, the first account that a user is added to will additionally become the default account for that user.

Example 8-8: Deactivating an account

```
$ mam-modify-account -I chemistry

Successfully modified 1 account
```

> ℹ In order for account validity enforcement to occur, the Values property for the UsageRecord Account attribute must be set to '@Account'.
>
> ```
> $ mam-shell Attribute Modify Object==UsageRecord Name==Account Values=@Account
> ```

Example 8-9: Adding users as members of an account

```
$ mam-modify-account --add-users jsmith,barney chemistry

Successfully added 2 users
```

Example 8-10: Deactivating a user in an account

```
$ mam-modify-account --mod-user -dave chemistry

Successfully modified 1 user
```

8.4 Deleting Accounts

To delete an account, use the command *mam-delete-account*.

mam-delete-account {[-a] <account_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 8-11: Deleting an account

```
$ mam-delete-account chemistry

Successfully deleted 1 account
```

Related Topics

- mam-delete-account

8.5 Account Auto-Generation

It is possible to have accounts be created automatically when first encountered in a usage function (charge, reserve or quote). It is also possible for accounts to be automatically created when initially added as a member of another object. In order for account auto-generation to occur, the `AutoGen` property for the Account object must be set to `True`. This is the default. For account auto-generation to occur when initially encountered in a usage function, the `Values` property of the account attribute of the UsageRecord object must be set to `@Account`. Additionally, for account auto-generation to occur when an account is added as a member of another object (such as the Organization object) via an association table (e.g. OrganizationAccount), the `Values` property for the account attribute of the Association (e.g. Name) must be set to `@Account`, indicating that that value should be constrained to be a valid instance of the Account object. The auto-creation of accounts can be completely disabled by setting the `AutoGen` property for the Account object to `False`.

Example 8-12: Enable auto-generation of accounts when initially seen in a charge

```
$ mam-shell Attribute Modify Object==UsageRecord Name==Account Values=@Account

Successfully modified 1 attribute
```

Example 8-13: Disable all auto-generation of accounts

```
$ mam-shell Object Modify Name==Account AutoGen=False

Successfully modified 1 object
```

See Object Auto-Generation for more information about the auto-generation of objects.

Related Topics

- 23.6 Object Auto-Generation - page 227

8.6 Default Account

It is possible to set a global default account to which usage would be ascribed in quotes, liens, or charges where no account is specified. This can be accomplished by setting the `DefaultValue` property for the `Account` object to the desired account name.

A per-user default account can be established by setting the `DefaultAccount` property for the user. If the `user.firstaccountdefault` server parameter is set to true (the default), the first account that a user is added to will automatically become the default account for that user. Otherwise, you can use the *mam-modify-user* command along with the `-a` option to set or change the default account for the user.

It is also possible to set a custom account default for a specific object, which will result in usage being ascribed to the specified account when the object is attributed to the usage. This is done by creating a default usage override modifier. For example, to specify a default account of testing for the beta organization, you might first create an attribute called `DefaultAccount` for the `Organization` Object with the `Values` property of `@?=Account`. Then you would populate the new `DefaultAccount` property for the beta organization with the value of testing. See Customizing Objects for more information on default and other usage override modifiers.

Example 8-14: Assign a global default account

```
$ mam-shell Object Modify Name==Account DefaultValue=common

Successfully modified 1 object
```

Related Topics

- Chapter 23: Customizing Objects - page 221

Chapter 9: Managing Organization

An organization is a virtual organization in which accounts are grouped. An account may only belong to a single organization while an organization may have multiple accounts. For example, an account may represent a project or cost-center while an organization may represent an institutional department or business division. The purpose of defining organizations is to support the ability to produce reporting for higher-order organizational entities beyond the individual account. Default organization properties include a name and a description. An organization can be created, queried, modified, and deleted.

9.1 Creating Organizations

To create a new organization, use the command *mam-create-organization*.

mam-create-organization {[-o] <organization_name>} [-d <description>]
[-X, --extension <property>=<value>]... [--debug] [--site <site_man>]
[--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 9-1: Creating an organization

```
$ mam-create-organization -d "Sciences Department" sciences

Successfully created 1 organization
```

Related Topics

- mam-create-organization

9.2 Querying Organizations

To display organization information, use the command *mam-list-organizations*.

mam-list-organizations [[-o] <organization_pattern>] [-X, --extension <property>=<value>]... [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--debug] [--site <site_man>] [--help] [--man] [--quiet] [--version] [--about]

Example 9-2: Listing all organization names

```
$ mam-list-organizations --show Name -q

+ arts
+ sciences
```

Related Topics

- mam-list-organizations

9.3 Modifying Organizations

To modify an organization, use the command *mam-modify-organization*.

mam-modify-organization {[-o] <organization_name>} [-d <description>]
[-X, --extension <property>=<value>]... [--debug] [--site <site_
name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 9-3: Changing an organization's description

```
$ mam-modify-organization -d "Art College" art

Successfully modified 1 organization
```

Related Topics

- mam-modify-organization

9.4 Deleting Organizations

To delete an organization, use the command *mam-delete-organization*.

mam-delete-organization {[-o] <organization_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 9-4: Deleting an organization

```
$ mam-delete-organization arts

Successfully deleted 1 organization
```

Related Topics

- mam-delete-organization

9.5 Organization Auto-Generation

It is possible to have organizations be created automatically when initially added as a member of another object. In order for organization auto-generation to occur, the `AutoGen` property for the Organization object must be set to `True`. This is the default. For organization auto-generation to occur when initially encountered in a usage function, the Values property of the organization attribute of the UsageRecord object must be set to `@Organization`. Additionally, for organization auto-generation to occur when an organization is added as a member of another object (such as a hypothetical Site object) via an association table (e.g. SiteOrganization), the `Values` property for the organization attribute of the Association (e.g. Name) must be set to `@Organization`, indicating that that value should be constrained to be a valid instance of the Organization object. The auto-creation of organizations can be completely disabled by setting the `AutoGen` property for the Organization object to `False`.

Example 9-5: Enable auto-generation of organizations when initially seen in a charge

```
$ mam-shell Attribute Modify Object==UsageRecord Name==Organization
Values=@Organization

Successfully modified 1 attribute
```

Example 9-6: Disable all auto-generation of organizations

```
$ mam-shell Object Modify Name==Organization AutoGen=False

Successfully modified 1 object
```

See Object Auto-Generation for more information about the auto-generation of objects.

Related Topics

9.6 Default Organization

It is possible to set a global default organization to which usage would be ascribed in quotes, liens, or charges where no organization is specified. This can be accomplished by setting the `DefaultValue` property for the `Organization` object to the desired organization name.

It is also possible to set an organization default for a specific object, which will result in usage being ascribed to the specified organization when the object is attributed to the usage. This is done by creating a default usage override modifier. For example, to specify that retail be the default organization for usage associated with the user amy, you might first create an attribute called `DefaultOrganization` for the User Object with the `Values` property of `@?=Organization`. Then you would populate the new `DefaultOrganization` property for the `amy` user with the value of retail. See the chapter on Customizing Objects for more information on default and other usage override modifiers.

Example 9-7: Assign a global default organization

```
$ mam-shell Object Modify Name==Organization DefaultValue=sciences

Successfully modified 1 object
```

Chapter 10: Managing Funds

A fund is a container for a time-bounded reference currency called credits for which the usage is restricted by constraints that define how the credits must be used. Much like with a bank, a fund is a repository for these resource or service credits which are added through deposits and debited through withdrawals and charges. Each fund has a set of constraints designating which entities (such as Users, Accounts, Machines, Classes, Organizations, etc.) may access the encapsulated credits or for which aspects of usage the funds are intended (QualityOfService, GeographicalArea, etc.). Fund constraints may also be negated with an exclamation point leading the constraint value.

In this chapter:

10.1 About Funds

Funds may have a name that is not necessarily unique for the fund. Funds may also have a priority that will influence the order of fund selection when charging. A default deposit amount can be set for a fund, which is used when the amount is not specified for a deposit. Derived properties such as Allocated, Balance, Effective, Available, Capacity, PercentRemaining, PercentUsed and Used can be displayed via the `mam-list-funds` or `mam-balance` commands (see the commands reference for mam-list-funds or mam-balance for more details). Operations include creating, querying, modifying, deleting and resetting funds as well as making deposits, withdrawals, transfers and balance queries. By default, a standard user may only query and view the balance for funds that pertain to them.

Credits are added to a fund via a deposit. If no amount is specified for the deposit, the fund's default deposit value is used for the deposit amount. When credits are deposited into a fund, they are associated with a time period within which they are valid. These time-bounded pools of credits are known as allocations. The initial deposit into a fund will create a new allocation having the specified or default time boundaries.

A fund may be reset, causing the currently active allocation to end and creating a new allocation. When a fund is reset, the default deposit amount will be used to determine the amount of the new allocation. A zero default deposit amount will result in the creation of an allocation with a zero balance. A negative default deposit amount can be used to stipulate that the allocations in the fund should be ended if the fund is reset. An empty default deposit amount stipulates that no change will be made to the allocations if the fund is reset. As an alternative to resetting funds, allocations with predesignated start and end times may be created ahead of time. By using one of these methods to implement periodic allocations, it is possible to establish an allocation cycle. See Chapter 11: Managing Allocations - page 115 for more information on periodic allocations, as well as, credit limits and infinite allocations.

Funds may be nested. Hierarchically nested funds may be useful for the delegation of management roles and responsibilities. Deposit shares may be established that assist to automate a trickle-down effect for credits deposited at higher level funds. Additionally, an optional overflow feature allows charges against lower level funds to trickle up the hierarchy.

Some fund operations (Query, Deposit, Withdraw, and Refund) allow the specification of filter options that narrow down the funds that will be acted on for that operation. There are three fund filter types that can be employed: ExactMatch, Exclusive, and NonExclusive. The NonExclusive filter type will be used by default if no filter type is specified.

- If an exact-match filter type is used, the query will return only the funds for which the specified filters exactly match the fund constraints. For example, `Fund Query FilterType:=ExactMatch Filter:=User=amy` would only return a fund with the sole constraint User=amy.

- If an exclusive filter type is used, the query will return only the funds for which the specified filters meet all constraints for usage. Another way to think of an exclusive filter is to ask: "If usage were to be posted given only the specified filter options as ACLs, which funds would be eligible for charging?". For example, `Fund Query FilterType:=Exclusive Filter:=User=amy` would not return a fund with the sole constraint `Machine=blue`

because `Machine=blue` was not included in the filters. Not only must the filters be a non-conflicting superset of the fund constraints, but all constraint dependencies must also be satisfied (for example, an appropriate user may need to be specified with the account).

- If a non-exclusive filter type is used, the query will return all funds for which the filters do not specifically exclude the constraints. The query assumes that if constraints are not specified within the filters, they can be assumed as a wildcard and will return all funds that are not specifically excluded by the filter. For example, `Fund Query FilterType:=NonExclusive Filter:=User=amy` would return a fund whose only constraint was `Machine=blue` (because it does not conflict) but would not return a fund with the constraint `User=bob` (because it does conflict).

10.2 Creating Funds

mam-create-fund is used to create a new fund. You can specify a fund name, a description, and any number of fund constraints. If a name is not specified and constraints are specified, a name will be automatically generated based on the constraints. A new unique id is automatically generated for the fund.

mam-create-fund [-n <fund_name>] [--priority <fund_priority>] [--default-deposit <deposit_amount>] [-d <description>] [-X, --extension <property>=<value>]... [-u <user_name>,...]... [-g <group_name>,...]... [-a <account_name>,...]... [-o <organization_name>,...]... [-c <class_name>,...]... [-m <machine_name>,...]... [--constraint <constraint_name>=[!]<constraint_value>,...]... [--parent <parent_fund_id>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

> 🛈 It is possible to have funds be created automatically when accounts are created by setting the Fund object's AutoGen property to true (see Fund Auto-generation). The auto-generated fund will be associated with the new account.

Example 10-1: Creating a fund valid for the `chemistry` *account*

```
$ mam-create-fund -a chemistry -n "Chemistry"

Successfully created 1 fund with id 7 and 1 constraint
```

Example 10-2: Creating a wide-open fund that can be used by anyone for anything

```
$ mam-create-fund -n "Windfall"

Successfully created 1 fund with id 8
```

Example 10-3: Creating a fund valid toward all `biology` *account members except for* `dave` *and just the machine colony*

```
$ mam-create-fund --constraint Account=biology,User=!dave,Machine=colony -n "Biology on Colony not for Dave"

Successfully created 1 fund with id 9 and 3 constraints
```

Related Topics

- mam-create-fund

10.3 Querying Funds

To display fund information, use the command *mam-list-funds*.

mam-list-funds [[-f] <fund_id>] [-A | -I] [-n <fund_name>] [-X, --extension <property>=<value>]... [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours] [--debug] [--site <site_man>] [--help] [--man] [--quiet] [--version] [--about]

Example 10-4: Listing all info about all funds with multi-valued fields displayed in a multi-line format

```
$ mam-list-funds --long

Id Name              Constraints        Allocated Balance   DefaultDeposit Description
-- ----------------- ------------------ --------- --------  -------------- -----------
1  biology           Account=biology    25000000  25000000      25000000
2  chemistry for amy User=amy           35000000  34802392      35000000
                     Account=chemistry
3  chemistry not amy User=!amy          50000000  50000000      50000000
                     Account=chemistry
4  film on colony    Account=film              0         0
                     Machine=colony
```

Example 10-5: Wide listing all info about all funds useable by amy

```
$ mam-list-funds -u amy

Id Name              Constraints                Allocated Balance   DefaultDeposit
Description
-- ----------------- -------------------------- --------- --------  -------------- ---
--------
1  biology           Account=biology            25000000  25000000      25000000
2  chemistry for amy Account=chemistry,User=amy 35000000  34802392      35000000
4  film on colony    Machine=colony,Account=film        0         0
```

Related Topics

- mam-list-funds

10.4 Modifying Funds

To modify a fund, use the command *mam-modify-fund*.

mam-modify-fund [[-f] <fund_id>] [-u <user_name>] [-g <group_name>]
[-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] {{[-n <fund_name>] [--priority <fund_priority>] [--default-deposit <deposit_amount>] [-d <description>] [-X, --extension <property>=<value>]... [—add-constraint <constraint_name>=[!]<constraint_value>,...] [--del-constraint(s) <constraint_name>[=<constraint_value>],...]... [--parent <parent_fund_id>]} | {--reset [--all]}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

*Example 10-6: Adding a constraint to a fund so that it can only be used by the *acme* organization*

```
$ mam-modify-fund --add-constraint Organization=acme 7

Successfully created 1 constraint
```

Example 10-7: Setting the default deposit amount for a fund

```
$ mam-modify-fund --default-deposit 5000000 -f 1

Successfully modified 1 fund
```

Example 10-8: Resetting a fund

```
$ mam-modify-fund --reset 1

Successfully deposited 5000000 credits into fund 1
Successfully stopped 1 allocation
Successfully created 1 allocation
```

Related Topics

- mam-modify-fund

10.5 Making Deposits

mam-deposit is used to deposit time-bounded resource credits into a fund resulting in the creation or increase of an allocation. (See Chapter 11: Managing Allocations - page 115 for information about managing allocations). The start time will default to `-infinity` and the end time will default to `infinity` if not specified. Filter options can be specified to help select a unique fund for the deposit. If multiple funds are matched by the filters, the matching funds will be listed and you will be prompted to respecify the deposit with one of the fund ids. If an allocation for the deposit fund is found having the start and end times for the deposit, the amount of the allocation will be increased by the deposit amount. Otherwise, a new allocation will be created for the fund with the amount of the deposit. If no funds match your criteria, if fund auto-generation is enabled, a fund will be created and the deposit made into it. Otherwise, the deposit will fail (the fund will need to be first created using *mam-create-fund*).

Deposits may be used to extend the debit ceiling by specifying an amount for the deposit (with the `-z` option) or extend the credit floor by specifying a credit limit for the deposit (with the `-L` option) or a combination of both options may be used. Additionally, `Infinity` may be used for either of these option values when Moab Accounting Manager is coupled with a database that supports IEEE Standard 754 for Floating-Point Arithmetic (e.g. PostgreSQL).

To make a deposit, use the command *mam-deposit*.

mam-deposit [-f <fund_id>] [-i <allocation_id>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filterType ExactMatch|Exclusive|NonExclusive] [[-z] <deposit_amount>] [-L <credit_limit>] [-s <start_time>] [-e <end_time>] [--reset] [-d <description>] [--create-fund True|False] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 10-9: Making a deposit into `fund 1`

```
$ mam-deposit -z 360000000 -f 1

Successfully created 1 allocation
```

Example 10-10: Making a deposit "into" an account

If an account has a single fund then a deposit can be made against the account.

```
$ mam-deposit -z 360000000 -a chemistry

Successfully deposited 360000000 credits into fund 2
```

Example 10-11: Creating a credit allocation

```
$ mam-deposit -L 10000000000 -f 3

Successfully created 1 allocation
```

Example 10-12: Making a reset deposit

Stop the active allocation within a fund and create a new allocation.

```
$ mam-deposit -f 4 -z 36000000 --reset

Successfully deposited 36000000 credits into fund 4
Successfully stopped 1 allocation
Successfully created 1 allocation
```

Example 10-13: Creating an infinite allocation

```
$ mam-deposit -z Infinity -f 5

Successfully deposited inf credits into fund 5
Successfully created 1 allocation
```

> ⓘ The use of infinite allocations requires the use of a database that supports the IEEE Standard 754 for Floating-Point Arithmetic (e.g. PostgreSQL).

Example 10-14: Creating a future quarterly allocation

```
$ mam-deposit -s 2018-10-01 -e 2019-01-01 -z 25000000 -a biology

Successfully created 1 allocation
```

Related Topics

- mam-deposit

10.6 Querying the Balance

To display balance information, use the command *mam-balance*.

mam-balance [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filterType ExactMatch|Exclusive|NonExclusive] [--ignore-ancestors] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 10-15: Querying amy*'s balance*

```
$ mam-balance -u amy

ID  Name      Balance Reserved Effective CreditLimit Available
--  --------- ------- -------- --------- ----------- ---------
1   biology   2785.87  103.22   2682.65        0.00   2682.65
2   chemistry 1785.87    0.00   1785.87        0.00   1785.87
```

Example 10-16: List the available balances that amy *can charge against along with the constraints on those balances*

```
$ mam-balance -u amy —show Available,Constraints

Available Constraints
--------- ---------------------------
 25000000 Account=biology
 34802392 Account=chemistry,User=amy
        0 Machine=colony,Account=film
```

Related Topics

- mam-balance

10.7 Personal Balance

The *mybalance* has been provided as a wrapper script to show users their personal balance. It provides a list of balances for the funds that they can charge to:

mybalance [--hours] [--help] [--man]

Example 10-17: List my fund balances

```
$ mybalance

Name              Available
----------------- ---------
biology            25000000
chemistry for amy  34802392
```

Example 10-18: List my balance in (Processor) hours

```
$ mybalance --hours

Name              Available
----------------- ---------
biology             6944.44
chemistry for amy   9667.33
```

Related Topics

- mybalance

10.8 Making Withdrawals

A withdrawal can be used to debit a fund without being associated with the usage charge from some item. To issue a withdrawal, use the command `mam-withdraw`.

`mam-withdraw` [`-f` <fund_id>] [`-i` <allocation_id>] [`-u` <user-name>] [`-g` <group_name>] [`-a` <account_name>] [`-o` <organization_name>] [`-c` <class_name>] [`-m` <machine_name>] [`--filter` <filter_name>=<filter_value>]... [`--filter-type` ExactMatch|Exclusive|NonExclusive] {[`-z`] <withdrawal_amount>} [`-d` <description>] [`--hours`] [`--debug`] [`--site` <site_name>] [`--help`] [`--man`] [`--quiet`] [`--verbose`] [`--version`] [`--about`]

Example 10-19: Making a withdrawal

```
$ mam-withdraw -z 12800 -f 1 -d "Grid Tax"

Successfully withdrew 12800 credits from fund 1
```

Example 10-20: Making a withdrawal "from" an account

If an account has a single fund then a withdrawal can be made against the account.

```
$ mam-withdraw -z 12800 -a biology

Successfully withdrew 12800 credits from fund 1
```

If more than one fund exists for the account or filter, you will be asked to be more specific:

```
$ mam-withdraw -z 12800 -a chemistry

Multiple funds were matched for the withdrawal.
Please respecify using one of the following fund ids:
2    [chemistry for amy]
3    [chemistry not amy]
```

Related Topics

- mam-withdraw

10.9 Making Transfers

To issue a transfer between funds, use the command *mam-transfer*. If the allocation id is specified, then only credits associated with the specified allocation will be transferred, otherwise, only active credits will be transferred. Fund transfers preserve the allocation time periods associated with the resource or service credits from the source to the destination funds. Source and destination filters may be used if they result in a single source fund and single destination fund.

mam-transfer {--from-fund <source_fund_id> &| --from-allocation <source_allocation_id> &| --from-filter <filter_name>=<filter_value>...} {--to-fund <destination_fund> &| --to-allocation <destination_allocation_id> &| --to-filter <filter_name>=<filter_value>...} [--filter-type ExactMatch|Exclusive|NonExclusive] {[-z] <transfer_amount>} [-d <description>] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 10-21: Transferring credits between two funds

```
$ mam-transfer --from-fund 1 --to-fund 2 10000

Successfully transferred 10000 credits from fund 1 to fund 2
```

Example 10-22: Transferring credits between two single-fund accounts

```
$ mam-transfer --from-filter Account=biology --to-filter Account=chemistry 10000

Successfully transferred 10000 credits from fund 1 to fund 2
```

Related Topics

- mam-transfer

10.10 Obtaining a Fund Statement

To generate a fund statement, use the command *mam-statement*. For a specified time frame it displays the beginning and ending balances as well as the total credits and debits to the fund over that period. This is followed by an itemized report of the debits and credits. Summaries of the debits and credits will be displayed instead of the itemized report if the --summarize option is specified. If filter options are specified instead of a fund, then the statement will consist of information merged from all funds valid toward the specified entities.

mam-statement [[-f] <fund_id>] [-n <fund_name>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] [-s <start_time>] [-e <end_time>] [--summarize] [--hours] [--debug] [--site <site_man>] [--help] [--man] [--version] [--about]

Example 10-23: Generating a fund statement for all chemistry *funds for the fourth quarter of 2017*

```
$ mam-statement -a chemistry -s 2017-10-01 -e 2018-01-01 --summarize

###########################################################################
#
# Includes fund 3 (chemistry not amy)
# Includes fund 2 (chemistry for amy)
# Generated on Mon Feb  7 18:44:23 2018.
# Reporting fund activity from 2017-10-01 to 2018-01-01.
#
###########################################################################
Beginning Balance:         0
-------------------------- ------------------------------
Total Credits:             90122212
Total Debits:              -5308668
-------------------------- ------------------------------
Ending Balance:            84813544
############################# Credit Summary #############################
Object        Action     Amount
----------    --------   --------
Fund          Deposit    90100000
UsageRecord   Refund        22212
############################# Debit Summary #############################
Object         Action     Account    User   Machine    Amount    Count
-------------  ---------  ---------  ----   --------   -------   -----
UsageRecord    Charge     chemistry  amy    colony     -19744    239

############################# End of Report #############################
```

> ℹ The fields which are used as default discriminators in the detail section of the mam-statement command (which are by default Account, User, and Machine) can be customized by setting the statement.show configuration parameter in mam-client.conf.

Related Topics

- mam-statement

10.11 Deleting Funds

To delete a fund, use the command `mam-delete-fund`.

`mam-delete-fund` `{[-f] <fund_id>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]`

Example 10-24: Deleting a fund

```
$ mam-delete-fund 2

Successfully deleted 1 fund
```

Related Topics

- mam-delete-fund

10.12 Fund Auto-generation

It is possible to enable the auto-generation of funds by setting the `AutoGen` property of the Fund object to `True`. When creating a new account, if fund auto-generation is enabled, a fund will automatically be created for the account (unless overridden with the `--create-fund` option). The fund will be usable only by usage attributed to the new account. Additionally, if fund auto-generation is set, a deposit that does not match an existing fund will automatically generate a fund using the filters as constraint options. Objects associated with the constraint that have `AutoGen` set to `True` will be auto-generated as well (unless overridden with the `--create-fund` option).

Example 10-25: Enable auto-generation of funds

```
$ mam-shell Object Modify Name==Fund AutoGen=True

Successfully modified 1 object
```

10.13 Hierarchical Funds

A hierarchy can be established between funds. When creating a fund or by modifying it later, one can specify a parent fund id via the `--parent` option to establish the object fund as a child of the specified parent fund. A fund may have multiple children funds but only a single parent fund.

Example 10-26: Establishing a child relationship with another fund

```
$ mam-modify-fund --parent 3 -f 6

Successfully added 1 parent
```

Deposit shares may be established between the parent fund and its children that assist to automate a trickle-down effect for funds deposited at higher level funds (`DepositShare` is an attribute of the FundFund association object). Deposit shares are integers and are treated as a percentage of each deposit and the sum of the deposit shares for a fund's children may not exceed 100. If the deposit shares for the children of a fund totals less than 100, the difference is taken to be the share of the deposit that will be allocated to the parent. When a deposit is made into a parent fund, for each child fund that has a non-zero deposit share a recursive deposit amounting to the designated percentage of the parent deposit is issued to that child. After the share amounts have been deposited to each of the child funds, the remaining percentage of the deposit is allocated to the parent fund. This effect is recursive with each child. If a start time and/or end time are specified in the original deposit, these time frames will be recursively applied to all descendant deposits. You have to use the mam-shell interactive control program to manage deposit shares. For the FundFund association object, the Fund is the parent and the Id is the child.

Example 10-27: Establishing a 10% deposit share between a parent and a child fund

```
$ mam-shell FundFund Modify Fund==3 Id==6 DepositShare=10

Fund    Id    DepositShare  Overflow
------  ----  ------------  --------
3        6    10            False

Successfully modified 1 funFund
```

An overflow policy may be established between the parent fund and its children to enable a trickle-up effect for charges, liens, and quotes from the lower level funds (Overflow is an attribute of the FundFund association object). The Overflow attribute is a boolean value (`True` or `False`). If the overflow value between a child and its parent is set to `True`, any charges, liens, or quotes issued against the child fund that cannot be satisfied by the balance in the child fund, will recursively issue the unsatisfied portion of the charge, lien, or quote against the parent fund. If the charge, lien, or quote cannot be satisfied by the ancestors, no charges, liens, or quotes will result against any of funds. The balance in the descendant funds will be depleted before ancestor funds. This effect is recursive with each parent. If a parent fund is linked with overflow to a child fund and a charge, lien, or quote overflows to the parent fund, the constraints of the parent fund will not be checked against the properties of the item. One must use the mam-shell control program to manage the overflow policy. For the FundFund association object, the Fund is the parent and the Id is the child.

Example 10-28: Enabling overflow between a parent and a child fund

```
$ mam-shell FundFund Modify Fund==3 Id==6 Overflow=True

Fund    Id    DepositShare  Overflow
------  ----  ------------  --------
3       6     10            True

Successfully modified 1 fundFund
```

10.14 Fund Priority

By default, when an item can charge to multiple funds, funds with more constraints are chosen over funds with fewer constraints. For example, if the user `amy` is charging against the chemistry account for usage of an item and there are two viable funds, one with a single constraint (e.g. `Account=chemistry`) and another with two constraints (e.g. `Account=chemistry` and `User=amy`), credits will be taken from the more specific fund (with 2 constraints) before they are taken the more general fund (with 1 constraint). To override this behavior, it is possible to give a priority to a fund. The priority factor of a fund has higher precedence than the specificity (constraint count) of the fund. Thus, all else being equal, if a fund with a lower number of constraints is given a higher priority than a fund with a higher number of constraints, the higher priority fund will be depleted first. Other factors, such as the end time of the allocation or whether there is an existing lien for the item against a fund, have a higher precedence than the specificity of the fund. If you want the allocations in a particular fund to be chosen before allocations that expire sooner or that have a lien, you may need to specify fund priorities that are in the millions (see Allocation Precedence for a discussion of the manner of sorting allocations for charging).

Example 10-29: Setting a fund priority

```
$ mam-modify-fund -f 3 --priority 1

Successfully  modified 1 fund
```

Chapter 11: Managing Allocations

An allocation is a time-bound pool of credits associated with a fund. A fund may have multiple allocations, each for use during a different time period. Normally, only a single allocation will be active within a fund at any given time.

11.1 About Allocations

Allocations are normally created via a fund deposit. An allocation has an amount, an initial deposit, and an allocated value. The `Amount` attribute tracks the current amount of credits in the allocation. The `InitialDeposit` attribute stores the amount originally deposited into an allocation when it is initially created. The `Allocated` attribute stores the current adjusted allocated amount. It is begins set to the initial deposit amount and is incremented with each crediting deposit or incoming transfer and decremented with each withdrawal or outgoing transfer. When a deposit is made to a fund, if a matching allocation already exists with the appropriate time period, the existing allocation is modified. Otherwise, a new allocation is created. A resetting deposit will end the currently active allocation and create a new allocation.

An allocation has a start time and an end time that defines the time period during which the allocation may be used. If a start time or end time is specified when making a deposit, an existing allocation having the specified boundary times will be credited. If no start time or end time is specified, the active allocation will be credited. If no matching or active allocations can be found, a new allocation will be created with the specified or default start and end time (the start time defaults to the present and the end time defaults to infinity). An active flag is automatically updated to `True` if the allocation is within its valid time frame or `False` if it is not. An allocation that becomes active because the current time is greater than its start time undergoes an activation that normally registers as a credit to the fund. An allocation that becomes inactive because the current time is greater than its end time undergoes a deactivation that normally registers as a debit to the fund.

By using multiple allocations that expire in regular intervals it is possible to implement a use-it-or-lose-it policy and establish an allocation cycle. There are two primary methods to implement periodic allocations. In the first method, called Resetting Allocations, funds are reset (ending the current allocation and creating a new one) at the beginning of each allocation period. By setting and maintaining an appropriate default deposit amount for each fund, the process of resetting funds can be simplified. The periodic reset can be performed either by making a resetting deposit for each fund (e.g. `mam-deposit -f 1 --reset`), which allows you to override default deposit amounts, by calling the reset action for each fund (e.g. `mam-modify-fund -f 1 --reset`), which allows you to select which funds to reset, or by invoking a reset across all funds (e.g. `mam-modify-fund --reset --all`). The effect of any of these commands is to end the currently active allocation in th ffund and then make a fresh deposit. The fund's default deposit amount is used any time the amount is not specified in a deposit (as in the case of a fund reset command). If the default deposit amount is positive, the currently active allocation is ended and a new allocation is created with the default amount. If the default deposit amount is set to a value of zero, the active allocation is ended and no new allocation is created. If the default deposit amount is not set, the fund's allocations are not affected. The reset can be performed via a scheduled event or via a cron script. If default deposit amounts are kept up-to-date (including being zeroed out for funds that are slated to end and being unset for funds that you do not want affected by the reset), automation of this method can be as simple as creating a single periodic event with a FireCommand of "Fund Reset (see Creating Events). In the second method, called Expiring Allocations, funds with predesignated start and end times are created head of time. When the beginning of an allocation period is reached, the currently active allocation automatically expires and the next one automatically becomes active. A future allocation is created by making a deposit while specifying a start time and an end time in the future (e.g. `mam-deposit -f 1 -s 2018-10-01 -e 2019-01-01`). This method can also take advantage of default deposit amounts. The overall effect of either of these methods is very similar.

By default, Moab Accounting Manager attempts to enforce Discrete allocations, or ensure that allocations within a fund are non-overlapping (in time) and non-reusable (each allocation period should use a distinct allocation). This behavior is designated by the `allocation.enforediscrete` server configuration parameter. If set to `true`, this policy prevents new allocations within a fund from overlapping existing ones. Enabling this policy helps to improve clarity when reporting on allocation usage during a particular period. If set to `false`, overlapping allocations within a fund can be created. This might be useful if you want to allow the remaining balance from a prior allocation period to carry over into the new allocation period. With overlapping allocations, it is harder to describe what percentage of a group's allocation has been used. This policy is applied when making deposits that create new allocations, when making transfers that create new allocations, or when modifying the start and end times of an existing allocation. It is possible to override the configured policy for an individual command by specifying the `EnforcedDiscrete` option (e.g. `mam-deposit --option name=EnforceDiscrete value=False`).

An allocation may have a credit limit representing the amount by which it can go negative. Thus, by having a positive balance in the `Amount` field, the fund is like a debit account, implementing a pay-first use-later model. By establishing a credit limit instead of depositing an initial balance, the fund will be like a credit account, implementing a use-first pay-later model. These strategies can be combined by depositing some amount of funds coupled with a credit limit, implementing a form of overdraft protection where the funds will be used down to the negative of the credit limit.

It is possible for the allocation `Amount` or `CredLimit` to be set to Infinity (via a deposit). If the amount is infinite, debits will not decrease the balance. An infinite deposit will result in an infinite Allocated amount. If the credit limit is infinite, there will be no negative limit for debits. It is not possible to have infinite charges, liens, quotes, withdrawals, refunds, or transfers. However, it is possible to have infinite allocation activations, deactivations, and deletions. This capability is only available when using a database that supports IEEE Standard 754 for Floating-Point Arithmetic (e.g. PostgreSQL).

Operations include querying, modifying, and deleting allocations. Allocations can be created by a fund deposit, creating a fund with allocation auto-generation enabled, refunding a usage record, or a transfer between funds. Allocations may also be indirectly modified via charges, withdrawals, transfers, or refunds. By default, a standard user may only query allocations that pertain to them.

Allocation queries allow the specification of filter options that filter the allocations to those with funds meeting the specified fund constraints. There are three allocation filter types that can be employed: ExactMatch, Exclusive, and NonExclusive. The NonExclusive filter type will be used by default if no filter type is specified.

- If an exact-match filter type is used, the query will return only the allocations relating to the funds for which the specified filters exactly match the constraints. For example, `Allocation Query FilterType:=ExactMatch Filter:=User=bob` would only return a fund with the sole constraint User=bob.

- If an exclusive filter type is used, the query will return only allocations relating to funds for which the specified filters meet all constraints. For example, `Allocation Query FilterType:=Exclusive Filter:=User=amy` would not return an allocation for a fund with the sole constraint `Machine=blue`.

- If a non-exclusive filter type is used, the query will return all allocations relating to funds for which the filters do not specifically exclude the constraints. The query assumes that if

constraints are not specified within the filters, they can be assumed as a wildcard and will return all allocations involving funds that are not specifically excluded by the filter. For example, `Allocation Query FilterType:=NonExclusive Filter:=User=amy` would return an allocation with a fund whose only constraint was `Machine=blue` but would not return an allocation with a fund with the constraint `User=bob`.

11.2 Creating Allocations

Allocations are normally created by making fund deposits via the `mam-deposit` command (See Making Deposits).

11.3 Querying Allocations

To display allocation information, use the command *mam-list-allocations*.

mam-list-allocations [[-i] <allocation_id>] [-f <fund_id>] [-A | -I |
{[-s <start_time>] [-e <end_time>]}] [-X, --extension
<property>=<value>]... [-u <user_name>] [-g <group_name>] [-
a <account_name>] [-o <organization_name>] [-c <class_name>] [-
m <machine_name>] [--filter <filter_name>=<filter_value>]... [--
filter-type ExactMatch|Exclusive|NonExclusive] [--include-ancestors]
[--full] [--show <attribute_name>,...] [--format csv|raw|standard] [-
-hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--
version] [--about]

Example 11-1: Listing allocations for fund 1

```
$ mam-list-allocations -f 1

Id Fund Active StartTime  EndTime     InitialDeposit Allocated CreditLimit Remaining
PercentUsed
-- ---- ------ ---------- ---------- -------------- --------- ----------- --------- --
---------
1  1    True   2018-01-01 2018-04-01       25000000  25000000           0  24974400
     0.10
2  1    False  2018-04-01 2018-07-01       25000000  25000000           0  25000000
     0.00
3  1    False  2018-07-01 2018-10-01       25000000  25000000           0  25000000
     0.00
4  1    False  2018-10-01 2019-01-01       25000000  25000000           0  25000000
     0.00
```

Related Topics

- mam-list-allocations

11.4 Modifying Allocations

To modify an allocation, use the command *mam-modify-allocation*.

mam-modify-allocation {[-i] <allocation_id>} [-s <start_time>] [-e <end_time>] [-L <credit_limit>] [-d <description>] [-X, --extension <property>=<value>]... [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 11-2: Changing the end time for an allocation

```
$ mam-modify-allocation -e "2019-01-01" 4

Successfully modified 1 allocation
```

Example 11-3: Changing the credit limit for an allocation

```
$ mam-modify-allocation -L 500000000000 -i 2

Successfully modified 1 allocation
```

Related Topics

- mam-modify-allocation

11.5 Deleting Allocations

To delete an allocation, use the command *mam-delete-allocation*.

mam-delete-allocation {-I | {[-i] <allocation_id>}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 11-4: Deleting an allocation

```
$ mam-delete-allocation 4

Successfully deleted 1 allocation
```

Example 11-5: Purging inactive allocations

```
$ mam-delete-allocation -I

Successfully deleted 2 allocations
```

Related Topics

- mam-delete-allocation

11.6 Allocation Auto-generation

It is possible to enable the auto-generation of allocations by setting the `AutoGen` property of the Allocation object to `True`. When creating a new fund, if allocation auto-generation is enabled, an allocation will automatically be created for the fund via a deposit. The deposit will use the default amount and default credit limit (defined in the `DefaultValue` property of the Allocation Amount and Allocation CreditLimit attributes). The default action for allocation auto-generation is to create an allocation with an infinite credit limit.

Example 11-6: Enable auto-generation of allocations

```
$ mam-shell Object Modify Name==Allocation AutoGen=True

Successfully modify 1 object
```

11.7 Allocation Precedence

When issuing a charge (or lien or quote) for the usage of a resource or service, the feasible allocations are sorted according to a weight given to them for that transaction. The weight for each allocation is calculated as follows.

- Independent of precedence, if the instance has current liens against one or more allocations, the reserved allocations will be debited first in order to avoid double booking.

- For the remaining non-nested funds, allocations will be are given a value of `100 + int ((2147483647 - <end_epoch_time>) / 86400) + 10 * <fund_priority> + <constraint_count>`. Thus, sooner expiring allocations will be used before later expiring allocations.

- Fund priority will be the next highest factor (assuming small priority values of 1-10), followed by the number of constraints on the fund (more specific funds will be used before more general funds). Of course, since priority is configurable, a sufficiently large priority (in the millions) can be used to override the precedence of earlier expiring allocations.

- Lastly, nested funds that become feasible because of overflow to ancestor funds have a negative weighting and are used last, with the earliest expiring allocations being used before later expiring allocations and closer level ancestors being depleted before ancestor funds that are at more distant levels. These allocations are given a weight of `<distance * 100000> - <end_epoch_time>`.

After all feasible allocations are sorted according to the above rules, the charge (or lien or quote) will be applied against the allocations one by one in sorted order (highest value first) until the request is fulfilled, or until it fails due to insufficient funds.

If a transaction is not able to be satisfied in whole, then

- for a *charge*, partial debits will be applied and the entire transaction will succeed regardless of the amount successfully debited.

- for a *quote* or a *lien*, the entire transaction will fail and no partial debits will be applied.

Chapter 12: Managing Liens

A lien is a hold placed against an allocation. Before usage of a resource or service begins, a lien is made against one or more allocations within the requesting user's applicable fund(s).Subsequent usage requests will also post liens while the available balance (active allocations minus liens) allows. When the usage ends, the lien is removed and the actual charge is made to the allocation(s). This procedure ensures that usage will only be permitted so long as the requestors have sufficient funds.

12.1 Creating Liens

Liens are normally created with the *mam-reserve* command (See Making a Usage Lien).

However, it is also possible to create a manual lien against specified allocations using the *mam-create-lien* command. A lien object and its allocation associations will be created. Unlike *mam-reserve*, no calculated lien amount will be returned nor will a usage record be created or updated with the lien. Furthermore, *mam-create-lien* will not perform any checking to ensure that the specified allocations have a sufficient active balance to support the lien.

mam-create-lien [-J <instance_name>] [-s <start_time>] {-e <end_time> | -t <lien_duration>} [-d <description>] [-X, --extension <property>=<value>]... {-A <allocation_id><-<fund_id>=<sublien_ amount>,...}... [--debug] [--site <site_name>] [--help] [--man] [-- quiet] [--verbose] [--version] [--about]

Example 12-1: Creating a manual lien

```
$ mam-create-lien -J weekend_run -t 84600 -A "5<-2=3600"

Successfully created 1 lien
```

⚠ Use of the *mam-create-lien* command bypasses the normal mechanisms that prevent more liens from being placed against an allocation than it can support. Use mam-reserve instead if you wish to avoid the possibility of oversubscribing the allocations.

Related Topics

- mam-create-lien

12.2 Querying Liens

To display lien information, use the command *mam-list-liens*.

mam-list-liens [[-l] <lien_id>] [-A | -I] [-J <instance_pattern>] [-
X, --extension <property>=<value>]... [-u <user_name>] [-g <group_
name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>]
[-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--
filter-type AttributedTo|ImpingesUpon] [--full] [--show <attribute_
name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours] [-
-debug] [--site <site_name>] [--help] [--man] [--quiet] [--version]
[--about]

Example 12-2: Listing all info about all liens for amy

```
$ mam-list-liens -u amy


Id   Instance     Amount   StartTime             EndTime               UsageRecord  Funds
     Description
---  -----------  -------  --------------------  --------------------  -----------  -----
---- -----------  -------------
3    PBS.1234.4   57600    2018-04-06 21:21:48   2018-04-06 22:31:48   7            2
```

Example 12-3: Listing all info about all liens that impinge against dave*'s balance*

```
$ mam-list-liens -u dave --filter-type ImpingesUpon


Id   Instance     Amount   StartTime             EndTime               UsageRecord  Funds
     Description
---  -----------  -------  --------------------  --------------------  -----------  -----
---- -----------
4    batch.12        7600  2018-04-06 15:30:34   2018-04-06 15:41:50   244          3
```

Example 12-4: Listing total of lien amounts broken down by attributed account

```
$ mam-list-liens --show "GroupBy(Account),Sum(Amount)=Reserved"

Account    Reserved
---------  --------
biology        1.00
chemistry      4.00
```

Related Topics

- mam-list-liens

12.3 Modifying Liens

To modify a lien, use the command `mam-modify-lien`.

`mam-modify-lien` {[`-l`] <lien_id>} [`-s` <start_time>] [`-e` <end_time>] [`-t` <lien_duration>] [`-d` <description>] [`-X`, --extension <property>=<value>]... [`--debug`] [`--site` <site_name>] [`--help`] [`--man`] [`--quiet`] [`--verbose`] [`--version`] [`--about`]

Example 12-5: Changing the expiration time of a lien

```
$ mam-modify-lien -e "2018-06-06 14:43:02" 1

Successfully modified 1 lien
```

Related Topics

- mam-modify-lien

12.4 Deleting Liens

To delete a lien, use the command *mam-delete-lien*.

mam-delete-lien {-I | {-J <instance_name>} | {[-l] <lien_id>}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 12-6: Deleting a lien by instance (or job id)

```
$ mam-delete-lien -J PBS.1234.0

Successfully deleted 1 lien
```

Example 12-7: Deleting a lien by Lien Id

```
$ mam-delete-lien 1

Successfully deleted 1 lien
```

Example 12-8: Purging stale liens

```
$ mam-delete-lien -I

Successfully deleted 2 liens
```

Related Topics

- mam-delete-lien

Chapter 13: Managing Quotes

A quotation provides a way to determine beforehand how much would be charged for a job. When a guaranteed quote is requested, the charge rates applicable to the usage request are saved and a quote id is returned. Charge rates may be specified with the quote or the standard rates may be used in the quote calculation. When the lien and the final charge are issued, the quote id can be referenced to ensure that the saved quote charge rates are used instead of current standard values. A quotation has an expiration time after which it cannot be used. A quotation may also be used to verify that the given job has sufficient funds and meets the policies necessary for the charge to succeed.

13.1 About Quotes

Associated with a quote is the id, the instance name (name of the item being used such as the job id), the amount quoted (assuming full use of the quoted resources or services), the usage record (which contains the usage details), a start and end time for the quote, a duration (how long the item is expected to be used), a boolean indicating whether the quote is pinned or unpinned, and a description. Each guaranteed quote will be associated with one or more saved charge rates. Operations include creating, querying, modifying and deleting quotes. By default, a standard user may only query quotes attributed to them.

Quote queries allow the specification of filter options which narrow down the quotes that will be returned. The query will return all quotes associated with usage records satisfying the filters. For example, `Quote Query Filter:=User=scottmo` will return all quotes for resources or services allocated to `scottmo`.

A quote may be pinned (restricted to a particular instance) or unpinned (allowed to be used by any number of different instances). If a quote is pinned and has not been tied to a particular instance when initially created, it will be tied to the first instance that claims it. Once pinned to an instance, it can then be used repeatedly by that same instance until the quote expires, but not by any other instance. If a quote is not pinned, any instances may use the quoted rates while the quote is active.

A quote may be granted a grace period, which is defined as the difference between the validity period of the quote (end time minus start time) and the expected duration of the usage in seconds. The purpose of a grace period is to account for the fact that we may not know precisely when the usage will begin and the quote needs to be valid during the time of completion of the usage in order for the guaranteed charge rates to be applied. One can apply a desired grace period for a quote by setting the end time longer than the specified duration. Alternatively, a grace duration option can be specified with the duration when creating a quote via *mam-quote* as a helper to computing a relatively adjusted end time.

A distinction may be made between quotes and quote templates, both of which use the Quote object. A quote will always return a cost estimate and will be associated with a specific usage record. A quote template provides a way to bundle together a package of special charge rates that can be applied to quotes, liens, and charges. Quote templates use the same Quote object as regular quotes but they are not associated with a usage record and do not generate a quote amount.

In calculating a price, a quote will use (in order of lower to higher precedence) the standard charge rates, the charge rates from a specified quote template, the specified override charge rates, or an externally specified charge amount. In saving guaranteed charge rates, the standard charge rates pertaining to the specified usage record properties will be used unless overridden by a specified quote template or specified charge rates.

There are several key purposes for using quotes and quote templates. First, a quote may be requested to discover the cost of using a resource or service. If this is your sole purpose, then you may want to use the mam-quote command with the `--costOnly` option. Second, a quote can be used to check whether the requestor has sufficient access and funds to use the requested resource. This may be accomplished by invoking the mam-quote command without the `--costOnly` option. Third, a quote or a quote template can be used to lock-in current or specified charge rates for use in future liens and charges. If the details of the usage are known and you would like to get a quote amount with a quote id that can be referenced to guarantee the quoted charge rates, you may use

the mam-quote command with the `--guarantee` option. Override charge rates may be factored in to the cost estimate of the quote by using the mam-quote command with the `--rate` option. If specific override charge rates need to be saved or guaranteed for future use within a quote, lien, or charge without generating a cost estimate, create a pinned quote template by using the *mam-create-quote* command with the `--pin` and `--rate` options. If it is necessary to create a quote template that can be used to override the standard charge rates for multiple instances, use the mam-create-quote command with the `--nopin` and `--rate` options.

13.2 Creating Quotes

Quotes are normally generated by the resource management system with the *mam-quote* command before an instance uses requested resources or services (see Obtaining Usage Quotes).

13.3   Creating Quote Templates

Quote templates may be created by using the *mam-create-quote* command. Quote templates provide a way to bundle together a package of special charge rates that can be applied to quotes, liens, and charges.

mam-create-quote [[--pin] [-J <instance_name>] | --nopin] [-s <start_ time>] {-e <end_time> | -t <quote_duration>} [-d <description>] [-X, --extension <property>=<value>]... {--rate <charge_rate_name> [{<charge_rate_value>}]=<charge_rate_amount>,...}... [--debug] [-- site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 13-1: Creating a pinned quote template

```
$ mam-create-quote --pin -J vpc.1 -t 86400 --rate

Processors=1.5/s,QualityOfService{Premium}=*1.7
Successfully created 1 quote template with id 17
```

Example 13-2: Creating an unpinned quote template

```
$ mam-create-quote --nopin -t 86400 --rate Disk=2.5/s,License{Matlab}=4/s

Successfully created 1 quote template with id 18
```

> ℹ️ Use of the *mam-create-quote* command will not result in a cost estimate or the creation of a usage record. Use *mam-quote* instead if you wish to obtain a quote for usage.

Related Topics

- mam-create-quote

13.4 Querying Quotes

To display quote information, use the command *mam-list-quotes*.

mam-list-quotes [[-q] <quote_id>] [-J <instance_name>] [-A | -I] [-X, --extension <property>=<value>]... [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours]] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 13-3: Listing all quotes for user amy *on machine colony*

```
$ mam-list-quotes -u amy -m colony


Id  Amount  Pinned  Instance  UsageRecord  StartTime           EndTime             D
uration   ChargeRates   Description
--- ------- ------  --------- -----------  ------------------- ------------------- -
-------  -------------  --------------------
1   57600   True               242         2018-04-06 12:49:53 2018-04-
13 13:09:58  3600        Processors:1/s
```

Related Topics

- mam-list-quotes

13.5 Modifying Quotes

To modify a quote, use the command *mam-modify-quote*.

mam-modify-quote {[-q] <quote_id>} [-s <start_time>] [-e <end_time>]
[-d <description>] [-X, --extension <property>=<value>]... [--debug]
[--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 13-4: Changing the expiration time of a quote

```
$ mam-modify-quote -e "2018-05-01" 1

Successfully modified 1 quote
```

Related Topics

- mam-modify-quote

13.6 Deleting Quotes

To delete a quote, use the command *mam-delete-organization*.

mam-delete-quote {-I | {[-q] <quote_id>}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 13-5: Deleting a quote

```
$ mam-delete-organization 1

Successfully deleted 1 quote
```

Example 13-6: Purging stale quotes

```
$ mam-delete-organization -I

Successfully deleted 2 quotes
```

Related Topics

- mam-delete-quote

Chapter 14: Managing Usage Records

Moab Accounting Manager can track the usage of resources and services on your system, recording the charge and the details of the usage in a usage record. A usage record is created when a resource or service manager requests a guaranteed quote for usage, places a lien for usage, or charges for the usage of an item. Usage records can also be created directly via UsageRecord Create (`mam-create-usagerecord`). A refund can be invoked to credit a charge amount back to the originating fund. Usage records can also be queried, modified, or deleted. By default, a standard user may only query usage records attributed to them.

In a typical use case, a quote might be used to discover how much it would cost to use an item (resource or service) and to verify the user had sufficient access to the item and funds to cover the requested usage. Just before the item is about to be used, a lien (or hold) might be placed against the user's allocated credits for the requested usage. After the usage is complete, a charge for the actual usage can be debited from their fund and the lien removed.

As is the case for other Moab Accounting Manager objects, usage records are highly customizable. One may remove most usage record properties and add new usage record properties. Refer to the section Customizing the Usage Record Object for examples of customizing usage records.

In this chapter:

14.1 Creating a Usage Record

In most cases, usage records will be created by the resource management system via the API or with the *mam-quote*, the *mam-reserve* or the *mam-charge* command.

However, it is also possible to create usage records directly using the *mam-create-usagerecord* command.

```
mam-create-usagerecord {-J <instance_name>} [-n <designated_name>] [-
T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-
a <account_name>] [-o <organization_name>] [-c <class_name>] [-
Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-
P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-
E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "
{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_
name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_
amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"]
[-W <requested_duration>] [-t <actual_duration>] [-s <start_time>] [-
e <end_time>] [-x <exit_code>] [--stage <lifecycle_stage>] [-
d <description>] [-X --extension <property>=<value>]... [--debug] [--
site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version]
[--about]
```

Example 14-1: Creating a usage record

```
$ mam-create-usagerecord -u jsmith -a chem -m cluster -X Charge=2468 -P 2 -t 1234 -J
PBS.1234.0

Successfully created 1 usage record with id 246
```

> ⓘ The fields which are displayed by default by this command can be customized by setting the `usagerecord.show` configuration parameter in `mam-client.conf`.

> ⓘ Use of the mam-create-usagerecord command to record usage will not result in the debiting of a user's allocation. Use mam-charge instead if you wish to charge for the usage.

Related Topics

- mam-create-usagerecord

14.2 Querying Usage Records

To display usage record information, use the command `mam-list-usagerecords`.

`mam-list-usagerecords` [[`-j`] <usage_record_id>] [`-J` <instance_name_pattern>] [`-T` <usage_record_type>] [`-u` <user_name>] [`-g` <group_name>] [`-a` <account_name>] [`-o` <organization_name>] [`-c` <class_name>] [`-Q` <quality_of_service>] [`-m` <machine_name>] [`--stage` <lifecycle_stage>] [`-X`, --extension <property>=<value>]... [`-s` <start_time>] [`-e` <end_time>] [`--full`] [`--show` <attribute_name>,...] [`--format` csv|raw|standard] [`--hours`] [`--debug`] [`--site` <site_name>] [`--help`] [`--man`] [`--quiet`] [`--version`] [`--about`]

Example 14-2: Show specific info about usage tallied by `amy`

```
$ mam-list-usagerecords --show=Type,Instance,Account,Machine,Charge -u amy

Type   Instance    Account     Machine    Charge
-----  ----------  ----------  ---------- -------
Job    PBS.1234.0  chemistry   colony     22212
```

Example 14-3: Show breakdown of charges by account and user

```
$ mam-list-usagerecords --show "GroupBy(Account),GroupBy(User),Sum(Charge)"

Account    User    Charge
---------  ------- ------
biology    bob       5.00
chemistry  amy       5.00
chemistry  bob       1.00
```

Example 14-4: Show number of jobs per quality of service

```
$ mam-list-usagerecords --show "Count(Instance)=Jobs,GroupBy(QualityOfService)"

Jobs QualityOfService
---- ----------------
40
1    premimum
9    windfall
```

Example 14-5: Show number of jobs using the bigmem node feature

```
$ mam-list-usagerecords --show "Count(Features{bigmem})"

bigmem
------
147
```

Example 14-6: Show number of matlab licenses used by the chemistry account

```
$ mam-list-usagerecords -a chemistry --show "Sum(Licenses{matlab})"

matlab
```

```
------
407
```

Related Topics

- [mam-list-usagerecords](#)

14.3 Modifying a Usage Record

It is possible to modify a usage record by using the command *mam-modify-usagerecord*.

mam-modify-usagerecord {[-j] <usage_record_id> | -J <instance_name>} [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [-t <actual_duration>] [-s <start_time>] [-e <end_time>] [-x <exit_code>] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property_name>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 14-7: Changing a usage record

```
$ mam-modify-usagerecord -Q HalfPrice -X Charge=1234 -d "Benchmark" -J PBS.1234.0

Successfully modified 1 usage record
```

> 🛈 Changing a recorded charge in this manner will not change the allocated balance (see Issuing Usage Refunds to refund a charge).

Related Topics

- mam-modify-usagerecord

14.4 Deleting a Usage Record

To delete a usage record, use the command *mam-delete-usagerecord*.

mam-delete-usagerecord {[-j] <usage_record_id> | -J <instance_name>}
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose]
[--version] [--about]

Example 14-8: Deleting a usage record

```
$ mam-delete-usagerecord -J PBS.1234.0

Successfully deleted 1 usage record
```

Related Topics

- mam-create-usagerecord

14.5 Obtaining Usage Quotes

Usage quotes can be used to determine how much it will cost to use a resource. Provided the cost-only option is not specified, this step will additionally verify that the submitter has sufficient funds and meets all the allocation policy requirements for the usage, and can be used at the submission of the usage request as an early filter to prevent the usage from getting blocked when it tries to obtain a lien to start later. If a guaranteed quote is requested, a quote id is returned and can be used in the subsequent charge to guarantee the rates that were used to form the original quote. A guaranteed quote has the side effect of creating a quote record and a permanent usage record. A quote id will be returned which can be used with the lien and charge to claim the quoted charge rates. A cost-only quote can be used to determine how much would be charged for usage without verifying sufficient funds or checking to see if the charge could succeed. A breakdown of the charges in the quote can be returned by specifying the `--itemize` option with the `--verbose` option.

To request a usage quote, use the command `mam-quote`.

```
mam-quote [-J <instance_name>] [[-j] <usage_record_id>] [-q <quote_template_id>] [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property>=<value>]... [-zt <quote_duration> [-G <grace_duration>]] [-zs <quote_start_time>] [-z <quote_amount>] [--cost-only | --guarantee] [---rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]
```

Example 14-9: Requesting a quote

```
$ mam-quote -a chemistry -u amy -m colony -P 2 -W 3600

Successfully quoted 7200 credits
```

Example 14-10: Requesting a guaranteed quote

```
$ mam-quote -a chemistry -u amy -m colony -P 16 -W 3600 --guarantee

Successfully quoted 57600 credits with quote id 1 and usage record id 86

$ mam-list-quotes


Id  Amount    UsageRecord  StartTime              EndTime               Duration  Used  Cha
rgeRates       Description
```

```
-------------------------------------------------------------------------------
--- ------- ----------- ------------------- ------------------- --------- ----- ---
------------- -------------
1   57600     86           2018-04-06 10:09:58  2018-04-
06 11:09:58 3600     0     Processors:1/s
-------------------------------------------------------------------------------
```

ℹ It is possible to establish a system default machine, project or user to be used in job functions (charge, reserve or quote) when left unspecified. See 25.2 Server Configuration - page 257 for more information.

Related Topics

- mam-quote

14.6 Making a Usage Lien

A usage lien can be used to place a hold on the user's fund before usage starts to ensure that the credits will be there when it completes. The replace option may be specified if you want the new lien to replace existing liens of the same instance name (associated with the same usage record). The modify option may be specified to dynamically extend any existing lien with the same instance name with the specified characteristics instead of creating a new one. See Managing Liens for more information about these options.

To create a usage lien use the command `mam-reserve`.

```
mam-reserve {-J <instance_name>} [[-j] <usage_record_id>] [-q <quote_
id>] [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>]
[-g <group_name>] [-a <account_name>] [-o <organization>] [-c <class_
name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-
P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-
E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "
{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_
name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_
amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"]
[-W <requested_duration>] [-s <start_time>] [--stage <lifecycle_
stage>] [-d <description>] [-X, --extension <property=value>]... [-
zt <lien_duration> [-zs <lien_start_time> [-G <grace_duration>]] [-
z <lien_amount>] [--modify | --replace] [--rate <charge_rate_name>
[{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--
itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [-
-verbose] [--version] [--about]
```

Example 14-11: Creating a lien

```
$ mam-reserve -J PBS.1234.0 -a chemistry -u amy -m colony -P 2 -W 3600

Successfully reserved 7200 credits with lien id 37 for instance PBS.1234.0 and created
usage record id 87
```

Related Topics

- mam-reserve

14.7 Charging for Usage

A usage charge debits the appropriate allocations based on the attributes of the usage. The charge is calculated based on factors including the resources used, the usage time, and other quality-based factors (see Managing Charge Rates). By default, any liens associated with the charge will be removed. The incremental option may be specified if you want associated liens to be reduced instead of removed. If a usage record already exists for the instance being charged it will be updated with the data properties passed in with the charge request, otherwise a new usage record will be created.

A quote id can be specified to use a previously quoted set of charge rates. This will also ensure the charge will update the usage record instantiated with the quote. A lien id can be specified to help match up a charge with its lien (this may assist in deleting the correct lien if instance ids are not unique). This will also ensure the charge will update the usage record that may have been instantiated by the lien.

Although, by default, Moab Accounting Manager will calculate the charge for the usage using its default charge rates or using the charge rates saved by a referenced quote or quote template, it is possible to specify override charge rates via the rate option. Alternatively, it is possible to designate an externally calculated charge by specifying the charge amount with the Charge option (`-z` option to *mam-charge*).

To charge for a usage use the command *mam-charge*.

mam-charge {-J <instance_name>} [[-j] <usage_record_id>] [-n <designated_name>] [-q <quote_id>] [-l <lien_id>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [-t <actual_duration>] [-s <start_time>] [-e <end_time>] [-x exit_code] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property>=<value>]... [-zt <charge_duration>] [-zs <charge_start_time>] [-z <charge_amount>] [-f <fund_id>] [--incremental] [--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 14-12: Issuing a usage charge

```
$ mam-charge -J PBS.1234.0 -a chemistry -u amy -m colony -P 2 -t 1234

Successfully charged 2468 credits for instance PBS.1234.0
1 lien was removed
```

Related Topics

- mam-charge

14.8 Issuing Usage Refunds

A charged amount can be credited back in part or in whole by issuing a usage refund. This action attempts to lookup the referenced usage record to ensure that the refund does not exceed the original charge and so that the charge entry can be updated. If multiple matches are found (such as the case when instance names (such as job ids) are non-unique), this command will return the list of matched usage records with unique ids so that the correct usage record can be specified for the refund.

To issue a refund for a usage charge, use the command *mam-refund*.

mam-refund {-J <instance_name> | [-j] <usage_record_id>} [-z <refund_amount>] [-i <allocation_id>] [-d <description>] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 14-13: Issuing a usage refund

```
$ mam-refund -J PBS.1234.0

Successfully refunded 19744 credits for instance PBS.1234.0
```

Related Topics

- mam-refund

14.9 Customizing the Usage Record Object

The usage record object as natively defined can be customized with the attributes you want to track in your use cases. The chapter on Customizing Objects goes into some detail on the customization syntax. However, since this may be a common requirement, this section will provide a few examples on modifying, adding and deleting usage record attributes and getting them to be tracked and show up in queries.

Usage record discriminators are those properties which are considered primary differentiators between usage, lien, and quote records. Usage record discriminators are used in the dynamic web portal as filters for the listing, modification, and deletion of usage records, liens, and quotes. The default usage record discriminators are Type, User, Group, Account, Organization, Class, QualityOfService, and Machine. Any new attributes added to the usage record object will become usage record discriminators. Removing a discriminator attribute from the usage record object will necessarily remove it as a usage record discriminator as well. It will be necessary to log out and back in after adding or removing a discriminator in order for it to be reflected in the web GUI.

Example 14-14: Adding an application field (and discriminator)

Let's say you would like to track the application run by the jobs. First, you would add `Application` as an Attribute of the UsageRecord Object.

```
$ mam-shell Attribute Create Object=UsageRecord Name=Application DataType=String

Successfully created 1 attribute
```

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Application,Charge,Stage,User,Group,Account,Organization,Class,Qualit
yOfService,Machine,Nodes,Processors,Memory,Duration,SubmitTime,StartTime,EndTime,Descr
iption
```

If you want to filter the usage records by `Application`, (such as listing all usage records associated with the specified application), use the `-X` (or `--extension`) option in *mam-list-usagerecords*.

```
$ mam-list-usagerecords -X Application=foo --
show=Type,Instance,Charge,User,Application

Type Instance    Charge User Application
---- ---------- ------ ---- -----------
Job  PBS.1234.0 19744  amy  foo
```

You could also use `Application` as the basis of a ChargeRate. See Chapter 16: Managing Charge Rates - page 167 for how to do this.

Although the initial step above allows the application value to be tracked in the usage record, it is also possible to add it as an attribute of the Transaction table so that it will be automatically pop-

ulated from actions having assignments, conditions, options and data values referring to the `Application`.

```
$ mam-shell Attribute Create Object=Transaction Name=Application DataType=String

Successfully created 1 attribute
```

Additionally, the *mam-statement* client command can show `Application` as one of its discriminators (which are Account, User, and Machine by default) in its debit detail. These statement discriminators are specified by the `--show` argument to mam-statement and can be configured with the `statement.show` configuration parameter in `mam-client.conf`.

Example 14-15: Tracking the user-specified job name

The following example demonstrates how to add a Name attribute to the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=Name DataType=String
Description="\"User-Specified Name\""

Successfully created 1 attribute
```

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Name,Charge,Stage,User,Group,Account,Organization,Class,QualityOfServ
ice,Machine,Nodes,Processors,Memory,Duration,SubmitTime,StartTime,EndTime,Description
```

Example 14-16: Tracking accelerator usage

The following examples demonstrate how to track hardware accelerator usage (e.g. GPUs and/or MICS) within the usage record.

- To track GPUs:

  ```
  $ mam-shell Attribute Create Object=UsageRecord Name=GPUs DataType=Integer
  Description="\"Number of GPUs Allocated\""

  Successfully created 1 attribute
  ```

- To track MICs:

  ```
  $ mam-shell Attribute Create Object=UsageRecord Name=MICs DataType=Integer
  Description="\"Number of MICs Allocated\""

  Successfully created 1 attribute
  ```

If you want the new attributes to show up in *mam-list-usagerecords*, you must add them to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
```

```
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,GPUs,MICs,Memory,Duration,SubmitTime,StartTime,EndTime,Descrip
tion
```

Once you have added them to the usage record, you may charge for them by adding an affiliated charge rate. See for how to do this.

Example 14-17: Tracking energy used

The following example demonstrates how to add an Energy attribute to the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=Energy DataType=Float
Description="\"Energy Used\""

Successfully created 1 attribute
```

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the usagerecord.show string in mam-client.conf.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,Memory,Energy,Duration,SubmitTime,StartTime,EndTime,Descriptio
n
```

Example 14-18: Tracking node features

The following example demonstrates how to add a Features attribute to the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=Features DataType=JSON
Description="\"Node Features Allocated\""

Successfully created 1 attribute
```

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the usagerecord.show string in mam-client.conf.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,Memory,Duration,Features,SubmitTime,StartTime,EndTime,Descript
ion
```

Example 14-19: Tracking NUMA properties

The following examples demonstrate how tow to track NUMA properties (e.g.Sockets, NumaNodes, Cores, Threads) with the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=Sockets DataType=Integer
Description="\"Number of NUMA Sockets Allocated\""

Successfully created 1 attribute
```

```
$ mam-shell Attribute Create Object=UsageRecord Name=NumaNodes DataType=Integer
```

```
Description="\"Number of NUMA Nodes Allocated\""

Successfully created 1 attribute
```

```
$ mam-shell Attribute Create Object=UsageRecord Name=Cores DataType=Integer
Description="\"Number of NUMA Cores Allocated\""

Successfully created 1 attribute
```

```
$ mam-shell Attribute Create Object=UsageRecord Name=Threads DataType=Integer
Description="\"Number of NUMA Threads Allocated\""

Successfully created 1 attribute
```

If you want the new attributes to show up in `mam-list-usagerecords`, you must add them to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,Sockets,NumaNodes,Cores,Threads,Memory,Duration,SubmitTime,Sta
rtTime,EndTime,Description
```

Example 14-20: Adding a ProcessorEquivalents field

The following example demonstrates how to track processor equivalents (PE) with the usage record. See the information on PE in Scheduling Environment in the *Moab Workload Manager Administrator Guide* for a description of what processor equivalent means.

```
$ mam-shell Attribute Create Object=UsageRecord Name=ProcessorEquivalents
DataType=Float Description="\"Processor Equivalents\""

Successfully created 1 attribute
```

If you want the new attribute to show up in `mam-list-usagerecords`, you must add it to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,ProcessorEquivalents,Memory,Duration,SubmitTime,StartTime,EndT
ime,Description
```

You could also use ProcessorEquivalents as the basis of a ChargeRate. See Chapter 16: Managing Charge Rates - page 167 for how to do this.

Example 14-21: Adding a BlockedProcessors field

The following example demonstrates how to track blocked processors with the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=BlockedProcessors
DataType=Integer Description="\"Number of Processors Blocked by the Job\""

Successfully created 1 attribute
```

14.9 Customizing the Usage Record Object

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,BlockedProcessors,Memory,Duration,SubmitTime,StartTime,EndTime
,Description
```

You could also use BlockedProcessors as the basis of a ChargeRate. SeeChapter 16: Managing Charge Rates - page 167 for how to do this.

Example 14-22: Tracking queued duration

The following example demonstrates how to track the effective duration that a job was in the idle state by adding a QueueDuration attribute to the usage record.

```
$ mam-shell Attribute Create Object=UsageRecord Name=QueueDuration DataType=Integer
Description="\"Queue Duration\""

Successfully created 1 attribute
```

If you want the new attribute to show up in *mam-list-usagerecords*, you must add it to the `usagerecord.show` string in `mam-client.conf`.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,Class,QualityOfService,M
achine,Nodes,Processors,Memory,Duration,QueueDuration,SubmitTime,StartTime,EndTime,Des
cription
```

Example 14-23: Enabling reservation statistics

The following example demonstrates how to track reservation statistics with the usage record. In this example, we will show how to track reserved processor seconds and idle processor seconds within a reservation.

```
$ mam-shell Attribute Create Object=UsageRecord Name=ReservedProcessorSeconds
DataType=Integer Description="\"Reserved Processor Seconds\""

Successfully deleted 1 attribute
```

```
$ mam-shell Attribute Create Object=UsageRecord Name=IdleProcessorSeconds
DataType=Integer Description="\"Unused Processor Seconds\""

Successfully deleted 1 attribute
```

Once you are able to track idle processor seconds, you may use the `IdleProcessorSeconds` property to charge for the unused cycles in a reservation. See Example 16-6: Charging for the unused cycles in reservations - page 171 for how to do this.

Example 14-24: Removing the UsageRecord class field

Let's say you were not interested in tracking the class. First, you would delete `Class` as an Attribute of the UsageRecord Object.

```
$ mam-shell Attribute Delete Object==UsageRecord Name==Class

Successfully deleted 1 attribute
```

Next, we need to make sure *mam-list-usagerecords* doesn't try to list the class.

```
$ vi /opt/mam/etc/mam-client.conf

usagerecord.show =
Id,Type,Instance,Charge,Stage,User,Group,Account,Organization,QualityOfService,Machine
,Nodes,Processors,Memory,Duration,SubmitTime,StartTime,EndTime,Description
```

If the attribute you want to delete is also an attribute in the Transaction table, you could delete it from there as well.

Example 14-25: Setting VM as the default usage record type

As installed, the usage record type defaults to `Job`. The default value can be set to `NULL` if there should be no default value, or to any other default value. This example will demonstrate how to set the default usage record type to `VM`.

```
$ mam-shell Attribute Modify Object=UsageRecord Name=Type DefaultValue=VM

Successfully modified 1 attribute
```

14.10 Usage Record Property Verification

If a usage record property has an object associated with it, you may want to verify that when that usage record property is specified in a scheduling action (Charge, Reserve, Quote), it verifies that that property is a valid instance of its object type. You can apply a simple verification to a usage record property by setting the property's `Values` attribute to an `@` sign followed by the name of the object.

Example 14-26: Ensure that an organization specified in a charge actually exists

```
$ mam-shell Attribute Modify Object==UsageRecord Name==Organization
Values=@Organization

Successfully modified 1 attribute
```

See Managing Attributes for more information about setting the `Values` attribute.

14.11 Usage Record Property Defaults

It is possible to set defaults for usage record properties when they are not specified in the usage data for a charge, lien, or quote. There are two cases which must be considered — when the property has an object associated with it and when the property does not.

If a property does not have an object associated with it, simply set the `DefaultValue` attribute for the property's UsageRecord Attribute object to the desired value.

Example 14-27: Setting a system-wide simple default class of batch for usage functions

```
$ mam-shell Attribute Modify Object==UsageRecord Name==Class DefaultValue=batch

Successfully modified 1 attribute
```

If a property does have an object associated with it, you will need to both set the `DefaultValue` attribute for the property's `UsageRecord` Attribute object to the desired value AND set the `DefaultValue` attribute for the corresponding object to the desired value.

Example 14-28: Setting a system-wide simple default user of anonymous for usage functions

```
$ mam-shell Attribute Modify Object==UsageRecord Name==User DefaultValue=anonymous

Successfully modified 1 attribute
```

See Global Object-Based Defaults for more information about setting default values for objects. See Local Attribute-Based Defaults for more information about setting default values for attributes.

Related Topics

- 23.7 Global Object-Based Defaults - page 228

14.12 Usage Record Property Auto-Generation

It is possible for usage record properties which have object definitions to automatically create the referenced objects the first time they are encountered in a usage function (charge, reserve or quote). To do this, the referenced object must be set to `AutoGen=True` and the `Values` attribute for the UsageRecord attribute corresponding to the object must be set to a string consisting of the `@` sign followed by the object name.

Example 14-29: Setting the Usage Record Type to auto-generate Items for usage functions

For example, let's assume there were many usage record types that could be charged for (Food, Book, Haircut) and that you had already created an Item object. It would be possible to automatically generate a new Item instance each time a new usage record type was referenced in a charge operation.

```
$ mam-shell Object Modify Name==Item AutoGen=True

Successfully modified 1 object

$ mam-shell Attribute Modify Object==UsageRecord Name==Type Values=@Item

Successfully modified 1 attribute
```

See Object Auto-Generation for more information about the auto-generation of objects.

Related Topics

14.13 Usage Record Property Instantiators

It is possible to establish a dynamic correlation between usage record properties in which one usage record property can instantiate another. For example, if a user is specified in a charge but no account is specified then the user's default account should be applied to the fund constraints and logged; or if an account is specified in a charge but not its organization then the organization corresponding to that account should be looked up and applied to the fund constraints and logged. Three usage record property instantiator types are currently supported and are configured by prefixing the property instance's Values foreign object reference with the appropriate characters: Assign if not defined (`@?=`), Assign if not different (`@!=`), Assign always (`@:=`). We shall look at each of these individually and in different terms.

Applying a correlated default (`@?=`) — If property X is specified with the value `x` in the usage record and property Y is not specified in the usage record and if the object instance referred to by `x` has a correlated default value of `y'` for property Y', then `y'` will be applied as the default value for property Y in the usage record. For example, we could establish the notion of a default account for a user.

Example 14-30: Establishing a default account for a user

First we add a `DefaultAccount` attribute (the name is arbitrary) to the User object and give it a `Values` property of `@?=Account`.

```
$ mam-shell Attribute Create Object=User Name=DefaultAccount DataType=String
Values="\"@?=Account\"" Description="\"Default Account\""

Successfully created 1 attribute
```

Then we can establish the default account for user `scottmo` to be `chemistry`.

```
User Modify Name==scottmo DefaultAccount=chemistry

Successfully modified 1 user
```

Subsequently, when a Charge, Lien, or Quote is issued that specifies the User `scottmo` but does not specify the Account, the `chemistry` Account will be applied to the charge as if originally specified in the usage record charge data.

Applying a correlated verification (`@!=`) — If property X is specified with the value `x` in the usage record and property Y is specified with the value `y` in the usage record and if the object instance referred to by `x` has a correlated verification value of `y'` for the property Y' and if `y'` does not equal `y`, then fail with an error message. Additionally, if property X is specified with the value `x` in the usage record and property Y is not specified in the usage record and if the object instance referred to by x has a correlated verification value of `y'` for property Y', then `y'` will be applied as the default value for property Y in the usage record. For example, we could establish a parent-child relationship between organizations and accounts in which explicitly specified incongruities result in a failure.

Example 14-31: Establishing an override hierarchy with accounts and organizations

First we add a `VerifyOrganization` attribute (the name is arbitrary) to the Account object and give it a `Values` property of `@!=Organization`.

```
$ mam-shell Attribute Create Object=Account Name=VerifyOrganization DataType=String
Values="\"@!=Organization\"" Description="\"Verify Organization\""

Successfully created 1 attribute
```

Then we can establish the verify organization for account `chemistry` to be `sciences`.

```
$ mam-shell Account Modify Name==chemistry VerifyOrganization=sciences

Successfully modified 1 account
```

Subsequently, when a Charge, Lien, or Quote is issued that specifies the Account `chemistry` and specifies the wrong Organization (e.g. arts), the transaction will fail with an error message. Additionally, when a Charge, Lien, or Quote is issued that specifies the Account `chemistry` but does not specify the Organization, the Organization sciences will be applied to the charge as if originally specified in the usage record charge data.

Applying a correlated override (`@:=`) — If property X is specified with the value `x` in the usage record and if the object instance referred to by `x` has a correlated override value of `y'` for property Y', then `y'` will be applied as the override value for property Y in the usage record. For example, we could establish a parent-child relationship between organizations and accounts in which explicitly specified incongruities are silently overridden with the value from the child.

Example 14-32: Establishing an override hierarchy with accounts and organizations

First we add an `OverrideOrganization` attribute (the name is arbitrary) to the Account object and give it a `Values` property of `@:=Organization`.

```
$ mam-shell Attribute Create Object=Account Name=OverrideOrganization DataType=String
Values="\"@:=Organization\"" Description="\"Override Organization\""

Successfully created 1 attribute
```

Then we can establish the override organization for account `chemistry` to be sciences.

```
$ mam-shell Account Modify Name==chemistry OverrideOrganization=sciences

Successfully modified 1 account
```

Subsequently, when a Charge, Reserve or Quote is issued that specifies the Account chemistry and specifies either the wrong Organization (e.g. arts) or no Organization, the Organization `sciences` will be silently applied to the charge as if originally specified in the usage record charge data.

Chapter 15: Managing Itemized Charges

The itemized charge table provides an ability to display the components of a composite charge in a line item format. Each charge transaction will write the components of its charge into the charge record so that you can get a line-item breakdown of each charge for usage including the names, values, rates, scaling factors, charge amounts and details listed for each component of the charge. This capability is enabled by setting `charge.itemization = true` in the `mam-server.conf` (it is `false` by default).

Itemized charges may only be queried. They are created automatically in charge transactions and there are no command line clients to change or remove them.

Additionally, an itemize option can be specified for quotes, liens, and charges to include an itemized charge breakdown in the response data instead of a single line with the amount.

In this chapter:

15.1 Querying Itemized Charges

To display itemized charge information, use the command *mam-list-itemizedcharges*.

mam-list-itemizedcharges [-j <usage_record_id>] [-J <instance_name>]
[-n <usage_property_name>] [-s <start_time>] [-e <end_time>] [--full]
[--show <attribute_name>,...] [--format csv|raw|standard] [--hours]
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version]
[--about]

Example 15-1: Listing all itemized charge information

```
$ mam-list-itemizedcharges


UsageRecord Instance Name       Value Duration Rate      ScalingFactor Amount Creatio
nTime       Description
----------- -------- ---------- ----- -------- --------- ------------- ------ -------
----------- -----------
24          job.1    Storage    100   86400    1.157e-07 1                  1 2018-
04-05 17:49:41
25          job.2    Processors 4     86400    5.787e-07 1                 20 2018-
04-05 17:49:42
25          job.2    Memory     4096  86400    1.13e-08  1                  4 2018-
04-05 17:49:42
26          job.3    Processors 1     86400    5.787e-05 1                  5 2018-
04-05 17:49:43
26          job.3    Memory     1004  86400    1.13e-08  1                  1 2018-
04-05 17:49:43
```

Related Topics

- mam-list-itemizedcharges

15.2 Displaying Itemized Charges for a Transaction

In addition to the itemized charge table, Moab Accounting Manager captures the itemized charges for usage record charges, liens, and guaranteed quotes in the details of the transaction. The itemized charges show the details for the formula used to calculate the charge for the transaction. To display the itemized charges for a scheduling transaction, parse the details from the command `mam-list-transactions --full -A Charge|Reserve|Quote`:

Example 15-2: Extract the itemized charges for a job charge

```
$ mam-list-transactions -A Charge -J PBS.1234.1 -q --show Details | perl -pe 's/.*
(ItemizedCharges[^,]*).*/\1/'
ItemizedCharges:=4 [Processors] * 5.787e-05 [ChargeRate{Processors}] * 86400
[Duration] + 4096 [Memory] * 1.13e-08
[ChargeRate{Memory}] * 86400 [Duration] = 24
```

Chapter 16: Managing Charge Rates

Charge rates establish how much to charge for usage. A charge rate consists of its name, an optional value, and the amount. Charge rates are applied when usage properties matching the charge rate names are found in the usage data. In order for a charge rate of a given name to be applied, a usage record attribute of the same name must exist. For example, a charge rate having the name Processors will be applied if Processors is defined as a Usage Record attribute and the incoming usage data for the charge request contains a property called Processors that matches the value specified in the charge rate.

In this chapter:

16.1 About Charge Rates

There are two basic types of charge rates: Name-valued charge rates and Numeric-valued charge rates.

- Name-valued charge rates charge rates are used for usage properties that take strings for values (e.g. `QualityOfService=premium` or `Account=chemistry`). The charge rate that is applied will be determined by a lookup of the usage property value to see if there is a matching charge rate value. A default rate may be specified by creating a name-valued charge rate with an empty charge rate value. Multiple values may be assigned to the same rate via separate charge rate definitions or by combining the values in a single charge rate value separated by commas.

- Numeric-valued charge rates are used for usage properties that take numbers for values (e.g. `Processors=2` or `CPUTime=12.67`). The charge rate amount that is applied will be multiplied by the usage property value. The charge rate value is commonly left blank to be taken as the default rate for the full range of usage property values. A particular value may also be specified as the charge rate value which means that that rate will only be used if the usage property value exactly matches the charge rate value. A half-bounded expression may be used by specifying a less than or greater than sign with an optional equal sign, followed by the number. For example, the charge rate value `<=4` would match a usage property value of x if $x <= 4$. A charge rate value may also be specified as a range (of the form `<number>[-<number>]`). For example, the range `1-4` would be match a usage property value of x if $1 <= x <= 4$. If you need to be more specific about the boundedness of the ranges, you may replace the dash with a less than sign with an optional equal sign on either side of it to indicate whether the endpoints are included. For example, the range `1<4` would match if $1 < x < 4$, `1=<4` would match if $1 <= x < 4$, `1<=4` would match if $1 < x <=4$ and `1=<=4` would match if $1 <= x <= 4$. So you might use ranges like `1=<2`, `2=<4`, `4=<8`, and `>=8`. Multiple values or value ranges having the same charge rate may be specified in a single expression separated by commas.

A charge rate amount may have an operation modifier which dictates the way the rate is factored into the charge calculation. For example, consumption-based charge rates or usage fees will often be additive in nature while quality-based charge rates may be multiplicative. The additive charge rates may be further distinguished by whether they should be added before or after the multiplicative charge rates are applied. The charge formula can be represented in the following form: (Σ(Pre-Additive Rates)*Π(Multiplicative Rates))+Σ(Post-Additive Rates). Thus, there are three operation modifiers: Pre-Additive, Multiplicative and Post-Additive.

Composite type	Description
Pre-Additive	Pre-additive modifiers are applied to charge rates that should be added together before any charge multipliers are applied. A pre-additive modifier is specified by prepending a plus sign '+' to the charge rate amount. Since pre-additive is the most commonly specified operation modifier, a charge rate amount without an operation modifier will be assumed to be pre-additive by default.

Composite type	Description
Multiplicative	Multiplicative modifiers are applied to charge rates that should be multiplied together with other multiplicative charge rates and with the sum of the pre-additive charge rates. A multiplicative modifier is specified by prepending an asterisk '*' to the charge rate amount.
Post-Additive	Post-additive modifiers are applied to charge rates that should be added together after any charge multipliers are applied. A post-additive modifier is specified by appending a plus sign '+' to the charge rate amount.

A pre-additive charge rate may have a time-based modifier which dictates that the charge should be multiplied by the amount of time the feature was used. For example, it is common for the processor charge to be multiplied by the the amount of time the processors were used. A time-based modifier is specified by appending a forward slash '/' to the charge rate amount, followed by one of the following time designators: s (per-second), m (per-minute), h (per-hour), d (per-day), W (per-week), M (per-month), Y (per-year). As an example, a per-hour time-based modifier is specified by appending '/h' to the charge rate amount and will cause a charge to be multiplied by the number of hours the feature was used. Technically, a rate with a time-based modifier will be multiplied by the number of seconds the feature was used, then divided by the number of seconds corresponding to the time designator (e.g. 3600), and will ultimately be rounded to the number of decimal places in the currency precision.

A pre-additive charge rate may have a divisor modifier which dictates that the charge should be divided by the specified integer. A divisor modifier is specified by appending a forward slash '/' to the charge rate amount, followed by an integer number. A divisor modifier can be used in lieu of expressing a small decimal fraction charge rate such as when converting a value from MegaBytes to GigaBytes. If a divisor modifier is used in conjunction with a time-based modifier, the divisor modifier must precede the time-based modifier.

A charge rate may have one or more conditions which dictates additional qualifications that must be met in order for the charge rate to be applied. A condition is specified by prepending `<propertyName>=<propertyValue>` followed by a question mark '?' to the value field of the charge rate. If you want Processors to apply a special charge rate (e.g. .5/s) for user amy, the charge rate value should consist of the string `"User=amy?"`. Additionally, you may combine charge rate conditions with either a pipe symbol '|' for or, or an ampersand symbol '&' for and. For example, `User=amy|User=dave?` or `User=amy&Project=chemistry?`. You may not combine ands and ors in the same charge rate value.

16.2 Creating Charge Rates

To create a new charge rate, use the command *mam-create-chargerate*.

mam-create-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_ value>] {-z <charge_rate_amount>} [-d <description>] [--debug] [-- site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

> ⚠️ You must first create the usage record property before you can charge by it. See Customizing the Usage Record Object for instructions on how to create a usage record property.

Example 16-1: Charging for requested memory

For consumable resources, we use a time-based modifier (e.g. '/s') to multiply the memory by the duration the resource was used (in this case, seconds). We also divide the result by 1024 since Moab reports memory in MegaBytes but we want to charge for GigaBytes.

```
$ mam-create-chargerate -n Memory -z 1/1024/s -d "1 credit per requested GigaByte of
memory per second.

Successfully created 1 charge rate
```

Example 16-2: Charging for GPUs (and/or MICs)

If you intend to have the accelerator charge multiplied by the amount of time that was used, use the appropriate time modifier. Use the name GPUs if charging for GPUs or use the name MICs if charging by MICs. Alternately, you may create separate charge rates for each if both are present in your system.

```
$ mam-create-chargerate -n GPUs -z 1/s -d "1 credit per GPU-second"

Successfully created 1 charge rate
```

Example 16-3: Charging for CPU time

Since CPU time already incorporates the element of time in its value, we do not need to include a time-based modifier in the charge rate.

```
$ mam-create-chargerate -n CPUTime -z 1 -d "1 credit per utilized cpu-second"

Successfully created 1 charge rate
```

Example 16-4: Charging for blocked processors (jobs only)

It is possible to charge for blocked processors rather than allocated processors. For example, all of the processors in an entire node may be blocked by a job using a node-exclusivity policy (e.g. a node access policy of "single-job") even though a lesser number of processors were actually requested and allocated to the job.

```
$ mam-create-chargerate -n BlockedProcessors -x Type=Job? -z 1/s -d "1 credit per
```

```
blocked processor second"

Successfully created 1 charge rate
```

Example 16-5: Charging for processor equivalents

Some sites may wish to charge for processor equivalents rather than allocated processors. Processor equivalents scale the allocated processors by the most constrained consumable resource (e.g. memory or CPU).

```
$ mam-create-chargerate -n ProcessorEquivalents -z 1/s -d "1 credit per processor
equivalent per second"

Successfully created 1 charge rate
```

Example 16-6: Charging for the unused cycles in reservations

If your resource manager supports it, and if configured to do so, you may charge for the unused cycles in administrative or standing reservations. If using Moab Workload Manager, you must first enable reservation charging. See Reservation Policies in the *Moab Workload Manager Administrator Guide* for how to do this.

It will also be necessary to add the required reservation statistics to the usage record object (e.g. IdleProcessorSeconds and ReservedProcessorSeconds). See Example 14-23: Enabling reservation statistics - page 155e for how to do this.

After adding the necessary usage record attributes, you must create a charge rate that charges for the unused cycles in the reservation. The following charge rate will charge for processor seconds that were not blocked by jobs running within the reservation.

```
$ mam-create-chargerate -n IdleProcessorSeconds -x 'Type=Reservation?' -z 1 -d "1
credit per unused processor second in reservations"

Successfully created 1 charge rate
```

If also charging for jobs, it is recommended that you charge jobs for the blocked processors with a condition of 'Type=Job?' since this is the best counterpart to the IdleProcessorSeconds metric which charges for unblocked processors. See Example 16-4: Charging for blocked processors (jobs only) - page 170 for how to do this.

Example 16-7: Creating a name-valued pre-additive charge rate

```
$ mam-create-chargerate -n License -x matlab -z 5

Successfully created 1 charge rate
```

Example 16-8: Creating a numeric-valued multiplicative charge rate

```
$ mam-create-chargerate -n Discount -z *1

Successfully created 1 charge rate
```

Example 16-9: Charging for quality of service

We want to multiply the resource charge by a value that depends on the quality of service applied to the job. Thus we must create a set of name-valued multiplicative charge rates with a default value.

```
$ mam-create-chargerate -n QualityOfService -x Premium -z *2

Successfully created 1 charge rate
```

```
$ mam-create-chargerate -n QualityOfService -J BottomFeeder -z *0.5

Successfully created 1 charge rate
```

```
$ mam-create-chargerate -n QualityOfService -z *1

Successfully created 1 charge rate
```

Example 16-10: Charging for licenses

```
$ mam-create-chargerate -n Licenses -x Matlab -z +20

Successfully created 1 charge rate
```

Example 16-11: Charging for generic resources

```
$ mam-create-chargerate -n Resources -x graphics -z 5

Successfully created 1 charge rate
```

Example 16-12: Charging for job variables

```
$ mam-create-chargerate -n Variables -x foo:bar -z 10

Successfully created 1 charge rate
```

Example 16-13: Creating a numeric-valued post-additive charge rate

```
$ mam-create-chargerate -n Shipping -z 25+

Successfully created 1 charge rate
```

Example 16-14: Creating a name-valued post-additive charge rate

```
$ mam-create-chargerate -n Zone -x Asia -z 200+

Successfully created 1 charge rate
```

Example 16-15: Creating a couple of conditional numeric-valued pre-additive charge rates

```
$ mam-create-chargerate -n Disk -x User=dave? -z 0.2/s

Successfully created 1 charge rate
```

```
$ mam-create-chargerate -n Disk -x User=mike? -z 0.5/s

Successfully created 1 charge rate
```

Example 16-16: Creating some numeric-valued pre-additive charge rate ranges and a default

```
$ $ mam-create-chargerate -n Processors -x 1-4 -z 2/s

Successfully created 1 charge rate
```

```
$ mam-create-chargerate -n Processors -x 5-8 -z 1.5/s

Successfully created 1 charge rate
```

```
$ mam-create-chargerate -n Processors -z 1/s

Successfully created 1 charge rate
```

Example 16-17: Creating some numeric-valued pre-additive rate ranges for floating point values (without time-based modifiers)

```
$ $ mam-create-chargerate -n Power -x '<2' -z 0.005

Successfully created 1 charge rate
```

```
$ $ mam-create-chargerate  -n Power -x '2=<4' -z 0.004

Successfully created 1 charge rate
```

```
$ $ mam-create-chargerate -n Power -x '>=4' -z 0.003

Successfully created 1 charge rate
```

Example 16-18: Assigning multiple classes to run for free

```
$ $ mam-create-chargerate -n Class -x dev,test -z *0

Successfully created 1 charge rate
```

Related Topics

- mam-create-chargerate

16.3 Querying Charge Rates

To display charge rate information, use the command *mam-list-chargerates*.

mam-list-chargerates [[-n] <charge_rate_name>] [-x <charge_rate_value>] [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 16-19: Listing all charge rates

```
$ mam-list-chargerates

Name               Value          Amount  Description
-----------------  ----------     -------  ------------
Class              dev,test       *0
CPUTime                           1
Discount                          *1
Disk               User=dave?     0.2/s
Disk               User=mike?     0.5/s
License            Matlab         5/s
Memory                            1/1024/s
Power              <2             0.005
Power              2=<4           0.004
Power              >=4            0.003
Processors                        1/s
Processors         1-4            2/s
Processors         5-8            1.5/s
QualityOfService                  *1
QualityOfService   BottomFeeder   *0.5
QualityOfService   Premium        *2
Shipping                          25+
Zone               Asia           200+
```

Related Topics

- mam-list-chargerates

16.4 Modifying Charge Rates

To modify a charge rate, use the command *mam-modify-chargerate*.

mam-modify-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_value>] [-z <charge_rate_amount>] [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 16-20: Changing a charge rate

```
$ mam-modify-chargerate -n License -x Matlab -z 4/s

Successfully modified 1 charge rate
```

Related Topics

- mam-modify-chargerate

16.5 Deleting Charge Rates

To delete a charge rate, use the command *mam-delete-chargerate*.

mam-delete-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_value>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 16-21: Deleting a charge rate

```
$ mam-delete-chargerate -n Memory

Successfully deleted 1 charge rate
```

Related Topics

- mam-delete-chargerate

Chapter 17: Managing Transactions

Moab Accounting Manager logs all modifying transactions in a detailed transaction journal (queries are not recorded). Previous transactions can be queried but not modified or deleted. By default, a standard user may only query transactions performed by them.

17.1 Querying Transactions

To display transaction information, use the command *mam-list-transactions*.

mam-list-transactions [[-T] <transaction_id>] [-R <request_id>] [-O <object>] [-A <action>] [-k <primary_key_value>] [-U <actor>] [-f <fund_id>] [-i <allocation_id>] [-u <user_name>] [-a <account_name>] [-m <machine_name>] [-j <usage_record_id>] [-J <instance_name>] [-s <start_time>] [-e <end_time>] [-X, --extension <property>=<value>]... [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 17-1: List all deposits made in 2018

```
$ mam-list-transactions -A Deposit -s 2018-01-01 -e 2019-01-01
```

Example 17-2: List refund totals broken down by fund

```
$ mam-list-transactions -A Refund --show "Sum(Amount),GroupBy(Fund)"
```

Example 17-3: List usage and charge totals broken down by account and user

```
$ mam-list-transactions -A Charge --show "GroupBy(Account),GroupBy(User),Sum
(ProcHours),Sum(Amount)=Charged"
```

Example 17-4: List every transaction performed by amy *since the beginning of 2018*

```
$ mam-list-transactions -U amy -s 2018-01-01
```

Example 17-5: List all transactions related to job moab.1

```
$ mam-list-transactions -J moab.1
```

Example 17-6: List all transactions affecting charge rates

```
$ mam-list-transactions -O ChargeRate
```

Related Topics

- mam-list-transactions

17.2 Customizing the Transaction Object

The transaction record as natively defined can be customized with the attributes you want to track in your use cases. It is possible to add additional attributes to the Transaction table so that it will be automatically populated from actions having assignments, conditions, options and data values referring to the attribute.

Transaction discriminators are those properties which are considered primary differentiators between transaction records (besides the metadata differentiators of object, action, and instance). Transaction discriminators are used in the dynamic web portal as filters for the listing of transaction records. Any new attributes added to the Transaction object will become transaction discriminators. Removing a discriminator attribute from the transaction object will necessarily remove it as a transaction discriminator as well. It will be necessary to log out and back in after adding or removing a discriminator in order for it to be reflected in the web GUI.

Example 17-7: Adding an Organization field to the Transaction record (which also makes it a discriminator)

```
$ mam-shell Attribute Create Object=Transaction Name=Organization DataType=String

Successfully created 1 attribute
```

Chapter 18: Managing Events

Moab Accounting Manager has an internal event scheduler that can be configured to execute Moab Accounting Manager actions at a designated time in the future or on a periodic basis. Valid actions on an event include Create, Query, Fire, Modify, Refresh and Delete. Event attributes include Id, FireCommand, ArmTime, FireTime, RearmPeriod, EndTime, Notify, RearmOnFailure, FailureCommand, CatchUp and Description.

18.1 About Events

There are two server configuration parameters which affect event scheduling: `event.scheduler` which specifies whether the event scheduler is enabled or not (it is disabled by default) and `event.pollinterval` which is the period in minutes that the event scheduler uses to fire events. The poll interval must divide evenly into the number of minutes in a day (1440).

> ℹ In order for events to fire, you must set *`event.scheduler = true`* in mam-server.conf and restart the MAM Server.

The command(s) to be fired by an event are expressed in a serialized form of the request identical to the syntax used in the interactive control program (mam-shell). There are two commands that can be configured in an event: the *FireCommand* which is the command to be executed when the event is fired, and the *FailureCommand* which is the command to be executed if the fired command results in an unsuccessful response status. The `FireTime` is the target time for the event to be triggered by the event scheduler. The actual fire time may be dependent on the state of the server and will be recorded in the `CreationTime` property of the corresponding "Event Fire" Transaction. An event may also be fired manually with the Event Fire action.

The `RearmPeriod` is a time period expression specifying when the event will be rearmed. This period expression is of the form: "`<period>[[@<instant>][~|^]|!]`". The time period is expressed as an integer number followed by a designator of minute(s), hour(s), day(s), week(s), month(s), or years(s). For example the period might be 1 day, 2 hours, or 5 minutes. The optional Instant locks the period to a specific instant within the time period such as `1 day @ hour 12` or `1 month @ day 3`. The modifiers indicate whether the time period should be relative to now (!), or relative to the start of this (~) designator (month or minute, etc.), or relative to the start of the first (^) designator (month or minute, etc.). For example, assuming the FireTime was 7:15, if you specified `4 hours !` as the rearm period it would be rearmed at 11:15, if you specified `4 hours ~` as the rearm period it would be rearmed at 11:00, and if you specified `4 hours ^` as the rearm period it would be rearmed at 8:00.

The `ArmTime` is the time the event was last armed or fired. This field is used as a reference time to be able to derive how long the event has been waiting to happen. This field will be initially set to mark the moment the first FireTime is set and updated thereafter to indicate the last time the event was fired. In the case where an event does not have a FireTime set, this field may be set manually and used in a similar manner. If we consider the time between event firings as "laps", this could be thought of as the Lap Start Time. If the `RearmOnFailure` boolean is set to `False`, the event will not be rearmed if the command was unsuccessful. If set to `True`, the event will be evaluated for rearming even if the command response has a status of `Failure`. The standard default is `False`. If the `CatchUp` boolean is set to `True` and the server was down during the time this event should have fired, the event scheduler will attempt to make up for the past due events by progressively firing them (rearming based on previous arm time) until catching up to the present. The actions will still show as having occurred in the present rather than in the past. If set to `False`, and the server is brought back up after an outage, the event scheduler will still fire immediately for a past due event, but it will only fire once and then rearm relative to the current time.

A Notification method can be specified via the `Notify` parameter and is of the form: `[+-=]` `[<delivery_method>:] [<recipient>][,[+-=][<delivery_method>:]`

`[<recipient>]]`*. If the term is a –, the notification is sent only on failure. If the term is a +, the notification is sent only on success. Otherwise the notification is always sent. There can be multiple notify expressions separated by a comma. All applicable notifications will be sent. See the chapter on Managing Notifications for more information about delivery method and recipient.

18.2 Creating Events

To create a new event, use the command *mam-create-event*.

mam-create-event [--fire-command <fire_command>] [-s <fire_time>] [-e <end_time>] [--rearm-period <rearm_period>] [--rearm-on-failure <boolean>] [--failure-commmand <failure_command>] [--notify <notification_url>] [--catch-up <boolean>] [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 18-1: Creating an automatic allocation renewal event

```
$ mam-create-event --fire-command "Fund Reset" -s "2019-01-01" --rearm-period "3
months^"

Successfully created 1 event
```

> 🛈 In order for events to fire, you must set *event.scheduler = true* in mam-server.conf and restart the MAM Server.

Related Topics

- mam-create-event

18.3 Querying Events

To display event information, use the command *mam-list-events*.

mam-list-events [[-E] <event_id>] [-s <start_time>] [-e <end_time>]
[--full] [--show <attribute_name>,...] [--format <csv|raw|standard>]
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version]
[--about]

Example 18-2: Listing all events

```
$ mam-list-events

Id FireCommand FireTime    ArmTime             RearmPeriod EndTime Notify
RearmOnFailure FailureCommand CatchUp CreationTime        Description
-- ----------- ---------- ------------------- ----------- ------- ------ -------------
- -------------- ------- ------------------- -----------
1  Fund Reset  2019-01-01 2018-11-09 10:31:28 3 months^                  False
               True    2018-11-09 10:31:28
```

Related Topics

- mam-list-events

18.4 Modifying Events

To modify an event, use the command `mam-modify-event`.

`mam-modify-event` {[-E] <event_id>} [--fire-command <fire_command>] [-s <fire_time>] [-e <end_time>] [--rearm-period <rearm_period>] [--rearm-on-failure True|(False)] [--failure-command <failure_command>] [--notify <notification_url>] [--catch-up (True)|False] [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 18-3: Changing an events's rearm period to be monthly

```
$ mam-modify-event --rearm-period "1 month" 1

Successfully modified 1 event
```

Related Topics

- mam-modify-event

18.5 Deleting Events

To delete an event, use the command `mam-delete-event`.

`mam-delete-event` {[-E] <event_id>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 18-4: Deleting an event

```
$ mam-delete-event 1

Successfully deleted 1 event
```

Related Topics

- mam-delete-event

Chapter 19: Managing Notifications

When event commands are executed (asynchronously), the success or failure of the operation is communicated back to the initiator via a notification. When an event is created, you may specify the Notify option which will associate a notification method with the event. Currently there is only one `DeliveryMethod` implemented which is `Store`. With the Store delivery method, command response information is stored as instances of the Notification object. These messages can later be retrieved by the initiator via a Notification Query. Payments can also route a notification method down to their associated events via a Notify option.

The notification attributes include `Id` (auto-generated), `Type`, `Event`, `Status`, `Code`, `Message`, `Key`, `Recipient`, `EndTime` and `CreationTime`. Stored notifications can be queried on any of these conditions. The notification type distinguishes what type of command resulted in the notification (Fire or Failure). The notification key is the value of the primary key of the object instance that the command acted on (e.g. the Payment Id). The recipient could be a user name or any tag that identifies the intended reader for the notification. The Notification Query supports a `Delete` option, which if set to `True`, will delete the notifications after they have been queried. Additionally, stored notifications have an EndTime after which they are automatically deleted by MAM. The Notification actions include Send, Refresh, Create, Query, Delete and Modify.

There are two server configuration parameters which affect notifications: `notification.deliverymethod` which dictates which deliverymethod is used by default if unspecified and `notification.duration` which defines how long notifications stick around if the Store delivery method is used.

In this chapter:

19.1 Querying Notifications

To display notification information, use the command *mam-list-notifications*.

mam-list-notifications [[-N] <notification_id>] [-E <event_id>] [-T <notification_type>] [-k <primary_key_value>] [-u <recipient>] [-x <status>] [-s <start_time>] [-e <end_time>] [--delete] [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 19-1: Listing all failure notifications

```
$ mam-list-notifications -x Failure

Id Event Type Status  Code Message

                                                        Key Recipient EndTim
e           CreationTime
-- ----- ---- ------- ---- ----------------------------------------------------
-------------------------------------------------------------------------------
----------------------------------------------------------- --- --------- ------
------------- ------------------

4  20    Fire Failure 782  Payment Begin failed starting payment: Failed creating paym
ent starting lien: Insufficient balance to reserve usage
(Instance Moab.1)\nClearing the event fire time.\nThe controlling event has been delet
ed.       9    amy       2018-04-23 13:35:01 2018-04-09 13:35:01
```

Related Topics

- mam-list-notifications

19.2 Deleting Notifications

To delete a notification, use the command `mam-delete-notification`.

`mam-delete-notification` {[`-N`] notification_id} [`--debug`] [`--site` <site_name>] [`--help`] [`--man`] [`--quiet`] [`--verbose`] [`--version`] [`--about`]

Example 19-2: Deleting a notification

```
$ mam-delete-notification 4

Successfully deleted 1 notification
```

Example 19-3: Deleting all successful notifications

To delete many notifications, query them with the `--delete` option:

```
$ mam-list-notifications -x Success --delete

Id Event Type Status  Code Message


                                                                    Key Re
cipient EndTime          CreationTime
-- ----- ---- ------- ---- -----------------------------------------------------
-----------------------------------------------------------------------------
-----------------------------------------------------------------------------
----------------------------------------------------------------- --- --
------- ------------------ ------------------
4  20    Fire Failure 782  Payment Begin failed starting payment: Failed creating paym
ent starting lien: Insufficient balance to reserve usage
(Instance Moab.1)\ nClearing the event fire time.\ nThe controlling event has been del
eted.
        9   amy        2018-04-23 13:35:01 2018-04-09 13:35:01

1  11    Fire Success 000  Payment Begin: Successfully charged 10 credits for instance
 Moab.1\ nSuccessfully charged 20 credits for instance Moab.2\ nSuccessfully charged 2
0 credits for instance Moab.3\ nSuccessfully started payment
(6) and created 3 liens\ nClearing the event fire time.\ nThe controlling event has be
en deleted.      6   scottmo   2018-04-23 13:28:02 2018-04-09 13:28:02

2  14    Fire Success 000  Payment Begin: Successfully charged 10 credits for instance
 Moab.1\ nSuccessfully charged 20 credits for instance Moab.2\ nSuccessfully charged 2
0 credits for instance Moab.3\ nSuccessfully started payment
(7) and created 3 liens\ nClearing the event fire time.\ nThe controlling event has be
en deleted.      7   amy       2018-04-23 13:31:02 2018-04-09 13:31:02

3  17    Fire Success 000  Payment Begin: Successfully charged 10 credits for instance
 Moab.1\ nSuccessfully charged 20 credits for instance Moab.2\ nSuccessfully charged 2
0 credits for instance Moab.3\ nSuccessfully started payment
(8) and created 3 liens\ nClearing the event fire time.\ nThe controlling event has be
en deleted.      8   amy       2018-04-23 13:32:02 2018-04-09 13:32:02
Successfully deleted 3 notifications
```

Related Topics

- [mam-delete-notification](#)

Chapter 20: Managing Roles

Moab Accounting Manager uses instance-level role-based access controls to determine what users can perform what functions. Named roles are created, actions are associated with the roles, and users are assigned to these roles.

The actions for a role consist of a set of tuples of object, action and instance permitted by the role. In other words, each role action defines an object (whether specific or `ANY`), the action that can be taken on that object (whether specific or `ANY`) and the instance of the object that action can be taken on (whether specific or `ANY`).

In the base configuration, there are three default roles: `SystemAdmin`, `Anonymous` and `OVERRIDE`. Other configurations, such as the bank configuration, add additional roles. Roles can be added as desired. The three base roles are required for proper function of Moab Accounting Manager and should not be deleted. By default, the `SystemAdmin` role can perform any action on any object. This role is usually assigned to the super user. The `Anonymous` role is intended to define the actions available to your standard unprivileged user. This may include the ability to set your password, query certain public objects and modify objects that belong to you (implemented via the `OVERRIDE` role). The `OVERRIDE` role is a special role type that defines those actions that should use special business logic intrinsic to the routine that handles that object and action. For example, in the bank configuration, the `OVERRIDE` logic for the Account Query routine will only allow the standard user to see information about accounts for which he or she is a member. A given user's privileges will be the superset of the actions of all roles that apply to that user.

The instance indicates which specific instances of the object the action can be performed on. There are several special instance types that can be used in certain situations. The `ANY` instance is supported by all objects and permits the specified action on all instances of the specified object. The `SELF` instance applies to the user's own instance if the object is User, or to objects that have a User attribute associated with the user. The `MEMBERS` instance applies to objects for which the user is a direct member. The `ADMIN` instance applies to objects for which the user is designated as an administrator. Unless otherwise specified, the instance will default to a value of `ANY`.

20.1 Creating Roles

To create a new role, use the command *mam-create-role*. Users and actions may be associated with the role at creation time. When assigning actions to a role, the object, action and instance must be specified in the form shown. Multiple actions or users may be specified for the role.

mam-create-role {[-r] <role_name>} [-d <description>] [-u <user_name>,...]... [-A "<object_name>-><action_name>[{<instance_name>}]",...]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 20-1: Creating a Manager role

```
$ mam-create-role -r Manager -d "Manages Roles and Responsibilities"

Successfully created 1 role
```

Related Topics

- mam-create-role

20.2 Querying Roles

To display the role information, use the command *mam-list-roles*.

mam-list-roles [[-r] <role_name>] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

Example 20-2: Listing all roles along with users and descriptions

```
$ mam-list-roles --show=Name,Users,Description

Name         Users     Description
-----------  -----     -------------------------------------------------------
------
AccountAdmin           Can update or view an  account they are admin for
Anonymous    ANY       Things that can be done by anybody
OVERRIDE     ANY       A custom authorization method will be invoked
Schedule     root      Scheduler relevant Transactions
SystemAdmin  scottmo   Can update or view any object
UserServices           User Services
```

Example 20-3: Listing information about the scheduler role

```
$ mam-list-roles --long Scheduler

Name         Users  Actions                   Description
-----------  -----  ------------------------  ----------------------------------------
-------------
Scheduler    root   UsageRecord->Create(ANY)  Scheduler relevant Transactions
                    UsageRecord->Quote(ANY)
                    UsageRecord->Reserve(ANY)
                    UsageRecord->Charge(ANY)
                    Lien->Delete(ANY)
```

Related Topics

- mam-list-roles

20.3 Modifying Roles

To modify a role, use the command *mam-modify-role*.

mam-modify-role {[-r] <role_name>} [-d <description>] [--add-user(s) <user_name>,...]... [--add-action(s) "<object_name>-><action_name> [{<instance_name>}]",...]... [--del-user(s) <user_name>,...]... [--del-action(s) "<object_name>-><action_name>[{<instance_ name>}]",...]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Users may be added to a role or removed from a role. Actions also may be added to a role or removed from a role. When specifying actions, the instance will default to a value of ANY.

Example 20-4: Adding a user to a role

Let's add dave to our new Manager role:

```
$ mam-modify-role --add-user dave -r Manager

Successfully added 1 user
```

Example 20-5: Associating an action with a role

Allow the Manager to change role responsibilities:

```
$ mam-modify-role --add-action "RoleAction->ANY" Manager -v

Successfully added 1 action
```

Related Topics

- mam-modify-role

20.4 Deleting Roles

To delete a role, use the command *mam-delete-role*.

mam-delete-role {[-r] <role_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Users may be added to a role or removed from a role. Actions also may be added to a role or removed from a role. When specifying actions, the instance will default to a value of ANY.

Example 20-6: Deleting the Manager role

Let's add dave to our new Manager role:

```
$ mam-delete-role Manager

Successfully deleted 1 role and 2 associations
```

Related Topics

- mam-delete-role

Chapter 21: Managing Passwords

Passwords must be established for each user who wishes to use the web-based GUI. Passwords must be at least eight characters and are stored in encrypted form. A `mam-set-password` command line client exists to aid a user or administrator in setting or changing a password. Other operations (deleting or listing password entries) must be performed using the interactive control program (mam-shell). By default, a standard user may only set or change his or her own password. A system administrator may set or change any user's password.

> ⚠️ Because Moab Accounting Manager caches password information for faster responsiveness, it will be necessary to restart the server after running mam-set-password for the GUI to accept that password change.

21.1 Setting Passwords

To set a new password, use the command *mam-set-password*. If the user name is not specified via an option or as the unique argument, then the invoking user will be taken as the user whose password will be set. The invoker will be prompted for the new password.

mam-set-password [[-u] <user_name>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

Example 21-1: Setting a password

```
$ mam-set-password amy

Enter your new password:
Successfully created 1 password
```

Related Topics

- mam-set-password

21.2 Querying Passwords

To display password information, use the command *mam-shell Password Query*:

mam-shell Password Query [Show:=<"Field1,Field2,...">] [User==<User Name>] [ShowUsage:=True]

Example 21-2: List the users who have set passwords

```
$ mam-shell Password Query Show:=User

User
------
amy
mam
```

21.3 Deleting Passwords

To delete a password, use the command *mam-shell Password Delete*:

mam-shell Password Delete User==<User Name>]

> ⓘ The mam-shell control program allows you to make powerful and sweeping modifications to Moab Accounting Manager objects. Misuse of this command could result in the inadvertent deletion of all passwords.

Example 21-3: Deleting a password

```
$ mam-shell Password Delete User==amy

User    Password
------  ----------------------------------------
amy     HZYzwD20o1XIE/gxRYyFKP2sumkCluHm
Successfully deleted 1 password
```

Chapter 22: Using the MAM Shell (mam-shell)

`mam-shell` is an interactive control program that can access all of the advanced functionality in Moab Accounting Manager.

> ⓘ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. Inadvertent mistakes could result in modifications that are very difficult to reverse.

In this chapter:

22.1 Usage

mam-shell commands can be invoked directly from the command line as arguments, or read from stdin (interactively or redirected from a file).

mam-shell [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about] [<command>]

Example 22-1: Specifying the command as direct arguments

```
$ mam-shell System Query

Name                    Version  Description
---------------------   -------  ------------------
Moab Accounting Manager  9.1.3.0  Commercial Release
```

Example 22-2: Using the interactive prompt

```
$ mam-shell

mam> System Query

Name                    Version Description
---------------------   ------- -----------------
Moab Accounting Manager  9.1.3.0 Commercial Release
mam> quit
```

Example 22-3: Reading commands from a file

```
$ cat >commands.mam <<EOF
System Query
quit
EOF
$ mam-shell <commands.mam

Name                    Version  Description
---------------------   -------  ------------------
Moab Accounting Manager  9.1.3.0  Commercial Release
```

Related Topics

- mam-shell

22.2 Command Syntax

`mam-shell` commands are of the form:
```
<Object> [=<Alias>] [,<Object> [=<Alias>]...] <Action> [
[<Conjunction>] [<Open_Parenthesis>...] [<Object>.] <Name> <Operator>
[<Subject>.] <Value> [<Close_Parenthesis>...] ...]
```

The basic form of a command is `<Object> <Action> [<Name><Operator><Value>]*`.
When an action is performed on more than one object, such as in a multi-object query, the objects
are specified in a comma-separated list. Commands may accept zero or more predicates which may
function as fields to return, conditions, update values, processing options, etc. Predicates, in their
simplest form, are expressed as Name, Operator, Value tuples. Predicates may be combined via con-
junctions with grouping specified with parentheses. When performing multi-object queries, names
and values may need to be associated with their respective objects.

Valid conjunctions include:

Conjunction	Meaning
&&	and
||	or
&!	and not
|!	or not

Open parentheses may be any number of literal open parentheses '('.

Name is the name of the condition, assignment, or option. When performing a multi-object query,
an attribute name may need to be prepended by its associated object separated by a period
(`<object>.<attribute>`). When specifying a partial condition, the name will consist of the
attribute followed by the part enclosed in curly braces (`<attribute>{<part>}`).

Valid operators include:

Operator	Meaning
==	equals
<	less than
>	greater than
<=	less than or equal to

Operator	Meaning
>=	greater than or equal to
!=	not equal to
~	matches
=	is assigned
+=	is incremented by
-=	is decremented by
:=	option
:!	not option

Value is the value of the selection list, condition, assignment, or option. When performing a multi-object query, a value may need to be prepended by its associated object (called the subject) separated by a period.

Close parentheses may be any number of literal closing parentheses ')'.

22.3 Valid Objects

To list the objects available for use with commands in `mam-shell` commands, use the mam-shell command: *Object Query*

Example 22-4: Listing all objects

```
mam> Object Query Show:="Sort(Name)"

Name
---------------------------
Account
AccountUser
Action
Allocation
Attribute
ChargeRate
Constrainst
Fund
FundFund
Lien
LienAllocation
Object
Organization
Password
Quote
QuoteChargeRate
Role
RoleAction
RoleUser
System
Transaction
UsageRecord
User
```

22.4 Valid Actions for an Object

To list the actions that can be performed on an object, use the *mam-shell* command: *Action Query*

Example 22-5: Listing all actions associated with the Fund object

```
mam> Action Query Object==Fund Show:="Sort(Name)"

Name
------------
Create
Delete
Deposit
Modify
Query
Transfer
Undelete
Withdraw
```

22.5 Valid Predicates for an Object and Action

By appending the option `ShowUsage:=True` to a command, the syntax of the command is returned, expressed in SSSRMAP XML Message Format.

Example 22-6: Show the usage for Allocation Query

```
mam> Allocation Query ShowUsage:=True

<Request action="Query">
    <Object>Allocation<Object>
    [<Get name="Id" [op="Sort|Tros|Count|GroupBy|Max|Min"]></Get>]
    [<Get name="Fund" [op="Sort|Tros|Count|GroupBy|Max|Min"]></Get>]
    [<Get name="StartTime" [op="Sort|Tros|Count|GroupBy|Max|Min"]></Get>]
    [<Get name="EndTime" [op="Sort|Tros|Count|GroupBy|Max|Min"]></Get>]
    [<Get name="Amount" [op="Sort|Tros|Count|GroupBy|Max|Min|Sum|Average"]></Get>]
    [<Get name="CreditLimit"
[op="Sort|Tros|Count|GroupBy|Max|Min|Sum|Average"]></Get>]
    [<Get name="InitialDeposit"
[op="Sort|Tros|Count|GroupBy|Max|Min|Sum|Average"]></Get>]
    [<Get name="Allocated" [op="Sort|Tros|Count|GroupBy|Max|Min|Sum|Average"]></Get>]
    [<Get name="Active" [op="Sort|Tros|Count|GroupBy"]></Get>]
    [<Get name="Description" [op="Sort|Tros|Count|GroupBy|Max|Min"]></Get>]
    [<Where name="Id" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>Integer Number}</Where>]
    [<Where name="Fund" [op="EQ|NE|GT|GE|LT|LE|Match|NotMatch (EQ)"] [conj="And|Or
(And)"] [group="<Integer Number>Fund Name}</Where>]
    [<Where name="StartTime" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>YYYY-MM-DD[ hh:mm:ss]|-infinity|infinity|now</Where>]
    [<Where name="EndTime" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>YYYY-MM-DD[ hh:mm:ss]|-infinity|infinity|now</Where>]
    [<Where name="Amount" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>Decimal Number}</Where>]
    [<Where name="CreditLimit" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>Decimal Number}</Where>]
    [<Where name="InitialDeposit" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>Decimal Number}</Where>]
    [<Where name="Allocated" [op="EQ|NE|GT|GE|LT|LE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>Decimal Number}</Where>]
    [<Where name="Active" [op="EQ|NE (EQ)"] [conj="And|Or (And)"]
[group="<Integer Number>True|False</Where>]
    [<Where name="Description" [op="EQ|NE|GT|GE|LT|LE|Match|NotMatch (EQ)"]
[conj="And|Or (And)"] [group="<Integer Number>Description}</Where>]
    [<Option name="Filter">True|False (False)</Option>]
    [<Option name="FilterType">Exclusive|NonExclusive (NonExclusive)</Option>]
    [<Option name="IncludeAncestors">True|False (False)</Option>]
    [<Option name="Time">YYYY-MM-DD[ hh:mm:ss]</Option>]
    [<Option name="Unique">True|False (False)</Option>]
    [<Option name="ChuckSize">{Integer Number}</Option>]
    [<Option name="Limit">{Integer Number}</Option>]
    [<Option name="Offset">Integer Number}</Option>]
    [<Option name="ShowHidden">True|False (False)</Option>]
    [<Option name="ShowUsage">True|False (False)</Option>]
<Request>
```

22.6 Common Options

There are a number of options that may be specified for all commands. These options include:
`ShowUsage`

ShowUsage — This option may be included with any command to cause the command to return a usage message in SSSRMAP XML Message Format.

22.7 Common Actions Available for Most Objects

There are a number of actions that are available for most objects. These actions include Query, Create, Modify, Delete, and Undelete. Commands involving these actions inherit some common structure unique to the action type.

In this topic:

22.7.1 Query Action

The Query action is used to query objects. It accepts selections that describe the attributes (fields) to return (including aggregation operations on those attributes), conditions that select which objects to return the attributes for, and other options unique to queries.

Selections	Selections use the `Show` option to specify a list of the attributes to return for the selected object. If selections are not specified, a default set of attributes (defaulting to those not marked as hidden) will be returned.
	``` Name = Show Op = := Value = "selection1,selection2,selection3,..." ```
	Aggregation operators may be applied to attributes by enclosing the target attribute in parenthesis and prepending the name of the desired operator. The aggregation operators that can be applied depend on the datatype of the attribute.
	Valid selection operators include:
	Sort — Ascending sort
	Tros — Descending sort
	Count — Count
	Max — Maximum value
	Min — Minimum value
	Average — Average value
	Sum — Sum
	GroupBy — Group other aggregations by this attribute
	Partial values may be requested for complex (multi-valued) attributes. Partial values are specified in the form: `<attribute>{<part>}`.
	Additionally, aliases can be applied to selections so that columns can be renamed as desired. Aliases are expressed by adding "=`<Alias>`" to the target attribute name (and after the trailing parenthesis of the aggregation if specified).
	Examples:
	``` Allocation Query Show:="GroupBy(Fund),Sum(Amount)=Total" ```
	``` UsageRecord Query Show:="GroupBy(Account),Sum(Licenses{matlab})=Matlab_ Licenses_Used" ```

Conditions	Conditions are used to select which objects the action is to be performed on.
	Name = Name of the attribute to be tested
	Op = conditional operator
	Value = The object or value against which the attribute is tested.
	When expressing a condition that is part of a multi-object join, the name may consist of the attribute prepended with the object and a period (`<object>.<attribute>`).
	When expressing a condition for a part of a complex (multi-valued) attribute, the name will consist of the attribute followed by the part in curly braces (`<attribute>{<part>}`).
	Valid condition operators include:
	== Equal to
	!= Not equal to
	< Less than
	> Greater than
	<= Less than or equal to
	>= Greater than or equal to
	~ Matches
	!~ Does not match
	Matching uses the wildcards `*` and `?` (equivalent to SQL `%` and `_` respectively) in a manner similar to file globbing. `*` matches zero or more unspecified characters and `?` matches exactly one unspecified character. For example `mscf*` matches objects having the specified attributes whose values start with the letters `mscf`, while `mscf?` matches objects having the specified attributes whose values start with `mscf` and have a total of exactly five characters.
	Examples:
	`UsageRecord Query Application~"NWChem*"`
	`UsageRecord Query Metrics{temperature}>100.0`

|

Options	Options indicate processing options that affect the result.
	Name = Name of the option
	Op = :=
	Value = Value of the option
	Valid options for query actions include:
	ShowHidden:=True|False (False) Includes hidden attributes in the result
	Time:=`YYYY-MM-DD[ hh:mm:ss]` Run the command as if it were the specified time
	Unique:=True|False (False) Display only unique results (like DISTINCT in SQL)
	ChuckSize:={Integer Number} Number of records to return per page
	Limit:={Integer Number} Limit the results to the number of objects specified
	Offset:={Integer Number} Number or records to skip before starting to return data

> ℹ️ It is important to specify the sort order (using the Sort operator) when using the ChunkSize, Limit, or Offset options; otherwise the query may return records in a nondeterministic order (different order in different requests). It is especially important to specify the sort order when using the ChunkSize option, otherwise the same records might be returned in different chunks, while other records may not be returned at all. This is not a bug; it is a consequence of the behavior of the underlying database which does not promise to deliver the results of a query in any particular order unless ORDER BY is used to constrain the order.

Example 22-7: Return the number of inactive liens

```
mam> Lien Query EndTime<now Show:="Count(Id)"

Id
---
8
```

22.7.2 Create Action

The Create action is used to create a new object. It accepts assignments that describe the values of the attributes to be set.

Assignments	Assignments specify values to be assigned to attributes in the new object.
	Name = Name of the attribute being assigned a value
	Op = = (is assigned)
	Value = The new value being assigned to the attribute

Example 22-8: Add a new account member

```
mam> AccountUser Create Account=chemistry Name=scottmo

Account        Name        Active     Admin
-------------  ----------  ---------  -------
```

```
chemistry      scottmo      True        False
Successfully created 1 accountUser
```

22.7.3 Modify Action

The Modify action is used to modify existing objects. It accepts conditions that select which objects will be modified and predicates that describe the values of the attributes to be set.

Assignments	Assignments specify values to be assigned to attributes in the selected objects.
	Name = Name of the attribute being assigned a value
	Op = assignment operators {=, +=, -=}
	Value = The value being assigned to the attribute
	Valid assignment operators include:
	= is assigned
	+= is incremented by
	-= is decremented by
Conditions	Conditions are used to select which objects the action is to be performed on.
	Name = Name of the attribute to be tested
	Op = conditional operator
	Value = The object or value against which the attribute is tested
	Valid condition operators include:
	== Equal to
	!= Not equal to
	< Less than
	> Greater than
	<= Less than or equal to
	>= Greater than or equal to
	~ Matches
	!~ Does not match
	Matching uses the wildcards * and ? (equivalent to SQL % and _ respectively) in a manner similar to file globbing. * matches zero or more unspecified characters and ? matches exactly one unspecified character. For example mscf* matches objects having the specified attributes whose values start with the letters mscf, while mscf? matches objects having the specified attributes whose values start with mscf and have a total of exactly five characters.

Example 22-9: Change/set `scottmo`'s phone number and email address

```
mam> User Modify Name==scottmo PhoneNumber="(509) 376-2204"
EmailAddress="scottmo@adaptivecomputing.com"
```

```
-----------------------------------------------------------------------------------
Name        Active    CommonName                      PhoneNumber         EmailAddress
            DefaultAccount         Description
----------  --------- ----------------------------  -------------------- -----------------
------------- -------------------- ----------------
scottmo     True      Jackson, Scott M.             (509) 376-
2204         scottmo@adaptivecomputing.com
Successfully modified 1 user
-----------------------------------------------------------------------------------
```

Example 22-10: Extend all liens against account `chemistry` *by 10 days*

```
-----------------------------------------------------------------------------------
mam> Lien Modify EndTime+=864000 Instance=="job.1"


Id   Fund      Amount    Instance      UsageRecord  User    Project    Machine    EndTime
                Description
---  --------- --------- -----------   -----------  ------  ---------  ---------  -------
--------------------- ----------------
1    2         57600     PBS.1234.0  1              amy     chemistry  colony     2018-
04-16 10:47:30
Successfully modified 1 lien
-----------------------------------------------------------------------------------
```

22.7.4 Delete Action

The Delete action is used to delete objects. It accepts conditions that select which objects are to be deleted.

Conditions	Conditions are used to select which objects the action is to be performed on.
	Name = Name of the attribute to be tested
	Op = conditional operator
	Value = The object or value against which the attribute is tested
	Valid condition operators include:
	== Equal to
	!= Not equal to
	< Less than
	> Greater than
	<= Less than or equal to
	>= Greater than or equal to
	~ Matches
	!~ Does not match
	Matching uses the wildcards * and ? (equivalent to SQL % and _ respectively) in a manner similar to file globbing. * matches zero or more unspecified characters and ? matches exactly one unspecified character. For example `mscf*` matches objects having the specified attributes whose values start with the letters mscf, while `mscf?` matches objects having the specified attributes whose values start with `mscf` and have a total of exactly five characters.

Example 22-11: Get rid of the pesky Jacksons

```
mam> User Delete CommonName~"Jackson*"


Name        Active    CommonName                  PhoneNumber           EmailAddress
            DefaultAccount        Description
---------- --------- ------------------------ -------------------- -----------------
-------------- -------------------- ----------------
scottmo     True      Jackson, Scott M.         (509) 376-
2204        scottmo@adaptivecomputing.gov
Successfully deleted 1 user and 1 association
```

22.7.5 Undelete Action

The Undelete action is used to restore deleted objects. It accepts conditions that select which objects are to be undeleted.

Conditions	Conditions are used to select which objects the action is to be performed on.
	Name = Name of the attribute to be tested
	Op = conditional operator
	Value = The object or value against which the attribute is tested
	Valid condition operators include:
	== Equal to
	!= Not equal to
	< Less than
	> Greater than
	<= Less than or equal to
	>= Greater than or equal to
	~ Matches
	!~ Does not match
	Matching uses the wildcards * and ? (equivalent to SQL % and _ respectively) in a manner similar to file globbing. * matches zero or more unspecified characters and ? matches exactly one unspecified character. For example mscf* matches objects having the specified attributes whose values start with the letters mscf, while mscf? matches objects having the specified attributes whose values start with mscf and have a total of exactly five characters.

Example 22-12: Resurrect the deleted users that were active

```
mam> User Undelete Active==True


Name        Active    CommonName                  PhoneNumber           EmailAddress
            DefaultAccount        Description
---------- --------- ------------------------ -------------------- -----------------
-------------- -------------------- ----------------
scottmo     True      Jackson, Scott M.         (509) 376-
```

```
2204          scottmo@adaptivecomputing.com
Successfully undeleted 1 user and 1 association
```

22.8 Multi-Object Queries

mam-shell supports multi-object queries (table joins). Multiple objects are specified via a comma-separated list and attributes need to be prefixed by the associated object.

Example 22-13: Print the sums for active balance and allocated amounts grouped by account

```
mam> Allocation,Constraint Query
Show:="GroupBy(Constraint.Value)=Account,Sum(Allocation.Amount)=Balance,Sum
(Allocation.Deposited)=Allocation"
Constraint.Fund==Allocation.Fund Constraint.Name==Account
Allocation.Active==True


Account         Balance        Allocation
-------------   -------------  -------------
biology         193651124      360000000
chemistry       296167659      360000000
```

Example 22-14: Show all active accounts for `amy`'s privileges

```
mam> RoleUser,RoleAction Query
Show:="RoleAction.Object,RoleAction.Name=Action"
RoleUser.Role==RoleAction.Role && ( RoleUser.Name==amy ||
RoleUser.Name==ANY ) Unique:=True


Object              Action
------------------  -------
Account             Query
AccountUser         Query
Action              Query
Allocation          Query
Attribute           Query
ChargeRate          Query
Constraint          Query
Fund                Query
FundFund            Query
Lien                Query
LienAllocation      Query
Object              Query
Organization        Query
Password            ANY
Quote               Query
QuoteChargeRate     Query
Role                Query
RoleAction          Query
RoleUser            Query
System              Query
Transaction         Query
UsageRecord         Query
User                Query
```

> **ⓘ** Although the forgoing was a good example of a join request, it should be understood that it is not a straightforward way to determine the full extent of a user's privileges. Some of the actions may be tied to specific object instances and many of them are associated with an over-ride method which may not actually permit the user access to any instances of the object. Using
> `Show:="RoleUser.Role,RoleUser.Name=User,RoleAction.Object,RoleAction.Name=Action,RoleAction.Instance"` may be revealing in this regard. See the chapter on Managing Roles for more information about managing roles.

Chapter 23: Customizing Objects

Moab Accounting Manager provides the ability to dynamically create new objects or customize or delete existing objects through the interactive control program (mam-shell).

> ℹ The object customizations described in this chapter will be noticeable in subsequent mam-shell queries (and in the web GUI after a fresh login). Client commands may need to be modified to properly interact with changed objects or attributes.

> ℹ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. Inadvertent mistakes could result in modifications that are very difficult to reverse.

In this chapter:

23.1 Managing Objects

In Moab Accounting Manager, Objects correspond to tables in the repository which have Attributes (such as Name and Color) and Actions (such as Query and Modify). A specific instance of an object is described as an Instance and has Properties (the specific values of the attributes for that object). The instance is uniquely referred to via its primary key(s) (such as its Name or Id).

An object must have a name and may have a description. An object may be set to auto-generate its instances when first seen (see Object Auto-Generation) and/or a default value may be designated for the object (see Global Object-Based Defaults).

Objects may reference other objects. If a single instance of an object references only a single instance of another object (for example, a usage record may only have one user), then it is sufficient for the first object to have an attribute field for the second object (the UsageRecord object has an attribute called User). However, if there may be a many-to-many relationship between objects (for example, an account may have multiple users and a user may belong to multiple accounts), then it is necessary to maintain a separate object as an association table (e.g. AccountUser). When creating an association object, the object should be given an appropriate name (e.g. AccountUser), it should be marked as an association (`Association=True`), and an object needs to be designated for the parent (e.g. Account) and the child (e.g. User). The association object itself may have additional attributes that provide qualitative information about the association (e.g. a particular AccountUser association may be active or be an administrator).

23.2 Creating a Custom Object

To create a new object, use the command *mam-shell Object Create*. When an object is created, the 5 default actions are automatically created for the object: Create, Delete, Modify, Query and Undelete. A number of default metadata attributes are created as well: CreationTime, ModificationTime, Deleted, RequestId and TransactionId. These attributes are normally hidden in regular queries.

```
mam-shell Object Create Name=<Object Name> [AutoGen=True|(False)]
[DefaultValue=<Default Value>] [Description=<Description>]
[Association=True|False)] [Child=<Child Object>] [Parent=<Parent
Object>][ShowUsage:=True]
```

Example 23-1: Creating a Node Object

```
$ mam-shell Object Create Name=Node Description=\"Node Information\"

Successfully created 1 object and 5 actions
```

Example 23-2: Add a node name attribute

```
$ mam-shell Attribute Create Object=Node Name=Name DataType=String PrimaryKey=True

Successfully created 1 attribute
```

Example 23-3: Add a processor count attribute

```
$ mam-shell Attribute Create Object=Node Name=Processors DataType=Integer

Successfully created 1 attribute
```

23.3 Querying Objects

To display object information, use the command *mam-shell Object Query*.

mam-shell Object Query [Name=<Object Name>]
[Show:=Name,AutoGen,DefaultValue,Description,Association,Parent,Chil
d] [ShowUsage:=True]

Example 23-4: List Information for the Node Object

```
$ mam-shell Object Query Name==Node

Name   Association   Parent   Child   DefaultValue   AutoGen   Description
----   -----------   ------   -----   ------------   -------   ----------------+
Node   False                                         False     Node Information
```

23.4 Modifying an Object

It is possible to modify an object by using the command *mam-shell Object Modify*.

mam-shell Object Query [Name=<Object Name>] [AutoGen=True|False]
[DefaultValue=Default Value>] [Description=Description>]
[Association=True|(False)] [Child=Child Object>] [Parent=Parent
Object>] [ShowUsage:=True]

Example 23-5: Changing the Node object's description

```
$ mam-shell Object Modify Name==Node Description="\"Host Information\""

Successfully modified 1 object
```

23.5 Deleting an Object

To delete an object, use the command *mam-shell Object Delete*. When an object is deleted, all associated attributes, actions and other associations are automatically deleted as well.

mam-shell Object Delete [Name=<Object Name>] [ShowUsage:=True]

Example 23-6: Deleting the Node Object

```
$ mam-shell Object Delete Name==Node

Successfully deleted 1 object
```

> ℹ This is a very dangerous operation and could result in the deletion of all object definitions requiring database repair. The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. Be sure to specify conditions for the object you want to delete.

23.6 Object Auto-Generation

It is possible to have object instances be automatically generated the first time they are referenced in designated contexts. For example, you might want a user be auto-generated when newly added to an account. You could have an organization auto-generated when specified as the default for a user. You could have a cost-center be auto-generated when referenced in a usage record. To do this, the referenced object must be set to `AutoGen=True` and the `Values` property for the attribute that you want to trigger the auto-generation must be set to a string consisting of the `@` sign followed by the object name.

Example 23-7: Auto-generate an account's organization

For example, let's assume that your accounts belong to specific organizations that you may want to run a report against but you don't want to define all of the organizations up front. It would be possible to automatically generate a new organization instance each time an undefined organization is specified for an account.

```
$ mam-shell Object Modify Name==Organization AutoGen=True

Successfully modified 1 object
```

```
$ mam-shell Attribute Modify Object==Account Name==Organization Values=@Organization

Successfully modified 1 attribute
```

See Usage Record Property Auto-Generation for a discussion of auto-generating objects referenced in usage records.

Related Topics

- 14.12 Usage Record Property Auto-Generation - page 159

23.7 Global Object-Based Defaults

It is possible to set a global default for an object that will be applied to all attributes referencing this object. When a new instance of an object is being created which has an attribute referring to another object via its `Values` property, if that attribute has not been specified and you want it to default to the global default, you will need to set the `DefaultValue` attribute for the referenced object to the desired value.

Example 23-8: Setting a system-wide simple default organization called general

```
$ mam-shell Object Modify Name==Organization DefaultValue=general

Successfully modified 1 object
```

Thereafter each (non-association) object which has an attribute with a `Values` property set to `@Organization` will default to general if that attribute is not specified. Perhaps we would want the default value to be taken for the organization when a new account is created.

```
$ mam-shell Attribute Modify Object==Account Name==Organization Values=@Organization

Successfully modified 1 attribute
```

See Local Attribute-Based Defaults for more information about setting default values for attributes. See Usage Record Property Defaults for more information about setting default values for usage record properties.

Related Topics

- 23.13 Local Attribute-Based Defaults - page 235
- 14.11 Usage Record Property Defaults - page 158

23.8 Managing Attributes

Objects can have any number of fields called Attributes. When an object is first created, a number of attributes are created for the object by default. These are: `CreationTime` (time the object was first created), `ModificationTime` (time the object was last updated), `Deleted` (whether the object is deleted or not), `RequestId` (request id that resulted in the last modification of the object), `TransactionId` (transaction id that resulted in the last modification of the object).

An attribute must have a name and be associated with an object.

An attribute will have a data type which can be one of (`AutoGen`, `Boolean`, `Currency`, `Float`, `Integer`, `JSON`, `String`, `TimeStamp`) and defaults to String. A data type of AutoGen means the field will be a primary key of type integer which will assume the next auto-incremented value from the g_key_generator table. TimeStamps are epoch times stored in integer format. Booleans are strings constrained to the values of `True` or `False` (or unset). Float is used to store decimal or floating point values. Currency is like Float but may have special business logic for handling currency values. The JSON data type provides support for complex properties and must store a valid JSON value. The current implementation only provides support for simple JSON objects of the form `{key:value,...}` where key is a double-quoted string and value may be a number or a double-quoted string. One may also use the more nuanced forms (`JSON:Integer`, `JSON:Float`, or `JSON:String`, etc.) to indicate the expectation that the values of the JSON object will be of the designated variety. Using these forms may be useful for clients and web services to render partial queries in the anticipated data type.

An object may have zero or more attributes which are primary keys (`PrimaryKey==True`), the combination of which are used to uniquely identify an object instance. Moab Accounting Manager will try to ensure that there can only be one object instance with the exact same set of values of its primary keys.

A required attribute (`Required==True`), must be either specified or be derived via a default value or other dynamic mechanism when the object is created. It can also not be unset.

A fixed attribute (`Fixed==True`), may not be changed from its initial value.

An attribute may be constrained to certain values via the `Values` attribute. The values may be constrained to members of a list expressed as a parenthesized comma-delimited list of strings (i.e. `Values="(Brazil,China,France,Russia,USA)"`). Alternatively, the values may be constrained to be an instance of a particular object type (like a foreign key constraint) by assigning to the Values attribute the name of an object prefixed by the `@` sign (e.g. `Values="@Account"` -- which would constrain the value of this attribute to be a valid account name). Stronger versions of the `@`-prefixed object-constrained values may be used in Quote, Reserve and Charge actions to enforce dynamic interactions between usage record properties such as to assign default values if not defined (e.g. `Values="@?=Account"`), verification values which evoke an error if they differ (e.g. `Values="@!=Account"`), or designated values which always overwrite the value (e.g. `Values="@:=Account"`). See Usage Record Property Instantiators for more information.

A default value may be assigned to an attribute via the `DefaultValue` attribute. When a new instance of an object is created, if a property is not specified for the attribute, the default value will be used.

The `Sequence` attribute determines which order an object's attributes will be listed in for queries if no selection list is specified in the query. Attributes with smaller sequence numbers will appear before attributes with larger sequence numbers. The Sequence attribute is also used to enforce a proper attribute display ordering in the web GUI.

The `Hidden` attribute specifies whether an attribute should be shown in a query by default or not. Hidden attributes can be seen in queries by specifying the `ShowHidden` option with a value of `True`.

The `Description` field is a location to describe the meaning of the attribute and is used in the GUI for field descriptions.

23.9 Adding an Attribute to an Object

To create a new attribute for an object, use the command *mam-shell Attribute Create*:

mam-shell Attribute Create Object=<Object Name> Name=<Attribute Name> [DataType=AutoGen|TimeStamp|Boolean|Float|Integer|Currency|(String)] [PrimaryKey=True|(False)] [Required=True|(False)] [Fixed=True| (False)] [Values=<Foreign Key or List of Values>] [DefaultValue=<Default Value>] [Sequence=<Integer Number>] [Hidden=<True|(False)>] [Description=<Description>] [ShowUsage:=True]

Example 23-9: Adding a Country Attribute to User

```
$ mam-shell Attribute Create Object=User Name=Country Values="\"
(Brazil,China,France,Russia,USA)\"" DefaultValue=USA

Successfully created 1 attribute
```

Example 23-10: Tracking Submission Time in Usage records

```
$ mam-shell Attribute Create Object=UsageRecord Name=SubmissionTime DataType=TimeStamp

Successfully created 1 attribute
```

23.10 Querying Attributes

To display attribute information, use the command `mam-shell Attribute Query`:

`mam-shell Attribute Query Object=<Object Name> Name=<Attribute Name>`
`[Show:=Object,Name,DataType,PrimaryKey,Required,Fixed,Values,DefaultV`
`alue,Sequence,Hidden,Description] [ShowHidden:=True]`
`[ShowUsage:=True]`

Example 23-11: List the attributes of the Node object

```
$ mam-shell Attribute Query Object==Node


Object Name            DataType  PrimaryKey Required Fixed Values DefaultValue Sequen
ce Hidden Description
------ ---------------- --------- ---------- -------- ----- ------ ------------ ------
-- ------ ---------------------------

Node   Processors       Integer   False      False    False                      20
   False

Node   Name             String    True       True     True                       10
   False

Node   TransactionId    Integer   False      False    True                       990
   True   Last Modifying Transaction Id

Node   RequestId        Integer   False      False    True                       980
   True   Last Modifying Request Id

Node   Deleted          Boolean   False      False    True                       970
   True   Is this object deleted?

Node   ModificationTime TimeStamp False      False    True                       960
   True   Last Updated

Node   CreationTime     TimeStamp False      False    True                       950
   True   First Created
```

23.11 Modifying an Attribute

To modify an attribute, use the command *mam-shell Attribute Modify*:

mam-shell Attribute Modify Object==<Object Name> Name==<Attribute Name> [Required=True|(False)] [Fixed=True|(False)] [Values=<Foreign Key or List of Values>] [DefaultValue=<Default Value>] [Sequence=<Integer Number>] [Hidden=<True|(False)>] [Description=<Description>] [ShowUsage:=True]

> ⓘ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. A mistake made using this command could result in the inadvertent modification of all attributes.

Example 23-12: Change Account Organization values to not be restricted to the set of organization instances

```
$ mam-shell Attribute Modify Object==Account Name==Organization Values=NULL

Successfully modified 1 attribute
```

23.12 Removing an Attribute From an Object

To delete an attribute, use the command *mam-shell Attribute Delete*:

mam-shell Attribute Delete Object==<Object Name> Name==<Attribute Name> [ShowUsage:=True]

> ⓘ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. A mistake made using this command could result in the inadvertent deletion of all attributes.

> ⓘ When using Moab Accounting Manager as an accounting manager, certain objects and attributes are assumed to exist. For example, a call to UsageRecord Charge would fail if you had deleted the `Allocation Amount` attribute. The *Attribute Undelete* command might come in useful in such a case.

Example 23-13: Removing the Organization attribute from Account

```
$ mam-shell Attribute Delete Object==Account Name==Organization

Successfully deleted 1 attribute
```

Example 23-14: Perhaps we don't care to track the QualityOfService attribute in a Usage record

```
$ mam-shell Attribute Delete Object==UsageRecord Name==QualityOfService

Successfully deleted 1 attribute
```

23.13 Local Attribute-Based Defaults

It is possible to set a specific default for an object attribute that will be applied when an instance of that object is created but the attribute is not specified. This type of default is intended for attributes which do not refer to another object or which should vary from the object global default. This default value is assigned to an attribute via the `DefaultValue` attribute. When a new instance of the associated object is created, if a property is not specified for the attribute, the specified default value will be used. A local attribute default will have precedence over a global object default.

mam-shell Attribute Delete Object`==<Object Name> Name==<Attribute Name> [ShowUsage:=True]`

Example 23-15: Setting a default organization just for the account object

```
$ mam-shell Attribute Modify Object==Account Name==Organization
DefaultValue=university

Successfully modified 1 attribute
```

Example 23-16: Setting a default phone for the user object

```
$ mam-shell Attribute Modify Object==User Name==PhoneNumber DefaultValue="\"NoPhone\""

Successfully modified 1 attribute
```

See Global Object-Based Defaults for more information about setting default values for objects.

See Usage Record Property Defaults for more information about setting default values for usage record properties.

Related Topics

- 23.7 Global Object-Based Defaults - page 228
- 14.11 Usage Record Property Defaults - page 158

23.14 Managing Actions

Moab Accounting Manager defines which actions can be performed by which objects. When an object is first created, five basic actions are created for the object by default. These are: Create, Modify, Query, Delete and Undelete. Specific code must exist in Moab Accounting Manager modules in order for objects to support additional actions.

An action is uniquely specified by its name and the object with which it is associated. An action also has a description and a boolean display attribute which governs whether this action should be displayed in the web GUI or not.

23.15 Adding an Action to an Object

To specify that an action is allowed for an object, use the command *mam-shell Action Create*:

mam-shell Action Create Object=<Object Name> Name=<Action Name> [Display=True|(False)] [Description=<Description>] [ShowUsage:=True]

Example 23-17: Adding a Modify Action to Transaction

```
$ mam-shell Action Create Object=Transaction Name=Modify Description=Modify

Successfully created 1 action
```

23.16 Querying Actions

To display action information, use the command *mam-shell Action Query*:

mam-shell Action Query [Object==<Object Name>] [Name==<Attribute Name>] [Show:=Object,Name,Display,Description] [ShowUsage:=True]

Example 23-18: List the actions of the Node object

```
$ mam-shell Action Query Object==Node

Object Name      Display Description
------ --------  ------- -----------
Node   Create   False   Create
Node   Delete   False   Delete
Node   Modify   False   Modify
Node   Query    False   Query
Node   Undelete False   Undelete
```

23.17 Modifying an Action

To modify an action, use the command `mam-shell Action Modify`:

`mam-shell Action Modify` [Object==<Object Name>] [Name==<Attribute Name>] [Display=True|(False)] [Description=<Description>] [ShowUsage:=True]

> ℹ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. A mistake made using this command could result in the inadvertent modification of all actions.

Example 23-19: Display all Node actions but Undelete in the web GUI

```
$ mam-shell Action Modify Object==Node Name!=Undelete Display=True

Successfully modified 4 actions
```

23.18 Removing an Action From an Object

To delete an action from an object, use the command *mam-shell Action Delete*:

mam-shell Action Delete [Object==<Object Name>] [Name==<Attribute Name>] [ShowUsage:=True]

> ⓘ The mam-shell control program allows you to make powerful and sweeping modifications to many objects with a single command. A mistake made using this command could result in the inadvertent modification of all actions.

> ⚠ When using Moab Accounting Manager as an accounting manager, certain actions are assumed to exist. Be careful what you delete!

Example 23-20: Do not allow accounts to be deleted

```
$ mam-shell Action Delete Object==Account Name==Delete

Successfully deleted 1 action
```

23.19 Examples Creating Custom Objects

Creating a custom object normally involves defining a new object and adding attributes to the object.

Example 23-21: Creating a License object to track license usage and charges.

Invoke the Moab Accounting Manager control program in interactive mode.

```
$ mam-shell
```

Create the License Object.

```
mam> Object Create Name=License Description=License

Successfully created 1 object and 5 actions
```

Next, define its attributes. Give each record a unique id (so the record can be more easily modified), a license type that can be one of (Matlab,Mathematica,Compiler,AutoCAD,Oracle), the user who is using it, the start and end time, how many instances of the license were used, and how much was charged.

```
mam> Attribute Create Object=License Name=Id DataType=AutoGen PrimaryKey=True
Description="Record Id"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=Type DataType=String Required=True Values="
(Matlab,Mathematica,Compiler,AutoCAD,Oracle)" Fixed=True Description="License Type"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=User Required=True Values="@User"
Description="User Name"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=StartTime DataType=TimeStamp
Description="Start Time"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=EndTime DataType=TimeStamp Description="End
Time"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=Count DataType=Integer Description="Number
of Licenses Used"

Successfully created 1 attribute

mam> Attribute Create Object=License Name=Charge DataType=Currency Description="Amount
Charged"

Successfully created 1 attribute
```

Finally, since we would like to manage licenses from the web GUI, set `Display=True`.

```
mam> Action Modify Object==License Name!=Undelete Display=True

Successfully modified 4 actions
```

When done, exit the mam-shell prompt.

```
mam> quit
```

That's about it. Licenses should now be able to be managed via the GUI and mam-shell. The data source will need to use one of the methods of interacting with Moab Accounting Manager (see Methods of Interacting with Moab Accounting Manager) in order to push license record usage info to Moab Accounting Manager.

Apart from being used as an accounting manager, Moab Accounting Manager can be used as a generalized information service. It can be used to manage just about any object-oriented information over the web. For example, Moab Accounting Manager could be used to provide meta-schedulers with machine/user mappings, or node/resource information.

Example 23-22: Using Moab Accounting Manager as a Grid Map File.

Invoke the mam-shell control program in interactive mode.

```
$ mam-shell
```

Create the GridMap Object.

```
mam> Object Create Name=GridMap Description="Online Grid Map File"

Successfully created 1 object and 5 actions
```

Next, define its attributes. Each entry will consist of a userid (which will serve as the primary key) and a required public X.509 certificate.

```
mam> Attribute Create Object=GridMap Name=User PrimaryKey=True Values=@User
Description="User Name"

Successfully created 1 attribute

mam> Attribute Create Object=GridMap Name=Certificate DataType=String Required=True
Description="X.509 Public Key"

Successfully created 1 attribute
```

Exit the mam-shell prompt.

```
mam> quit
```

From this point, a peer service will need to use one of the methods of interacting with Moab Accounting Manager (see Interaction Methods) in order to query the GridMap information.

Chapter 24: Integration

Moab Accounting Manager (MAM) works in conjunction with a resource management system, such as Moab Workload Manager or Slurm. Moab Workload Manager incorporates direct support for MAM and offers a full-featured integration. Slurm can utilize the usage tracking or notional charging accounting modes via simple configuration and can be be adapted to support the strict allocation accounting mode with a small patch. MAM can generally be made to work with other resource management systems, so long as they provide support for prolog and epilog scripts.

24.1 Integrating with Moab Workload Manager

Moab Workload Manager can be configured to interact with Moab Accounting Manager to track and charge for resources utilized by jobs and reservations. You will need to have the Accounting Manager Licensing feature in Moab in order to have support for Moab Accounting Manager.

24.1.1 Select an Appropriate Accounting Management Interface Type

There are two accounting manager interface types that Moab can use to interact with Moab Accounting Manager: MAM, which makes direct calls to MAM over the SSS wire protocol, and Native, in which customizable scripts are invoked to communicate with Moab Accounting Manager. The MAM accounting manager interface is the default as it is usually faster. The Native accounting manager interface can be used if higher customizability is needed, or if you need to interface with a third party accounting or allocation system. See Accounting Manager Interface Types in the *Moab Workload Manager 9.1.3 Administrator Guide* for more information. Choose the accounting manager interface type that is right for your needs and remember it. This information will be used in a later step.

24.1.2 Run Configure --with-am

It may be necessary or advantageous when installing Moab Workload Manager to run configure with certain accounting related options.

Configure Moab to use the Moab Accounting Manager by running ./configure with the applicable options when installing Moab:

- `--with-am[=TYPE]` - Enable accounting management with the specified accounting manager interface type (mam or native) [mam].

- `--with-am-dir=DIR` - Uses the specified prefix directory for the accounting manager if installed in a non-default location.

The --with-am option specifies the accounting manager interface type that you want to use as either mam, which is the default, or native. Specifying this option will add essential entries into Moab configuration files. Although these entries may be added manually later, this step facilitates configuration by adding parameters appropriate for your selected accounting manager interface type.

Use --with-am-dir to specify the prefix directory for Moab Accounting Manager if it has been installed in a non-default location. This value is used to help the native accounting manager scripts find the Moab Accounting Manager libraries and server connection information.

Example 24-1: Configuring Moab to Use the Direct Accounting Manager Interface

```
$ ./configure --with-am
```

24.1.3 Edit the Moab Server Configuration File

Add or uncomment the essential AMCFG lines in the moab.cfg file.

Example 24-2: Configuring Moab to Use the MAM Accounting Manager Interface

If you are using the direct (MAM) accounting manager interface, at a minimum, you must tell Moab to use `AMCFG[] TYPE=MAM`. Additionally, if your Moab Accounting Manager server is running on a different host than the Moab Workload Manager server, you must specify the hostname via the AMCFG[] HOST parameter.

```
$ vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=MAM HOST=localhost
```

Example 24-3: Configuring Moab to Use the NativeAccounting Manager Interface

If you are using the script (Native) accounting manager interface, at a minimum, you must tell Moab to use `AMCFG[] TYPE=NATIVE`. Moab Workload Manager will default to using a set of stock scripts to interact with Moab Accounting Manager.

```
$ vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=NATIVE
```

24.1.4 Edit the Moab Private Configuration File

If you have chosen to use the direct MAM accounting manager interface type, you will need to configure Moab to have Moab Accounting Manager's symmetric key for secure authentication. This step is not necessary when using the Native accounting manager interface type since the secret key can be securely derived from Moab Accounting Manager and used via the connection libraries.

Example 24-4: Configuring Moab to Communicate Securely with Moab Accounting Manager

Add or uncomment a `CLIENTCFG[AM:mam] KEY` parameter line in moab-private.cfg. Copy the `token.value` parameter in `/opt/mam/etc/mam-site.conf` into the KEY value in `/opt/moab/etc/moab-private.cfg`.

```
# vi /opt/moab/etc/moab-private.cfg
CLIENTCFG[AM:mam] KEY=UiW7EihzKyUyVQg6dKirDhV3
```

24.1.5 Restart Moab Workload Manager

In order for the configuration changes to take effect, restart Moab.

- Red Hat 6-based or SUSE 11-based systems

```
# service moab restart
```

- Red Hat 7-based, SUSE 12-based, or Fedora systems

```
# systemctl restart moab.service
```

24.2 Integrating with Moab Web Services

Moab Web Services can be configured to interact with Moab Accounting Manager order to be able to perform RESTful web service queries against accounting objects in Moab Accounting Manager.

> In this topic:
>

24.2.1 Edit the MWS HPC Configuration File

Uncomment and set the following parameters in /opt/mws/etc/mws.d/mws-config-hpc.groovy:

- **mam.secretKey** - Set to the value of the token.value parameter in /opt/mam/etc/mam-site.conf

- **mam.server** - Set to the hostname of the MAM server

- **mam.port** - Set to the port of the MAM server (defaults to 7112)

Example 24-5: Configuring Moab Web Services to Communicate with MAM Accounting Manager

```
$ vi /opt/mws/etc/mws.d/mws-config-hpc.groovy

mam.secretKey = "UiW7EihzKyUyVQg6dKirDhV3"
mam.server = "localhost"
mam.port = 7112
```

24.2.2 Restart Moab Web Services

In order for the MWS configuration changes to take effect, restart Tomcat

- Red Hat 6-based systems

```
# service tomcat restart
```

- Red Hat 7-based, SUSE 12-based, or Fedora systems

```
# systemctl restart tomcat.service
```

24.3 Integrating With Slurm

Moab Accounting Manager can be configured to interact with Slurm to track and charge for resources utilized by jobs. The integration involves the use of an epilog script as well as a patch and the use of a prolog script if enforcing allocations.

24.3.1 Copy MAM's Slurm Contrib Scripts

If you installed MAM from tarball, the Slurm integration scripts can be found in the directory where you unpacked the tarball. If you installed from RPM, the Slurm integration scripts can be found in `/usr/share/moab-accounting-manager/contrib`. Copy MAM's Slurm contrib scripts to `/opt/slurm/etc` and ensure that they are owned and executable by the Slurm user.

Example 24-6: Copying the Slurm Contrib Scripts

```
[root]# cp /software/mam-<version>/contrib/slurm/mam-*.slurm.pl /opt/slurm/etc
[root]# chown slurm:slurm /opt/slurm/etc/mam-*.slurm.pl
[root]# chmod +x /opt/slurm/etc/mam-*.slurm.pl
```

24.3.2 Set Database Max Connections Appropriately

Each Slurm job will require a database connection when creating liens for jobs, and another when charging for the job completion. For array jobs, or a large batch of simultaneous submissions, this can result in a large number of simultaneous connections. If the database does not have enough connections configured, this will result in array job failures.

Be sure to increase your database connections enough to handle your expected workload. A good rule of thumb would be to set it to at least half the expected number of jobs that might be submitted within a short time. Also, be sure to monitor Slurm's logs for database connection failures from `scontrol`.

For PostgreSQL, as an example, the database connections are specified in `postgresql.conf` (usually found in the directory `/var/lib/pgsql/data/`). The number of connections is specified by the `max.connections` value. When increasing the connection count, it's a good idea to also increase the size of the pool for shared buffers (using the `shared_buffers` setting). Your needs may vary, but a good starting point is 32 MB of shared buffer space for every 100 connections.

Also, when making these adjustments you may also need to increase the kernel's `shmmax` setting. It is recommended that a qualified DBA review the database configuration.

24.3.3 Configure the Controller Epilog to Call the MAM Charge Script

If you do not intend to use the `slurmctld` epilog for any purpose other than for integration with MAM, you can configure Slurm to call the script directly by editing the Slurm configuration file, setting the `EpilogSlurmctld` to point to the `mam-charge.slurm.pl` file, and reconfiguring `slurmctld`.

Example 24-7: Setting the Controller Epilog to Call the Charge Script Directly

```
[root]# vi /opt/slurm/etc/slurm.conf

EpilogSlurmctld=/opt/slurm/etc/mam.charge.slurm.pl

[root]# scontrol reconfigure
```

If you already have a `slurmctld` epilog configured, the charge script may be called within your existing epilog script. Edit your `slurmctld` epilog script and add a section at the end of the epilog that calls the charge script and exits with the status returned by the charge script. The exit in this case is optional and may be excluded if desired, as its only use is for logging purposes.

Example 24-8: Editing the Existing Epilog Script to Call the Charge Script

```
[root]# vi <slurmctld_epilog_script>
```

If you are using a bash script for your slurmctld epilog, include an excerpt similar to the following:

```
/opt/slurm/etc/mam.charge.slurm.pl
exit $?
```

If you are using a Perl script, include an excerpt similar to the following:

```
my $cmd = "/opt/slurm/etc/mam.charge.slurm.pl";
my $output = `$cmd 2>&1` || `sh -c "$cmd 2>&1"`;
exit $? >> 8;
```

If you are using a Python script, include an excerpt similar to the following:

```
import subprocess
cmd = '/opt/slurm/etc/mam.charge.slurm.pl'
rc = subprocess.Popen(cmd).wait()
exit(rc)
```

24.3.4 Patch Slurm

If you intend to use the strict allocation accounting mode in MAM, you will need to patch Slurm in order for Slurm to enforce your configured failure action when unable to obtain a lien with MAM. This patch will need to be reapplied each time Slurm is upgraded.

Example 24-9: Patching Slurm

```
[root]# scontrol shutdown slurmctld
[root]# cd /software/slurm-<version>
[root]# patch -p 0 < /software/mam-<version>/contrib/slurm/slurm-mam.patch
[root]# make
[root]# make install
[root]# su - slurm -c "slurmctld"
```

24.3.5 Configure the Controller Prolog to Call the MAM Reserve Script

If you intend to use the strict allocation accounting mode in MAM, you will need to configure SLURM to call the reserve script from the `slurmctld` prolog.

If you do not intend to use the `slurmctld` prolog for any purpose other than for integration with MAM, you can configure Slurm to call the script directly by editing the Slurm configuration file, setting the `PrologSlurmctld` to point to the `mam-reserve.slurm.pl` file, and reconfiguring `slurmctld`.

Example 24-10: Setting the Controller Prolog to Call the Reserve Script Directly

```
[root]# vi /opt/slurm/etc/slurm.conf

PrologSlurmctld=/opt/slurm/etc/mam.reserve.slurm.pl

[root]# scontrol reconfigure
```

If you already have a `slurmctld` prolog configured, the reserve script may be called within your existing prolog script. Edit your `slurmctld` prolog script and add a section in the prolog that calls the reserve script and exits with an appropriate exit code.

Example 24-11: Editing the Existing Prolog Script to Call the Reserve Script

```
[root]# vi <slurmctld_prolog_script>
```

If you are using a bash script for your `slurmctld` prolog, include an excerpt similar to the following:

```
/opt/slurm/etc/mam.reserve.slurm.pl
rc=$?
if (( $rc >= 78 && $rc <= 103 )); then
    exit $?
fi
```

If you are using a Perl script, include an excerpt similar to the following:

```
my $cmd = "/opt/slurm/etc/mam.reserve.slurm.pl";
my $output = `$cmd 2>&1` || `sh -c "$cmd 2>&1"`;
my $rc =  $? >> 8;
exit $rc if ($rc >= 78 && $rc <= 103);
```

If you are using a Python script, include an excerpt similar to the following:

```
import subprocess
cmd = '/opt/slurm/etc/mam.reserve.slurm.pl'
rc = subprocess.Popen(cmd).wait()
if rc >= 78 and rc <= 103:
    exit(rc)
```

24.3.6 Customize the Reserve Script

If you intend to use the strict allocation accounting mode in MAM, edit the `mam.re-serve.slurm.pl` script and set the connection failure action, funds failure action, and general failure action values according to your desired policy.

Before starting a job, the prolog will call MAM to create a lien in order to verify and protect the funds required for the job run. If the lien fails, one of four failure actions can be applied:

- IGNORE – allows the job to start

- DEFER – delays the start of the job for 5 minutes

- HOLD – puts an administrative hold on the job

- CANCEL – cancels the job

A separate failure action can be configured for each of three different situations:

- The connection failure action is applied if there is a communication problem with the accounting manager.

- The funds failure action is applied if the lien request is rejected due to insufficient funds.

- The general failure action is applied if the accounting manager rejects the lien request for any other reason.

Example 24-12: Configuring the Failure Action Policies in the Reserve Script

```
[root]# vi /opt/slurm/etc/mam.reserve.slurm.pl

my $connectionFailureAction = 'DEFER';
my $fundsFailureAction = 'HOLD';
my $generalFailureAction = 'CANCEL';
```

When an accounting failure occurs in the prolog, the MAM response message and the resulting failure action is recorded in the job's comment field.

24.4 Methods of Interacting with Moab Accounting Manager

There are several ways of interacting with Moab Accounting Manager. Let's consider a simple usage charge in each of the different ways.

> In this topic:
>

24.4.1 Using the Appropriate Command-line Client

From inside a script, or by invoking a system command, you can use a command-line client (one of the "g" commands in the bin directory).

Example 24-13: To issue a charge at the completion of job usage, you could use mam-charge:

```
mam-charge -J Moab.1234 -a chemistry -u amy -m colony -P 2 -t 3600
```

24.4.2 Using the Interactive Control Program

The interactive control program, mam-shell, will issue a charge for a job expressed in xml.

Example 24-14: To issue a charge you must invoke the Charge action on the Job object:

```
mam-shell UsageRecord Charge
Data:="<UsageRecord><Instance>Moab.1234</Instance><Account>chemistry</Account><User>am
y</User><Machine>colony</Machine><Processors>2</Processors><Duration>3600</Duration></
UsageRecord>" Duration:=3600
```

24.4.3 Use the Perl API

The Perl API exposes the full functionality of Moab Accounting Manager. The client commands can be examined as sample code. Use perldoc on the modules in lib/mam for function documentation.

Example 24-15: To make a charge via this interface you might do something like:

```
use MAM;

my $request = new MAM::Request(object => "UsageRecord", action => "Charge");
```

```
my $usageRecord = new MAM::Datum("UsageRecord");
$usageRecord->setProperty("Instance", "Moab.1234");
$usageRecord->setProperty("Account", "chemistry");
$usageRecord->setProperty("User", "amy");
$usageRecord->setProperty("Machine", "colony");
$usageRecord->setProperty("Processors", "2");
$usageRecord->setProperty("Duration", "3600");
$request->addDatum($usageRecord);
$request->setOption("Duration", "3600");
my $response = $request->getResponse();
print $response->getStatus(), ": ", $response->getMessage(), "\n";
```

24.4.4 Use the Java API (no longer supported)

> ℹ Beginning with Moab Accounting Manager 9.0, Java API is no longer supported, use Perl API instead.

24.4.5 Communicating via the SSSRMAP Protocol

Finally, it is possible to interact with Moab Accounting Manager by directly using the SSSRMAP Wire Protocol and Message Format over the network (see SSS Resource Management and Accounting documentation). This will entail building the request body in XML, appending an XML digital signature, combining these in an XML envelope framed in an HTTP POST, sending it to the server, and parsing the similarly formed response. The Moab Workload Manager communicates with Moab Accounting Manager via this method.

Example 24-16: The message might look something like:

```
POST /SSSRMAP HTTP/1.1
Content-Type: text/xml; charset="utf-8"
Transfer-Encoding: chunked

190
<?xml version="1.0" encoding="UTF-8"?>
<Envelope>
  <Body>
    <Request action="Charge" actor="scottmo">
    <Object>UsageRecord</Object>
      <Data>
        <UsageRecord>
          <Instance>Moab.1234</Instance>
          <Account>chemistry</Account>
          <User>amyh</User>
          <Machine>colony</Machine>
          <Processors>2</Processors>
          <Duration>3600</Duration>
        </UsageRecord>
      </Data>
      <Option name="Duration">3600</Option>
    </Request>
  <//Body>
  <Signature>
```

```
     <DigestValue>azu4obZswzBt89OgATukBeLyt6Y=</DigestValue>
     <SignatureValue>YXE/C08XX3RX4PMU1bWju+5/E5M=</SignatureValue>
     <SecurityToken type="Symmetric"></SecurityToken>
   </Signature>
</Envelope>
0
```

Chapter 25: Configuration Files

Moab Accounting Manager uses four configuration files: one for the connection information (`mam-site.conf`), one for the server (`mam-server.conf`), one for the clients (`mam-client.conf`) and one for the graphical user interface (`mam-gui.conf`). For configuration parameters that have hard-coded defaults, the default value is specified within brackets.

After modifying configuration parameters used by the server (such as those in the site configuration or server configuration files), you must restart the mam-server for the new settings to take effect. Alternatively, for most parameters, you can force the server to reread its configuration by running `mam-server --reconfig` or by sending the HUP signal to the main server process.

25.1 Site Configuration

The site configuration file specifies the connection information for the current site such as the server host name, port, backup server, default security method and the symmetric key. Optionally, it may also have blocks that specify connection information for other sites. This file should be readable only by the accounting admin user.

Example 25-1: The following is an example `mam-site.conf` *file*

```
server.host = red-head1
backup.host = red-head2
server.port = 7071
token.type = Symmetric
token.value = pBaIapJqbfLd8NiyzTJefFXW

[white]
server.host = white-head1
server.port = 7071
token.value = Fl7wOkioUpyjdqJ8ckvWK_ta

[blue]
server.host = blue-head1
server.port = 7071
token.value = gVSeQ8Diz5O3pzj01y4inGWq
```

The following configuration parameters may be set in the site configuration file (`mam-site.conf`).

`backup.host` — The hostname of the backup server. Each site can have both a primary server and a hot-standby backup server. They should either point to the same database or separate instances of a replicated database. If backup.host is specified, clients will try communicating with the primary server first, and if the connection fails, they will try communicating with the backup server. Since both the primary and backup servers may run simultaneously, events are disabled for the backup server so they do not conflict with events triggered by the primary server.

`server.host` — The hostname of the primary server

`server.port` [7112] — The port that the server listens on

`token-type` [Symmetric] — Indicates the default security token type to be used in both authentication and encryption. Valid token types include `Password` and `Symmetric`. The default is `Symmetric`.

`token.value` — When using the `Symmetric` token type, `token.value` is the secret key. It is a base64-encoded symmetric key used between clients and the server for authentication and encryption.

25.2 Server Configuration

The following configuration parameters may be set in the server configuration file (`mam-server.conf`).

Parameter	Description
accounting.mode [strict-allocation]	The accounting mode can be one of `usage-tracking`, `notional-charging`, `fast-allocation`, or `strict-allocation`. If `usage-tracking` is specified, charges will simply result in the creation of usage records with no charge value. No charge will be calculated and allocations will not be debited. If `notional-charging` is specified, a charge will be calculated and recorded with the usage record, but allocations are not debited. If `fast-allocation` is specified, usage records will be updated with charge amounts and allocations will be debited, but liens will not be used to protect the allocation from simultaneous use. If `strict-allocation` is specified, usage records will be updated with charge amounts and allocations will be debited, and liens will be used to protect the allocation from simultaneous use.
currency.itemization [false]	Enables (`true`) or disables (`false`) the storing of itemized charges to the Charge table for charge transactions.
currency.precision [0]	Indicates the number of decimal places in the resource credit currency. For example, if you are will be dealing with an integer billable unit like processor-seconds, use 0 (which is the default). If you will be charging dollars and cents, then use 2. This parameter should be the same in the `mam-server.conf` and `mam-client.conf` files.
database.datasource [DBI:Pg:db-name=mam;host=localhost]	The Perl DBI data source name for the database you wish to connect to
database.password	The password to be used for the database connection (if any)
database.user	The username to be used for the database connection (if any)
event.scheduler [true]	Specifies whether the event scheduler is enabled (`true`) or not (`false`). ⚠ MAM relies on pre-configured events for refreshing stale allocations and notifications; disabling the event scheduler will prevent these updates from occurring.

Parameter	Description
event.pollinterval [5]	The period in minutes that the event scheduler uses to check and fire events. The poll interval must divide evenly into the number of minutes in a day (1440).
log4perl.appender.Log.filename	Used by `log4perl` to set the base name of the log file
log4perl.appender.Log.max	Used by `log4perl` to set the number of rolling backup logs
log4perl.appender.Log.size	Used by `log4perl` to set the size the log will grow to before it is rotated
log4perl.appender.Log.Threshold	Used by `log4perl` to set the debug level written to the log. The logging threshold can be one of TRACE, DEBUG, INFO, WARN, ERROR, and FATAL
log4perl.appender.Screen.Threshold	Used by `log4perl` to set the debug level written to the screen. The logging threshold can be one of TRACE, DEBUG, INFO, WARN, ERROR, and FATAL
notification.deliverymethod [store]	Specifies which delivery method is used by default if unspecified
notification.duration [1209600]	Defines how long in seconds that stored notifications persist before being automatically deleted. The default is two weeks.
response.chunksize [0]	Indicates the line length in the data response that will trigger message segmentation (or truncation). A value of 0 (zero) means unlimited, i.e. that the server will not truncate or segment large responses unless overridden by a chunksize specification in a client request. The response chunksize will be taken to be the smaller of the client and server chunksize settings.
security.authentication [true]	Indicates whether incoming message authentication is required
security.encryption [false]	Indicates whether incoming message encryption is required
super.user [root]	Removed beginning with the 9.1.0 release.
user.firstaccountdefault [true]	If set to true, the first account that a user is added to will become the default account for that user. This default value is true.

25.3 Client Configuration

The following configuration parameters may be set in the client configuration file (`mam-client.conf`):

Parameter	Description
account.show [Name,Active,Users,Organization,Description]	The default fields shown by `mam-list-accounts`.
accounting.mode [strict-allocation]	The accounting mode can be one of `usage-tracking`, `notional-charging`, `fast-allocation`, or `strict-allocation`. The value of this parameter may modify the default fields displayed by certain commands such as `mam-list-usagerecords`.
allocation.enforcedisdiscrete [true]	If enabled (the default), new allocations will be prevented from overlapping existing ones. This policy helps to improve clarity when reporting on allocation usage during a particular period.
allocation.show [Id,Fund,StartTime,EndTime,InitialDeposit,Allocated,CreditLimit,Remaining,PercentUsed]	The default fields shown by `mam-list-allocations`.
balance.show [Id,Name,Balance,Reserved,Effective,CreditLimit,Available]	The default fields shown by `mam-balance`.
currency.precision [0]	Indicates the number of decimal places in the credit currency. For example, if you will be dealing with integer billable units like processor-seconds, use 0 (which is the default). If you will be charging dollars and cents, then use 2. This parameter should be the same in the `mam-server.conf` and `mam-client.conf` files.

Parameter	Description
event.show	[Id,FireCommand,FireTime,ArmTime,RearmPeriod,EndTime,Notify,RearmOnFailure,FailureCommand,CatchUp,CreationTime,Description] -- The default fields shown by `mam-list-events`.
fund.show [Id,Name,Constraints,Allocated,Balance,DefaultDeposit,Description]	The default fields shown by `mam-list-funds`.
lien.show [Id,Instance,Amount,StartTime,EndTime,UsageRecord,Funds,Description]	The default fields shown by `mam-list-liens`.
log4perl.appender.Log.filename	Used by log4perl to set the base name of the log file.
log4perl.appender.Log.max	Used by log4perl to set the number of rolling backup logs.
log4perl.appender.Log.size	Used by log4perl to set the size the log will grow to before it is rotated.
log4perl.appender.Log.Threshold	Used by log4perl to set the debug level written to the log. The logging threshold can be one of TRACE, DEBUG, INFO, WARN, ERROR, and FATAL.
log4perl.appender.Screen.Threshold	Used by log4perl to set the debug level written to the screen. The logging threshold can be one of TRACE, DEBUG, INFO, WARN, ERROR, and FATAL.
notification.show	[Id,Event,Type,Status,Code,Message,Key,Recipient,EndTime,CreationTime] --The default fields shown by `mam-list-notifications`.

Parameter	Description
organization.show [Name,Description]	The default fields shown by *mam-list-organizations*.
quote.show [Id,A-moun-t,Pinned,In-stance,UsageRe-cord,StartTime,EndTime,Duration,ChargeRates,Description]	The default fields shown by *mam-list-quotes*.
response.chunking [false]	Indicates whether large responses should be chunked (segmented). If set to false, large responses will be truncated.
response.chunksize [1000]	Indicates the line length in the data response that will trigger message segmentation (or truncation). A value of 0 (zero) means unlimited, i.e., that the client will accept the chunksize set by the server. The response chunksize will be taken to be the smaller of the client and server chunksize settings.
security.authentication [true]	Indicates whether outgoing message are signed.
security.encryption [false]	Indicates whether outgoing messages are encrypted

Parameter	Description
security.promotion [suidperl]	When using the symmetric key for security authentication or encryption, since the site configuration file is readable only by the accounting admin user, a method must be employed to temporarily elevate privileges in order to encrypt the communication with the symmetric key. One of two security promotion methods may be selected: `suidperl` or `mamauth`. Suiperl allows a Perl script to temporarily elevate privileges to the owner of the script if the setuid bit is set on the file. This method is recommended when suidperl can be installed on the system. If you prefer not to use suidperl or if it is not available for your system (such as with Perl 5.12 and higher), you will need to use the mamauth security promotion method. mamauth is a setuid binary that allows the request body to be passed in as standard input and returns the authenticated digest and signature. Currently, only suidperl can be used for encryption of client communication. The security promotion method should be configured at install time by specifying the `--with-promotion` configuration parameter and defaults to suidperl when it is available.
statement.show [Account,User,Machine]	The default discriminator fields in `mam-statement`.
transaction.show [Id,Object,Action,Actor,Name,Child,Instance,Count,Amount,Delta,Balance,User,Account,Machine,Fund,Allocation,UsageRecord,Duration,Description]	The default fields shown by `mam-list-transactions`.

Parameter	Description
usagerecord.show [Id,Type,Instance,Charge,Stage,User-,Group,Account,Organization,Class,QualityOfService,Machine,Nodes,Processors,Memory,Duration,SubmitTime,StartTime,EndTime,Description]	The default fields shown by *mam-list-usagerecords*.
user.show [Name,Active,CommonName,PhoneNumber,EmailAddress,DefaultAccount,Description]	The default fields shown by *mam-list-users*.

25.4 GUI Configuration

The following configuration parameters may be set in the GUI configuration file (`mam-gui.conf`).

> ℹ️ gui.style [viewpoint] parameter is obsolete and has been removed. See the *Moab - Enterprise Edition Release Notes 9.0* for more information.

Parameter	Description
currency.enablehours [false]	If set to true, the graphical user interface will include a ShowHours radio button (defaulting to `True`) for certain panels (e.g. Fund Deposit, Query, Statement, Transfer, Withdraw) that will allow the currency inputs or outputs to be divided by 3600.
currency.precision [0]	Indicates the number of decimal places in the credit currency. For example, if you will be dealing with integer billable units like processor-seconds, use 0 (which is the default). If you will be charging dollars and cents, then use 2. This parameter should be the same in the `mam-server.conf` and `mam-client.conf` files.
log4perl.appender.Log.filename	Used by log4perl to set the base name of the log file
log4perl.appender.Log.max	Used by log4perl to set the number of rolling backup logs
log4perl.appender.Log.size	Used by log4perl to set the size the log will grow to before it is rotated
log4perl.appender.Log.Threshold	Used by log4perl to set the debug level written to the log. The logging threshold can be one of TRACE, DEBUG, INFO, WARN, ERROR, and FATAL.
response.chunking [false]	Indicates whether large responses should be chunked (segmented). If set to false, large responses will be truncated.
response.chunksize [1000]	Indicates the line length in the data response that will trigger message segmentation (or truncation). A value of 0 (zero) means unlimited, i.e. that the client will accept the chunksize set by the server. The response chunksize will be taken to be the smaller of the client and server chunksize settings.
security.authentication [true]	Indicates whether outgoing message are signed
security.encryption [false]	Indicates whether outgoing messages are encrypted

Parameter	Description
security.promotion [suidperl]	When using the symmetric key for security authentication or encryption, since the site configuration file is readable only by the accounting admin user, a method must be employed to temporarily elevate privileges in order to encrypt the communication with the symmetric key. One of two security promotion methods may be selected: `suidperl` or `mamauth`. Suidperl allows a Perl script to temporarily elevate privileges to the owner of the script if the setuid bit is set on the file. This method is recommended when suidperl can be installed on the system. If you prefer not to use suidperl, or if it is not available for your system (such as with Perl 5.12 and higher), you will need to use the mamauth security promotion method. mamauth is a setuid binary that allows the request body to be passed in as standard input and returns the authenticated digest and signature. Currently, only suidperl can be used for encryption of client communication. The security promotion method should be configured at install time by specifying the `--with-promotion` configuration parameter and defaults to suidperl when it is available.
statement.discriminators	The Fund Statement page will group summary entries in the debit detail by these transaction properties.

Chapter 26: Web Services

Moab Accounting Manager Web Services (MAMWS) provides a REST-like interface permitting access to the full Moab Accounting Manager API. MAMWS communicates with Moab Accounting Manager using the same wire protocol, message format, and Perl API as the MAM command-line clients and GUI interfaces. MAMWS runs under mod_perl from an apache httpd server.

> ⓘ Refer to the *Moab HPC Suite Installation and Configuration Guide* for instructions on installing and configuring MAM Web Services.

This chapter provides information about the Web Services API and contains specific examples of the accounting and the framework resources using that API.

26.1 Web Services API

MAMWS provides a web interface using REST (Representational State Transfer) concepts to create, query, modify, and delete objects in Moab Accounting Manager. MAMWS also supports additional actions and alternative syntax options for interacting with the web service.

This topic provides information on the Web Services API, including the format of the request and response syntax and authentication and error code details.

26.1.1 URL Format

- A MAMWS URL is composed of a resource URI and optional query string.

- The resource URI is composed of the prefix and a resource.

- The prefix is composed of the protocol (normally https), the MAM Web Services httpd server hostname or IP address, the location (/mamws), and an optional API version.

- The resource corresponds with MAM objects and instances of those objects. Thus, a MAMWS resource is composed of a MAM object optionally followed by one or more primary keys.

- The resource's object is specified in kebab-case and is normally pluralized. For example, /usage-records represents the UsageRecord object in MAM, while /usage-records/1 represents the instance of the UsageRecord object having the value 1 as the primary key. HTTP parameters and data are used as syntactical parameters and options for the API queries.

For example:

```
<mamws_url> ::= <mamws_uri>[<query_string>]
<mamws_uri> ::= <mamws_prefix><mamws_resource>
<mamws_prefix> ::= <protocol>://<mamws_server>/mamws[/<version>]
<mamws_resource> ::= <mam_object>[/<primary_key>...]
<query_string> := ?<parameter>[&<parameter>...]
```

An expanded URL is of the form:

```
<protocol>://<mamws_server>/mamws[/<version>]/<object>[/<primary_key>...][?<parameter>
[&<parameter>...]]
```

26.1.2 HTTP Methods

MAMWS supports the use of REST concepts utilizing HTTP (Hypertext Transfer Protocol) methods operating on endpoint URLs that describe resources.

The following table describes the HTTP methods used in MAMWS.

Method	Path Info	Description
GET	/<object>	Query for a list of resources.
GET	/<object>/<primary_key>…	Query a single resource.
POST	/<object	Create a resource (primary key(s) not included in path).
PUT	/<object>/<primary_key>…	Create a resource (primary key(s) included in path).
PATCH	/<object>/<primary_key>…	Modify a resource.
DELETE	/<object>/<primary_key>…	Delete a resource.
POST	/<object>?action=<action>	Other actions.

26.1.3 JSON Data Format

When HTTP data is included in the HTTP request or response, it is encoded in JSON object format.

- Input data for a POST or PATCH must be in JSON format with the top-level data type being a JSON object. The Content-Type header should be set to 'application/json'.

- Output data is always in JSON format and always consists of a JSON object with two or more key/value pairs. The output is "pretty-printed" by default.

Sample Request Data

```
POST /users
{
  "active" : true,
  "common-name" : "Amy Miller",
  "default-account" : "chemistry",
  "email-address" : "amy@hpc.com",
  "name" : "amy",
  "phone-number" : "(801) 555-1437"
}
```

Sample Response Data

```
GET /users/amy
{
  "code" : "000",
  "count" : 1,
  "data" : [
    {
      "active" : true,
      "common-name" : "Amy Miller",
      "default-account" : "chemistry",
      "description" : null,
      "email-address" : "amy@hpc.com",
      "name" : "amy:,
      "phone-number" : "(801) 555-1437"
    }
  ],
  "status" : "Success"
}
```

26.1.4 API Version

The Web Services API is supports versioning.

The version is optional and, when used, is appended to the prefix (effectively prepended to the resource) in the URL (i.e. https://<mamws_server>/mamws[/<version>]/<object>[/<primary_key>...][?<parameter>[&<parameter>...]]). For example:

```
GET https://localhost/mamws/v1/users
```

If you omit the version in the URL, the web services client will use the *current* version. If an invalid version is specified, the request will fail.

> ℹ With the MAM 9.1.0 release, the Web Services API is version 1 (v1) and comes with versioning support for future enhancements and possible compatibly-breaking changes.

26.1.5 Request Format

A MAMWS Request includes the object (or instance which consists of the object and primary keys) and the action (whether explicit or implied), and may provisionally include selections, assignments, conditions, options, data, and meta-options.

In this section:

26.1.5.A Object

The request object is specified in the URL path info. Some actions or methods additionally require or allow primary keys to be specified as additional path elements in the URL to specify the object instance.

> ℹ The values of the filters (object and primary-key) must be specified in UpperCamel case; the web service interface does *not* translate the case for values.

Examples

Specify the User object in a query (query all users).

```
GET /users
```

Specify the instance of the User object having the primary key 'amy' (query just the user amy).

```
GET /users/amy
```

List valid objects.

```
GET /objects?fields=name
```

List primary keys for the usage record object in sequential order.

```
GET /attributes?filter=object=UsageRecord,primary-key=True&fields=name&sort-by-sequence
```

List all attributes for the usage record object.

```
GET /attributes?filter=object=UsageRecord&fields=sort(name)
```

26.1.5.B Action

The request action may be specified via the action parameter. When not specified via the parameter, the action will be implied from the HTTP method as follows:

- The GET method implies the Query action.

- The PUT and POST methods imply the Create action.

- The PATCH method implies the Modify action.

- And the DELETE method implies the Delete action.

> ℹ The POST method will permit any supported action to be explicitly specified via the action parameter; all other methods are restricted to their default action.

Examples

Delete action implied by the DELETE method.

```
DELETE /users/amy
```

Refund action explicitly specified via the action parameter.

```
POST /usage-records/1?action=refund&id=1
```

List all actions available to the usage record object.

```
GET /actions?filter=object=UsageRecord&fields=sort(name)
```

26.1.5.C Other Request Components

Other request components may be specified via parameters in the query string or via JSON data.

The following table describes the parameter information for the other components.

Parameter Type	Description	Examples
Selections	Designate which properties of an object are returned in a query.	`fields=name,active`

Assignments	Specify new field values when creating and modifying objects.	`update=active=true`
		`active@=true`
		`{`
		`  "active" : true`
		`}`

Conditions	Specify which objects to query, update or delete.	`fields=active=true` `active==true`

Options	Specify additional business-logic parameters.	`options=show-`
		`hidden=true`
		`show-hidden:=true`
		`show-hidden=true`

Parameter Type	Description	Examples

Data	Although not a parameter type, JSON data may be used with some actions as assignment properties or as input data.	`{`
		`   "processors" : 2,`
		`   "account" :`
		`"chemistry"`
		`}`

Meta-Options	Options used to by the web services client and not forwarded in the MAM request.	`pretty=false` `!pretty`

The following table describes the actions and the supported parameter type for the other components.

> ⓘ The use of square brackets means this parameter type is optional.

Actions	Supported Parameter Type
Query	[Selections], [Conditions], [Options], [Meta-Options]
Create	Assignments*, [Options], [Meta-Options]
Modify	Assignments*, [Conditions], [Options], [Meta-Options]
Delete, Undelete	[Conditions], [Options], [Meta-Options]
Other actions	[Options], [Data], [Meta-Options]

* For these actions, Assignments may alternatively be specified in the JSON data.

26.1.5.D Selections

Selections designate the fields that are to be returned in a query. Besides simple field selection, selection criteria may also include sorting, extraction of partial values from complex data types, aliases, and aggregation (sum, average, min, max, etc).

Selections are expressed as a comma-separated list of desired object properties as the value of the fields parameter in the following form:

```
fields=[<aggregation_function>(]<name>[{<part>}][)][=<alias>],...
```

The following table describes the selection parameter components.

Selection Parameter Component	Description	Examples

aggregation_function	Designates sorting or an aggregation function to apply to the field. Valid values:
	
	• sum (sum of values)
	• average (average of values)
	• min (minimum value)
	• max (maximum value)
	• count (count of values)
	• group-by (group-by field)
	• sort (descending sort)
	• tros (increasing sort)
		`fields=sum(amount),group-by`
		`(account)`
		`fields=sort(name)`

name	Name of the field or object property to display or use in an aggregation.	`fields=name,email-address`
part	Name of the part to extract from the complex object property.	`fields=resources{telescope}`
alias	Designates what to call the returned property or aggregation value.	`fields=name=user` `fields=sum(amount)=total`

> ⓘ Aliases for the fields parameter include select, show, and get.

26.1.5.E Assignments

Assignments designate the new values in the creation *or* modification of objects. Besides simple assignment, assignments may alternatively increment or decrement the value.

Assignments may be expressed in one of three different ways:

- using the update parameter
- directly with the property name as the parameter name with an assignment operator
- as a JSON object in the request data.

Expressed Using the Update Parameter

When using the update parameter, assignments are expressed as a comma-separated list of update expressions in the following form:

```
update=<name><op><value>,...
```

The following table describes the assignment parameter components when expressed using the update parameter.

Assignment Parameter Component	Description	Examples
name	Name of the object property to set or update.	```update=name=amy,active=true```
op	Designates whether the specified value should be assigned to the property, or used to increment or decrement it: • = (assignment) • += (increment) • -= (decrement)	```update=duration+=3600```
value	Designates the value to assign as the new value of the property or the amount to increment or decrement it. Use null to unset the object property.	```update=email-address=null```

> 🛈 Aliases for the update parameter include assign and set.

Expressed Directly

Assignments may be expressed directly with the property name as the parameter name with an assignment operator in the form:

```
<name><op><value>
```

The following table describes the assignment operator components when expressed directly.

Assignment Operator Component	Description	Examples
name	Name of the object property to set or update.	```active@=true```

Assignment Operator Component	Description	Examples
op	Designates whether the specified value should be assigned to the property, or used to increment or decrement it: • @= (assignment) • += (increment) • -= (decrement)	`duration+=3600`
value	Designates the value to assign as the new value of the property or the amount to increment or decrement it. Use null to unset the object property.	`email-address@=null`

Expressed as a JSON Object in the Request Data

The properties to be assigned may be expressed as a JSON object in the HTTP request data in the form:

```
{
   <name> : <value>,...
}
```

ℹ This form *cannot* be used to increment or decrement the object property.

The following table describes the assignment data components when expressed as a JSON object.

Assignment Data Component	Description	Examples

name	Name of the object property to set or update.	`{`
		`   "name" : "amy"`
		`   "active" : true`
		`}`

value	Designates the value to assign as the new value of the property or the amount to increment or decrement it. Use null to unset the object property.	`{`
		`   "email-address" :`
		`null`
		`}`

26.1.5.F Conditions

Conditions allow filtering of the objects to be queried, updated, or acted upon. Besides simple equality conditions, condition criteria may include filtering on part names of a complex value,

comparisons (greater-than, not equal, etc), pattern matching, conjunctions (and, or), and grouping.

Conditions may be expressed in one of two different ways:

- using the filter parameter
- directly with the property name as the parameter name with a condition operator

Expressed Using the Filter Parameter

When using the filter parameter, conditions are expressed as a list of filter expressions (delimited with the respective conjunction symbol) in the following form:

```
filter=[<pre-group>]<name>[{<part>}]<op><value>[<post-group>]<conjunction>...
```

The following table describes the condition parameter components when expressed using the filter parameter.

Condition Parameter Component	Description	Examples
pre-group	Zero or more open parentheses used for grouping of ANDed and ORed conditions.	`filter=` `(instance~j1|charge>10),id<5&fields=id`
name	Name of the object property used in determining which objects to include in the query or update.	`filter=active=true`
part	Designates to only include objects having an individual named part with the specified value.	`filter=resources{telescope}==2`

Condition Parameter Component	Description	Examples
| op | Comparison or matching operator employed to determine whether objects having the specified name are included in the operation:<br><br>• == or = (equality)<br>• > (greater than)*<br>• >= (greater than or equal to)*<br>• < (less than)*<br>• <= (less than or equal to)*<br>• != (not equal to)<br>• ~ (matches)**<br>• !~ (does not match)**<br><br>* These operators are *only* valid with attributes having numeric data types (e.g. AutoGen, Currency, Float, Integer, TimeStamp).<br><br>** These operators are *only* valid with attributes having string data types (e.g. String).<br><br>The following wildcards are supported with matching operators:<br><br>• ? (matches any one character)<br>• * (matches zero or more of any characters) | ```
filter=processors>=4
filter=account~chem*
``` |
| value | Value of the specified object property. Use null to include objects whose specified property is unset. | ```
filter=email-address==null
``` |
| post-group | Zero or more open parentheses used for grouping ANDed and ORed conditions. | ```
filter=instance~j1|
(charge>10,id<5)&fields=id
``` |

Condition Parameter Component	Description	Examples
conjunction	Symbol used to connect condition groups indicating whether the current and preceding condition group should be ANDed or ORed: • , (and) • | (or)	``` filter=user==amy,account==chemistry filter=account==chemistry|account==biology ```

> **ℹ** Aliases for the filter parameter include query and where.

Expressed Directly

Conditions may be expressed directly with the property name as the parameter name with a condition operator in the form:

```
<name><op><value>
```

> **ℹ** This form *cannot* be used to specify parts, conjunctions, or grouping.

The following table describes the condition operator components when expressed directly.

Condition Operator Component	Description	Examples
name	Name of the object property used to filter objects included in the operation.	``` active==true ```

Condition Operator Component	Description	Examples
op	Comparison or matching operator employed to determine whether objects having the specified name are included in the operation: • == (equality) • > (greater than)* • >= (greater than or equal to)* • < (less than)* • <= (less than or equal to)* • != (not equal to) • ~ (matches)** • !~ (does not match)** * These operators are *only* valid with attributes having numeric data types (e.g. AutoGen, Currency, Float, Integer, TimeStamp). ** These operators are *only* valid with attributes having string data types (e.g. String). The following wildcards are supported with matching operators: • ? (matches any one character) • * (matches zero or more of any characters)	`processors>=4` `account~chem*`
value	Value of the specified object property. Use null to include objects whose specified property is unset.	`email-` `address==null`

26.1.5.G Options

Options specify additional business-logic options the may affect the behavior of the request or resulting response.

Options may be expressed in one of three different ways:

• using the options parameter

• directly with the option name as the parameter name with an option operator

• using the meta-option operator (=) where there is no similarly-named meta-option

Expressed using the Options Parameter

When using the options parameter, options are expressed as a list of comma-delimited option expressions in the following form:

```
options=<name>=<value>,...
```

The following table describes the option parameter components when expressed using the options parameter.

Option Parameter Component	Description	Examples
name	Name of the option.	`options=job-id=2,amount=1.5`
value	Value of the option.	`options=show-hidden=true`

Expressed Directly

Options may be expressed directly with the option name as the parameter name with an option operator in the form:

```
<name><op><value>
```

The following table describes the option operator components when expressed directly.

Option Operator Component	Description	Examples
name	Name of the option.	`active:=true`
op	Option operator: • := (assertion)	`filter-type:=NonExclusive`
value	Value of the option. As a shorthand notation for a boolean value of true, the operator and the value can be omitted. As a shorthand notation for a boolean value of false, the name can be preceded by an exclamation point (!) and the operator and value omitted.	`active` `!active`

Expressed Using the Meta-Option Operator

Parameters of the form <name>=<value> that are *not* interpreted as meta-options will be taken as request options. However, when using this form, care must be taken to avoid conflict with the meta-options.

For example, filter=User=amy should not be used to express the Filter request option with the value User=amy, since this expression would be interpreted as specifying the filter meta-option for the User condition with value amy. In this case, you would need to either use the constraint-filter

meta-option (constraint-filter=User=amy), the options meta-option (options=filter=User=amy) or the option operator (filter:=User=amy).

26.1.5.H Data

Some actions require input data with the request (e.g. Charge, Reserve and Quote require a usage record as input data). Other actions, such as Create and Modify, allow the newly created or updated fields to be passed in via the data as an alternate form of expressing the assignment fields.

In all cases, data is expressed as a JSON object in the following form:

```
{
   <name> : <value>,...
}
```

The following table describes the request data components.

Request Data Component	Description	Examples
name	Name of the object property.	```{ "name" : "amy", "active" : true }```
value	Value of the object property. In some cases, the value itself can be a simple JSON object (e.g., complex usage record fields).	```{ "class" : null, "amount" : 12.5, "resources" : { "telescope" : 2 } }```

26.1.5.I Meta-Options

Meta-options are HTTP parameters used by the web services client and not forwarded in the request to Moab Accounting Manager. Meta-options include fields, update, filter, options. It also includes their respective aliases as described in previous sections of this topic.

The following table describes the supplemental meta-options.

Meta-Option Name	Function	Examples
force	In some situations, asserting the force parameter may allow an action to do something potentially dangerous or bend RESTful rules, such as allowing the PATCH or DELETE methods to operate on multiple instances.	`force=true` `force`
pretty	Pretty-printing is enabled by default. To disable it, de-assert the pretty parameter.	`pretty=false` `!pretty`
suppress-nulls	When rendering the response data in JSON, null-valued fields are explicitly shown as having the null value by default. Asserting the suppress-nulls parameter will avoid printing fields having a null value.	`suppress-nulls=true` `suppress-nulls`

26.1.6 Response Format

A MAMWS Response has an HTTP status code and HTTP data. The HTTP data is in the form of a JSON object with key value pairs that includes a MAM status and code (different from the HTTP status code), and may optionally include a message, a count, and JSON data. The MAM response is expressed in the HTTP response data as a JSON object of the following form:

```
{
   "code" : <code>,
   ["count" : <count>,]
   ["data" : <data>,]
   ["message" : <message>,]
   "status" : <status>
}
```

The following table describes the response data components.

Response Data Component	Description	Examples
code	MAM SSSRMAP* Status Code.	`"code" : "740"`
count	Usually the number of objects returned or affected; but sometimes is used to return other key values such as amount charged.	`"count" : 24`

Response Data Component	Description	Examples
data	Response data as a JSON object.	```"data" : [{ "name" : "amy", "active" : true }, { "name" : "bob", "active" : false }]```
message	Response message.	```"message" : "Successfully modified 2 users"```
status	Status: • Success • Warning • Failure	```"status" : "Failure"```

* SSSRMAP stands for Scalable Systems Software Resource Manager and Accounting Protocol

26.1.6.A HTTP Codes

The following table describes the HTTP codes that may be returned with the HTTP response.

HTTP Status Code	Description	When Used
200	OK	Successful response received from MAM server.
400	Bad Request	Invalid request on the client side or any business-logic or miscellaneous problem that the server could not successfully fulfill.
401	Unauthorized	User did not successfully authenticate.
403	Forbidden	User is not authorized to perform the request.
404	Not Found	The specified resource does not exist.

HTTP Status Code	Description	When Used
405	Method Not Allowed	The HTTP method is not used in the API.

26.1.6.B Status Codes

MAMWS uses 3 digit SSSRMAP status codes in the JSON response object.

Refer to "SSSRMAP Message Format 3.0.4" at http://www.adaptivecomputing.com/products/open-source/gold/gold- product-documentation/ for the list and meanings of these codes.

26.1.7 Authentication

MAMWS uses HTTP Basic Authentication for all REST API requests. The required username and password is forwarded to the MAM server for authentication and authorization. Thus, each user that wants to be able to use MAM Web Services must first set a password in MAM (e.g with the mam-set-password client command).

The username and password in the Basic Authentication header are encoded but *not* encrypted. It is *strongly* recommended that MAMWS be run under an httpd server with SSL enabled.

Related Topics

26.2 MAM Actions Mapping

This topic provides an aid in mapping MAM actions to HTTP methods and resources in MAM Web Services.

26.2.1 Query Action

Use the GET method to query an object. In MAM, there is no fundamental difference between querying a single instance of an object *or* multiple instances of the object. Querying a single object simply includes a query filter using the object's primary keys. With REST, these are differentiated by the presence of additional path info nodes in the request URL.

The following table describes the methods and resources used for the Query action.

HTTP Method	MAMWS Resource	Description	Example
GET	/<object>	Query multiple objects.	GET /users
GET	/<object>{/<primary_key>...}	Query a single object.	GET /users/amy

In a MAMWS query response, the selected object properties are returned in the JSON data field as an array of objects. This is true both when querying in the single object form or in the multiple object form, and is done this way so that a client can use the same parsing routine for both cases.

26.2.2 Create Action

Use the POST method or the PUT method to create resources (objects) in MAMWS.

- When using POST, the resource URI should *not* include the primary keys with the object in the path info.

- When using PUT, the resource URI must include the primary keys with the object in the path info. Thus, PUT may only be used when you know the primary keys that will uniquely define the object instance being created.

The POST method is considered the primary method since it is considered more straightforward to put all of the new object properties in a single location (the request data).

The following table describes the methods and resources used for the Create action.

HTTP Method	MAMWS Resource	Description	Example
POST	/<object>	Create an object (primary key(s) not included in path).	```POST /users { "name" : "amy" "active" : true }```
PUT	/<object>{/<primary_key>…}	Create an object (primary key(s) included in path).	```PUT /users/amy { "active" : true }```

26.2.3 Modify Action

Use the PATCH method to modify an object.

The following table describes the methods and resources used for the Modify action.

HTTP Method	MAMWS Resource	Description	Example
PATCH	/<object>{/<primary_key>…}	Modify an object.	```PATCH /users/amy { "active" : false }```

26.2.4 Delete Action

Use the DELETE method to delete an object.

The following table describes the methods and resources used for the Delete action.

HTTP Method	MAMWS Resource	Description	Example
DELETE	/<object>{/<primary_key>…}	Delete an object.	```DELETE /users/amy```

26.2.5 Other Actions

All other actions are implemented using the POST method and using the action parameter.

The following table describes the methods and resources used for all other actions.

HTTP Method	MAMWS Resource	Description	Example
POST	/<object>	Perform an action against an object.	```POST /users?action=undelete&filter=name=amy```

Related Topics

-
-

26.3 Accounting Resources

This section provides information on available MAMWS accounting resources.

26.4 Accounts Resource

This section provides information on the supported actions for the Accounts accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query accounts	GET	/accounts[/<name>]
Create an account	POST	/accounts
Modify an account	PATCH	/accounts/<name>
Delete an account	DELETE	/accounts/<name>
Query account users	GET	/account-users[/<account>[/<user>]]
Add a user to an account	POST	/account-users
Modify an account user	PATCH	/account-users/<account>/<user>
Remove a user from an account	DELETE	/account-users/<account>/<user>

Related Topics

- 26.3 Accounting Resources - page 299

26.5 Query Accounts

26.5.1 Synopsis

```
GET /accounts[/<name>][?<parameter>[&<parameter>...]]
```

26.5.2 Parameters

Parameter	Description	Example
constraint-filter	Applies meta-filters to the query (user: include only accounts having the specified user)	`GET /accounts?constraint-filter=user=amy`
fields	Designates the properties to be returned in the query	`GET /accounts?fields=name`
filter	Filters the objects to be returned in the query	`GET /accounts?filter=organization=sciences`
limit	Limits the results to the number of objects specified	`GET /accounts?limit=100`
offset	Number of objects to skip before starting to return data	`GET /accounts?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /accounts?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /accounts?fields=organization&unique=true`

26.5.3 Sample Request

```
GET /accounts/amy
```

26.5.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "description" : "Chemistry Department",
            "name" : "chemistry",
            "organization" : "sciences"
        }
    ],
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300
- 26.3 Accounting Resources - page 299

26.6 Create an Account

26.6.1 Synopsis

```
POST /accounts[?<parameter>]
{
    <name> : <value>,...
}
```

26.6.2 Parameters

Parameter	Description	Example

create-fund	Overrides the fund auto-generation setting	`POST /accounts?create-fund=true`
		`{`
		`    "name" : "chemistry"`
		`}`

26.6.3 Sample Request

```
POST /accounts
{
    "description" : "Chemistry Department",
    "name" : "chemistry",
    "organization" : "sciences"
}
```

26.6.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
```

```
            "active" : true,
            "description" : "Chemistry Department",
            "name" : "chemistry",
            "organization" : "sciences"
        }
    ],
    "message" : "Successfully created 1 account",
    "status" : "Success"
}
```

Related Topics

26.7 Modify an Account

26.7.1 Synopsis

```
PATCH /accounts/<name>
{
    <name> : <value>,...
}
```

26.7.2 Sample Request

```
PATCH /accounts/chemistry
{
    "active" : false
}
```

26.7.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : false,
            "description" : "Chemistry Department",
            "name" : "chemistry",
            "organization" : "sciences"
        }
    ],
    "message" : "Successfully modified 1 account",
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300

- 26.3 Accounting Resources - page 299

26.8 Delete an Account

In this topic:

26.8.1 Synopsis

```
DELETE /accounts/<name>
```

26.8.2 Sample Request

```
DELETE /accounts/chemistry
```

26.8.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "description" : "Chemistry Department",
            "name" : "chemistry",
            "organization" : "sciences"
        }
    ],
    "message" : "Successfully deleted 1 account",
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300

- 26.3 Accounting Resources - page 299

26.9 Query Account Users

26.9.1 Synopsis

```
GET /account-users[/<account>[/<user>]][?<parameter>[&<parameter>...]]
```

26.9.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /account-users/chemistry?fields=name`
filter	Filters the objects to be returned in the query	`GET /account-users?filter=name=amy`
limit	Limits the results to the number of objects specified	`GET /account-users?limit=100`
offset	Number of objects to skip before starting to return data	`GET /account-users?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /account-users?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /account-users?fields=name&unique=true`

26.9.3 Sample Request

```
GET /account-users/chemistry?fields=name
```

26.9.4 Sample Response

```
{
    "code" : "000",
    "count" : 2,
    "data" : [
        {
            "name" : "amy"
        },
        {
            "name" : "dave"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.10 Add a User to an Account

In this topic:

26.10.1 Synopsis

```
POST /account-users
{
    <name> : <value>,...
}
```

26.10.2 Sample Request

```
POST /account-users
{
    "account" : "chemistry",
    "active" : true,
    "admin" : true,
    "name" : "amy"
}
```

26.10.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "active" : true,
            "admin" : true,
            "name" : "amy"
        }
    ],
    "message" : "Successfully created 1 account user",
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300

- 26.3 Accounting Resources - page 299

26.11 Modify an Account User

26.11.1 Synopsis

```
PATCH /account-users/<account>/<user>
{
    <name> : <value>,...
}
```

26.11.2 Sample Request

```
PATCH /account-users/chemistry/amy
{
    "active" : false
}
```

26.11.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "active" : false,
            "admin" : true,
            "name" : "amy"
        }
    ],
    "message" : "Successfully modified 1 account user",
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300

- 26.3 Accounting Resources - page 299

26.12 Remove a User from an Account

In this topic:

26.12.1 Synopsis

```
DELETE /account-users/<account>/<user>
```

26.12.2 Sample Request

```
DELETE /accounts-users/chemistry/amy
```

26.12.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "active" : false,
            "admin" : true,
            "name" : "amy"
        }
    ],
    "message" : "Successfully deleted 1 account user",
    "status" : "Success"
}
```

Related Topics

- 26.4 Accounts Resource - page 300
- 26.3 Accounting Resources - page 299

26.13 Allocations Resource

This section provides information on the supported actions for the Allocations accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query allocations	GET	/allocations[/<id>]
Modify an allocation	PATCH	/allocations/<id>
Delete an allocation	DELETE	/allocations/<id>

Related Topics

26.14 Query Allocations

26.14.1 Synopsis

```
GET /allocations[/<id>][?<parameter>[&<parameter>...]]
```

26.14.2 Parameters

Parameter	Description	Example
constraint-filter	Displays allocations whose fund constraints comply with the specified filters	```GET /allocations?constraint-filter=user=amy```
fields	Designates the properties to be returned in the query	```GET /allocations?fields=id,amount```
filter	Filters the objects to be returned in the query	```GET /allocations?filter=active=true```
filter-type	Designates the constraint filter type	```GET /allocations?constraint-filter=user=amy&filter-type=ExactMatch```

Parameter	Description	Example
limit	Limits the results to the number of objects specified	```GET /allocations?limit=100```
offset	Number of objects to skip before starting to return data	```GET /allocations?offset=100```
show-hidden	Includes hidden attributes in the result	```GET /allocations?show-hidden=true```
unique	Displays only unique results (like DISTINCT in SQL)	```GET /allocations?fields=fund&unique=true```

26.14.3 Sample Request

```
GET /allocations/2
```

26.14.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "allocated" : 3000,
            "amount" : 3000,
            "credit-limit" : 0,
            "description" : null,
            "end-time" : "Infinity",
            "fund" : 2,
            "id" : 2,
            "initial-deposit" : 3000,
            "start-time" : "2018-06-15 18:29:44"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.15 Modify an Allocation

In this topic:

26.15.1 Synopsis

```
PATCH /allocations/<id>
{
    <name> : <value>,...
}
```

26.15.2 Sample Request

```
PATCH /allocations/2
{
    "credit-limit" : 1000
}
```

26.15.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "allocated" : 3000,
            "amount" : 3000,
            "credit-limit" : 1000,
            "description" : null,
            "end-time" : "Infinity",
            "fund" : 2,
            "id" : 2,
            "initial-deposit" : 3000,
            "start-time" : "2018-06-15 18:29:44"
        }
    ],
    "message" : "Successfully modified 1 allocation",
    "status" : "Success"
}
```

Related Topics

26.16 Delete an Allocation

In this topic:

26.16.1 Synopsis

```
DELETE /allocations/<id>
```

26.16.2 Sample Request

```
DELETE /allocations/2
```

26.16.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "allocated" : 3000,
            "amount" : 3000,
            "credit-limit" : 0,
            "description" : null,
            "end-time" : "Infinity",
            "fund" : 2,
            "id" : 2,
            "initial-deposit" : 3000,
            "start-time" : "2018-06-15 18:29:44"
        }
    ],
    "message" : "Successfully deleted 1 allocation",
    "status" : "Success"
}
```

Related Topics

- 26.13 Allocations Resource - page 312

- 26.3 Accounting Resources - page 299

26.17 Charges Resource

This section provides information on the supported actions for the Charges accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query itemized charges	GET	/charges

Related Topics

- 26.3 Accounting Resources - page 299

26.18 Query Itemized Charges

26.18.1 Synopsis

```
GET /charges[?<parameter>[&<parameter>...]]
```

26.18.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /charges?fields=sum(amount)`
filter	Filters the objects to be returned in the query	`GET /charges?filter=usage-record=1`
limit	Limits the results to the number of objects specified	`GET /charges?limit=100`
offset	Number of objects to skip before starting to return data	`GET /charges?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /charges?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /charges?fields=name&unique=true`

26.18.3 Sample Request

```
GET /charges?filter=usage-record=1
```

26.18.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : 1,
            "description" : null,
            "details" : "12 [Processors] * 0.000277777777777778 [ChargeRate{Processors}]
 * 300 [Duration]",
            "duration" : 300,
            "instance" : "24809",
            "name" : "Processors",
            "rate" : "1/h",
            "scaling-factor" : 1,
            "usage-record" : 1,
            "value" : "12"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.19 Charge Rates Resource

This section provides information on the supported actions for the Charge Rates accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query charge rates	GET	/charge-rates[/<name>[/<value>]]
Create a charge rate	POST	/charge-rates
Modify a charge rate	PATCH	/charge-rates/<name>/<value>
Delete a charge rate	DELETE	/charge-rates/<name>/<value>

Related Topics

-

26.20 Query Charge Rates

26.20.1 Synopsis

```
GET /charge-rates[/<name>[/<value>]][?<parameter>[&<parameter>...]]
```

26.20.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /charge-rates?fields=name`
filter	Filters the objects to be returned in the query	`GET /charge-rates?filter=name=Processors`
limit	Limits the results to the number of objects specified	`GET /charge-rates?limit=100`
offset	Number of objects to skip before starting to return data	`GET /charge-rates?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /charge-rates?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /charge-rates?fields=name&unique=true`

26.20.3 Sample Request

```
GET /charge-rates
```

26.20.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : "1/h",
            "description" : "1 credit per processor-hour",
            "name" : "Processors",
            "value" : null
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.21 Create a Charge Rate

26.21.1 Synopsis

```
POST /charge-rates
{
    <name> : <value>,...
}
```

26.21.2 Sample Request

```
POST /charge-rates
{
    "amount" : "1/h",
    "description" : "1 credit per processor-hour",
    "name" : "Processors"
}
```

26.21.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : "1/h",
            "description" : "1 credit per processor-hour",
            "name" : "Processors",
            "value" : null
        }
    ],
    "message" : "Successfully created 1 charge rate",
    "status" : "Success"
}
```

Related Topics

- 26.19 Charge Rates Resource - page 322
- 26.3 Accounting Resources - page 299

26.22 Modify a Charge Rate

26.22.1 Synopsis

```
PATCH /charge-rates/<name>/<value>
{
    <name> : <value>,...
}
```

26.22.2 Sample Request

```
PATCH /charge-rates/Processors/null
{
    "amount" : "2/h"
}
```

26.22.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : "2/h",
            "description" : "1 credit per processor-hour",
            "name" : "Processors",
            "value" : null
        }
    ],
    "message" : "Successfully modified 1 charge rate",
    "status" : "Success"
}
```

Related Topics

- 26.19 Charge Rates Resource - page 322

- 26.3 Accounting Resources - page 299

26.23 Delete a Charge Rate

In this topic:

26.23.1 Synopsis

```
DELETE /charge-rates/<name>/<value>
```

26.23.2 Sample Request

```
DELETE /charge-rates/Processors/null
```

26.23.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : "1/h",
            "description" : "1 credit per processor-hour",
            "name" : "Processors",
            "value" : null
        }
    ],
    "message" : "Successfully deleted 1 charge rate",
    "status" : "Success"
}
```

Related Topics

- 26.19 Charge Rates Resource - page 322
- 26.3 Accounting Resources - page 299

26.24 Funds Resource

This section provides information on the supported actions for the Funds accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query funds	GET	/funds[/<id>]
Create a fund	POST	/funds
Modify a fund	PATCH	/funds/<id>
Delete a fund	DELETE	/funds/<id>
Query fund constraints	GET	/constraints[/<id>]
Add a fund constraint	POST	/constraints
Remove a fund constraint	DELETE	/constraints/<id>
Deposit into a fund	POST	/funds?action=deposit
Withdraw from a fund	POST	/funds?action=withdraw
Transfer between funds	POST	/funds?action=transfer
Reset a fund	POST	/funds?action=reset

Related Topics

26.25 Query Funds

26.25.1 Synopsis

```
GET /funds[/<id>][?<parameter>[&<parameter>...]]
```

26.25.2 Parameters

Parameter	Description	Example
constraint-filter	Displays funds whose constraints do not conflict with the specified filters	```GET /funds?constraint-filter=user=amy```
fields	Designates the properties to be returned in the query	```GET /funds?fields=id,name```
filter	Filters the objects to be returned in the query	```GET /funds?filter=priority>0```
filter-type	Designates the constraint filter type	```GET /funds?constraint-filter=user=amy&filter-type=ExactMatch```
limit	Limits the results to the number of objects specified	```GET /funds?limit=100```
offset	Number of objects to skip before starting to return data	```GET /funds?offset=100```

Parameter	Description	Example
show-hidden	Includes hidden attributes in the result	`GET /funds?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /funds?fields=priority&unique=true`

26.25.3 Sample Request

```
GET /funds/2
```

26.25.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "default-deposit" : -1,
            "description" : null,
            "id" : 2,
            "name" : "chemistry",
            "priority" : 0
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.26 Create a Fund

26.26.1 Synopsis

```
POST /funds[?<parameter>]
{
    <name> : <value>,...
}
```

26.26.2 Parameters

Parameter	Description	Example
constraint	Specifies a constraint for the fund	`POST /funds?constraint=account=chemistry`

26.26.3 Sample Request

```
POST /funds?constraint=account=chemistry
{
    "default-deposit" : 5000
}
```

26.26.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "default-deposit" : 5000,
            "description" : null,
            "id" : 2,
            "name" : "chemistry",
            "priority" : 0
```

```
        }
    ],
    "message" : "Successfully created 1 fund with id 2 and 1 constraint",
    "status" : "Success"
}
```

Related Topics

26.27 Modify a Fund

In this topic:

26.27.1 Synopsis

```
PATCH /funds/<id>
{
    <name> : <value>,...
}
```

26.27.2 Sample Request

```
PATCH /funds/2
{
    "default-deposit" : -1
}
```

26.27.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "default-deposit" : -1,
            "description" : null,
            "id" : 2,
            "name" : "chemistry",
            "priority" : 0
        }
    ],
    "message" : "Successfully modified 1 fund",
    "status" : "Success"
}
```

Related Topics

- 26.24 Funds Resource - page 328
- 26.3 Accounting Resources - page 299

26.28 Delete a Fund

26.28.1 Synopsis

```
DELETE /funds/<id>
```

26.28.2 Sample Request

```
DELETE /funds/2
```

26.28.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "default-deposit" : -1,
            "description" : null,
            "id" : 2,
            "name" : "chemistry",
            "priority" : 0
        }
    ],
    "message" : "Successfully deleted 1 fund",
    "status" : "Success"
}
```

Related Topics

- 26.24 Funds Resource - page 328
- 26.3 Accounting Resources - page 299

26.29 Query Fund Constraints

In this topic:

26.29.1 Synopsis

```
GET /constraints[/<id>][?<parameter>[&<parameter>...]]
```

26.29.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	```GET /constraints?fields=fund,name,value```
filter	Filters the objects to be returned in the query	```GET /constraints?filter=name=Account,value=chemistry```
limit	Limits the results to the number of objects specified	```GET /constraints?limit=100```
offset	Number of objects to skip before starting to return data	```GET /constraints?offset=100```
show-hidden	Includes hidden attributes in the result	```GET /constraints?show-hidden=true```
unique	Displays only unique results (like DISTINCT in SQL)	```GET /constraints?fields=name&unique=true```

26.29.3 Sample Request

```
GET /constraints?filter=fund=2
```

26.29.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "fund" : 2,
            "id" : 2,
            "name" : "Account",
            "value" : "chemistry"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.30 Add a Fund Constraint

In this topic:

26.30.1 Synopsis

```
POST /constraints
{
    <name> : <value>,...
}
```

26.30.2 Sample Request

```
POST /constraints
{
    "fund" : 2,
    "name" : "Account",
    "value" : "chemistry"
}
```

26.30.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "fund" : 2,
            "id" : 2,
            "name" : "Account",
            "value" : "chemistry"
        }
    ],
    "message" : "Successfully created 1 constraint",
    "status" : "Success"
}
```

Related Topics

- 26.24 Funds Resource - page 328
- 26.3 Accounting Resources - page 299

26.31 Remove a Fund Constraint

In this topic:

26.31.1 Synopsis

```
DELETE /constraints/<id>
```

26.31.2 Sample Request

```
DELETE /constraints/2
```

26.31.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "fund" : 2,
            "id" : 2,
            "name" : "Account",
            "value" : "chemistry"
        }
    ],
    "message" : "Successfully deleted 1 constraint",
    "status" : "Success"
}
```

Related Topics

- 26.24 Funds Resource - page 328
- 26.3 Accounting Resources - page 299

26.32 Deposit into a Fund

26.32.1 Synopsis

```
POST /funds?action=deposit[&<parameter>...]
```

26.32.2 Parameters

Para-meter	Descrip-tion	Example
alloc-ation	Specifies that the deposit should go into the specified allocation	```POST /funds?action=deposit&allocation=2&amount=1000```
amount	Amount to deposit	```POST /funds?action=deposit&id=2&amount=1000```
con-straint-filter	Restricts the fund to one whose con-straints do not conflict with the specified filters	```POST /funds?action=deposit&constraint-filter=account=chemistry&amount=1000```

Para-meter	Descrip-tion	Example
credit-limit	Credit limit for the new allocation	```POST /funds?action=deposit&id=2&credit-limit=1000```
filter-type	Designates the con-straint fil-ter type	```POST /funds?action=deposit&constraint-filter=account=chemistry&filter-type=ExactMatch&amount=1000```
id	ID of the fund into which the deposit will be made	```POST /funds?action=deposit&id=2&amount=1000```
reset	Ends the current allocation and cre-ates a new allocation with the deposit	```POST /funds?action=deposit&id=2&amount=1000&reset=true```

26.32.3 Sample Request

```
POST /funds?action=deposit&id=2&amount=1000
```

26.32.4 Sample Response

```
{
    "code" : "000",
    "count" : 1000,
    "message" : "Successfully deposited 1000.00 credits into fund 2",
    "status" : "Success"
}
```

Related Topics

26.33 Withdraw from a Fund

In this topic:

26.33.1 Synopsis

```
POST /funds?action=withdraw[&<parameter>...]
```

26.33.2 Parameters

Parameter	Description	Example
allocation	The credits will be withdrawn from the specified allocation only	```POST /funds?action=withdraw&allocation=2&amount=1000```
amount	Amount to withdraw	```POST /funds?action=withdraw&id=2&amount=1000```
constraint-filter	Restricts the fund to one whose constraints do not conflict with the specified filters	```POST /funds?action=withdraw&constraint-filter=account=chemistry&amount=1000```

Para-meter	Descrip-tion	Example
filter-type	Designates the constraint filter type	```POST /funds?action=withdraw&constraint-filter=account=chemistry&filter-type=ExactMatch&amount=1000```
id	ID of the fund from which the withdrawal will be made	```POST /funds?action=withdraw&id=2&amount=1000```

26.33.3 Sample Request

```
POST /funds?action=withdraw&id=2&amount=1000
```

26.33.4 Sample Response

```
{
    "code" : "000",
    "count" : 1000,
    "message" : "Successfully withdrew 1000.00 credits from fund 2",
    "status" : "Success"
}
```

Related Topics

26.34 Transfer Between Funds

26.34.1 Synopsis

```
POST /funds?action=transfer[&<parameter>...]
```

26.34.2 Parameters

Parameter	Description	Example
amount	Amount to transfer	`POST /funds?action=transfer&from-id=2&to-id=3&amount=1000`
from-alloc-ation	The credits will be transferred from the specified allocation only	`POST /funds?action=transfer&from-allocation=2&to-id=3&amount=1000`
from-id	Fund to be debited	`POST /funds?action=transfer&from-id=2&to-id=3&amount=1000`
to-allocation	The credits will be transferred to the specified allocation only	`POST /funds?action=transfer&from-id=2&to-allocation=3&amount=1000`
to-id	Fund to be credited	`POST /funds?action=transfer&from-id=2&to-id=3&amount=1000`

26.34.3 Sample Request

```
POST /funds?action=transfer&from-id=2&to-id=1&amount=1000
```

26.34.4 Sample Response

```
{
    "code" : "000",
    "count" : 1000,
    "message" : "Successfully transferred 1000.00 credits from fund 2 to fund 1",
    "status" : "Success"
}
```

Related Topics

26.35 Reset a Fund

In this topic:

26.35.1 Synopsis

```
POST /funds?action=reset[&<parameter>...]
```

26.35.2 Parameters

Para-meter	Descrip-tion	Example
con-straint-fil-ter	Restricts the fund to one whose con-straints do not conflict with the spe-cified filters	```POST /funds?action=reset&constraint-filter=account=chemistry```
filter-type	Designates the con-straint filter type	```POST /funds?action=reset&constraint-filter=account=chemistry&filter-type=ExactMatch```
id	ID of the fund to reset	```POST /funds?action=reset&id=2```

26.35.3 Sample Request

```
POST /funds?action=reset&id=1
```

26.35.4 Sample Response

```
{
    "code" : "000",
    "count" : 5000,
    "message" : "Successfully deposited 5000.00 credits into fund 1\nSuccessfully
stopped 1 allocation\nSuccessfully created 1 allocation",
    "status" : "Success"
}
```

Related Topics

- 26.24 Funds Resource - page 328
- 26.3 Accounting Resources - page 299

26.36 Liens Resource

This section provides information on the supported actions for the Liens accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query liens	GET	/liens[/<id>]
Modify a lien	PATCH	/liens/<id>
Delete a lien	DELETE	/liens/<id>
Query lien allocations	GET	/lien-allocations[/<lien>[/<allocation>]]

Related Topics

26.37 Query Liens

26.37.1 Synopsis

```
GET /liens[/<id>][?<parameter>[&<parameter>...]]
```

26.37.2 Parameters

Parameter	Description	Example
active	Displays only unexpired liens	```GET /liens?active=true```
constraint-filter	Displays liens whose constraints comply with the specified filters	```GET /liens?constraint-filter=user=amy```
fields	Designates the properties to be returned in the query	```GET /liens?fields=id,amount```
filter	Filters the objects to be returned in the query	```GET /liens?filter=usage-record=1```
filter-type	Designates the constraint filter type	```GET /liens?constraint-filter=user=amy&filter-type=ImpingesUpon```
limit	Limits the results to the number of objects specified	```GET /liens?limit=100```

Parameter	Description	Example
offset	Number of objects to skip before starting to return data	`GET /liens?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /liens?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /liens?fields=usage-record&unique=true`

26.37.3 Sample Request

```
GET /liens?active=true
```

26.37.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : null,
            "duration" : 600,
            "end-time" : "2018-06-15 18:39:47",
            "id" : 1,
            "instance" : "24809",
            "start-time" : "2018-06-15 18:29:47",
            "usage-record" : 1
        }
    ],
    "status" : "Success"
}
```

Related Topics

-
-

26.38 Modify a Lien

26.38.1 Synopsis

```
PATCH /liens/<id>
{
    <name> : <value>,...
}
```

26.38.2 Sample Request

```
PATCH /liens/1
{
    "end-time" : "2018-06-16"
}
```

26.38.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : null,
            "duration" : 600,
            "end-time" : "2018-06-16",
            "id" : 1,
            "instance" : "24809",
            "start-time" : "2018-06-15 18:29:47",
            "usage-record" : 1
        }
    ],
    "message" : "Successfully modified 1 lien",
    "status" : "Success"
}
```

Related Topics

- 26.36 Liens Resource - page 348

- 26.3 Accounting Resources - page 299

26.39 Delete a Lien

26.39.1 Synopsis

```
DELETE /liens/<id>
```

26.39.2 Sample Request

```
DELETE liens/2
```

26.39.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : null,
            "duration" : 600,
            "end-time" : "2018-06-15 18:39:47",
            "id" : 1,
            "instance" : "24809",
            "start-time" : "2018-06-15 18:29:47",
            "usage-record" : 1
        }
    ],
    "message" : "Successfully deleted 1 lien",
    "status" : "Success"
}
```

Related Topics

- 26.36 Liens Resource - page 348
- 26.3 Accounting Resources - page 299

26.40 Query Lien Allocations

26.40.1 Synopsis

```
GET /lien-allocations[/<lien>[/<allocation>]][?<parameter>[&<parameter>...]]
```

26.40.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	```GET /lien-allocations/chemistry?fields=sum(amount),group-by(fund)```
filter	Filters the objects to be returned in the query	```GET /lien-allocations?filter=fund=4```
limit	Limits the results to the number of objects specified	```GET /lien-allocations?limit=100```
offset	Number of objects to skip before starting to return data	```GET /lien-allocations?offset=100```
show-hidden	Includes hidden attributes in the result	```GET /lien-allocations?show-hidden=true```

Parameter	Description	Example
unique	Displays only unique results (like DISTINCT in SQL)	`GET /lien-allocations?fields=fund&unique=true`

26.40.3 Sample Request

```
GET /lien-allocations?fields=sum(amount),group-by(fund)
```

26.40.4 Sample Response

```
{
    "code" : "000",
    "count" : 2,
    "data" : [
        {
            "amount" : 2,
            "fund" : 2
        },
        {
            "amount" : 10.56,
            "fund" : 4
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.41 Organizations Resource

This section provides information on the supported actions for the Organizations accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query organizations	GET	/organizations[/<name>]
Create an organization	POST	/organizations
Modify an organization	PATCH	/organizations/<name>
Delete an organization	DELETE	/organizations/<name>

Related Topics

26.42 Query Organizations

26.42.1 Synopsis

```
GET /organizations[/<name>][?<parameter>[&<parameter>...]]
```

26.42.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /organizations?fields=name`
filter	Filters the objects to be returned in the query	`GET /organizations?filter=name~sci*`
limit	Limits the results to the number of objects specified	`GET /organizations?limit=100`
offset	Number of objects to skip before starting to return data	`GET /organizations?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /organizations?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /organizations?fields=name&unique=true`

26.42.3 Sample Request

```
GET /organizations/sciences
```

26.42.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Sciences College",
            "name" : "sciences"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.43 Create an Organization

26.43.1 Synopsis

```
POST /organizations
{
    <name> : <value>,...
}
```

26.43.2 Sample Request

```
POST /organizations
{
    "description" : "Sciences College",
    "name" : "sciences"
}
```

26.43.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Sciences College",
            "name" : "sciences"
        }
    ],
    "message" : "Successfully created 1 organization",
    "status" : "Success"
}
```

Related Topics

- 26.41 Organizations Resource - page 355
- 26.3 Accounting Resources - page 299

26.44 Modify an Organization

In this topic:

26.44.1 Synopsis

```
PATCH /organizations/<name>
{
    <name> : <value>,...
}
```

26.44.2 Sample Request

```
PATCH /organizations/sciences
{
    "description" : "Sciences Department"
}
```

26.44.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Sciences Department",
            "name" : "sciences"
        }
    ],
    "message" : "Successfully modified 1 organization",
    "status" : "Success"
}
```

Related Topics

- 26.41 Organizations Resource - page 355

- 26.3 Accounting Resources - page 299

26.45 Delete an Organization

26.45.1 Synopsis

```
DELETE /organizations/<name>
```

26.45.2 Sample Request

```
DELETE /organizations/sciences
```

26.45.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Sciences College",
            "name" : "sciences"
        }
    ],
    "message" : "Successfully deleted 1 organization",
    "status" : "Success"
}
```

Related Topics

- 26.41 Organizations Resource - page 355
- 26.3 Accounting Resources - page 299

26.46 Quotes Resource

This section provides information on the supported actions for the Quotes accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query quotes	GET	/quotes[/<id>]
Modify a quote	PATCH	/quotes/<id>
Delete a quote	DELETE	/quotes/<id>
Query quote charge rates	GET	/quote-charge-rates[/<quote>[/<name>[/<value>]]]

Related Topics

26.47 Query Quotes

26.47.1 Synopsis

```
GET /quotes[/<id>][?<parameter>[&<parameter>...]]
```

26.47.2 Parameters

Parameter	Description	Example
active	Displays only unexpired quotes	`GET /quotes?active=true`
constraint-filter	Displays quotes whose constraints comply with the specified filters	`GET /quotes?constraint-filter=user=amy`
fields	Designates the properties to be returned in the query	`GET /quotes?fields=id,amount`
filter	Filters the objects to be returned in the query	`GET /quotes?filter=usage-record=1`
limit	Limits the results to the number of objects specified	`GET /quotes?limit=100`
offset	Number of objects to skip before starting to return data	`GET /quotes?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /quote?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /quotes?fields=usage-record&unique=true`

26.47.3 Sample Request

```
GET /quotes?active=true
```

26.47.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : 0.56,
            "description" : null,
            "duration" : 1000,
            "end-time" : "2018-08-23 18:16:18",
            "id" : 1,
            "instance" : "j1",
            "pinned" : true,
            "start-time" : "2018-08-23 17:59:38",
            "usage-record" : 12
        }
    ],
    "status" : "Success"
}
```

Related Topics

- 26.46 Quotes Resource - page 361
- 26.3 Accounting Resources - page 299

26.48 Modify a Quote

26.48.1 Synopsis

```
PATCH /quotes/<id>
{
    <name> : <value>,...
}
```

26.48.2 Sample Request

```
PATCH /quotes/1
{
    "end-time" : "2018-08-24"
}
```

26.48.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : 0.56,
            "description" : null,
            "duration" : 1000,
            "end-time" : "2018-08-24",
            "id" : 1,
            "instance" : "j1",
            "pinned" : true,
            "start-time" : "2018-08-23 17:59:38",
            "usage-record" : 12
        }
    ],
    "message" : "Successfully modified 1 quote",
    "status" : "Success"
}
```

Related Topics

26.49 Delete a Quote

26.49.1 Synopsis

```
DELETE /quotes/<id>
```

26.49.2 Sample Request

```
DELETE /quotes/1
```

26.49.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : 0.56,
            "description" : null,
            "duration" : 1000,
            "end-time" : "2018-08-23 18:16:18",
            "id" : 1,
            "instance" : "j1",
           "pinned" : true,
            "start-time" : "2018-08-23 17:59:38",
            "usage-record" : 12
        }
    ],
    "message" : "Successfully deleted 1 quote",
    "status" : "Success"
}
```

Related Topics

26.50 Query Quote Charge Rates

In this topic:

26.50.1 Synopsis

```
GET /quote-charge-rates[/<quote>[/<name>[/<value>]]][?<parameter>[&<parameter>...]]
```

26.50.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /quote-charge-rates?fields=name`
filter	Filters the objects to be returned in the query	`GET /quote-charge-rates?filter=name=Processors`
limit	Limits the results to the number of objects specified	`GET /quote-charge-rates?limit=100`
offset	Number of objects to skip before starting to return data	`GET /quote-charge-rates?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /quote-charge-rates?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /quote-charge-rates?fields=name&unique=true`

26.50.3 Sample Request

```
GET /quote-charge-rates/1
```

26.50.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : "1/h",
            "name" : "Processors",
            "quote" : 1,
            "value" : null
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.51 Transactions Resource

This section provides information on the supported actions for the Transactions accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query transactions	GET	/transactions[/<id>]

Related Topics

- 26.3 Accounting Resources - page 299

26.52 Query Transactions

26.52.1 Synopsis

```
GET /transactions[/<id>][?<parameter>[&<parameter>...]]
```

26.52.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	```GET /transactions?filter=action=Charge&fields=sum (amount)```
filter	Filters the objects to be returned in the query	```GET /transactions?filter=action=Charge```
limit	Limits the results to the number of objects specified	```GET /transactions?limit=100```
offset	Number of objects to skip before starting to return data	```GET /transactions?offset=100```
show-hidden	Includes hidden attributes in the result	```GET /transactions?show-hidden=true```
unique	Displays only unique results (like DISTINCT in SQL)	```GET /transactions?fields=account&unique=true```

26.52.3 Sample Request

```
GET /transactions?filter=usage-record=1,action=Charge
```

26.52.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "action" : "Charge",
            "actor" : "scottmo",
            "allocation" : 2,
            "amount" : 1,
            "balance" : 2999,
            "child" : "24809",
            "count" : 1,
            "delta" : -1,
            "description" : null,
            "duration" : 300,
            "fund" : 2,
            "id" : 334,
            "instance" : "24809",
            "key" : "1",
            "machine" : "colony",
            "object" : "UsageRecord",
            "remaining" : 2999,
            "usage-record" : 1,
            "user" : "amy"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.53 Usage Records Resource

This section provides information on the supported actions for the Usage Records accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query usage records	GET	/usage-records[/<id>]
Create a usage record	POST	/usage-records
Modify a usage record	PATCH	/usage-records/<id>
Delete a usage record	DELETE	/usage-records/<id>
Quote for usage	POST	/usage-records?action=quote
Reserve for usage	POST	/usage-records?action=reserve
Charge for usage	POST	/usage-records?action=charge
Refund usage	POST	/usage-records?action=refund

Related Topics

- 26.3 Accounting Resources - page 299

26.54 Query Usage Records

26.54.1 Synopsis

```
GET /usage-records[/<id>][?<parameter>[&<parameter>...]]
```

26.54.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /usage-records?fields=account,charge`
filter	Filters the objects to be returned in the query	`GET /usage-records?filter=instance=24809`
limit	Limits the results to the number of objects specified	`GET /usage-records?limit=100`
offset	Number of objects to skip before starting to return data	`GET /usage-records?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /usage-records?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /usage-records?fields=account&unique=true`

26.54.3 Sample Request

```
GET /usage-records?filter=instance=24809
```

26.54.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "c-p-u-time" : 1800,
            "charge" : 0,
            "class" : "batch",
            "description" : null,
            "duration" : 300,
            "end-time" : "2018-06-15 18:34:47",
            "exit-code" : null,
            "group" : "research",
            "id" : 1,
            "instance" : "24809",
            "licenses" : null,
            "machine" : "colony",
            "memory" : null,
            "metrics" : null,
            "nodes" : 1,
            "organization" : "sciences",
            "processors" : 12,
            "quality-of-service" : "normal",
            "requested-duration" : 600,
            "resources" : "{\"gres\":1,\"color\":2}",
            "stage" : null,
            "start-time" : "2018-06-15 18:29:47",
            "submit-time" : null,
            "type" : "Job",
            "user" : "amy",
            "variables" : null
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.55 Create a Usage Record

26.55.1 Synopsis

```
POST /usage-records
{
    <name> : <value>,...
}
```

26.55.2 Sample Request

```
POST /usage-records
{
    "account" : "chemistry",
    "c-p-u-time" : 1800,
    "class" : "batch",
    "duration" : 300,
    "end-time" : "2018-06-15 18:34:47",
    "group" : "research",
    "instance" : "24809",
    "machine" : "colony",
    "nodes" : 1,
    "organization" : "sciences",
    "processors" : 12,
    "quality-of-service" : "normal",
    "requested-duration" : 600,
    "resources" : "{\"gres\":1,\"color\":2}",
    "start-time" : "2018-06-15 18:29:47",
    "type" : "Job",
    "user" : "amy",
}
```

26.55.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "c-p-u-time" : 1800,
            "charge" : 0,
```

```
            "class" : "batch",
            "description" : null,
            "duration" : 300,
            "end-time" : "2018-06-15 18:34:47",
            "exit-code" : null,
            "group" : "research",
            "id" : 1,
            "instance" : "24809",
            "licenses" : null,
            "machine" : "colony",
            "memory" : null,
            "metrics" : null,
            "nodes" : 1,
            "organization" : "sciences",
            "processors" : 12,
            "quality-of-service" : "normal",
            "requested-duration" : 600,
            "resources" : "{\"gres\":1,\"color\":2}",
            "stage" : null,
            "start-time" : "2018-06-15 18:29:47",
            "submit-time" : null,
            "type" : "Job",
            "user" : "amy",
            "variables" : null
        }
    ],
    "message" : "Successfully created 1 usage-record",
    "status" : "Success"
}
```

Related Topics

26.56 Modify a Usage Record

In this topic:

26.56.1 Synopsis

```
PATCH /usage-records/<id>
{
    <name> : <value>,...
}
```

26.56.2 Sample Request

```
PATCH /usage-records/1
{
    "group" : "staff"
}
```

26.56.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "account" : "chemistry",
            "c-p-u-time" : 1800,
            "charge" : 0,
            "class" : "batch",
            "description" : null,
            "duration" : 300,
            "end-time" : "2018-06-15 18:34:47",
            "exit-code" : null,
            "group" : "staff",
            "id" : 1,
            "instance" : "24809",
            "licenses" : null,
            "machine" : "colony",
            "memory" : null,
            "metrics" : null,
            "nodes" : 1,
            "organization" : "sciences",
            "processors" : 12,
            "quality-of-service" : "normal",
```

```
        "requested-duration" : 600,
        "resources" : "{\"gres\":1,\"color\":2}",
        "stage" : null,
        "start-time" : "2018-06-15 18:29:47",
        "submit-time" : null,
        "type" : "Job",
        "user" : "amy",
        "variables" : null
      }
  ],
  "message" : "Successfully modified 1 usage record",
  "status" : "Success"
}
```

Related Topics

26.57 Delete a Usage Record

In this topic:

26.57.1 Synopsis

```
DELETE /usage-records/<id>
```

26.57.2 Sample Request

```
DELETE /usage-records/1
```

26.57.3 Sample Response

```
{
   "code" : "000",
   "count" : 1,
   "data" : [
      {
         "account" : "chemistry",
         "c-p-u-time" : 1800,
         "charge" : 0,
         "class" : "batch",
         "description" : null,
         "duration" : 300,
         "end-time" : "2018-06-15 18:34:47",
         "exit-code" : null,
         "group" : "research",
         "id" : 1,
         "instance" : "24809",
         "licenses" : null,
         "machine" : "colony",
         "memory" : null,
         "metrics" : null,
         "nodes" : 1,
         "organization" : "sciences",
         "processors" : 12,
         "quality-of-service" : "normal",
         "requested-duration" : 600,
         "resources" : "{\"gres\":1,\"color\":2}",
         "stage" : null,
         "start-time" : "2018-06-15 18:29:47",
         "submit-time" : null,
         "type" : "Job",
```

```
            "user" : "amy",
            "variables" : null
        }
    ],
    "message" : "Successfully deleted 1 usage record",
    "status" : "Success"
}
```

Related Topics

26.58 Quote for Usage

26.58.1 Synopsis

```
POST /usage-records?action=quote[&<parameter>...]
{
    <name> : <value>,...
}
```

26.58.2 Parameters

Para-meter	Descrip-tion	Example
charge	Specifies the quote amount if calculated externally	```POST /usage-records?action=quote&charge=1\n{\n "instance" : "j1"\n}```
cost-only	Returns the cost, ignoring all balance and validity checks	```POST /usage-records?action=quote&cost-only=true\n{\n "processors" : 1,\n "requested-duration" : 3600\n}```
duration	Incremental duration for the quote in seconds	```POST /usage-records?action=quote&duration=3600\n{\n "processors" : 1\n}```
end-time	End time for the quote	```POST /usage-records?action=quote&start-time=2018-08-23&end-time=2018-08-24\n{\n "processors" : 1\n}```

Para-meter	Descrip-tion	Example
grace-dur-ation	Grace period in seconds	```POST /usage-records?action=quote&id=1&duration=3600&grace-duration=3600 { "processors" : 1 }```
id	Usage record for the quote (if usage record already cre-ated)	```POST /usage-records?action=quote&id=1 { "processors" : 1, "requested-duration" : 3600 }```
itemize	Returns the composite charge inform-ation in the response data	```POST /usage-records?action=quote&itemize=true { "processors" : 1, "requested-duration" : 3600 }```
quote	Quote tem-plate used to override standard charge rates	```POST /usage-records?action=quote&quote=1 { "processors" : 1, "requested-duration" : 3600 }```
rate	Uses the spe-cified charge rate in the quote	```POST /usage-records?action=quote&rate=Processors=2/h { "processors" : 1, "requested-duration" : 3600 }```
start-time	Start time for the quote	```POST /usage-records?action=quote&start-time=2018-08-23&duration=3600 { "processors" : 1 }```

26.58.3 Sample Request

```
POST /usage-records?action=quote
{
    "account" : "chemistry",
    "class" : "batch",
    "group" : "research",
    "machine" : "colony",
```

```
    "nodes" : 1,
    "processors" : 12,
    "quality-of-service" : "normal",
    "requested-duration" : 600,
    "user" : "amy"
}
```

26.58.4 Sample Response

```
{
    "code" : "000",
    "count" : 2,
    "data" : [
        {
            "amount" : 2
        }
    ],
    "message" : "Successfully quoted 2.00 credits",
    "status" : "Success"
}
```

Related Topics

26.59 Reserve for Usage

26.59.1 Synopsis

```
POST /usage-records?action=reserve[&<parameter>...]
{
    <name> : <value>,...
}
```

26.59.2 Parameters

Para-meter	Descrip-tion	Example
charge	Specifies the lien amount if calculated externally	```POST /usage-records?action=reserve&charge=1 { "instance" : "j1" }```
duration	Incremental duration for the lien in seconds	```POST /usage-records?action=reserve&duration=3600 { "processors" : 1 }```
end-time	End time for the lien	```POST /usage-records?action=reserve&start-time=2018-08-23&end-time=2018-08-24 { "processors" : 1 }```
grace-dur-ation	Grace period in seconds	```POST /usage-records?action=reserve&id=1&duration=3600&grace-duration=3600 { "processors" : 1 }```

Para-meter	Descrip-tion	Example
id	Usage record for the lien (if usage record already created)	```POST /usage-records?action=quote&id=1
{
"processors" : 1,
"requested-duration" : 3600
}```
itemize	Returns the composite charge information in the response data	```POST /usage-records?action=reserve&itemize=true
{
"processors" : 1,
"requested-duration" : 3600
}```
modify	Augments existing liens instead of creating new ones	```POST /usage-records?action=reserve&modify=true
{
"processors" : 1,
"requested-duration" : 3600
}```
rate	Uses the specified charge rate in the lien	```POST /usage-records?action=reserve&rate=Processors=2/h
{
"processors" : 1,
"requested-duration" : 3600
}```
replace	Similarly named liens will be deleted before this lien is created	```POST /usage-records?action=reserve&replace=true
{
"processors" : 1,
"requested-duration" : 3600
}```
start-time	Start time for the lien	```POST /usage-records?action=reserve&start-time=2018-08-23&duration=3600
{
"processors" : 1
}```

26.59.3 Sample Request

```
POST /usage-records?action=reserve
{
    "account" : "chemistry",
```

```
    "class" : "batch",
    "group" : "research",
    "instance" : "j1",
    "machine" : "colony",
    "nodes" : 1,
    "processors" : 12,
    "quality-of-service" : "normal",
    "requested-duration" : 600,
    "user" : "amy"
}
```

26.59.4 Sample Response

```
{
    "code" : "000",
    "count" : 2,
    "data" : [
        {
            "amount" : 2,
            "instance" : "j1",
            "lien" : 17,
            "usage-record" : 14
        }
    ],
    "message" : "Successfully reserved 2.00 credits with lien id 17 for instance j1 and
created usage record 14",
    "status" : "Success"
}
```

Related Topics

- 26.53 Usage Records Resource - page 372
- 26.3 Accounting Resources - page 299

26.60 Charge for Usage

26.60.1 Synopsis

```
POST /usage-records?action=charge[&<parameter>...]
{
    <name> : <value>,...
}
```

26.60.2 Parameters

Para-meter	Descrip-tion	Example
charge	Specifies the charge amount if calculated externally	```POST /usage-records?action=charge&charge=1 { "instance" : "j1" }```
duration	Incremental duration for the charge in seconds	```POST /usage-records?action=charge&duration=3600 { "processors" : 1 }```
end-time	End time for the charge	```POST /usage-records?action=charge&start-time=2018-08-23&end-time=2018-08-24 { "processors" : 1 }```
fund	Fund to charge	```POST /usage-records?action=charge&fund=2 { "processors" : 1, "duration" : 3600 }```

Para-meter	Descrip-tion	Example
id	Usage record for the charge (if usage record already created)	```POST /usage-records?action=charge&id=1 { "processors" : 1, "duration" : 3600 } ```
incre-mental	Any asso-ciated liens will be deb-ited instead of removed	```POST /usage-records?action=charge&incremental=true { "processors" : 1, "requested-duration" : 3600 } ```
itemize	Returns the composite charge inform-ation in the response data	```POST /usage-records?action=charge&itemize=true { "processors" : 1, "requested-duration" : 3600 } ```
rate	Uses the spe-cified charge rate in the charge	```POST /usage-records?action=charge&rate=Processors=2/h { "processors" : 1, "requested-duration" : 3600 } ```
start-time	Start time for the charge	```POST /usage-records?action=charge&start-time=2018-08-23&duration=3600 { "processors" : 1 } ```

26.60.3 Sample Request

```
POST /usage-records?action=charge
{
    "account" : "chemistry",
    "class" : "batch",
    "c-p-u-time" : 1800,
    "duration" : 300,
    "end-time" : "2018-06-15 18:34:47",
    "group" : "research",
    "instance" : "j1",
    "machine" : "colony",
    "nodes" : 1,
    "processors" : 12,
    "quality-of-service" : "normal",
```

```
    "start-time" : "2018-06-15 18:29:47",
    "user" : "amy"
}
```

26.60.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "amount" : 1,
            "instance" : "j1",
            "usage-record" : 15
        }
    ],
    "message" : "Successfully charged 1.00 credits for instance j1 and created usage
record 15",
    "status" : "Success"
}
```

Related Topics

26.61 Refund Usage

26.61.1 Synopsis

```
POST /usage-records?action=refund[&<parameter>...]
```

26.61.2 Parameters

Parameter	Description	Example
allocation	Allocation to be credited	`POST /usage-records?action=refund&id=1&allocation=2`
amount	Amount to refund	`POST /usage-records?action=refund&id=1&amount=0.5`
id	Usage record to be refunded	`POST /usage-records?action=refund&id=1`
instance	Instance to be refunded	`POST /usage-records?action=refund&instance=j1`

26.61.3 Sample Request

```
POST /usage-records?action=refund&instance=j1
```

26.61.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
```

```
        "amount" : 1,
        "id" : "1",
        "instance" : "j1"
    }
 ],
 "message" : "Successfully refunded 1.00 credits to usage record 1 for instance j1",
 "status" : "Success"
}
```

Related Topics

26.62 Users Resource

This section provides information on the supported actions for the Users accounting resource.

Supported Actions

Action	HTTP Method	Resource
Query users	GET	/users[/<name>]
Create a user	POST	/users
Modify a user	PATCH	/users/<name>
Delete a user	DELETE	/users/<name>

Related Topics

- 26.3 Accounting Resources - page 299

26.63 Query Users

In this topic:

26.63.1 Synopsis

```
GET /users[/<name>][?<parameter>[&<parameter>...]]
```

26.63.2 Parameters

Parameter	Description	Example
constraint-filter	Applies meta-filters to the query (account: include only users associated with the specified account)	`GET /users?constraint-filter=account=chemistry`
fields	Designates the properties to be returned in the query	`GET /users?fields=name,email-address`
filter	Filters the objects to be returned in the query	`GET /users?filter=active=true`
limit	Limits the results to the number of objects specified	`GET /users?limit=100`
offset	Number of objects to skip before starting to return data	`GET /users?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /users?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /users?fields=default-account&unique=true`

26.63.3 Sample Request

```
GET /users/amy
```

26.63.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "common-name" : "Amy Miller",
            "default-account" : "chemistry",
            "description" : null,
            "email-address" : "amy@hpc.com",
            "name" : "amy",
            "phone-number" : "(801) 555-1437"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.64 Create a User

In this topic:

26.64.1 Synopsis

```
POST /users
{
    <name> : <value>,...
}
```

26.64.2 Sample Request

```
POST /users
{
    "common-name" : "Amy Miller",
    "default-account" : "chemistry",
    "email-address" : "amy@hpc.com",
    "name" : "amy",
    "phone-number" : "(801) 555-1437"
}
```

26.64.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "common-name" : "Amy Miller",
            "default-account" : "chemistry",
            "description" : null,
            "email-address" : "amy@hpc.com",
            "name" : "amy",
            "phone-number" : "(801) 555-1437"
        }
    ],
    "message" : "Successfully created 1 user",
    "status" : "Success"
}
```

Related Topics

26.65 Modify a User

26.65.1 Synopsis

```
POST /users/<name>
{
    <name> : <value>,...
}
```

26.65.2 Sample Request

```
PATCH /users/amy
{
    "email-address" : "amy@htc.org"
}
```

26.65.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "common-name" : "Amy Miller",
            "default-account" : "chemistry",
            "description" : null,
            "email-address" : "amy@htc.org",
            "name" : "amy",
            "phone-number" : "(801) 555-1437"
        }
    ],
    "message" : "Successfully modified 1 user",
    "status" : "Success"
}
```

Related Topics

- 26.62 Users Resource - page 392

- 26.3 Accounting Resources - page 299

26.66 Delete a User

26.66.1 Synopsis

```
DELETE /users/<name>
```

26.66.2 Sample Request

```
DELETE /users/amy
```

26.66.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "active" : true,
            "common-name" : "Amy Miller",
            "default-account" : "chemistry",
            "description" : null,
            "email-address" : "amy@hpc.com",
            "name" : "amy",
            "phone-number" : "(801) 555-1437"
        }
    ],
    "message" : "Successfully deleted 1 user",
    "status" : "Success"
}
```

Related Topics

- 26.62 Users Resource - page 392

- 26.3 Accounting Resources - page 299

26.67 Framework Resources

This section provides information on available MAMWS framework resources.

26.68 Actions Resource

This section provides information on the supported actions for the Actions framework resource.

Supported Actions

Action	HTTP Method	Resource
Query actions	GET	/actions[/<object>[/<name>]]
Create an action	POST	/actions
Modify an action	PATCH	/actions/<object>/<name>
Delete an action	DELETE	/actions/<object>/<name>

Related Topics

-

26.69 Query Actions

26.69.1 Synopsis

```
GET /actions[/<object>[/<name>]][?<parameter>[&<parameter>...]]
```

26.69.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /actions/UsageRecord?fields=name`
filter	Filters the objects to be returned in the query	`GET /actions?filter=name=Refund`
limit	Limits the results to the number of objects specified	`GET /actions?limit=100`
offset	Number of objects to skip before starting to return data	`GET /actions?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /actions?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /actions?fields=object&unique=true`

26.69.3 Sample Request

```
GET /actions/UsageRecord/Charge
```

26.69.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Charge for Usage",
            "display" : false,
            "name" : "Charge",
            "object" : "UsageRecord"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.70 Create an Action

26.70.1 Synopsis

```
POST /actions
{
    <name> : <value>,...
}
```

26.70.2 Sample Request

```
POST /actions
{
    "description" : "Modify",
    "name" : "Modify",
    "object" : "Transaction"
}
```

26.70.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Modify",
            "display" : false,
            "name" : "Modify",
            "object" : "Transaction"
        }
    ],
    "message" : "Successfully created 1 action",
    "status" : "Success"
}
```

Related Topics

- 26.68 Actions Resource - page 400
- 26.67 Framework Resources - page 399

26.71 Modify an Action

In this topic:

26.71.1 Synopsis

```
PATCH /actions/<object>/<name>
{
    <name> : <value>,...
}
```

26.71.2 Sample Request

```
PATCH /actions/Transaction/Modify
{
    "display" : true
}
```

26.71.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
          "description" : "Modify",
          "display" : true,
          "name" : "Modify",
          "object" : "Transaction"
        }
    ],
    "message" : "Successfully modified 1 action",
    "status" : "Success"
}
```

Related Topics

- 26.68 Actions Resource - page 400
- 26.67 Framework Resources - page 399

26.72 Delete an Action

26.72.1 Synopsis

```
DELETE /actions/<object>/<name>
```

26.72.2 Sample Request

```
DELETE /actions/Transaction/Modify
```

26.72.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
          "description" : "Modify",
          "display" : false,
          "name" : "Modify",
          "object" : "Transaction"
        }
    ],
    "message" : "Successfully deleted 1 action",
    "status" : "Success"
}
```

Related Topics

- 26.68 Actions Resource - page 400
- 26.67 Framework Resources - page 399

26.73 Attributes Resource

This section provides information on the supported actions for the Attributes framework resource.

Supported Actions

Action	HTTP Method	Resource
Query attributes	GET	/attributes[/<object>[/<name>]]
Create an attribute	POST	/attributes
Modify an attribute	PATCH	/attributes/<object>/<name>
Delete an attribute	DELETE	/attributes/<object>/<name>

Related Topics

-

26.74 Query Attributes

26.74.1 Synopsis

```
GET /attributes[/<object>[/<name>]][?<parameter>[&<parameter>...]]
```

26.74.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /attributes/UsageRecord?fields=name`
filter	Filters the objects to be returned in the query	`GET /attributes/ChargeRate?filter=primary-key=True`
limit	Limits the results to the number of objects specified	`GET /attributes?limit=100`
offset	Number of objects to skip before starting to return data	`GET /attributes?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /attributes?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /attributes?fields=name&unique=true`

26.74.3 Sample Request

```
GET /attributes/Account/Organization
```

26.74.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "data-type" : "String",
            "default-value" : null,
            "description" : "Organization",
            "fixed" : false,
            "hidden" : false,
            "name" : "Organization",
            "object" : "Account",
            "primary-key" : false,
            "required" : false,
            "sequence" : 30,
            "values" : "@!=Organization"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.75 Create an Attribute

In this topic:

26.75.1 Synopsis

```
POST /attributes
{
    <name> : <value>,...
}
```

26.75.2 Sample Request

```
POST /attributes
{
    "data-type" : "String",
    "description" : "Organization",
    "name" : "Organization",
    "object" : "Account",
    "values" : "@!=Organization"
}
```

26.75.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "data-type" : "String",
            "default-value" : null,
            "description" : "Organization",
            "fixed" : false,
            "hidden" : false,
            "name" : "Organization",
            "object" : "Account",
            "primary-key" : false,
            "required" : false,
            "sequence" : 30,
            "values" : "@!=Organization"
        }
    ],
    "message" : "Successfully created 1 attribute",
    "status" : "Success"
}
```

```
}
```

Related Topics

26.76 Modify an Attribute

26.76.1 Synopsis

```
PATCH /attributes/<object>/<name>
{
    <name> : <value>,...
}
```

26.76.2 Sample Request

```
PATCH /attributes/Account/Organization
{
    "default-value" : "university"
}
```

26.76.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "data-type" : "String",
            "default-value" : "university",
            "description" : "Organization",
            "fixed" : false,
            "hidden" : false,
            "name" : "Organization",
            "object" : "Account",
            "primary-key" : false,
            "required" : false,
            "sequence" : 30,
            "values" : "@!=Organization"
        }
    ],
    "message" : "Successfully modified 1 attribute",
    "status" : "Success"
}
```

Related Topics

26.77 Delete an Attribute

26.77.1 Synopsis

```
DELETE /attributes/<object>/<name>
```

26.77.2 Sample Request

```
DELETE /attributes/Account/Organization
```

26.77.3 Sample Response

```
{
   "code" : "000",
   "count" : 1,
   "data" : [
      {
         "data-type" : "String",
         "default-value" : null,
         "description" : "Organization",
         "fixed" : false,
         "hidden" : false,
         "name" : "Organization",
         "object" : "Account",
         "primary-key" : false,
         "required" : false,
         "sequence" : 30,
         "values" : "@!=Organization"
      }
   ],
   "message" : "Successfully deleted 1 attribute",
   "status" : "Success"
}
```

Related Topics

- 26.73 Attributes Resource - page 406
- 26.67 Framework Resources - page 399

26.78 Events Resource

This section provides information on the supported actions for the Events framework resource.

Supported Actions

Action	HTTP Method	Resource
Query events	GET	/events[/<id>]
Create an event	POST	/events
Modify an event	PATCH	/events/<id>
Delete an event	DELETE	/events/<id>

Related Topics

-

26.79 Query Events

26.79.1 Synopsis

```
GET /events[/<id>][?<parameter>[&<parameter>...]]
```

26.79.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /events?fields=id,description`
filter	Filters the objects to be returned in the query	`GET /events?filter=fire-time>now`
limit	Limits the results to the number of objects specified	`GET /events?limit=100`
offset	Number of objects to skip before starting to return data	`GET /events?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /events?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /events?fields=rearm-period&unique=true`

26.79.3 Sample Request

```
GET /events/1
```

26.79.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "arm-time" : "2018-05-31 16:29:05",
            "catch-up" : false,
            "description" : "Delete Stale Notifications",
            "end-time" : null,
            "failure-command" : null,
            "fire-command" : "Notification Refresh",
            "fire-time" : "2018-05-31 16:29:05",
            "id" : 1,
            "notify" : "Store:",
            "rearm-on-failure" : true,
            "rearm-period" : "1 day @ hour 2"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.80 Create an Event

26.80.1 Synopsis

```
POST /events
{
    <name> : <value>,...
}
```

26.80.2 Sample Request

```
POST /events
{
    "catch-up" : false,
    "description" : "Delete Stale Notifications",
    "fire-command" : "Notification Refresh",
    "fire-time" : "Now",
    "rearm-on-failure" : true,
    "rearm-period" : "1 day @ hour 2"
}
```

26.80.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "arm-time" : "2018-05-31 16:29:05",
            "catch-up" : false,
            "description" : "Delete Stale Notifications",
            "end-time" : null,
            "failure-command" : null,
            "fire-command" : "Notification Refresh",
            "fire-time" : "2018-05-31 16:29:05",
            "id" : 1,
            "notify" : "Store:",
            "rearm-on-failure" : true,
            "rearm-period" : "1 day @ hour 2"
        }
    ],
    "message" : "Successfully created 1 event",
```

```
    "status" : "Success"
}
```

Related Topics

-
-

26.81 Modify an Event

In this topic:

26.81.1 Synopsis

```
PATCH /events/<id>
{
    <name> : <value>,...
}
```

26.81.2 Sample Request

```
PATCH /events/1
{
    "rearm-period" : "12 hours^"
}
```

26.81.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "arm-time" : "2018-05-31 16:29:05",
            "catch-up" : false,
            "description" : "Delete Stale Notifications",
            "end-time" : null,
            "failure-command" : null,
            "fire-command" : "Notification Refresh",
            "fire-time" : "2018-05-31 16:29:05",
            "id" : 1,
            "notify" : "Store:",
            "rearm-on-failure" : true,
            "rearm-period" : "12 hours^"
        }
    ],
    "message" : "Successfully modified 1event",
    "status" : "Success"
}
```

Related Topics

26.82 Delete an Event

26.82.1 Synopsis

```
DELETE /events/<id>
```

26.82.2 Sample Request

```
DELETE /events/1
```

26.82.3 Sample Response

```
{
   "code" : "000",
   "count" : 1,
   "data" : [
      {
         "arm-time" : "2018-05-31 16:29:05",
         "catch-up" : false,
         "description" : "Delete Stale Notifications",
         "end-time" : null,
         "failure-command" : null,
         "fire-command" : "Notification Refresh",
         "fire-time" : "2018-05-31 16:29:05",
         "id" : 1,
         "notify" : "Store:",
         "rearm-on-failure" : true,
         "rearm-period" : "1 day @ hour 2"
      }
   ],
   "message" : "Successfully deleted 1 event",
   "status" : "Success"
}
```

Related Topics

- 26.78 Events Resource - page 414
- 26.67 Framework Resources - page 399

26.83 Notifications Resource

This section provides information on the supported actions for the Notifications framework resource.

Supported Actions

Action	HTTP Method	Resource
Query notifications	GET	/notifications[/<id>]
Delete a notification	DELETE	/notifications/<id>

Related Topics

26.84 Query Notifications

26.84.1 Synopsis

```
GET /notifications[/<id>][?<parameter>[&<parameter>...]]
```

26.84.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /notifications?fields=message`
filter	Filters the objects to be returned in the query	`GET /notifications?filter=status=Failure`
limit	Limits the results to the number of objects specified	`GET /notifications?limit=100`
offset	Number of objects to skip before starting to return data	`GET /notifications?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /notifications?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET notifications?fields=type&unique=true`

26.84.3 Sample Request

```
GET /notifications/1
```

26.84.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "code" : "000",
            "end-time" : "2018-09-23 13:55:00",
            "event" : 1,
            "id" : 1,
            "key" : null,
            "message" : "No stale events were located for deletion",
            "recipient" : null,
            "status" : "Success",
            "type" : "Fire"
        }
    ],
    "status" : "Success"
}
```

Related Topics

- 26.83 Notifications Resource - page 422
- 26.67 Framework Resources - page 399

26.85 Delete a Notification

26.85.1 Synopsis

```
DELETE /notifications/<id>
```

26.85.2 Sample Request

```
DELETE /notifications/1
```

26.85.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "code" : "000",
            "end-time" : "2018-09-23 13:55:00",
            "event" : 1,
            "id" : 1,
            "key" : null,
            "message" : "No stale events were located for deletion",
            "recipient" : null,
            "status" : "Success",
            "type" : "Fire"
        }
    ],
    "message" : "Successfully deleted 1 notification",
    "status" : "Success"
}
```

Related Topics

- 26.83 Notifications Resource - page 422

- 26.67 Framework Resources - page 399

26.86 Objects Resource

This section provides information on the supported actions for the Objects framework resource.

Supported Actions

Action	HTTP Method	Resource
Query objects	GET	/objects[/<name>]
Create an object	POST	/objects
Modify an object	PATCH	/objects/<name>
Delete an object	DELETE	/objects/<name>

Related Topics

-

26.87 Query Objects

26.87.1 Synopsis

```
GET /objects[/<name>][?<parameter>[&<parameter>...]]
```

26.87.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /objects?fields=name`
filter	Filters the objects to be returned in the query	`GET /objects?filter=association=True`
limit	Limits the results to the number of objects specified	`GET /objects?limit=100`
offset	Number of objects to skip before starting to return data	`GET /objects?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /objects?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /objects?fields=child&unique=true`

26.87.3 Sample Request

```
GET /objects/Organization
```

26.87.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "association" : false,
            "auto-gen" : true,
            "child" : null,
            "default-value" : null,
            "description" : "Virtual Organization",
            "name" : "Organization",
            "parent" : null"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.88 Create an Object

In this topic:

26.88.1 Synopsis

```
POST /objects
{
    <name> : <value>,...
}
```

26.88.2 Sample Request

```
POST /objects
{
    "auto-gen" : true,
    "description" : "Virtual Organization",
    "name" : "Organization",
}
```

26.88.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "association" : false,
            "auto-gen" : true,
            "child" : null,
            "default-value" : null,
            "description" : "Virtual Organization",
            "name" : "Organization",
            "parent" : null
        }
    ],
    "message" : "Successfully created 1 object",
    "status" : "Success"
}
```

Related Topics

26.89 Modify an Object

In this topic:

26.89.1 Synopsis

```
PATCH /objects/<name>
{
    <name> : <value>,...
}
```

26.89.2 Sample Request

```
PATCH /objects/Organization
{
    "auto-gen" : false
}
```

26.89.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "association" : false,
            "auto-gen" : false,
            "child" : null,
            "default-value" : null,
            "description" : "Virtual Organization",
            "name" : "Organization",
            "parent" : null"
        }
    ],
    "message" : "Successfully modified 1 object",
    "status" : "Success"
}
```

Related Topics

- 26.86 Objects Resource - page 426

- 26.67 Framework Resources - page 399

26.90 Delete an Object

26.90.1 Synopsis

```
DELETE /objects/<name>
```

26.90.2 Sample Request

```
DELETE /objects/Organization
```

26.90.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "association" : false,
            "auto-gen" : true,
            "child" : null,
            "default-value" : null,
            "description" : "Virtual Organization",
            "name" : "Organization",
            "parent" : null"
        }
    ],
    "message" : "Successfully deleted 1 object",
    "status" : "Success"
}
```

Related Topics

- 26.86 Objects Resource - page 426
- 26.67 Framework Resources - page 399

26.91 Passwords Resource

This section provides information on the supported actions for the Passwords framework resource.

Supported Actions

Action	HTTP Method	Resource
Query password	GET	/passwords[/<user>]
Create a password	POST	/passwords
Modify a password	PATCH	/passwords/<user>
Delete a password	DELETE	/passwords/<user>

Related Topics

- 26.67 Framework Resources - page 399

26.92 Query Passwords

26.92.1 Synopsis

```
GET /passwords[/<user>][?<parameter>[&<parameter>...]]
```

26.92.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /passwords?fields=user`
filter	Filters the objects to be returned in the query	`GET /passwords?filter=user~a*`
limit	Limits the results to the number of objects specified	`GET /passwords?limit=100`
offset	Number of objects to skip before starting to return data	`GET /passwords?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /passwords?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /passwords?fields=user&unique=true`

26.92.3 Sample Request

```
GET /passwords/amy
```

26.92.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "password" : "LWL9zkOOyv1ekGCRfFuuMeOHp4EtRdjX",
            "user" : "amy"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.93 Create a Password

In this topic:

26.93.1 Synopsis

```
POST /passwords
{
    <name> : <value>,...
}
```

26.93.2 Sample Request

```
POST /passwords
{
    "password" : "changeme!",
    "user" : "amy"
}
```

26.93.3 Sample Response

```
{
    "code" : "080",
    "count" : 1,
    "data" : [
        {
            "password" : "LWL9zkOOyv1ekGCRfFuuMeOHp4EtRdjX",
            "user" : "amy"
        }
    ],
    "message" : "Successfully created 1 password",
    "status" : "Success"
}
```

Related Topics

- 26.91 Passwords Resource - page 433
- 26.67 Framework Resources - page 399

26.94 Modify a Password

26.94.1 Synopsis

```
PATCH /passwords/<name>
{
    <name> : <value>,...
}
```

26.94.2 Sample Request

```
PATCH /passwords/amy
{
    "password" : "changeme2"
}
```

26.94.3 Sample Response

```
{
    "code" : "080",
    "count" : 1,
    "data" : [
        {
            "password" : "TDB5dM5sKdpti8N730cMWxoJx6XUksq1",
            "user" : "amy"
        }
    ],
    "message" : "Successfully modified 1 password",
    "status" : "Success"
}
```

Related Topics

- 26.91 Passwords Resource - page 433
- 26.67 Framework Resources - page 399

26.95 Delete a Password

26.95.1 Synopsis

```
DELETE /passwords/<name>
```

26.95.2 Sample Request

```
DELETE /passwords/amy
```

26.95.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "password" : "LWL9zkOOyv1ekGCRfFuuMeOHp4EtRdjX",
            "user" : "amy"
        }
    ],
    "message" : "Successfully deleted 1 password",
    "status" : "Success"
}
```

Related Topics

- 26.91 Passwords Resource - page 433
- 26.67 Framework Resources - page 399

26.96 Roles Resource

This section provides information on the supported actions for the Roles framework resource.

Supported Actions

Action	HTTP Method	Resource
Query roles	GET	/roles[/<name>]
Create a role	POST	/roles
Modify a role	PATCH	/roles/<name>
Delete a role	DELETE	/roles/<name>
Query role actions	GET	/role-actions[/<role>[/<object>[/<name>]]]
Add an action to a role	POST	/role-actions
Remove an action from a role	DELETE	/role-actions/<role>/<object>/<name>
Query role users	GET	/role-users[/<role>[/<user>]]
Add a user to a role	POST	/role-users
Remove a user from a role	DELETE	/role-users/<role>/<user>

Related Topics

- 26.67 Framework Resources - page 399

26.97 Query Roles

In this topic:

26.97.1 Synopsis

```
GET /roles[/<name>][?<parameter>[&<parameter>...]]
```

26.97.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /roles?fields=name`
filter	Filters the objects to be returned in the query	`GET /roles?filter=name~Account*`
limit	Limits the results to the number of objects specified	`GET /roles?limit=100`
offset	Number of objects to skip before starting to return data	`GET /roles?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /roles?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /roles?fields=name&unique=true`

26.97.3 Sample Request

```
GET /roles/UserServices
```

26.97.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "User Services",
            "name" : "UserServices"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.98 Create a Role

In this topic:

26.98.1 Synopsis

```
POST /roles
{
    <name> : <value>,...
}
```

26.98.2 Sample Request

```
POST /roles
{
    "description" : "User Services",
    "name" : "UserServices"
}
```

26.98.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "User Services",
            "name" : "UserServices"
        }
    ],
    "message" : "Successfully created 1 role",
    "status" : "Success"
}
```

Related Topics

26.99 Modify a Role

26.99.1 Synopsis

```
PATCH /roles/<name>
{
    <name> : <value>,...
}
```

26.99.2 Sample Request

```
PATCH /roles/UserServices
{
    "description" : "Help Desk"
}
```

26.99.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Help Desk",
            "name" : "UserServices"
        }
    ],
    "message" : "Successfully modified 1 role",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.100 Delete a Role

In this topic:

26.100.1 Synopsis

```
DELETE /roles/<name>
```

26.100.2 Sample Request

```
DELETE /roles/UserServices
```

26.100.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "User Services",
            "name" : "UserServices"
        }
    ],
    "message" : "Successfully deleted 1 role",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.101 Query Role Actions

26.101.1 Synopsis

```
GET /role-actions[/<role>[/<object>[/<name>]]][?<parameter>[&<parameter>...]]
```

26.101.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	```GET /role-actions/UserServices?fields=object,name,instance```
filter	Filters the objects to be returned in the query	```GET /role-actions?filter=object=UsageRecord```
limit	Limits the results to the number of objects specified	```GET /role-actions?limit=100```
offset	Number of objects to skip before starting to return data	```GET /role-actions?offset=100```
show-hidden	Includes hidden attributes in the result	```GET /role-actions?show-hidden=true```
unique	Displays only unique results (like DISTINCT in SQL)	```GET /role-actions?fields=object&unique=true```

26.101.3 Sample Request

```
GET /role-actions/UserServices/UsageRecord
```

26.101.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "instance" : "ANY",
            "name" : "Refund",
            "object" : "UsageRecord",
            "role" : "UserServices"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.102 Add an Action to a Role

In this topic:

26.102.1 Synopsis

```
POST /role-actions
{
    <name> : <value>,...
}
```

26.102.2 Sample Request

```
POST /role-actions
{
    "name" : "Refund",
    "object" : "UsageRecord",
    "role" : "UserServices"
}
```

26.102.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "instance" : "ANY",
            "name" : "Refund",
            "object" : "UsageRecord",
            "role" : "UserServices"
        }
    ],
    "message" : "Successfully created 1 role action",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.103 Remove an Action from a Role

26.103.1 Synopsis

```
DELETE /role-actions/<role>/<object>/<name>
```

26.103.2 Sample Request

```
DELETE /role-actions/UserServices/UsageRecord/Refund
```

26.103.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "instance" : "ANY",
            "name" : "Refund",
            "object" : "UsageRecord",
            "role" : "UserServices""
        }
    ],
    "message" : "Successfully deleted 1 role action",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.104 Query Role Users

In this topic:

26.104.1 Synopsis

```
GET /role-users[/<role>[/<user>]][?<parameter>[&<parameter>...]]
```

26.104.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	`GET /role-users/UserServices?fields=name`
filter	Filters the objects to be returned in the query	`GET /role-users?filter=name=amy`
limit	Limits the results to the number of objects specified	`GET /role-users?limit=100`
offset	Number of objects to skip before starting to return data	`GET /role-users?offset=100`
show-hidden	Includes hidden attributes in the result	`GET /role-users?show-hidden=true`
unique	Displays only unique results (like DISTINCT in SQL)	`GET /role-users?fields=name&unique=true`

26.104.3 Sample Request

```
GET /role-users/UserServices/amy
```

26.104.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "name" : "amy",
            "role" : "UserServices"
        }
    ],
    "status" : "Success"
}
```

Related Topics

26.105 Add a User to a Role

26.105.1 Synopsis

```
POST /role-users
{
    <name> : <value>,...
}
```

26.105.2 Sample Request

```
POST /role-users
{
    "name" : "amy",
    "role" : "UserServices"
}
```

26.105.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "name" : "amy",
            "role" : "UserServices"
        }
    ],
    "message" : "Successfully created 1 role user",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.106 Remove a User from a Role

26.106.1 Synopsis

```
DELETE /role-users/<role>/<user>
```

26.106.2 Sample Request

```
DELETE /role-users/UserServices/amy
```

26.106.3 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "name" : "amy",
            "role" : "UserServices"
        }
    ],
    "message" : "Successfully deleted 1 role user",
    "status" : "Success"
}
```

Related Topics

- 26.96 Roles Resource - page 439
- 26.67 Framework Resources - page 399

26.107 System Resource

This section provides information on the supported actions for the System framework resource.

Supported Actions

Action	HTTP Method	Resource
Query system properties	GET	/system

Related Topics

- 26.67 Framework Resources - page 399

26.108 Query the System

26.108.1 Synopsis

```
GET /system[?<parameter>[&<parameter>...]]
```

26.108.2 Parameters

Parameter	Description	Example
fields	Designates the properties to be returned in the query	```GET /system?fields=version```
show-hidden	Includes hidden attributes in the result	```GET /system?show-hidden=true```

26.108.3 Sample Request

```
GET /system
```

26.108.4 Sample Response

```
{
    "code" : "000",
    "count" : 1,
    "data" : [
        {
            "description" : "Commercial Release",
            "name" : "Moab Accounting Manager",
            "version" : "9.1.3"
        }
    ],
    "status" : "Success"
}
```

Related Topics

Appendix A: Commands Reference

Moab Accounting Manager provides a server daemon and client commands for use by admin-
istrators and end users.

A.0.1 Common Command Options

Most Moab Accounting Manager commands support the following common options.

Option	Description
--help	Brief command option summary.
--format `<output-format>`	Data output format. Valid values include standard, raw and csv. • csv – Fields are delimited by commas (CSV = comma-separated values). Fields containing commas are double-quoted. • raw – Fields are delimited by the pipe character ('|') . • standard (default) – Fields are aligned to fixed-width columns; widths are dynamically calculated based on the widest value in a column (including the header).
--man	Full command documentation.
--site	Obtain response from specified site.
--version	Display product version.

A.0.2 List of Commands

Command	Description
mam-balance	Display balance information
mam-charge	Create a usage charge

Command	Description
mam-create-account	Create a new account
mam-create-chargerate	Create a new charge rate
mam-create-event	Create a new event
mam-create-fund	Create a new fund
mam-create-lien	Create a lien
mam-create-organization	Create a new organization
mam-create-quote	Create a quote template
mam-create-role	Create a new role
mam-create-usagerecord	Create a new usage record
mam-create-user	Create a new user
mam-delete-account	Delete an account
mam-delete-allocation	Delete an allocation or purge stale allocations
mam-delete-chargerate	Delete a charge rate
mam-delete-event	Delete an event
mam-delete-fund	Delete a fund
mam-delete-lien	Delete a lien
mam-delete-notification	Delete a stored notification
mam-delete-organization	Delete an organization
mam-delete-quote	Delete a quote

Command	Description
mam-delete-role	Delete a role
mam-delete-usagerecord	Delete a usage record
mam-delete-user	Delete a user
mam-deposit	Issue a deposit
mam-list-accounts	Query accounts
mam-list-allocations	Query allocations
mam-list-chargerates	Query charge rates
mam-list-events	Query events
mam-list-itemizedcharges	Query charges
mam-list-funds	Query funds
mam-list-liens	Query liens
mam-list-notifications	Query stored notifications
mam-list-organizations	Query organizations
mam-list-quotes	Query quotes
mam-list-roles	Query roles
mam-list-transactions	Query transactions
mam-list-usagerecords	Query usage records
mam-list-users	Query users
mam-modify-account	Modify an account

Command	Description
mam-modify-allocation	Modify an allocation
mam-modify-chargerate	Modify a charge rate
mam-modify-event	Modify an event
mam-modify-fund	Modify a fund
mam-modify-lien	Modify a lien
mam-modify-organization	Modify an organization
mam-modify-quote	Modify a quote
mam-modify-role	Modify a role
mam-modify-usagerecord	Modify a usage record
mam-modify-user	Modify a user
mam-quote	Quote for usage
mam-read-configuration	Query configuration
mam-refund	Issue a usage refund
mam-reserve	Reserve for usage
mam-server	Moab Accounting Manager server
mam-set-password	Set a user password
mam-shell	Interactive shell for MAM
mam-statement	Display fund statement
mam-transfer	Issue a transfer

Command	Description
mam-withdraw	Issue a withdrawal
mybalance	Display personal balance information

A.1 mam-balance

A.1.1 Synopsis

mam-balance [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filterType ExactMatch|Exclusive|NonExclusive] [--ignore-ancestors] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.1.2 Overview

`mam-balance` displays balance information for funds having active allocations.

A.1.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays the balance available to the specified account.

-c	
Format	-c <class_name>
Default	---
Description	Displays the balance available to the specified class.

-g	
Format	-g <group_name>

-g	
Default	---
Description	Displays the balance available to the specified group.

-m	
Format	`-m <machine_name>`
Default	---
Description	Display the s balance available to the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays the balance available to the specified organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays the balance available to the specified user.

--debug	
Format	`--debug`
Default	---
Description	Logs debugging information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Displays the balance for funds where constraints do not conflict with the specified filters. You can use multiple filter options by logically ANDing them together.

--filter-type

Format	`--filter-type ExactMatch|Exclusive|NonExclusive`
Default	`NonExclusive`
Description	Selects the filtering type. • If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. • If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). • If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--format

Format	`--format <output_type>`
Default	standard
Description	Data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--ignore-ancestors

Format	`--ignore-ancestors`
Default	---
Description	Does not include hierarchical ancestor funds in the result.

--long

Format	`--long`
Default	---
Description	Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays the full documentation.

--hours

Format	`--hours`
Default	---

--hours	
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--quiet	
Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show	
Description	Displays only the specified attributes in the order you specified. Valid attributes: ● `Allocated` – Adjusted allocation. This value stores the effective allocated amount based on the initial deposit and subsequent allocation adjustments via deposits, withdrawals or transfers. ● `Available` – Total amount currently available for charging (Balance - Reserved + CreditLimit). ● `Balance` – Sum of active allocation amounts remaining within this fund. It does not take into account current liens. ● `Capacity` – Total expendable amount (Allocated + CreditLimit). ● `Constraints` – Contraints on fund usage. ● `CreationTime` – Time this fund was created. ● `CreditLimit` – Sum of active credit limits within this fund. ● `Deleted` – Boolean indicating whether this fund is deleted. ● `Description` – Fund description ● `Effective` – Effective allocation total not blocked by liens (Balance - Reserved). ● `Id` – Fund id. ● `ModificationTime` – Time this fund was last modified. ● `Name` – Fund name. ● `PercentRemaining` – Percentage of allocation remaining (Balance * 100 / Capacity). ● `PercentUsed` – Percentage of allocation used (Used * 100 / Capacity). ● `RequestId` – Id of the last modifying request. ● `Reserved` – Sum of active lien amounts against this fund. ● `TransactionId` – Id of the last modifying transaction. ● `Used` – Total amount used from this allocation (Allocated - Balance). Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form of `operator(attribute_name) [=alias]`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--wide	
Format	`--wide`
Default	---
Description	Displays multi-valued fields in a single-line, comma-separated format.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.2 mam-charge

A.2.1 Synopsis

mam-charge {-J <instance_name>} [[-j] <usage_record_id>] [-n <designated_name>] [-q <quote_id>] [-l <lien_id>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [-t <actual_duration>] [-s <start_time>] [-e <end_time>] [-x exit_code] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property>=<value>]... [-zt <charge_duration>] [-zs <charge_start_time>] [-z <charge_amount>] [-f <fund_id>] [--incremental] [--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.2.2 Overview

mam-charge charges for resource usage.

A.2.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Account to charge.

-c	
Format	-c <class_name>

-c	
Default	---
Description	Class of queue used.

-C	
Format	`-C <cpu_time>`
Default	---
Description	CPU time used. `cpu_time` may be an expression of the form of `[cumulative_cpu_time]` `[(incremental_cpu_time)]`. • If both `incremental_cpu_time` and `cumulative_cpu_time` are specified, then `incremental_cpu_time` will be used for the charge and `cumulative_cpu_time` will be recorded as the cumulative value used in the usage record. • If only `incremental_cpu_time` is specified, this value will be used for the charge *only* and no cpu time value will be recorded in the usage record. • If only `cumulative_cpu_time` is specified, this value will be used both in the charge *and* recorded in the usage record.

-d	
Format	`-d <description>`
Default	---
Description	Description of the usage.

-D	
Format	`-D <disk>`
Default	---
Description	Amount of disk space used.

-e	
Format	`-e <end_time>`
Default	`Now`
Description	End time for the usage in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-E	
Format	`-E <energy>`
Default	`---`
Description	Amount of energy used.

-f	
Format	`-f <fund_id>`
Default	`---`
Description	Fund ID to charge.

-F	
Format	`-F "{\"<feature_name>\":<feature_count>,...}"`
Default	`---`
Description	Allocated node features. Features represent counts of the node features allocated to the job.

-g	
Format	`-g <group_name>`
Default	`---`
Description	Name of the group to charge.

-j

Format	`[-j] <usage_record_id>`
Default	---
Description	Usage record ID for the charge (if already created with *mam-create-usagerecord*, *mam-quote*, *mam-reserve* or a previous *mam-charge*). Use `-j` to charge an existing usage record if the instance name (such as a job ID) is ambiguous, or if a usage has already been debited and you want to charge an additional amount to the same usage record.

-J

Format	`-J <instance_name>`
Default	---
Description	Instance name (or job ID) for the charge, if known. This can sometimes be non-unique (such as when a resource manager recycles job IDs) and does not always unambiguously identify a usage record to charge. In such cases, look up and specify the usage record ID for the charge.

-l

Format	`-l <lien_id>`
Default	---
Description	Lien ID, which MAM will use to match up the right usage record ID and remove the correct lien, if ambiguous.

-L

Format	`-L "{\"<license_name>\":<license_count>,...}"`
Default	---
Description	Licenses used. Licenses represent software licenses that are used (in integer units).

-m	
Format	`-m <machine_name>`
Default	---
Description	Name of the cluster.

-M	
Format	`-M <memory>`
Default	---
Description	Amount of memory used.

-n	
Format	`-n <designated_name>`
Default	---
Description	User-specified job name.

-N	
Format	`-N <nodes>`
Default	---
Description	Number of nodes used.

-o	
Format	`-o <organization_name>`
Default	---
Description	Organization name.

-P	
Format	`-P <processors>`
Default	---
Description	Number of processors used.

-q	
Format	`-q <quote_id>`
Default	---
Description	Quote MAM should use to determine charge rates.

-Q	
Format	`-Q <quality_of_service_name>`
Default	---
Description	Quality of service used.

-R	
Format	`-R "{\"<resource_name>\":<resource_count>,...}"`
Default	---
Description	Consumable resources allocated. Resources represent consumable resources that may be allocated (in integer units).

--rate	
Format	`--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...`
Default	---

--rate

Description	Charge rate expressions. Multiple charge rate expressions may be passed to the `--rate` option in a comma-delimited list. Alternately, multiple `--rate` options may be specified.

-s

Format	`-s <start_time>`
Default	`---`
Description	Start time for the usage in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

--stage

Format	`--stage <lifecycle_stage>`
Default	`---`
Description	Latest stage in the object's accounting lifecycle (Create, Start, Continue, End).

-t

Format	`-t <actual_duration>`
Default	`---`
Description	Total actual duration (in seconds).

-T

Format	`-T <usage_record_type>`
Default	`---`
Description	Usage record type (Job, Reservation, etc.).

-u	
Format	`-u <user_name>`
Default	---
Description	User name.

-V	
Format	`-V "{\"<variable_name>\":\"<variable_value>\",...}"`
Default	---
Description	Job variables. Variables represent arbitrary variables passed into the job.

-W	
Format	`-W <requested_duration>`
Default	---
Description	Total estimated wallclock duration (in seconds).

-x	
Format	`-x exit_code`
Default	---
Description	Exit code.

-X, --extension	
Format	`-X` or `--extension <property>=<value>`
Default	---

-X, --extension

Description	Extension property. You can specify any number of extra usage properties with the charge.
	When expressing accumulating properties, value may be an expression in the form of `[cumulative_value][(incremental_value)]`.
	- If both `incremental_value` and `cumulative_value` are specified, then `incremental_value` will be used for the charge and `cumulative_value` will be recorded as the cumulative value used in the usage record.
	- If only `incremental_value` is specified, this value will be used for the charge *only* and no cumulative value will be recorded in the usage record.
	- If only `cumulative_value` is specified, this value will be used both in the charge *and* recorded in the usage record.

-z

Format	`-z <charge_amount>`
Default	---
Description	Charge amount if calculated externally.

-zs

Format	`-zs <charge_start_time>`
Default	`Now - <charge_duration>` (if unable to derive by other means)
Description	Start time for the charge in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`. This is *only* needed for incremental charges when the start of the charge interval differs from the original start time *and* is used to determine the appropriate allocation to the charge.

-zt

Format	`-zt <charge_duration>`
Default	---
Description	Incremental duration of the charge (in seconds). This is *only* needed for incremental charges when the incremental duration differs from the total actual duration and is used to compute the incremental charge amount.

-Z	
Format	`-Z "{\"<metric_name>\":<metric_amount>,...}"`
Default	---
Description	Generic metrics. Metrics represent floating point metrics of the job *or* average metrics values across the nodes in the job.

--hours	
Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--itemize	
Format	`--itemize`
Default	---
Description	Returns the composite charge information in the response data. This must be used in conjunction with the --verbose flag to display the data.

--debug	
Format	`--debug`
Default	---
Description	Logs debugging information to the screen.

--help	
Format	`--help`

--help

Default	---
Description	Displays a brief help message.

--incremental

Format	`--incremental`
Default	---
Description	Debits any associated liens instead of removing them.

--man

Format	`--man`
Default	---
Description	Displays the full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.3 mam-create-account

A.3.1 Synopsis

mam-create-account {[-a] <account_name>} [-A | -I] [-o <organization_name>] [-d <description>] [-X, --extension <property>=<value>]... [-u [^|!][+|-]<user_name>,...]... [--create-fund True|False] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.3.2 Overview

mam-create-account creates a new account. Users may be associated with the account. If you turn on auto-generation for the Fund object or assert the --create-fund flag, a fund will automatically be created for the account.

A.3.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Specifies the name of the new account

-A	
Format	-A
Default	---
Description	Activates the account

-d	
Format	-d <description>

-d	
Default	---
Description	Specifies an account description.

-I	
Format	-I
Default	---
Description	Deactivates the account.

-o	
Format	-o <organization_name>
Default	---
Description	Specifies the name of the organization to which the account belongs.

-u	
Format	-u [^|!][+|-]<user_name>[,[^|!][+|-]<user_name>...]
Default	---
Description	Defines user members of the account. The optional caret or exclamation symbol indicates whether the user should be created as an admin (^) or not (!) for the account. The optional plus or minus sign can precede each member to indicate whether the member should be created in the active (+) or inactive (-) state. By default, a user will be created in the active state but not an admin. You may pass multiple users may be passed to the -u option in a comma-delimited list or by specifying multiple -u options.

-X	
Format	-X or --extension <property>=<value>

-X	
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--create-fund	
Format	`--create-fund True|False`
Default	---
Description	Overrides the fund auto-generation setting. Setting this option to `True` creates a default fund for this account. Setting this option to `False` inhibits the creation of a default fund for this account.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Displays modified object details.

--version

Format	--version
Default	---
Description	Displays the product version.

--about

Format	--about
Default	---
Description	Displays product information.

Related Topics

A.4 mam-create-chargerate

A.4.1 Synopsis

mam-create-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_value>] {-z <charge_rate_amount>} [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.4.2 Overview

mam-create-chargerate creates a new charge rate.

A.4.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a charge rate description.

-n	
Format	[-n] <charge_rate_name>
Default	---
Description	Specifies the name of the usage record property for which the rate is charging, such as Processors or QualityOfService.

-x	
Format	-x <charge_rate_value>
Default	---

-x	
Description	Specifies charge rate value. For name-valued charge rates, this is the usage property value corresponding to the rate. For numeric-valued charge rates, this is the range of values corresponding to the rate. A blank value will function as a default charge rate. See Chapter 16: Managing Charge Rates - page 167 for more information.

-z	
Format	`-z <charge_rate_amount>`
Default	---
Description	Specifies the rate for the charge. This is an integer or decimal number and may include operators that indicate how the charge is applies as well as divisors and time-based units. See Chapter 16: Managing Charge Rates - page 167 for more information.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---

--man

Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Displays modified object details.

--version

Format	--version
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 16.2 Creating Charge Rates - page 170
- Appendix A: Commands Reference - page 457

A.5 mam-create-event

A.5.1 Synopsis

mam-create-event [--fire-command <fire_command>] [-s <fire_time>] [-e <end_time>] [--rearm-period <rearm_period>] [--rearm-on-failure <boolean>] [--failure-commmand <failure_command>] [--notify <notification_url>] [--catch-up <boolean>] [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.5.2 Overview

mam-create-event creates a new event.

A.5.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies an event description.

-e	
Format	-e <end_time>
Default	---
Description	Specifies the time that this event becomes inactive in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now.

-s	
Format	-s <fire_time>

-s	
Default	---
Description	Specifies the target time for the event scheduler to trigger the event. The actual fire time may be dependent on the state of the server and will be recorded in the CreationTime property of the corresponding "Event Fire" Transaction. An event may also be fired manually with the mam-shell Event Fire action.

--catch-up	
Format	`--catch-up <boolean>`
Default	`True`
Description	If you set `--catch-up` to `True` and the server was down during the time this event should have fired, the event scheduler will attempt to make up for the past-due events by progressively firing them (rearming based on previous arm time) until it catches up to the present. The actions will still appear to have occurred in the present rather than in the past. If you set it to `False` and the server is brought back up after an outage, the event scheduler will still fire immediately for a past due event, but it will only fire once and then rearm relative to the current time.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--failure-command	
Format	`--failure-command <failure_command>`
Default	---
Description	Specifies the command MAM should execute if the fired command results in an unsuccessful response status. This command is expressed in a serialized form of the request identical to the syntax used in the interactive control program (mam-shell). You must appropriately quote and/or escape the option argument to avoid misinterpretation or alteration by the shell.

--fire-command

Format	`--fire-command <fire_command>`
Default	---
Description	Specifies the command MAM should execute.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--notify

Format	`--notify [+-=][<delivery_method>:][<recipient>]`
Default	Log all event statuses to the Notification table.
Description	Causes MAM to log the result of the fired command. If the term is a –, the notification is sent only on failure. If the term is a +, the notification is sent only on success. Otherwise the notification is always sent. See Chapter 19: Managing Notifications - page 189 for more information about delivery method and recipient.

--quiet

Format	`--quiet`

--quiet	
Default	---
Description	Suppresses headers and success messages.

--rearm-on-failure	
Format	`--rearm-on-failure <Boolean>`
Default	`False`
Description	If you set `--rearm-on-failure` to `False`, MAM will not rearm the event if the command was unsuccessful. If you set it to `True`, the event will be evaluated for rearming even if the command response has a status of Failure.

--rearm-period	
Format	`--rearm-period <period>[[@instant][~|^]|!]`
Default	---
Description	Specifies when the event will be rearmed. This period expression is in the form of `<period>` `[[@instant][~|^]|!]`. The `<period>` is expressed as an integer number followed by a designator of minute(s), hour(s), day(s), week(s), month(s), or year(s). For example, the period might be 1 day, 2 hours, or 5 minutes. The optional `instant` locks the period to a specific instant within the time period such as 1 day @ hour 12 or 1 month @ day 3. The modifiers indicate whether the time period should be relative to now (`!`), relative to the start of this (`~`) designator (month, minute, or other unit), or relative to the start of the first (`^`) designator (month, minute, or other unit). For example, assuming the FireTime was 7:15, if you specified `4 hours !` as the rearm period it would be rearmed at 11:15; if you specified `4 hours ~` as the rearm period, it would be rearmed at 11:00; and if you specified `4 hours ^` as the rearm period it would be rearmed at 8:00.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.6 mam-create-fund

A.6.1 Synopsis

mam-create-fund [-n <fund_name>] [--priority <fund_priority>] [--default-deposit <deposit_amount>] [-d <description>] [-X, --extension <property>=<value>]... [-u <user_name>,...]... [-g <group_name>,...]... [-a <account_name>,...]... [-o <organization_name>,...]... [-c <class_name>,...]... [-m <machine_name>,...]... [--constraint <constraint_name>=[!]<constraint_value>,...]... [--parent <parent_fund_id>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.6.2 Overview

mam-create-fund creates new funds. MAM automatically generates a new ID for the fund. It essentially creates a new container into which time-bounded credits valid toward a specific set of constraints can be later credited and debited.

A.6.3 Options

-a	
Format	-a <account_name>[,<account_name>...]
Default	---
Description	Specifies the account required by the fund. You may pass multiple accounts to the -a option in a comma-delimited list or by specifying multiple -a options may be specified.

-c	
Format	-c <class_name>[,<class_name>...]
Default	---
Description	Specifies the class or queue required by the fund. You may pass multiple classes may be passed to the -c option in a comma-delimited list or by specifying multiple -c options.

--constraint	
Format	`--constraint <constraint_name>=<constraint_value>[,<constraint_name>=<constraint_value>...]`
Default	---
Description	Specifies a constraint for the fund. The constraint value may be a perl5 regular expression. You may prepend an exclamation point to the constraint value to express a negation of the constraint. You may specify multiple constraint options. For example, `--constraint User=amy --constraint Machine=colony` will make the credits in this fund valid only for the user `amy` on the machine colony. You m ay pass multiple constraints to the `--constraint` option in a comma-delimited list or specify multiple `--constraint`.

-d	
Format	`-d <description>`
Default	---
Description	Specifies a fund description.

--default-deposit	
Format	`--default-deposit <default_amount>`
Default	---
Description	Sets the default amount for any deposit that is made to this fund that does not specify a deposit amount. • A zero value will result in the creation of an allocation with a zero balance (or add nothing if an allocation already exists and a reset is not being requested). • A negative value can be used to stipulate that the allocations in the fund should be ended if the fund is reset. • An empty value ('') or NULL can be used to stipulate that no change will be made to the allocations if the fund is reset.

-g	
Format	`-g <group_name>[,<group_name>...]`

-g	
Default	---
Description	Specifies the group required by the fund. You can pass multiple groups to the -g option in a comma-delimited list or by specifying multiple -g options.

-m	
Format	-m <machine_name>[,<machine_name>...]
Default	---
Description	Specifies the machine (cluster) the fund requires. You may pass multiple machines to the -m option in a comma-delimited list or by specifying multiple -m options .

-n	
Format	-n <fund_name>
Default	---
Description	Specifies the fund name.

-o	
Format	-o <organization_name>[,<organization_name>...]
Default	---
Description	Specifies the organization the fund requiresd. You may pass multiple organizations may be passed to the -o option in a comma-delimited list or by specifying multiple -o options.

--parent	
Format	--parent <parent_fund_id>
Default	---

--parent

Description	Associates the newly created fund as a child of the specified parent fund.

--priority

Format	`--priority <fund_priority>`
Default	---
Description	Specifies the fund priority.

-u

Format	`-u <user_name>[,<user_name>...]`
Default	---
Description	Specifies the user required by the fund. You may pass mulltiple users may be passed to the `-u` option in a comma-delimited list or by specifying multiple `-u` options.

-X, --extension

Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.7 mam-create-lien

A.7.1 Synopsis

mam-create-lien [-J <instance_name>] [-s <start_time>] {-e <end_time> | -t <lien_duration>} [-d <description>] [-X, --extension <property>=<value>]... {-A <allocation_id><-<fund_id>=<sublien_ amount>,...}... [--debug] [--site <site_name>] [--help] [--man] [-- quiet] [--verbose] [--version] [--about]

A.7.2 Overview

mam-create-lien creates a lien against specified allocations. MAM will create a lien object and its allocation associations. Unlike mam-reserve, MAM will not return a calculated lien amount or create a usage record with the lien.

> ⓘ This command bypasses the normal mechanisms that prevent more liens from being placed against an allocation than it can support.

A.7.3 Options

-A	
Format	-A <allocation_id><-<fund_id>=<sublien_amount>[,<allocation_id><-<fund_ id>=<sublien_amount>...]
Default	---
Description	Creates subliens against the specified allocations. You must specify at least one allocation expression with the lien. You may pass multiple allocation expressions to the -A option in a comma-delimited list or by specifying multiple -A options.

-d	
Format	-d <description>
Default	---
Description	Specifies a description for the lien.

-e	
Format	`-e <end_time>`
Default	`Now`
Description	Specifies the expiration time for the lien in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-J	
Format	`[-J] <instance_name>`
Default	`---`
Description	Specifies the instance name (e.g job id) for the lien.

-s	
Format	`-s <start_time>`
Default	`Now`
Description	Specifies a new start time for the lien in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-t	
Format	`-t <lien_duration>`
Default	Lien end time minus start time
Description	Specifies the duration of the lien in seconds.

-X, --extension	
Format	`-X or --extension <property>=<value>`

-X, --extension

Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.8 mam-create-organization

A.8.1 Synopsis

mam-create-organization {[-o] <organization_name>} [-d <description>]
[-X, --extension <property>=<value>]... [--debug] [--site <site_man>]
[--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.8.2 Overview

`mam-create-organization` creates a new organization.

A.8.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a description for the organization.

-o	
Format	-o <organization_name>
Default	---
Description	Specifies the name of the organization.

-X, --extension `<property>`	
Format	-X or --extension <property>=<value>
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.9 mam-create-quote

A.9.1 Synopsis

mam-create-quote [[--pin] [-J <instance_name>] | --nopin] [-s <start_time>] {-e <end_time> | -t <quote_duration>} [-d <description>] [-X, --extension <property>=<value>]... {--rate <charge_rate_name> [{<charge_rate_value>}]=<charge_rate_amount>,...}... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.9.2 Overview

mam-create-quote creates a new chargeable quote template. MAM will create a quote object and its associated charge rates. Instances referencing the quote will use the override charge rates specified in the command. Unlike mam-quote, *mam-create-quote* will not return a calculated quote amount or create a usage record with the quote.

A.9.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a description of the quote.

-e	
Format	-e <end_time>
Default	---
Description	Specifies the expiration time for the quote template in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`. The rates associated with this quote may not be claimed after this time. If you do not specify an end time but did specify a duration, MAM will calculate the end time as start time + duration. If you specify both end time and duration but they are inconsistent, MAM will ignore the duration specification.

-J	
Format	`-J <instance_name>`
Default	---
Description	Specifies the instance name (e.g job id) for the quote. You cannot specify an instance name if the quote is unpinned.

--rate	
Format	`--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>` `[,<charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>...]`
Default	---
Description	Charge rate expressions. Multiple charge rate expressions may be passed to the `--rate` option in a comma-delimited list. Alternately, multiple `--rate` options may be specified.

-s	
Format	`-s <start_time>`
Default	`Now`
Description	Specifies a beginning time for the quote template in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`. The rates associated with this quote may not be claimed before this time.

-t	
Format	`-t <quote_duration>`
Default	---
Description	Specifies the amount of time in seconds the rates in the quote template may be used. MAM uses the duration to calculate an end time (start time + duration) as an alternative to specifying the end time.

-X, --extension `<property>`

Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--nopin

Format	`--nopin`
Default	Not set

--nopin

Description	Indicates that the quote is not to be pinned to a specific instance. An instance may use an unpinned quote while the quote is active.

--pin

Format	`--pin`
Default	Set
Description	Indicates that the quote will be pinned to a specific instance. If you do not specify the instance when you create the quote, the first instance to claim it will become the pinned instance. Once a quote is pinned to a particular instance, no other instances may use the quote.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 13.3 Creating Quote Templates - page 135
- Appendix A: Commands Reference - page 457

A.10 mam-create-role

A.10.1 Synopsis

mam-create-role {[-r] <role_name>} [-d <description>] [-u <user_name>,...]... [-A "<object_name>-><action_name>[{<instance_name>}]",...]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.10.2 Overview

mam-create-role creates a new role. You can associate users and actions with the role at creation time.

A.10.3 Options

-A	
Format	-A "<object_name>-><action_name>[{<instance_name>}]][,<object_name>-><action_name>[{<instance_name>}]...]"
Default	ANY
Description	Adds actions to the role. You must specify the object, action and instance in the form shown. You may pass multiple actions to the -A option in a comma-delimited list or specify multiple -A options.

-d	
Format	-d <description>
Default	---
Description	Specifies a role description.

-r	
Format	[-r] <role_name>

-r	
Default	---
Description	Specifies the name of the new role.

-u	
Format	`-u <user_name>[,<user_name>...]`
Default	---
Description	Adds users to the role. You may pass multiple users to the `-u` option in a comma-delimited list or specify multiple `-u` options.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.11 mam-create-usagerecord

A.11.1 Synopsis

mam-create-usagerecord {-J <instance_name>} [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [-t <actual_duration>] [-s <start_time>] [-e <end_time>] [-x <exit_code>] [--stage <lifecycle_stage>] [-d <description>] [-X --extension <property>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.11.2 Overview

mam-create-usagerecord creates a new usage record.

A.11.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Account name.

-c	
Format	-c <class_name>
Default	---
Description	Class or queue.

-C	
Format	`-C <cpu_time>`
Default	---
Description	CPU time used.

-d	
Format	`-d <description>`
Default	---
Description	Description of the usage.

-D	
Format	`-D <disk>`
Default	---
Description	Amount of disk used.

-e	
Format	`-e <end_time>`
Default	---
Description	Date and time the usage ended in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-E	
Format	`-E <energy>`
Default	`---`

-E	
Description	Amount of energy used.

-F	
Format	`-F "{\"<feature_name>\":<feature_count>,...}"`
Default	`---`
Description	Allocated node features. Features represent counts of the node features allocated to the job.

-g	
Format	`-g <group_name>`
Default	`---`
Description	Group name.

-L	
Format	`-L "{\"<license_name>\":<license_count>,...}"`
Default	`---`
Description	Licenses used. Licenses represent software licenses that are used (in integer units).

-J	
Format	`-J <instance_name>`
Default	`---`
Description	Instance name or job ID of the new usage record.

-m	
Format	`-m <machine_name>`
Default	---
Description	Name of the cluster.

-M	
Format	`-M <memory>`
Default	---
Description	Amount of memory used.

-n	
Format	`-n <designated_name>`
Default	---
Description	User-specified job name.

-N	
Format	`-N <nodes>`
Default	---
Description	Number of nodes used.

-o	
Format	`-o <organization_name>`
Default	---
Description	Organization name.

-P

Format	`-P <processors>`
Default	---
Description	Number of processors used.

-Q

Format	`-Q <quality_of_service>`
Default	---
Description	Quality of service used.

-R

Format	`-R "{\"<resource_name>\":<resource_count>,...}"`
Default	---
Description	Consumable resources allocated. Resources represent consumable resources that may be allocated (in integer units).

-s

Format	`-s <start_time>`
Default	---
Description	Date and time the usage started in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now.`

--stage

Format	`--stage <lifecycle_stage>`

--stage	
Default	---
Description	Latest stage in the object's accounting lifecycle (Create, Start, Continue, End).

-t	
Format	`-t <actual_duration>`
Default	---
Description	Total actual duration (in seconds).

-T	
Format	`-T <usage_record_type>`
Default	---
Description	Usage record type (Job or reservation, for example).

-u	
Format	`-u <user_name>`
Default	---
Description	User name.

-V	
Format	`-V "{\"<variable_name>\":\"<variable_value>\",...}"`
Default	---
Description	Job variables. Variables represent arbitrary variables passed into the job.

-W

Format	`-W <requested_duration>`
Default	---
Description	Total estimated wallclock duration (in seconds).

-x

Format	`-x <exit_code>`
Default	---
Description	Exit code.

-X, --extension

Format	`-X or --extension <property>=<value>`
Default	---
Description	Extension property. You can specify any number of extra field assignments.

-Z

Format	`-Z "{\"<metric_name>\":<metric_amount>,...}"`
Default	---
Description	Generic metrics. Metrics represent floating point metrics of the job *or* average metrics values across the nodes in the job.

--debug

Format	`--debug`
Default	---

--debug

Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.12 mam-create-user

A.12.1 Synopsis

mam-create-user {[-u] <user_name>} [-A | -I] [-n <common_name>] [--phone <phone_number>] [--email <email_address>] [-a <default_account>] [-d <description>] [-X, --extension <property>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.12.2 Overview

mam-create-user creates a new user.

A.12.3 Options

-a	
Format	-a <default_account>
Default	---
Description	Account MAM will charge when no account is specified.

-A	
Format	-A
Default	---
Description	Activates the user.

-d	
Format	-d <description>
Default	---
Description	User description.

--email

Format	`--email <email_address>`
Default	---
Description	Email address.

-l

Format	`-I`
Default	---
Description	Deactivates the user.

n

Format	`-n <common_name>`
Default	---
Description	Common name for the user.

--phone

Format	`--phone <phone_number>`
Default	---
Description	Phone number.

-u

Format	`[-u] <user_name>`
Default	---
Description	User's ID or name.

-X, --extension `<property>`

Format	-X or --extension <property>=<value>
Default	---
Description	Extension property. You can specify any number of extra field assignments.

--debug

Format	--debug
Default	---
Description	Logs debug information to the screen.

--help

Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	--man
Default	---
Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.13 mam-delete-account

A.13.1 Synopsis

mam-delete-account {[-a] <account_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.13.2 Overview

mam-delete-account deletes an account.

A.13.3 Options

-a	
Format	[-a] <account_name>
Default	---
Description	Specifies the name of the account to be deleted.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.14 mam-delete-allocation

A.14.1 Synopsis

mam-delete-allocation {-I | {[-i] <allocation_id>}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.14.2 Overview

`mam-delete-allocation` deletes an allocation or purges stale allocations.

A.14.3 Options

-i	
Format	`[-i] <allocation_id>`
Default	---
Description	Specifies the allocation to be deleted.

-l	
Format	`-I`
Default	---
Description	Deletes inactive allocations.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	--man
Default	---
Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.15 mam-delete-chargerate

A.15.1 Synopsis

mam-delete-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_value>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.15.2 Overview

mam-delete-chargerate deletes a charge rate.

A.15.3 Options

-n	
Format	[-n] <charge_rate_name>
Default	---
Description	Specifies the charge rate to delete.

-x	
Format	-x <charge_rate_value>
Default	---
Description	Specifies the charge rate value to delete. If you do not specify a value, MAM will only delete a charge rate with an empty value.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 16.5 Deleting Charge Rates - page 176
- Appendix A: Commands Reference - page 457

A.16 mam-delete-event

A.16.1 Synopsis

mam-delete-event {[-E] <event_id>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.16.2 Overview

mam-delete-event deletes an event.

A.16.3 Options

-E	
Format	[-E] <event_id>
Default	---
Description	Specifies the ID of the event to be deleted.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.17 mam-delete-fund

A.17.1 Synopsis

mam-delete-fund {[-f] <fund_id>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.17.2 Overview

mam-delete-fund deletes a fund.

A.17.3 Options

-f	
Format	[-f] <fund_id>
Default	---
Description	Specifies the fund to be deleted.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.18 mam-delete-lien

A.18.1 Synopsis

mam-delete-lien {-I | {-J <instance_name>} | {[-l] <lien_id>}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.18.2 Overview

mam-delete-lien deletes a lien or purges stale liens.

A.18.3 Options

-I	
Format	-I
Default	---
Description	Deletes expired liens.

-J	
Format	-J <instance_name>
Default	---
Description	Specifies that the liens with the specified instance name,, or job ID, will be deleted.

-l	
Format	[-l] <lien_id>
Default	---
Description	Specifies the lien to be deleted.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.19 mam-delete-notification

A.19.1 Synopsis

mam-delete-notification {[-N] notification_id} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.19.2 Overview

mam-delete-notification deletes a stored notification.

A.19.3 Options

-N	
Format	-N <notification_id>
Default	---
Description	Deletes expired liens.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	--man
Default	---
Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Display modified object details.

--version

Format	--version
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.20 mam-delete-organization

A.20.1 Synopsis

mam-delete-organization {[-o] <organization_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.20.2 Overview

`mam-delete-organization` deletes an organization.

A.20.3 Options

-o	
Format	`[-o] <organization_name>`
Default	---
Description	Specifies the organization to delete.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	--man
Default	---
Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Display modified object details.

--version

Format	--version
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.21 mam-delete-quote

A.21.1 Synopsis

mam-delete-quote {-I | {[-q] <quote_id>}} [--debug] [--site <site_
name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.21.2 Overview

mam-delete-quote deletes a quote or purges expired quotes.

A.21.3 Options

-I	
Format	-I
Default	---
Description	Deletes expired quotes.

-q	
Format	[-q] <quote_id>
Default	---
Description	Specifies the quote to be deleted.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 13.6 Deleting Quotes - page 138
- Appendix A: Commands Reference - page 457

A.22   mam-delete-role

A.22.1 Synopsis

mam-delete-role {[-r] <role_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.22.2 Overview

mam-delete-role deletes a role.

A.22.3 Options

-r	
Format	[-r] <role_name>
Default	---
Description	Specifies the role to delete.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.23 mam-delete-usagerecord

A.23.1 Synopsis

mam-delete-usagerecord {[-j] <usage_record_id> | -J <instance_name>}
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose]
[--version] [--about]

A.23.2 Overview

mam-delete-usagerecord deletes a usage record.

A.23.3 Options

-j

Format	[-j] <usage_record_id>
Default	---
Description	Specifies the ID of the usage record to delete. Instance names can be non-unique, because resource managers often recycle job IDs). This option enables specifying a unique usage record using the unique identifier.

-J

Format	-J <instance_name>
Default	---
Description	Specifies the instance name (e.g. job id) to delete. Since instance names are assigned by externally and may be non-unique (such as job IDs assigned by a resource manager), all usage records with the specified instance name will be deleted.

--debug

Format	--debug

--debug

Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.24 mam-delete-user

A.24.1 Synopsis

mam-delete-user {[-u] <user_name>} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.24.2 Overview

mam-delete-user deletes a user.

A.24.3 Options

-u	
Format	[-u] <user_name>
Default	---
Description	Specifies the name of the user to delete.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Display modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.25 mam-deposit

A.25.1 Synopsis

mam-deposit [-f <fund_id>] [-i <allocation_id>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filterType ExactMatch|Exclusive|NonExclusive] [[-z] <deposit_amount>] [-L <credit_limit>] [-s <start_time>] [-e <end_time>] [--reset] [-d <description>] [--create-fund True|False] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.25.2 Overview

mam-deposit makes time-bound deposits into funds. After applying all filter options, if there is exactly one debitable fund for the specified criteria, a deposit will be made into that fund. If multiple funds match the specified criteria, a list of matching funds will be displayed, and you will be prompted to respecify the deposit against one of the enumerated funds. If no funds match your criteria, if auto-generation is turned on for the fund object, or the `--create-fund` flag is asserted, a fund will be created and a deposit made into it; otherwise, the deposit will fail (the fund will need to be created with `mam-create-fund`).

The --reset option can be used to end the current allocation and create a new allocation with the deposit.

- If an amount is not specified for the deposit, the fund's default deposit amount will be used.

- A zero amount or a default deposit amount will result in the creation of an allocation with a zero balance (or add nothing if an allocation already exists and a reset is not being requested).

- A negative default deposit amount can be used to stipulate that the allocations in the fund should be ended if the fund is reset.

- An empty default deposit amount stipulates that no change will be made to the allocations if the fund is reset.

A.25.3 Options

-a	
Format	-a <account_name>

-a	
Default	---
Description	Restricts the fund for the deposit to one usable by the specified account.

-c	
Format	`-c <class_name>`
Default	---
Description	Restricts the fund for the depositto one usable by the specified class.

-d	
Format	`-d <description>`
Default	---
Description	Specifies the reason for the deposit. The annotation applies to the transaction description (seen via `mam-list-transactions`), not the allocation description.

-e	
Format	`-e <end_time>`
Default	`Infinity`
Description	Specifies the end time for the allocation to be credited in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-f	
Format	`-f <fund_id>`
Default	`Infinity`

-f

Description	Specifies the ID of the fund into which the deposit will be made.

-g

Format	-g <group_name>
Default	---
Description	Specifies that the fund for the deposit should be restricted to one usable by the specified group.

-i

Format	-i <allocation_id>
Default	---
Description	Specifies the allocation in which to make the deposit. This option is incompatible with the -L option.

-L

Format	-L <credit_limit>
Default	---
Description	Creates a new allocation with the specified credit limit. This option is incompatible with the -i option.

-m

Format	-m <machine_name>
Default	---
Description	Restricts the fund for the deposit to one usable by the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Restricts the fund for the deposit to one usable by the specified organization.

-s	
Format	`-s <start_time>`
Default	`Infinity`
Description	Specifies the start time for the allocation to be credited in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-u	
Format	`-u <user_name>`
Default	---
Description	Restricts the fund for the deposit to one usable by the specified user.

-z	
Format	`[-z] <deposit_amount>`
Default	---
Description	Specifies the amount to deposit.

--create-fund	
Format	`--create-fund True|False`
Default	---

--create-fund

Description	Overrides the fund auto-generation setting. Setting this option to `True` creates a default fund for this deposit. Setting this option to `False` inhibits the creation of a default fund for this deposit.

--hours

Format	`--hours`
Default	---
Description	Treats currency as specified in hours. In systems where the currency is measured in resource-seconds (like processor-seconds), this option allows you to specify the amount and credit limit in resource-hours.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Restricts the fund to one where constraints do not conflict with the specified filters. For example, `mam-modify-fund --filter User=amy` restricts the fund to one usable by the user `amy`. You may specify multiple filter options by logically ANDing them together.

--filter-type

Format	`--filter-type ExactMatch|Exclusive|NonExclusive`
Default	`NonExclusive`

--filter-type

Description	Specifies the filtering type.
	• If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints.
	• If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied).
	• If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--reset

Format	`--reset`

--reset

Default	---
Description	Ends the current allocation and creates a new allocation with the deposit.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 10.5   Making Deposits - page 101
- Appendix A: Commands Reference - page 457

A.26 mam-list-accounts

A.26.1 Synopsis

mam-list-accounts [[-a] <account_pattern>] [-A | -I] [-o <organization_name>] [-X, --extension <property>=<value>]... [-u <user_name>] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.26.2 Overview

mam-list-accounts displays account information.

You can customize the fields this command displays by default by setting the `account.show` configuration parameter in `mam-client.conf`.

A.26.3 Options

-a	
Format	`[-a] <account_pattern>`
Default	---

Description	Displays only accounts matching the pattern. Supported wildcards:
	    * Matches any number of characters.
	    ? Matches a single character.
	If no pattern is specified then all accounts are displayed.

-A	
Format	`-A`
Default	---
Description	Displays only active accounts.

-I	
Format	`-I`
Default	---
Description	Displays only inactive accounts.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only accounts belonging to the specified organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only accounts that have the specified user as a member.

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format

Format	`--format csv|raw|standard`
Default	standard
Description	Specifies the data output format.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--long

Format	`--long`
Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the specified order. Valid attributes: • `Active` – Boolean indicating whether this account is active or not. • `CreationTime` – Time this account was created. • `Deleted` – Boolean indicating whether this account is deleted or not. • `Description` – Account description. • `ModificationTime` – Time this account was last modified. • `Name` – Account name. • `Organization` – Organization to which the account belongs. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction. • `Users` – List of users belonging to the account. A caret prefixing a user name indicates that the user is an account admin. A minus sign prefixing a user name indicates that the user is an inactive member of the account.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`

--version	
Default	---
Description	Displays the product version.

--wide	
Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line, comma-separated format.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.27 mam-list-allocations

A.27.1 Synopsis

mam-list-allocations [[-i] <allocation_id>] [-f <fund_id>] [-A | -I |
{[-s <start_time>] [-e <end_time>]}] [-X, --extension
<property>=<value>]... [-u <user_name>] [-g <group_name>] [-
a <account_name>] [-o <organization_name>] [-c <class_name>] [-
m <machine_name>] [--filter <filter_name>=<filter_value>]... [--
filter-type ExactMatch|Exclusive|NonExclusive] [--include-ancestors]
[--full] [--show <attribute_name>,...] [--format csv|raw|standard] [-
-hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--
version] [--about]

A.27.2 Overview

mam-list-allocations displays allocation information.

You can customize the fields this command displays by defaul by setting the allocation.show configuration parameter in mam-client.conf.

A.27.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only allocations usable by the specified account.

-A	
Format	-A
Default	---
Description	Displays only active allocations.

-c	
Format	`-c <class_name>`
Default	---
Description	Displays only allocations usable by the specified class.

-e	
Format	`-e <end_time>`
Default	---
Description	Displays only allocations that start before the specified end time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-f	
Format	`-f <fund_id>`
Default	---
Description	Displays only the allocations associated with the specified fund.

-g	
Format	`-g <group_name>`
Default	---
Description	Displays only allocations usable by the specified group.

-i	
Format	`[-i] <allocation_id>`
Default	---

-i	
Description	Displays only the allocation with the specified ID.

-l	
Format	`-I`
Default	---
Description	Displays only inactive allocations.

-m	
Format	`-m <machine_name>`
Default	---
Description	Displays only allocations usable by the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only accounts usable to the specified organization.

-s	
Format	`-s <start_time>`
Default	---
Description	Displays only allocations that end after the specified start time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only allocations usable by the specified user.

-X	
Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter	
Format	`--filter <filter_name>=<`*`filter_value`*`>`
Default	---
Description	Displays allocations where fund constraints comply with the specified filters. For example, `mam-list-funds --filter User=amy` displays funds usable by the user `amy`. You can specify multiple filter options by logically ANDing them together.

--filter-type	
Format	`--filter-type ExactMatch|Exlusive|NonExclusive`

--filter-type

Default	NonExclusive
Description	Specifies the filtering type. • If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. • If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). • If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--format

Format	--format csv|raw|standard
Default	standard
Description	Specifies a data output format.

--hours

Format	--hours
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--full

Format	--full
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--include-ancestors

Format	`--include-ancestors`
Default	---
Description	Includes ancestors of the selected allocations.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show	
Description	Displays only the specified attributes in the order you specify. Valid attributes: `Active` – Boolean indicating whether this allocation is active or not. `Adjustments` – Total of subsequent adjustments to the initial deposit via deposits, withdrawals or transfers (Allocated - InitialDeposit). `Allocated` – Adjusted allocation. This value stores the effective allocated amount based on the initial deposit and subsequent allocation adjustments via deposits, withdrawals or transfers. `Available` – Amount currently available for charging. If theallocation is active, this is Remaining - Reserved + CreditLimit. If the allocation is inactive, this is zero. `Balance` – Active allocation balance. If the allocation is active, this is the remaining allocation amount (Remaining). If the allocation is inactive, this is zero. `Capacity` – Total expendable amount (Allocated + CreditLimit). `CreationTime` – Time this allocation was created. `CreditLimit` – Determines how far in the negative this allocation is permitted to be used (enforced in quotes and liens). `Deleted` – Boolean indicating whether this allocation is deleted or not. `Description` – Allocation description. `Effective` – Effective balance not blocked by liens. If the.allocation is active, this is Remaining - Reserved. If the allocation is inactive, this is zero. `EndTime` – Time this allocation becomes inactive. `Fund` – Fund ID. `FundName` – Fund name. `Id` – Allocation ID. `InitialDeposit` – Amount of the first deposit into this allocation. `ModificationTime` – Time this allocation was last modified. `PercentRemaining` – Percentage of allocation remaining (remaining * 100 / Capacity). `PercentUsed` – Percentage of allocation used (Used * 100 / Capacity). `Remaining` – Remaining allocation amount. `RequestId` – ID of the last modifying request. `Reserved` – Sum of active lien amounts against this allocation. `StartTime` – Time this allocation becomes active. `TransactionId` – ID of the last modifying transaction. `Used` – Amount used from this allocation (Allocated - Remaining). Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form of `operator(attribute_name)[=alias]`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.28 mam-list-chargerates

A.28.1 Synopsis

mam-list-chargerates [[-n] <charge_rate_name>] [-x <charge_rate_value>] [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.28.2 Overview

mam-list-chargerates displays charge rate information.

A.28.3 Options

-n	
Format	[-n] <charge_rate_name>
Default	---
Description	Displays only charge rates of the specified name.

-x	
Format	-x <charge_rate_value>
Default	---
Description	Displays only charge rates having the specified value.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--format

Format	`--format <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the order specified. Valid attributes: • `Amount` – Charge rate amount. The amount is an integer or decimal number and may include operators indicating how to apply the charge rate as well as divisors and time-based units. See Managing Charge Rates for more information. • `CreationTime` – Time this charge rate was created. • `Deleted` – Boolean indicating whether this charge rate is deleted or not. • `Description` – Charge rate description. • `ModificationTime` – Time this charge rate was last modified. • `Name` – Charge rate name (such as Processors or License). • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction. • `Value` – Charge rate value. For name-valued charge rates this is the usage property value corresponding to the rate. For numeric-valued charge rates this is the range of values corresponding to the rate. A blank value will function as a default charge rate. See Managing Charge Rates the for more information.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 16.3 Querying Charge Rates - page 174
- Appendix A: Commands Reference - page 457

A.29 mam-list-events

A.29.1 Synopsis

mam-list-events [[-E] <event_id>] [-s <start_time>] [-e <end_time>]
[--full] [--show <attribute_name>,...] [--format <csv|raw|standard>]
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version]
[--about]

A.29.2 Overview

mam-list-events displays event information.

You can customize the fields this command displays by default by setting the event.show configuration parameter in mam-client.conf.

A.29.3 Options

-e	
Format	-e <end_time>
Default	---
Description	Displays events with a prospective fire time occurring before the specified time in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now.

-E	
Format	[-E] <event_id>
Default	---
Description	Displays only the event with the specified ID.

-s	
Format	-s <start_time>

-s

Default	---
Description	Displays events with a prospective fire time occurring after the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format

Format	`--format csv|raw|standard`
Default	standard
Description	Specifies a data output format.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show

Description	Displays only the specified attributes in the specified order. Valid attributes:

- `ArmTime` – Time the event was last armed or fired. This field is used as a reference time to be able to derive how long the event has been waiting to happen. This field will be initially set to mark the moment the first FireTime is set and updated thereafter to indicate the last time the event was fired. In the case where an event does not have a FireTime set, this field may be set manually and used in a similar manner. If we consider the time between event firings as "laps," this could be thought of as the Lap Start Time.

- `CatchUp` – If set to `True` and MAM was down during the time this event should have fired, MAM will attempt to make up for the past due events by progressively firing them (rearming based on previous arm time) until catching up to the present. The actions will still show as having occurred in the present rather than in the past. If set to `False`, and MAM is brought back up after an outage, MAM will still fire immediately for a past due event, but it will only fire once and then rearm relative to the current time.

- `CreationTime` – Time this event was created.

- `Deleted` – Boolean indicating whether this event is deleted or not.

- `Description` – Event description.

- `EndTime` – Time after which an event having a rearm period will be deleted.

- `FailureCommand` – Serialized MAM request string to be executed if the fired command results in an unsuccessful response status. They syntax is the same as used to invoke commands within the mam-shell prompt.

- `FireCommand` – Serialized MAM request string to be executed when the event is fired. They syntax is the same as used to invoke commands within the mam-shell prompt.

- `FireTime` – Target time for the event to be triggered. The actual fire time may be dependent on the state of the server and will be recorded in the CreationTime property of the corresponding "Event Fire" Transaction.

- `Id` – Event ID.

- `ModificationTime` – Time this event was last modified.

- `Notify` – Expression specifying where to send a notification of the response for the fire command and the failure command. The notification expression is of the form: `[+-=][delivery_method:][recipient][,[+-=][delivery_method:][recipient]]*` (For example, `-store:amy`). If the term is a `–`, the notification is sent only on failure. If the term is a `+`, the notification is sent only on success. Otherwise the notification is always sent. There can be multiple notify expressions separated by a comma.

- `RearmPeriod` – Time period expression specifying when the event will be rearmed. This period expression is of the form: `period[[@instant][~|^]|!]` where period may be something like 1 day, 2 hours, or 5 minutes. Instant locks the period to a specific instant within the time period such as 1 day @ hour 12 or 1 month @ day 3. The modifiers indicate whether the time period should be relative to now (`!`), or relative to the start of this (`~`) designator (month or minute, etc.), or relative to the start of the first (`^`) designator (month or minute, etc.).

- `RearmOnFailure` – If set to `False`, the event will not be rearmed if the command was unsuccessful. If set to `True`, the event will be evaluated for rearming even if the command response has a status of Failure. The standard default value for this boolean is `False`.

- `RequestId` – ID of the last modifying request.

- `TransactionId` – ID of the last modifying transaction.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.30 mam-list-funds

A.30.1 Synopsis

mam-list-funds [[-f] <fund_id>] [-A | -I] [-n <fund_name>] [-X, --
extension <property>=<value>]... [-u <user_name>] [-g <group_name>]
[-a <account_name>] [-o <organization_name>] [-c <class_name>] [-
m <machine_name>] [--filter <filter_name>=<filter_value>]... [--
filter-type ExactMatch|Exclusive|NonExclusive] [--full] [--
show <attribute_name>,...] [--long] [--wide] [--format
csv|raw|standard] [--hours] [--debug] [--site <site_man>] [--help] [-
-man] [--quiet] [--version] [--about]

A.30.2 Overview

mam-list-funds displays fund information.

You can customize the fields this command displays by default by setting the fund.show configuration parameter in mam-client.conf.

A.30.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only funds valid toward the specified account.

-A	
Format	-A
Default	---
Description	Displays funds with active allocations.

-c	
Format	`-c <class_name>`
Default	---
Description	Displays only funds usable by the specified class.

-f	
Format	`[-f] <fund_id>`
Default	---
Description	Displays only the funds with the specified ID.

-g	
Format	`-g <group_name>`
Default	---
Description	Displays only funds usable by the specified group.

-l	
Format	`-I`
Default	---
Description	Displays only funds with inactive allocations.

-m	
Format	`-m <machine_name>`
Default	---
Description	Displays only funds valid toward the specified machine.

-n	
Format	`-n <fund_name>`
Default	---
Description	Displays only funds with the specified name.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only funds valid toward the specified organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only funds valid toward the specified user.

-X	
Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter	
Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Displays funds where constraints do not conflict with the specified filters. For example, `mam-list-funds -f User=amy` displays funds usable by the user `amy`. You may specify multiple filter options by logically ANDing them together.

--filter-type	
Format	`--filter-type ExactMatch|Exlusive|NonExclusive`
Default	`NonExclusive`
Description	Selects the filtering type. • If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. • If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). • If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--format	
Format	`--format csv|raw|standard <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full	
Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--long

Format	`--long`
Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--hours

Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--quiet

Format	`--quiet`
Default	---

--quiet	
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show	
Description	Displays only the specified attributes in the specified order. Valid attributes: • `Allocated` – Adjusted allocation. This value stores the effective allocated amount based on the initial deposit and subsequent allocation adjustments via deposits, withdrawals or transfers. • `Allocations` – Lists the active allocations in this fund in the format `id:amount:start_time:end_time`. • `Balance` – Sum of active allocation amounts within this fund. • `Children` – Lists the children funds in the format `id[(deposit_share)][^]` where the carat symbol (^) is displayed if Overflow is `True`. • `Constraints` – Contraints on fund usage. • `CreationTime` – Time this fund was created. • `CreditLimit` – Sum of active credit limits within this fund. • `DefaultDeposit` – Used as the default amount for any deposit that is made to this fund that does not specify a deposit amount. A zero value will result in the creation of an allocation with a zero balance (or add nothing if an allocation already exists and a reset is not being requested). A negative value can be used to stipulate that the allocations in the fund should be ended if the fund is reset. An empty value is used to stipulate that no change will be made to the allocations if the fund is reset. • `Deleted` – Boolean indicating whether this fund is deleted or not. • `Description` – Fund description. • `Id` – Fund ID. • `InitialDeposit` – Initial deposit for current allocation. • `ModificationTime` – Time this fund was last modified. • `Name` – Fund name. • `Parent` – Displays the parent fund in the format `id[(deposit_share)][^]` where the carat symbol (^) is displayed if Overflow is `True`. • `Priority` – Fund priority. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction. Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form: operator(attribute_name)[=alias]. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site	
Format	`--site <site_name>`
Default	---

--site	
Description	Obtains a response from specified site.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

--wide	
Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line comma-separated format.

Related Topics

A.31 mam-list-itemizedcharges

A.31.1 Synopsis

mam-list-itemizedcharges [-j <usage_record_id>] [-J <instance_name>]
[-n <usage_property_name>] [-s <start_time>] [-e <end_time>] [--full]
[--show <attribute_name>,...] [--format csv|raw|standard] [--hours]
[--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version]
[--about]

A.31.2 Overview

mam-list-itemizedcharges displays allocation information.

A.31.3 Options

-e	
Format	-e <end_time>
Default	---
Description	Displays charges occurring before the specified time in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now.

-j	
Format	-j <usage_record_id>
Default	---
Description	Displays only charges associated with the specified usage record.

-J	
Format	-J <instance_name>
Default	---

-J	
Description	Displays only charges against the specified instance (such as a job ID).

-n	
Format	`-n <usage_record_property_name>`
Default	---
Description	Displays only charges against the specified usage property.

-s	
Format	`-s <start_time>`
Default	---
Description	Displays charges occurring after the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format	
Format	`--format csv|raw|standard`
Default	standard
Description	Specifies a data output format.

--hours	
Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--full	
Format	`--full`
Default	---
Description	Displays all attributes.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet	
Format	`--quiet`
Default	---

--quiet	
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the specified order. Valid attributes: • `Amount` – Amount charged. • `CreationTime` – Time this charge was created. • `Deleted` – Boolean indicating whether this allocation is deleted or not. • `Description` – Charge description. • `Details` – Details of the formula used in calculating the charge. • `Duration` – Amount of time the item was used in seconds. • `Instance` – Instance name (such as job ID) for the charge. • `ModificationTime` – Time this charge was last modified. • `Name` – Usage record property name (also charge rate name). • `Rate` – Base charge rate. • `RequestId` – ID of the last modifying request. • `ScalingFactor` – Product of all applicable multipliers (discounts and premiums) applied to the base rate. • `TransactionId` – ID of the last modifying transaction. • `UsageRecord` – Usage record ID. • `Value` – Usage record property value.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version	
Format	`--version`

--version	
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 15.1 Querying Itemized Charges - page 164
- Appendix A: Commands Reference - page 457

A.32 mam-list-liens

A.32.1 Synopsis

mam-list-liens [[-l] <lien_id>] [-A | -I] [-J <instance_pattern>] [-X, --extension <property>=<value>]... [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type AttributedTo|ImpingesUpon] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.32.2 Overview

mam-list-liens displays lien information.

You can customize the fields this command displays by default by setting the lien.show configuration parameter in mam-client.conf.

A.32.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only liens against the specified account.

-A	
Format	-A
Default	---
Description	Displays only unexpired liens.

-c	
Format	`-c <class_name>`
Default	---
Description	Displays only liens against the specified class.

-g	
Format	`-g <group_name>`
Default	---
Description	Displays only liens against the specified group.

-I	
Format	`-I`
Default	---
Description	Displays only expired liens.

-J	
Format	`-J <instance_pattern>`
Default	---
Description	Displays only liens with the instance names (or job IDs) matching the pattern.

l	
Format	`[-l] <lien_id>`
Default	---
Description	Displays only the specified lien.

-m	
Format	`-m <machine_name>`
Default	---
Description	Displays only liens against the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only liens against the specified organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only liens against the specified user.

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Displays liens where constraints do not conflict with the specified filters. For example, `mam-list-liens -f User=amy` will display liens usable by the user `amy`. You may specify multiple filter options by logically ANDing them together.

--filter-type

Format	`--filter-type <filter_type>`
Default	`AttributedTo`
Description	Selects the filtering type. If you use the `AttributedTo` filter type, the query returns all liens associated with usage records satisfying the filters. If you use the `ImpingesUpon` filter type, the query returns all liens affecting the balances of funds satisfying the filters.

--format

Format	`--format <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`

--help

Default	---
Description	Displays a brief help message.

--long

Format	--long
Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	--man
Default	---
Description	Displays full documentation.

--hours

Format	--hours
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the order specified. Valid attributes: • `Allocations` – List of allocations that the lien has holds against in the format `<allocation_id><-<fund_id>=<reserved_amount>`. • `Amount` – Reserved amount. • `CreationTime` – Time this lien was created. • `Deleted` – Boolean indicating whether this lien is deleted or not. • `Description` – Lien description. • `Duration` – Expected duration of the reserved usage in seconds. • `EndTime` – Time the lien becomes inactive. • `Funds` – List of funds that the lien has holds against. • `Id` – Lien ID. • `Instance` – The lien is against the specified instance (for instance, job ID). • `ModificationTime` – Time this lien was last modified. • `RequestId` – ID of the last modifying request. • `StartTime` – Time the lien becomes active. • `TransactionId` – ID of the last modifying transaction. • `UsageRecord` – ID of the usage record associated with the lien and containing the usage properties. Additionally, unambiguous usage record properties may also be specified for display (User, Group, Account, Organization, Class, QuaulityOfService, Machine, Nodes, Processors, Memory, etc.). Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form of `operator(attribute_name)[=alias]`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

--wide	
Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line comma-separated format.

Related Topics

A.33 mam-list-notifications

A.33.1 Synopsis

mam-list-notifications [[-N] <notification_id>] [-E <event_id>] [-T <notification_type>] [-k <primary_key_value>] [-u <recipient>] [-x <status>] [-s <start_time>] [-e <end_time>] [--delete] [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.33.2 Overview

mam-list-notifications displays stored notification information.

You can customize the fields this command displays by default by setting the `notification.show` configuration parameter in `mam-client.conf`.

A.33.3 Options

-e	
Format	`-e <end_time>`
Default	---
Description	Displays the notifications sent before the specified time in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-E	
Format	`-E <event_id>`
Default	---
Description	Displays only the notifications associated with the specified event ID.

|

-k	
Format	`-k <primary_key_value>`
Default	---
Description	Displays only the notifications associated with the specified primary key value. This value of the primary key of the object instance that the command acted on.

-N	
Format	`[-N] <notification_id>`
Default	---
Description	Displays only the notifications with the specified ID.

-s	
Format	`-s <start_time>`
Default	---
Description	Displays notifications sent after the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-x	
Format	`-x <status>`
Default	---
Description	Displays notifications having the specified status (such as Success or Failure).

-T	
Format	`-T <notification_type>`
Default	---

-T	
Description	Displays notifications of the specified type (such as Fire or Failure).

-u	
Format	`-u <recipient>`
Default	---
Description	Displays notifications having the specified recipient. This could be a user name or any tag that identifies the intended reader of this notification.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--delete	
Format	`--delete`
Default	---
Description	Deletes a notification after it has been queried.

--format	
Format	`--format <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show

Description	Displays only the specified attributes in the specified order. Valid attributes:
	`Code` – Event command exit code. `CreationTime` – Time this notification was created. `Deleted` – Boolean indicating whether this notification is deleted or not. `EndTime` – Time after which a notification will be detected. `Event` – Event ID. `Key` – Object primary key value. `Id` – Notification ID. `Message` – Event command message. `ModificationTime` – Time this notification was last modified. `Recipient` – Recipient to notify. `RequestId` – ID of the last modifying request. `Status` – Event command status. `TransactionId` – ID of the last modifying transaction. `Type` – Displays the type of notification. Notifications may be created by event "Fire" commands or by event "Failure" commands.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`

--about	
Default	---
Description	Displays product information.

Related Topics

- 19.1 Querying Notifications - page 190
- Appendix A: Commands Reference - page 457

A.34 mam-list-organizations

A.34.1 Synopsis

mam-list-organizations [[-o] <organization_pattern>] [-X, --extension
<property>=<value>]... [--full] [--show <attribute_name>,...] [--
format csv|raw|standard] [--debug] [--site <site_man>] [--help] [--
man] [--quiet] [--version] [--about]

A.34.2 Overview

mam-list-organizations displays organization information.

You can customize the fields this command displays by default by setting the
organization.show configuration parameter in mam-client.conf.

A.34.3 Options

-o	
Format	-o <organization_pattern>
Default	---
Description	Displays only organizations matching the pattern. If no pattern is specified then all organizations are displayed. The following wildcards are supported: * – matches any number of characters ? – matches a single character

-X	
Format	-X or --extension <property>=<value>
Default	---
Description	Extension property. You may specify any number of extra custom conditions may be specified.

--debug

Format	--debug
Default	---
Description	Logs debug information to the screen.

--format

Format	--format <output_format>
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	--full
Default	---
Description	Displays all attributes.

--help

Format	--help
Default	---
Description	Displays a brief help message.

--man

Format	--man
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the order you specify. Valid attributes: • `CreationTime` – Time this organization was created. • `Deleted` – Boolean indicating whether this organization is deleted or not. • `Description` – Organization description. • `ModificationTime` – Time this organization was last modified. • `Name` – Organization name. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-
-

A.35 mam-list-quotes

A.35.1 Synopsis

mam-list-quotes [[-q] <quote_id>] [-J <instance_name>] [-A | -I] [-X, --extension <property>=<value>]... [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--hours]] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.35.2 Overview

mam-list-quotes displays quote information.

You can customize the fields this command displays by default by setting the quote.show configuration parameter in mam-client.conf.

A.35.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only quotes for the specified account.

-A	
Format	-A
Default	---
Description	Displays only unexpired quotes.

-c	
Format	`-c <class_name>`
Default	---
Description	Displays only quotes for the specified class.

-g	
Format	`-g <group_name>`
Default	---
Description	Displays only quotes for the specified group.

-l	
Format	`-I`
Default	---
Description	Displays only expired quotes.

-J	
Format	`-J <instance_name>`
Default	---
Description	Displays only quotes with the specified instance name or job ID.

-m	
Format	`-m <machine_name>`
Default	---
Description	Displays only quotes for the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only quotes for the specified organization.

-q	
Format	`[-q] <quote_id>`
Default	---
Description	Displays only information for the specified quote.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only quotes for the specified user.

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Displays quotes where constraints do not conflict with the specified filters. For example, `mam-list-quotes --filter User=amy` will display funds usable by the user `amy`. You may specify multiple filter options by logically ANDing them together.

--format

Format	`--format <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--long

Format	`--long`

--long

Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--hours

Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show

Description	Displays only the specified attributes in the specified order. Valid attributes include the following: • `Amount` – Quoted amount. • `ChargeRate` – Saved charge rates to be used when the quote is referenced. These are displayed in the format `<charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>` • `CreationTime` – Time this quote was created. • `Deleted` – Boolean indicating whether this quote is deleted or not. • `Description` – Quote description. • `Duration` – Expected duration of the quoted usage in seconds. • `EndTime` – Time the quote becomes inactive. • `Id` – Quote ID. • `Instance` – The quote may only be used by the specified instance. • `ModificationTime` – Time this quote was last modified. • `Pinned` – Boolean indicating whether the quote is pinned or not. • `RequestId` – ID of the last modifying request. • `StartTime` – Time the quote becomes active. • `TransactionId` – ID of the last modifying transaction. • `UsageRecord` – ID of the usage record associated with the quote and containing the usage properties. Additionally, unambiguous usage record properties may also be specified for display (User, Group, Account, Organization, Class, QuaulityOfService, Machine, Nodes, Processors, Memory, etc.). Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form of `operator(attribute_name)[=alias]`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`

--version

Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

--wide

Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line comma-separated format.

Related Topics

-
-

A.36 mam-list-roles

A.36.1 Synopsis

mam-list-roles [[--r] <role_name>] [--full] [--show <attribute_name>,...] [--long] [--wide] [--format csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.36.2 Overview

mam-list-roles displays role information.

A.36.3 Options

-r	
Format	[-r] <role_name>
Default	---
Description	Displays information for only the specified role.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--format	
Format	--format <output_format>
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--long

Format	`--long`
Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the specified order. Valid attributes: • `Actions` – List of actions permitted by the role. Actions are displayed in the format `object->action{instance}` • `CreationTime` – Time this role was created. • `Deleted` – Boolean indicating whether this role is deleted or not. • `Description` – Role description. • `ModificationTime` – Time this role was last modified. • `Name` – Role name. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction. • `Users` – List of users granted access to the role.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---

--about	
Description	Displays product information.

--wide	
Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line comma-separated format.

Related Topics

A.37 mam-list-transactions

A.37.1 Synopsis

mam-list-transactions [[-T] <transaction_id>] [-R <request_id>] [-O <object>] [-A <action>] [-k <primary_key_value>] [-U <actor>] [-f <fund_id>] [-i <allocation_id>] [-u <user_name>] [-a <account_name>] [-m <machine_name>] [-j <usage_record_id>] [-J <instance_name>] [-s <start_time>] [-e <end_time>] [-X, --extension <property>=<value>]... [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.37.2 Overview

mam-list-transactions displays transaction information.

You can customize the fields this command displays by default by setting the transaction.show configuration parameter in mam-client.conf.

A.37.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only transactions involving the specified account.

-A	
Format	-A <action>
Default	---
Description	Displays only transactions invoking the specified action.

-e	
Format	`-e <end_time>`
Default	---
Description	Displays transactions occurring before the specified time in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-f	
Format	`-f <fund_id>`
Default	---
Description	Displays only transactions involving the specified fund.

-i	
Format	`-i <allocation_id>`
Default	---
Description	Displays only transactions logged against the specific allocation.

-j	
Format	`-j <usage_record_id>`
Default	---
Description	Displays only transactions affecting the given usage record.

-J	
Format	`-J <instance_name>`
Default	---

-J	
Description	Displays only transactions affiliated with the given instance name (e.g. job ID).

-k	
Format	-k <primary_key_value>
Default	---
Description	Displays only transactions involving the objects having the specified primary key value (i.e. having the specified Id or Name) or associations with the given parent name.

-m	
Format	-m <machine_name>
Default	---
Description	Displays only transactions involving the specified machine.

-O	
Format	-O <object>
Default	---
Description	Displays only transactions performing actions on the given object type.

-R	
Format	-R <request_id>
Default	---
Description	Displays only transactions with the specified request ID. A unique request ID is associated with each request, while each request may be associated with more than one transaction.

-s	
Format	`-s <start_time>`
Default	---
Description	Displays transactions occurring on or after the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-T	
Format	`[-T] <transaction_id>`
Default	---
Description	Displays only transactions with the specified transaction ID. A transaction occurs when an action is invoked on an object. A complex request may involve multiple transactions.

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only transaction involving the specified user.

-U	
Format	`-U <actor>`
Default	---
Description	Displays only transactions invoked by the specified user.

-X	
Format	`-X` or `--extension <property>=<value>`

-X	
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format	
Format	`--format csv|raw_standard`
Default	standard
Description	Specifies a data output format.

--hours	
Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--full	
Format	`--full`
Default	---
Description	Displays all attributes.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet	
Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show	
Description	Displays only the specified attributes in the specified order. Valid attributes: • `Account` – Account name associated with the transaction. • `Action` – Action name. • `Actor` – User that performed the action. • `Allocation` – Allocation ID associated with the transaction. • `Amount` – Amount. • `Balance` – Effective active balance. If the allocation is active, this is the same as the remaining allocation amount (Remaining). If the allocation is inactive, this is zero. • `Child` – If the transaction object is an association, this is the value of the child. • `Count` – Number of objects affected by the transaction. • `CreationTime` – Time this transaction was created. • `Deleted` – Boolean indicating whether this transaction is deleted or not. • `Delta` – Change (positive or negative) to the effective active balance of an allocation (Balance). This may differ in some cases from the change in the actual allocation amount (Remaining). For example, if an allocation expires, a negative Delta will be recorded for the event, while the remaining allocation amount has not changed. On the other hand, a modification of the amount in an expired allocation will be recorded as a Delta of zero. • `Description` – Transaction description. • `Details` – Additional assignments, conditions, options, and other details of the transaction are recorded here when there is no applicable transaction property to store them in. • `Duration` – Expected duration of the transaction in seconds. • `Fund` – Fund ID associated with the transaction. • `Id` – Transaction ID. • `Instance` – Instance name. • `Key` – If the transaction object is an association, this is the value of the parent, otherwise this is the value of the primary key (ID or name) of the object. • `Machine` – Machine name associated with the transaction. • `ModificationTime` – Time this transaction was last modified. • `Object` – Object name. • `Remaining` – Remaining allocation amount. If an allocation amount has the potential for being affected by this transaction, this field stores the remaining allocation amount after the transaction completed. Note that for expired allocations, this will still record the allocation's actual remaining amount, even though the allocation's effective active balance (Balance) may be zero. Thus it is possible for the Remaining amount to change even though the Delta is zero or the Remaining amount to remain unchanged even though the Delta is non-zero. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction. • `UsageRecord` – ID of the usage record associated with the transaction. • `User` – User name associated with the transaction. Additionally, when the transaction refers to a Usage record, unambiguous usage record properties

--show

	may also be specified for display (Group, Organization, Class, QuaulityOfService, Nodes, Processors, Memory), as well the derived fields (NodeHours, NodeSeconds, ProcHours, and ProcSeconds). Aggregate values may be requested for specified attributes by using operators. Aliases may be used to specify the column name for the aggregated field. Aggregated fields are specified in the form of `operator(attribute_name)[=alias]`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.38 mam-list-usagerecords

A.38.1 Synopsis

mam-list-usagerecords [[-j] <usage_record_id>] [-J <instance_name_pattern>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [--stage <lifecycle_stage>] [-X, --extension <property>=<value>]... [-s <start_time>] [-e <end_time>] [--full] [--show <attribute_name>,...] [--format csv|raw|standard] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--version] [--about]

A.38.2 Overview

mam-list-usagerecords displays usage record information.

You can customize the fields this command displays by default by setting the usagerecord.show configuration parameter in mam-client.conf.

A.38.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only usage records affiliated with the specified account.

-c	
Format	-c <class_name>
Default	---
C	Specifies the class or queue name.

-e	
Format	`-e <end_time>`
Default	---
Description	Ended before the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-g	
Format	`-g <group_name>`
Default	---
Description	Displays only usage records affiliated with the specified group.

-j	
Format	`[-j] <usage_record_id>`
Default	---
Description	Displays the usage record with the specified ID.

-J	
Format	`-J <instance_name>`
Default	---
Description	Displays only usage records matching the specified instance name (e.g. job id) pattern. The following wildcards are supported: • `*` – matches any number of characters • `?` – matches a single character

-m	
Format	`-m <machine_name>`
Default	---
Description	Displays only usage records affiliated with the specified machine.

-o	
Format	`-o <organization_name>`
Default	---
Description	Displays only usage records affiliated with the specified organization.

-Q	
Format	`-Q <quality_of_service>`
Default	---
Description	Displays only usage records affiliated with the given quality of service .

-s	
Format	`-s <start_time>`
Default	---
Description	Ended on or after the specified time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

--stage	
Format	`--stage <lifecycle_stage>`
Default	---

--stage	
Description	Latest stage in the object's accounting lifecycle (e.g. Create, Start, Continue, End).

-T	
Format	`-T <usage_record_type>`
Default	---
Description	Displays only usage records associated with the specified type (such as Job or Reservation).

-u	
Format	`-u <user_name>`
Default	---
Description	Displays only usage records affiliated with the given user.

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format	
Format	`--format <output_format>`
Default	standard
Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full	
Format	`--full`
Default	---
Description	Displays all attributes.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--hours	
Format	`--hours`
Default	---

--hours

Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show

Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---

--show	
Description	Displays only the specified attributes in the order specified. Valid attributes include the following: • `Account` – Account name associated with the usage. • `BlockedProcessors` – Number of processors blocked by the job. • `Charge` – Cumulative amount charged. • `Class` – Class or queue name associated with the usage. • `CPUTime` – CPU time used. • `CreationTime` – Time this usage record was created. • `Deleted` – Boolean indicating whether this usage record is deleted or not. • `Description` – Usage description. • `Duration` – Expected duration of the usage. • `EndTime` – Overall end time of the usage. • `Features` – Allocated node features. Individual feature counts may be displayed using the partial value syntax `Features{<feature_part_name>}`. • `Group` – Group name associated with the usage. • `Id` – Usage record ID. • `Instance` – Instance name (job ID). • `Licenses` – Licenses used. Individual license counts may be displayed using the partial value syntax `Licenses{<license_part_name>}`. • `Machine` – Cluster name. • `Metrics` – Generic metrics. Individual metric values may be displayed using the partial value syntax `Metrics{<metric_part_name>}` • `Memory` – Amount of memory used. • `ModificationTime` – Time this usage record was last modified. • `Nodes` – Number of nodes used. • `NodeHours` – Nodes * Duration / 3600. • `NodeSeconds` – Nodes * Duration. • `Organization` – Organization name associated with the usage. • `Processors` – Number of cores or processors allocated. • `ProcessorEquivalents` – Number of processor equivalents allocated by the job. • `ProcHours` – Processors * Duration / 3600. • `ProcSeconds` – Processors * Duration. • `QualityOfService` – Quality of service associated with the usage. • `QueueDuration` – Duration the job was in the idle state. • `Quote` – Associated quote ID. • `RequestedDuration` – Requested wallclock limit. • `RequestId` – ID of the last modifying request. • `Resources` – Generic resources. Individual resource amounts may be displayed using the partial value syntax `Resources{<resource_part_name>}`. • `Stage` – Latest stage in the object's accounting lifecycle (Create, Start, Continue, End). • `StartTime` – Latest start time of the usage. • `SubmitTime` – Creation or submit time of the item.

--show

- `TransactionId` – ID of the last modifying transaction.
- `Type` – Usage record type.
- `User` – User name associated with the usage.
- `Variables` – Job variables. Individual variable values may be displayed using the partial value syntax `Variables{<variable_part_name>}`.

Aggregate values may be requested for specified attributes by using operators. Aggregated fields are specified in the form of `operator(attribute_name)`. Valid operators include Sum, Average, Count, Min, Max and GroupBy. When an operator is specified, fields without an explicit operator are assumed to have the GroupBy operator.

Partial values may be requested for complex (multi-valued) attributes. Partial values are specified in the form of `attribute_name{part_name}`.

Aliases may be used to specify the resultant column name. Aliases are specified in the form of `attribute_name=alias`.

Aggregate values, partial values and aliases may be combined, e.g. `operator(attribute_name {part_name})=alias`.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 14.2 Querying Usage Records - page 141
- Appendix A: Commands Reference - page 457

<div style="background:#1a3a5c;color:white;padding:10px;">

A.39 mam-list-users

</div>

A.39.1 Synopsis

mam-list-users [[-u] <user_pattern>] [-A | -I] [-X, --extension
<property>=<value>]... [-a <account_name>] [--full] [--
show <attribute_name>,...] [--long] [--wide] [--format
csv|raw|standard] [--debug] [--site <site_name>] [--help] [--man] [--
quiet] [--version] [--about]

A.39.2 Overview

mam-list-users displays user information.

You can customize the fields this command displays by default by setting the user.show con-figuration parameter in mam-client.conf.

A.39.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Displays only users affiliated with the specified account.

-A	
Format	-A
Default	---
Description	Displays only active users.

-I	
Format	-I

-l	
Default	---
Description	Displays only inactive users.

-u	
Format	`[-u] <user_pattern>`
Default	---
Description	Displays only users matching the pattern. If you do not specify a pattern then all users are displayed. The following wildcards are supported: • ***** – matches any number of characters • **?** – matches a single character

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra custom conditions.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format	
Format	`--format <output_format>`
Default	standard

--format

Description	Specifies a data output format. Valid values are standard, raw, and csv.

--full

Format	`--full`
Default	---
Description	Displays all attributes.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--long

Format	`--long`
Default	---
Description	Long format. Displays multi-valued fields in a multi-line format.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet	
Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--show	
Format	`--show <attribute_name>[,<attribute_name>...]`
Default	---
Description	Displays only the specified attributes in the order specified. Valid attributes include the following: • `Accounts` – List of accounts to which the user belongs. • `Active` – Boolean indicating whether this user is active or not. • `CommonName` – Common name for the user. • `CreationTime` – Time this user was created • `Deleted` – Boolean indicating whether this user is deleted or not. • `DefaultAccount` – Default account. • `Description` – User description. • `EmailAddress` Email address. • `ModificationTime` – Time this user was last modified. • `Name` – User name. • `PhoneNumber` – Phone number. • `RequestId` – ID of the last modifying request. • `TransactionId` – ID of the last modifying transaction.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

--wide

Format	`--wide`
Default	---
Description	Wide format. Displays multi-valued fields in a single-line comma-separated format.

Related Topics

- 7.2 Querying Users - page 73
- Appendix A: Commands Reference - page 457

A.40 mam-modify-account

A.40.1 Synopsis

mam-modify-account {[-a] <account_name>} [-A | -I] [-o <organization_name>] [-d <description>] [-X, --extension <property>=<value>]... [--add-user(s) [^|!][+|-]<user_name>,...]... [--del-user(s) <user_name>,...] [--mod-user(s) [^|!][+|-]<user_name>,...]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.40.2 Overview

mam-modify-account modifies an account.

A.40.3 Options

-a	
Format	[-a] <account_name>
Default	---
Description	Specifies the name of the account to modify.

-A	
Format	-A
Default	---
Description	Activates the account.

-d	
Format	-d <description>
Default	---

-d

Description	Modifies the account description.

-I

Format	`-I`
Default	---
Description	Deactivates the account.

-o

Format	`-o <organization_name>`
Default	---
Description	Modifies the name of the organization to which the account belongs.

-X, --extension

Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--add-user

Format	`--add-user [^|!][+|-]<user_name>[,[^|!][+|-]<user_name>...]`
Default	---
Description	Adds user members of the account. The optional caret or exclamation symbol indicates whether the user should be created as an administrator (^) or not (!) for the account. The optional plus or minus signs can precede each member to indicate whether the member should be created in the active (+) or inactive (-) state. By default, a user will be created in the active state but not an administrator. You can pass multiple users to the `--add-user` option in a comma-delimited list or you can specify multiple `--add-user` options.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--del-user

Format	`--del-user <user_name>[,<user_name>...]`
Default	---
Description	Removes user members from the account. You can pass multiple users to the `--del-user` option in a comma-delimited list or specify multiple `--del-user` options.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--mod-user

Format	`--mod-user [^|!][+|-]<user_name>[,[^|!][+|-]<user_name>...]`
Default	---

--mod-user

Description	Modifies user members of the account. The caret symbol or exclamation symbol indicates the user should be changed to become an administrator (`^`) or not (`!`) for the account. The plus or minus signs indicate whether the user should be changed to become active (+) or inactive (−). If you do not specify an active or admin modifier, that aspect of the user member will remain unchanged. You can pass multiple users to the `--mod-user` option in a comma-delimited list or you can specify multiple `--mod-user` options.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 8.3 Modifying Accounts - page 82
- Appendix A: Commands Reference - page 457

A.41 mam-modify-allocation

A.41.1 Synopsis

mam-modify-allocation {[-i] <allocation_id>} [-s <start_time>] [-e <end_time>] [-L <credit_limit>] [-d <description>] [-X, --extension <property>=<value>]... [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.41.2 Overview

mam-modify-allocation modifies an allocation. This includes changing the credit limit or description or adjusting the start time or end time.

A.41.3 Options

-d	
Format	-d <description>
Default	---
Description	Modifies the allocation description.

-e	
Format	-e <end_time>
Default	---
Description	Specifies a new end time in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now

-i	
Format	[-i]<allocation_id>
Default	---
Description	The ID of the allocation to modify.

-L

Format	`-L <credit_limit>`
Default	---
Description	Specifies a new credit limit.

-s

Format	`-s <start_time>`
Default	---
Description	Specifies a new start time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`

-X, --extension `<property>`

Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--hours

Format	`--hours`
Default	---
Description	Treats currency as specified in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), this option allows the credit limit to be specified in resource hours.

--debug

Format	`--debug`

--debug

Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.42 mam-modify-chargerate

A.42.1 Synopsis

mam-modify-chargerate {[-n] <charge_rate_name>} [-x <charge_rate_
value>] [-z <charge_rate_amount>] [-d <description>] [--debug] [--
site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version]
[--about]

A.42.2 Overview

mam-modify-chargerate modifies a charge rate. Only the amount or the description of a charge rate may be modified.

A.42.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-n	
Format	[-n] <charge_rate_name>
Default	---
Description	Specifies the name of the charge rate to change.

-x	
Format	-x <charge_rate_value>
Default	---

-x	
Description	Specifies the charge rate value expression to change. If you do not specify a value, an empty value is assumed.

-z	
Format	`-z <charge_rate_amount>`
Default	---
Description	Specifies a new amount for the charge rate. The amount is an integer or decimal and may include operators indicating how to apply the charge rate as well as divisors and time-based units. See Chapter 16: Managing Charge Rates - page 167 for more information.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about

Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.43 mam-modify-event

A.43.1 Synopsis

mam-modify-event {[-E] <event_id>} [--fire-command <fire_command>] [-s <fire_time>] [-e <end_time>] [--rearm-period <rearm_period>] [--rearm-on-failure True|(False)] [--failure-command <failure_command>] [--notify <notification_url>] [--catch-up (True)|False] [-d <description>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.43.2 Overview

mam-modify-event modifies an event.

A.43.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-e	
Format	-e <end_time>
Default	---
Description	Specifies the time this event becomes inactive in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now.

-E	
Format	-E <event_id>

-E	
Default	---
Description	Specifies the ID of the event to modify.

-s	
Format	`-s <fire_time>`
Default	---
Description	Specifies a new target time for the event to be triggered by the event scheduler. The actual fire time may be dependent on the state of the server and will be recorded in the CreationTime property of the corresponding "Event Fire" Transaction. An event may also be fired manually with the mam-shell Event Fire action.

--catch-up	
Format	`--catch-up <boolean>`
Default	`True`
Description	If you set the `--catch-up` boolean to `True` and the server was down during the time this event should have fired, the event scheduler will attempt to make up for the past due events by progressively firing them (rearming based on previous arm time) until it catches up to the present. The actions will still show as having occurred in the present rather than in the past. If set to `False` and the server is brought back up after an outage, the event scheduler will still fire immediately for a past due event, but it will only fire once and then rearm relative to the current time.

--debug	
Format	`--debug`
Default	---
Description	Logs debugging information to the screen.

--failure-command

Format	`--failure-command <failure_command>`
Default	---
Description	Specifies a new command MAM should execute if the fired command results in an unsuccessful response status. This command is expressed in a serialized form of the request identical to the syntax used in the interactive control program (mam-shell). The option argument will need to be appropriately quoted and/or escaped in order to avoid misinterpretation or alteration by the shell.

--fire-command

Format	`--fire-command <fire_command>`
Default	---
Description	Specifies the command MAM should execute.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays the full documentation.

--notify

Format	`--notify [+-=][<delivery_method>:][recipient]`

--notify	
Default	Logs all event statuses to the Notification table.
Description	A Notification method logs the result of the fired command. If the term is a –, the notification is sent only on failure. If the term is a +, the notification is sent only on success. Otherwise the notification is always sent. See Chapter 19: Managing Notifications - page 189 for more information about delivery method and recipient.

--quiet	
Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--rearm-on-failure	
Format	`--rearm-on-failure <boolean>`
Default	`False`
Description	If you set the `--rearm-on-failure` boolean to `False`, MAM will not rearm the event if the command was unsuccessful. If you set it to `True`, MAM will evaluate the event for rearming even if the command response has a status of `Failure`.

--rearm-period	
Format	`--rearm-period <period>[[@instant][~|^]|!]`
Default	---

--rearm-period

Description	The `--rearm-period` is a time period expression specifying when MAM will rearm the event. This period expression is in the form of `<period>[[@instant][~|^]|!]`. The `<period>` is expressed as an integer number followed by a designator of minute(s), hour(s), day(s), month(s), or year(s). For example, the period might be 1 day, 2 hours, or 5 minutes. The optional `instant` locks the period to a specific instant within the time period such as 1 day @ hour 12 or 1 month @ day 3. The modifiers indicate whether the time period should be relative to now (`!`), relative to the start of this (`~`) designator (month, minute, or other unit), or relative to the start of the first (`^`) designator (month, minute, or other unit). For example, assuming the FireTime was 7:15, if you specified `4 hours !` as the rearm period it would be rearmed at 11:15; if you specified `4 hours ~` as the rearm period, it would be rearmed at 11:00; and if you specified `4 hours ^` as the rearm period it would be rearmed at 8:00.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 18.4 Modifying Events - page 186
- Appendix A: Commands Reference - page 457

A.44 mam-modify-fund

A.44.1 Synopsis

mam-modify-fund [[-f] <fund_id>] [-u <user_name>] [-g <group_name>]
[-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] {{[-n <fund_name>] [--priority <fund_priority>] [--default-deposit <deposit_amount>] [-d <description>] [-X, --extension <property>=<value>]... [—add-constraint <constraint_name>=[!]<constraint_value>,...] [--del-constraint(s) <constraint_name>[=<constraint_value>],...]... [--parent <parent_fund_id>]} | {--reset [--all]}} [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version]
[--about]

A.44.2 Overview

mam-modify-fund modifies a fund. This includes adding to or deleting from constraints for the account. After applying all filter options, if there is exactly one applicable fund, that fund will be modified. Otherwise, a list of funds will be displayed for the specified filters and you will be prompted to rerun *mam-modify-fund* against one of the enumerated funds.

A.44.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given account.

-c	
Format	-c <class_name>
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given class.

-d	
Format	`-d <description>`
Default	---
Description	Specifies a new description

-f	
Format	`[-f] <fund_id>`
Default	---
Description	Specifies the ID of the fund to modify.

-g	
Format	`-g <group_name>`
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given group.

-m	
Format	`-m <machine_name>`
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given machine.

-n	
Format	`-n <fund_name>`
Default	---
Description	Specifies a new fund name.

-o	
Format	`-o <organization_name>`
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Specifies that the fund to modify should be restricted to one usable by the given user.

-X, --extension	
Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--add-constraint	
Format	`--add-constraint <constraint_name>=[!]<constraint_value>[,<constraint_name>=[!]<constraint_value>...]`
Default	---
Description	Adds a constraint to the fund. The constraint value may be a perl5 regular expression. You can pre-pend an exclamation point to the constraint value to express a negation of the constraint. You can pass multiple constraints to the `--add-constraint` option in a comma-delimited list or specify multiple `--add-constraint` options.

--all	
Format	`--all`

--all	
Default	---
Description	Specifies that you want to reset all active allocations for all funds when you use it with the --reset option.

--debug	
Format	--debug
Default	---
Description	Logs debug information to the screen.

--default-deposit	
Format	--default-deposit <deposit_amount>
Default	---
Description	Specifies the default amount for any deposit that is made to this fund that does not already specify a deposit amount. • A zero value will result in the creation of an allocation with a zero balance (or add nothing if an allocation already exists and a reset is not being requested). • A negative value can be used to stipulate that the allocations in the fund should be ended if the fund is reset. • An empty value ('') or NULL can be used to stipulate that no change will be made to the allocations if the fund is reset.

--del-constraint	
Format	--del-constraint <constraint_name>=<constraint_value>[,<constraint_name>[=<constraint_value>]...]
Default	---
Description	Removes a constraint from the fund. You can pass multiple constraints to the --del-constraint option in a comma-delimited list or by specifying multiple --del-constraint options.

--filter	
Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Restricts the fund to one without constraints that conflict with the specified filters. For example, `mam-modify-fund --filter User=amy` will restrict the fund to one usable by the user `amy`. You can specify multiple filter options by logically ANDing them together.

--filter-type	
Format	`--filter-type ExactMatch|Exlusive|NonExclusive`
Default	`NonExclusive`
Description	Selects the filtering type. - If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. - If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). - If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--parent

Format	`--parent <parent_fund_id>`
Default	---
Description	Sets a new parent fund, replacing the current parent fund if one exists.

--priority

Format	`--priority <fund_priority>`
Default	---
Description	Sets a new fund priority.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--reset

Format	`--reset`
Default	---
Description	Ends all active allocations and initiates a new default deposit. If the default deposit amount is positive, MAM creates a new allocation with this amount, otherwise no deposit is made and the fund becomes inactive. You may reset the allocations for a specified fund using the `-f` option, all funds using the `--all` option, or use filtering options to filter the funds to be reset. Do not use this option with any other modifying option.

--site

Format	`--site <site_name>`

--site	
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	--verbose
Default	---
Description	Displays modified object details.

--version	
Format	--version
Default	---
Description	Displays the product version.

--about	
Format	--about
Default	---
Description	Displays product information.

Related Topics

A.45 mam-modify-lien

A.45.1 Synopsis

mam-modify-lien {[-l] <lien_id>} [-s <start_time>] [-e <end_time>] [-t <lien_duration>] [-d <description>] [-X, --extension <property>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.45.2 Overview

mam-modify-lien modifies a lien.

A.45.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-e	
Format	-e <end_time>
Default	---
Description	Specifies a new expiration time in the format YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now

-l	
Format	[-l] <lien_id>
Default	---
Description	Specifies the ID of the lien to modify.

-s	
Format	`-s <start_time>`
Default	---
Description	Specifies a new start time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`

-t	
Format	`-t <lien_duration>`
Default	---
Description	Specifies the duration of the lien in seconds. Although the lien start time and end time are enforced, the duration is not authoritative. If the time frame between the end time and the start time is greater than the duration, the difference is the allotted grace period (which defaults to 10 minutes).

-X, --extension `<property>`	
Format	`-X` or `--extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`

--help

Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.46 mam-modify-organization

A.46.1 Synopsis

mam-modify-organization {[-o] <organization_name>} [-d <description>]
[-X, --extension <property>=<value>]... [--debug] [--site <site_
name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.46.2 Overview

`mam-modify-organization` modifies an organization.

A.46.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-o	
Format	-o <organization_name>
Default	---
Description	Specifies the name of the organization to modify.

-X, --extension `<property>`	
Format	-X or --extension <property>=<value>
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.47 mam-modify-quote

A.47.1 Synopsis

mam-modify-quote {[-q] <quote_id>} [-s <start_time>] [-e <end_time>]
[-d <description>] [-X, --extension <property>=<value>]... [--debug]
[--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.47.2 Overview

`mam-modify-quote` modifies a quote.

A.47.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-e	
Format	-e <end_time>
Default	---
Description	Specifies a new expiration time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`

-q	
Format	[-q] <quote_id>
Default	---
Description	Specifies the ID of the quote to modify.

-s	
Format	`-s <start_time>`
Default	---
Description	Specifies a new start time in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`

-X, --extension `<property>`	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Modifies an extension property. You can specify any number of extra field assignments.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---

--man

Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 13.5 Modifying Quotes - page 137
- Appendix A: Commands Reference - page 457

A.48 mam-modify-role

A.48.1 Synopsis

mam-modify-role {[-r] <role_name>} [-d <description>] [--add-user(s) <user_name>,...]... [--add-action(s) "<object_name>-><action_name> [{<instance_name>}]",...]... [--del-user(s) <user_name>,...]... [--del-action(s) "<object_name>-><action_name>[{<instance_ name>}]",...]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.48.2 Overview

mam-modify-role modifies a role. This can include adding or removing users from a role and adding removing actions from a role.

A.48.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies a new description.

-r	
Format	[-r] <role_name>
Default	---
Description	Specifies the name of the role to modify.

--add-action	
Format	--add-action "<object_name>-><action_name>[{<instance_name>}][,<object_ name>-><action_name>[{<instance_name>}]...]"

--add-action

Default	---
Description	Adds actions to the role. You must specify the object, action and instance in the form shown. Unless specified, the instance will default to a value of ANY. You may pass multiple actions to the `--add-action` option in a comma-delimited list or by specifying multiple `--add-action` options.

--add-user

Format	`--add-user <user_name>[,<user_name>...]`
Default	---
Description	Adds users to the role. You may pass multiple users to the `--add-user` option in a comma-delimited list or by specifying multiple `--add-user` options.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--del-action

Format	`--del-action "<object_name>-><action_name>[{<instance_name>}][,<object_name>-><action_name>[{<instance_name>}]...]"`
Default	---
Description	Removes actions from a role. You must specify the object and action; however, the instance is optional. You may pass multiple actions to the `--del-action` option in a comma-delimited list or by specifying multiple `--del-action` options.

--del-user

Format	`--del-user <user_name>[,<user_name>...]`

--del-user

Default	---
Description	Removes users from the role. You may pass multiple users to the `--del-user` option in a comma-delimited list or by specifying multiple `--del-user` options.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.49 mam-modify-usagerecord

A.49.1 Synopsis

mam-modify-usagerecord {[-j] <usage_record_id> | -J <instance_name>}
[-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-
g <group_name>] [-a <account_name>] [-o <organization_name>] [-
c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-
N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-
D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_
count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "
{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_
name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_
value>\",...}"] [-W <requested_duration>] [-t <actual_duration>] [-
s <start_time>] [-e <end_time>] [-x <exit_code>] [--stage <lifecycle_
stage>] [-d <description>] [-X, --extension <property_
name>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--
quiet] [--verbose] [--version] [--about]

A.49.2 Overview

mam-modify-usagerecord modifies a usage record.

A.49.3 Options

-a	
Format	-a <account_name>
Default	---
Description	New account name.

-c	
Format	-c <class_name>
Default	---
Description	New class or queue.

-C	
Format	`-C <cpu_time>`
Default	---
Description	New CPU time used.

-d	
Format	`-d <description>`
Default	---
Description	New description.

-D	
Format	`-D <disk>`
Default	---
Description	New amount of disk used.

-e	
Format	`-e <end_time>`
Default	---
Description	New date and time the usage ended in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

-E	
Format	`-E <energy>`
Default	---

-E	
Description	New energy used.

-F	
Format	`-F "{\"<feature_name>\":<feature_count>,...}"`
Default	---
Description	New allocated node features. Features represent counts of the node features allocated to the job.

-g	
Format	`-g <group_name>`
Default	---
Description	New group name.

-j	
Format	`[-j] <usage_record_id>`
Default	---
Description	ID of the usage record to modify. Instance names can be non-unique (resource managers often recycle job IDs). This option allows you to specify a usage record using the unique identifier.

-J	
Format	`[-J] <instance_name>`
Default	---
Description	Instance name (e.g. job id) of the usage record(s) to modify. If there is exactly one matching usage record, that usage record will be modified. Otherwise, a list of usage records will be displayed for the specified instance, and you will be prompted to rerun `mam-modify-usagerecord` against one of the enumerated usage records.

-L	
Format	`-L "{\"<license_name>\":<license_count>,...}"`
Default	---
Description	New licenses used. Licenses represent software licenses that are used (in integer units).

-m	
Format	`-m <machine_name>`
Default	---
Description	New name of the cluster.

-M	
Format	`-M <memory>`
Default	---
Description	New amount of memory used.

-n	
Format	`-n <designated_name>`
Default	---
Description	New user-specified job name.

-N	
Format	`-N <nodes>`
Default	---
Description	New number of nodes used.

-o	
Format	`-o <organization_name>`
Default	---
Description	New organization name.

-P	
Format	`-P <processors>`
Default	---
Description	New number of processors used.

-Q	
Format	`-Q <quality_of_service>`
Default	---
Description	New quality of service used.

-R	
Format	`-R "{\"<resource_name>\":<resource_count>,...}"`
Default	---
Description	New consumable resources allocated. Resources represent consumable resources that may be allocated (in integer units).

-s	
Format	`-s <start_time>`
Default	---

-s	
Description	Specifies a new date and time the usage started in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

--stage	
Format	`--stage <lifecycle_stage>`
Default	---
Description	New latest stage in the object's accounting lifecycle (Create, Start, Continue, End).

-t	
Format	`-t <actual_duration>`
Default	---
Description	New total actual duration (in seconds).

-T	
Format	`-T <usage_record_type>`
Default	---
Description	New usage record type (Job or Reservation, for example).

-u	
Format	`-u <user_name>`
Default	---
Description	New user name.

-V	
Format	`-V "{\"<variable_name>\":\"<variable_value>\",...}"`
Default	---
Description	New job variables. Variables represent arbitrary variables passed into the job.

-W	
Format	`-W <requested_duration>`
Default	---
Description	New total estimated wallclock duration (in seconds).

-x	
Format	`-x <exit_code>`
Default	---
Description	New exit code.

-X, --extension	
Format	`-X or --extension <property>=<value>`
Default	---
Description	New extension property. You can specify any number of extra field assignments.

-Z	
Format	`-Z "{\"<metric_name>\":<metric_amount>,...}"`
Default	---

-Z	
Description	New generic metrics. Metrics represent floating point metrics of the job *or* average metrics values across the nodes in the job.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---
Description	Displays full documentation.

--quiet	
Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.50 mam-modify-user

A.50.1 Synopsis

mam-modify-user {[-u] <user_name>} [-A | -I] [-n <common_name>] [--phone <phone_number>] [--email <email_address>] [-a <default_account>] [-d <description>] [-X, --extension <property>=<value>]... [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.50.2 Overview

mam-modify-user modifies a user.

A.50.3 Options

-a	
Format	-a <default_account>
Default	---
Description	Account MAM will charge when no account is specified.

-A	
Format	-A
Default	---
Description	Activates the user.

-d	
Format	-d <description>
Default	---
Description	New description.

--email

Format	`--email <email_address>`
Default	---
Description	New email address.

-l

Format	`-I`
Default	---
Description	Deactivates the user.

n

Format	`-n <common_name>`
Default	---
Description	Common name for the user.

--phone

Format	`--phone <phone_number>`
Default	---
Description	New phone number.

-u

Format	`[-u] <user_name>`
Default	---
Description	Name of the user to modify.

-X, --extension `<property>`

Format	`-X` or `--extension <property>=<value>`
Default	---
Description	New extension property. You can specify any number of extra field assignments.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.51 mam-quote

A.51.1 Synopsis

mam-quote [-J <instance_name>] [[-j] <usage_record_id>] [-q <quote_template_id>] [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property>=<value>]... [-zt <quote_duration> [-G <grace_duration>]] [-zs <quote_start_time>] [-z <quote_amount>] [--cost-only | --guarantee] [---rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.51.2 Overview

mam-quote obtains a quote for usage. This command and its options can estimate the cost of using resources, validate that a requester has sufficient access and funds to use the requested resources, and guarantee that the charge rates used to generate the quote do not change when applying subsequent liens and charges.

A.51.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Account name.

-c	
Format	`-c <class_name>`
Default	---
Description	Class or queue used.

-C	
Format	`-C <cpu_time>`
Default	---
Description	Estimated CPU time used.

--cost-only	
Format	`--cost-only`
Default	---
Description	Returns the cost, ignoring all balance and validity checks. This option is mutually exclusive with `--guarantee`.

-d	
Format	`-d <description>`
Default	---
Description	Description of the usage.

-D	
Format	`-D <disk>`
Default	---

-D	
Description	Amount of disk used.

-E	
Format	`-E <energy>`
Default	---
Description	Amount of energy used.

-F	
Format	`-F "{\"<feature_name>\":<feature_count>,...}"`
Default	---
Description	Allocated node features. Features represent counts of the node features allocated to the job.

-g	
Format	`-g <group_name>`
Default	---
Description	Group name.

-G	
Format	`-G <grace_duration>`
Default	---
Description	Grace period (in seconds). If you specify the quote duration but not the quote end time, MAM will calculate the quote end time as the quote start time plus the quote duration plus the grace duration.

--guarantee	
Format	`--guarantee`
Default	---
Description	Guarantees the quote and returns a quote ID to secure the current charge rates. This results in the creation of a quote record and a permanent usage record. This option is mutually exclusive with `--cost-only`.

-j	
Format	`[-j] <usage_record_id>`
Default	---
Description	Usage record ID for the quote if already created with *mam-create-usagerecord* or a previous *mam-quote*.

-J	
Format	`-J <instance_name>`
Default	---
Description	Instance name (e.g. job id) of the quote, if known.

-L	
Format	`-L "{\"<license_name>\":<license_count>,...}"`
Default	---
Description	Licenses used. Licenses represent software licenses that are used (in integer units).

-m	
Format	`-m <machine_name>`

-m	
Default	---
Description	Name of the cluster.

-M	
Format	-M <memory>
Default	---
Description	Amount of memory used.

-n	
Format	-n <designated_name>
Default	---
Description	User-specified job name.

-N	
Format	-N <nodes>
Default	---
Description	Number of nodes used.

-o	
Format	-o <organization_name>
Default	---
Description	Organization name.

-P	
Format	`-P <processors>`
Default	---
Description	Number of processors used.

-q	
Format	`-q <quote_template_id>`
Default	---
Description	Quote template used to override standard charge rates.

-Q	
Format	`-Q <quality_of_service>`
Default	---
Description	Quality of service used.

-R	
Format	`-R "{\"<resource_name>\":<resource_count>,...}"`
Default	---
Description	Consumable resources allocated. Resources represent consumable resources that may be allocated (in integer units).

--rate	
Format	`--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...]`

--rate	
Default	---
Description	Charge rate expressions. Multiple charge rate expressions may be passed to the `--rate` option in a comma-delimited list. Alternately, multiple `--rate` options may be specified.

--stage	
Format	`--stage <lifecycle_stage>`
Default	---
Description	Latest stage in the object's accounting lifecycle (Create, Start, Continue, End).

-T	
Format	`-T <usage_record_type>`
Default	---
Description	Usage record type, such as job or reservation.

-u	
Format	`-u <user_name>`
Default	---
Description	User name.

-V	
Format	`-V "{\"<variable_name>\":\"<variable_value>\",...}"`
Default	---
Description	Job variables. Variables represent arbitrary variables passed into the job.

-W	
Format	`-W <requested_duration>`
Default	---
Description	Total estimated wallclock duration (in seconds).

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Extension property. You can specify any number of extra usage properties with the quote.
	When expressing accumulating properties, value may be an expression in the form of `[cumulative_value][(incremental_value)]`.
	• If both `incremental_value` and `cumulative_value` are specified, then `incremental_value` will be used for the quote and `cumulative_value` will be recorded as the cumulative value used in the usage record.
	• If only `incremental_value` is specified, this value will be used for the quote *only* and no cumlative value will be recorded in the usage record.
	• If only `cumulative_value` is specified, this value will be used both in the quote *and* recorded in the usage record.

-z	
Format	`-z <quote_amount>`
Default	---
Description	Quote amount if calculated externally.

-zs	
Format	`-zs <quote_start_time>`
Default	---

-zs	
Description	Start time for the quote in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`. This is *only* needed for non-cost-only quotes *and* is used to determine the appropriate allocation to apply to quote to.

-zt	
Format	`-zt <quote_duration>`
Default	---
Description	Incremental duration for the quote (in seconds). This is *only* needed for incremental quotes when the incremental duration differs from the wallclock duration *and* is used to compute the incremental quote amount.

-Z	
Format	`-Z "{\"<metric_name>\":<metric_amount>,...}"`
Default	---
Description	Generic metrics. Metrics represent floating point metrics of the job *or* average metrics values across the nodes in the job.

--hours	
Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--itemize	
Format	`--itemize`

Appendix A: Commands Reference

--itemize

Default	---
Description	Returns the composite charge information in the response data. You must use this in conjunction with the `--verbose` flag to display the data.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site	
Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose	
Format	`--verbose`
Default	---
Description	Display modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.52 mam-read-configuration

A.52.1 Synopsis

mam-read-configuration [-c|-s|-g][-p parameter_pattern] [--help] [--man] [--quiet] [--version]

A.52.2 Overview

mam-read-configuration is used to display configuration information. It simply parses the configuration files and will only display enabled (uncommented) parameter values. If neither -c, -s, nor -g are specified, configuration parameters from all configuration files will be displayed.

A.52.3 Options

-c	
Format	-c
Default	---
Description	Display only client configuration parameters.

-g	
Format	-g
Default	---
Description	Display only GUI configuration parameters.

-s	
Format	-s
Default	---
Description	Display only server configuration parameters.

-p	
Format	`-p <parameter_pattern>`
Default	---
Description	Display only configuration parameters matching the specified pattern. The following wildcards are supported: ***** – Matches any number of characters **?** – Matches a single character

--help	
Format	`--help`
Default	---
Description	Brief help message.

--man	
Format	`--man`
Default	---
Description	Full documentation.

--quiet	
Format	`--quiet`
Default	---
Description	Suppress headers and parameter names.

--version	
Format	`--version`

--version	
Default	---
Description	Display product version.

Related Topics

- Appendix A: Commands Reference - page 457

A.53 mam-refund

A.53.1 Synopsis

mam-refund {-J <instance_name> | [-j] <usage_record_id>} [-z <refund_amount>] [-i <allocation_id>] [-d <description>] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.53.2 Overview

mam-refund issues a refund for the specified usage. The command will return a list of usage records if the usage search does not yield a unique match. If an amount is not specified, the appropriate allocations will be credited for the full amount the overall usage was charged. A lesser amount may be specified for a partial refund. The refund will go to the allocations that were charged unless an allocation is specified, in which case the specified allocation will be credited.

A.53.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies an explanatory message for the refund.

-i	
Format	-i <allocation_id>
Default	---
Description	Specifies the allocation to be credited. If this is omitted, the allocations that were debited in the original charges will be credited.

-j	
Format	`[-j] <usage_record_id>`
Default	---
Description	Specifies the unique usage record identifier the accounting manager assigns to distinguish between usage with non-unique instance names.

-J	
Format	`-J <instance_name>`
Default	---
Description	Specifies the name of the instance (e.g. job id). This ID might not be unique among the historical list of usage records the accounting manager manages.

-z	
Format	`-z <refund_amount>`
Default	---
Description	Specifies the amount to refund. This amount must be non-negative and less than or equal to the amount charged for the overall usage.

--hours	
Format	`--hours`
Default	---
Description	Treats currency as specified in hours. In systems where the currency is measured in resource-seconds (like processor-seconds), this option allows the amount to be specified in resource-hours.

--debug	
Format	`--debug`

--debug

Default	---
Description	Logs debug information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	By default, MAM refunds amounts to the same fund from which it took them. If you want to over-ride this, by specifying filters you may restrict the fund to be refunded to one whose constraints are consistent with the specified filters. For example, `mam-refund --filter User=amy` will refund the amount to the fund usable by the user `amy`. You may specify multiple filter options by logically ANDing them together.

--filter-type

Format	`--filter-type <filter_type>`
Default	`NonExclusive`
Description	Selects the filtering type. If you use the `Exclusive` filter type, a fund will only be matched if the specified filters meet all constraints. If you use the `NonExclusive` filter type, a fund will be matched as long as the specified filters do not conflict with the constraints.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`

--man

Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 14.8 Issuing Usage Refunds - page 150
- Appendix A: Commands Reference - page 457

A.54 mam-reserve

A.54.1 Synopsis

mam-reserve {-J <instance_name>} [[-j] <usage_record_id>] [-q <quote_id>] [-n <designated_name>] [-T <usage_record_type>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization>] [-c <class_name>] [-Q <quality_of_service>] [-m <machine_name>] [-N <nodes>] [-P <processors>] [-C <cpu_time>] [-M <memory>] [-D <disk>] [-E <energy>] [-F "{\"<feature_name>\":<feature_count>,...}"] [-R "{\"<resource_name>\":<resource_count>,...}"] [-L "{\"<license_name>\":<license_count>,...}"] [-Z "{\"<metric_name>\":<metric_amount>,...}"] [-V "{\"<variable_name>\":\"<variable_value>\",...}"] [-W <requested_duration>] [-s <start_time>] [--stage <lifecycle_stage>] [-d <description>] [-X, --extension <property=value>]... [-zt <lien_duration> [-zs <lien_start_time> [-G <grace_duration>]] [-z <lien_amount>] [--modify | --replace] [--rate <charge_rate_name> [{<charge_rate_value>}]=<charge_rate_amount>,...]... [--hours] [--itemize] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.54.2 Overview

mam-reserve obtains a lien for usage.

A.54.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Account name.

-c	
Format	-c <class_name>
Default	---

-c	
Description	Class or queue used.

-C	
Format	-C <cpu_time>
Default	---
Description	Estimated CPU time used.

-d	
Format	-d <description>
Default	---
Description	Description of the usage.

-D	
Format	-D <disk>
Default	---
Description	Amount of disk used.

-E	
Format	-E <energy>
Default	
Description	Amount of energy used.

-F

Format	`-F "{\"<feature_name>\":<feature_count>,...}"`
Default	---
Description	Allocated node features. Features represent counts of the node features allocated to the job.

-g

Format	`-g <group_name>`
Default	---
Description	Group name.

-G

Format	`-G <grace_duration>`
Default	---
Description	Grace period in seconds. If you specify the lien duration but not the lien end time, MAM will calculate the lien end time as the lien start time plus the lien duration plus the grace duration.

-j

Format	`[-j] <usage_record_id>`
Default	---
Description	Usage record ID for the lien (if already created with `mam-create-usagerecord`, `mam-quote`, or a previous `mam-reserve`). This is used to place a hold against an existing usage record if the instance name (e.g. job id) is ambiguous or if usage has already been debited and you want to reserve an additional amount associated with the same usage record.

-J

Format	`-J <instance_name>`

-J	
Default	---
Description	Instance name (e.g job ID) of the lien, if known. This can sometimes be non-unique, such as when a resource manager recycles job ids, and does not always unambiguously identify a usage record to reserve. In such cases, look up and specify the usage record ID for the lien.

-L	
Format	`-L "{\"<license_name>\":<license_count>,...}"`
Default	`---`
Description	Licenses used. Licenses represent software licenses that are used (in integer units).

-m	
Format	`-m <machine_name>`
Default	---
Description	Name of the cluster.

-M	
Format	`-M <memory>`
Default	---
Description	Amount of memory used.

--modify	
Format	`--modify`
Default	---

--modify	
Description	Causes the reserve operation to augment existing liens instead of creating new ones. This new option is mutually exclusive with the `--replace` option which deletes existing matching liens and recreates a new one. The default action is to create a new lien even if a lien for an instance of the same name exists. The modify behavior supports extending liens out dynamically and is often used with incremental charging.

-n	
Format	`-n <designated_name>`
Default	---
Description	User-specified job name.

-N	
Format	`-N <nodes>`
Default	---
Description	Number of nodes used.

-o	
Format	`-o <organization_name>`
Default	---
Description	Organization name.

-P	
Format	`-P <processors>`
Default	---
Description	Number of processors used.

-q

Format	`[-q] <quote_id>`
Default	---
Description	Quote used to determine charge rates.

-Q

Format	`-Q <quality_of_service>`
Default	---
Description	Quality of service used.

-R

Format	`-R "{\"<resource_name>\":<resource_count>,...}"`
Default	---
Description	Consumable resources allocated. Resources represent consumable resources that may be allocated (in integer units).

--rate

Format	`--rate <charge_rate_name>[{<charge_rate_value>}]=<charge_rate_amount>,...`
Default	---
Description	Charge rate expressions. Multiple charge rate expressions may be passed to the `--rate` option in a comma-delimited list. Alternately, multiple `--rate` options may be specified.

--replace

Format	`--replace`

--replace

Default	---
Description	If you specify this option, MAM will delete similarly named liens before creating this lien. The default action is to create a new lien while leaving any existing liens for instances of the same name. The replace option should be specified if you want this lien to replace existing liens for instances of the same name such as when a system reuses instance names. This new option is mutually exclusive with the `--modify` option which modifies any existing matching lien instead of creating a new one.

-s

Format	`-s <start_time>`
Default	---
Description	Start time for the usage in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`.

--stage

Format	`--stage <lifecycle_stage>`
Default	---
Description	Latest stage in the object's accounting lifecycle (Create, Start, Continue, End).

-T

Format	`-T <usage_record_type>`
Default	---
Description	Usage record type, such as job or reservation.

-u

Format	`-u <user_name>`

-u	
Default	---
Description	User name.

-V	
Format	`-V "{\"<variable_name>\":\"<variable_value>\",...}"`
Default	---
Description	Job variables. Variables represent arbitrary variables passed into the job.

-W	
Format	`-W <requested_duration>`
Default	---
Description	Total estimated wallclock duration (in seconds).

-X	
Format	`-X or --extension <property>=<value>`
Default	---
Description	Specifies an extension property. You can specify any number of extra usage properties with the lien.
	When expressing accumulating properties, value may be an expression in the form of `[cumulative_value][(incremental_value)]`.
	• If both `incremental_value` and `cumulative_value` are specified, then `incremental_value` will be used for the lien and `cumulative_value` will be recorded as the cumulative value used in the usage record.
	• If only `incremental_value` is specified, this value will be used for the lien *only* and no cumlative value will be recorded in the usage record.
	• If only `cumulative_value` is specified, this value will be used both in the lien *and* recorded in the usage record.

-z	
Format	`-z <lien_amount>`
Default	---
Description	Lien amount if calculated externally.

-zs	
Format	`-zs <lien_start_time>`
Default	`Now` (if unable to derive by other means)
Description	Start time for the lien in the format `YYYY-MM-DD[hh:mm:ss]|-Infinity|Infinity|Now`. This is *only* needed for incremental liens when the start of the lien interval differs from the original start time *and* is used to determine the appropriate allocation to reserve.

-zt	
Format	`-zt <lien_duration>`
Default	---
Description	Incremental duration for the lien (in seconds). This is *only* needed for incremental liens when the incremental duration differs estimated wallclock duration *and* is used to compute the incremental lien amount.

-Z	
Format	`-Z "{\"<metric_name>\":<metric_amount>,...}"`
Default	---
Description	Generic metrics. Metrics represent floating point metrics of the job *or* average metrics values across the nodes in the job.

--hours	
Format	`--hours`
Default	---
Description	Displays time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--itemize	
Format	`--itemize`
Default	---
Description	Returns the composite charge information in the response data. You must use this in conjunction with the `--verbose` flag to display the data.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man	
Format	`--man`
Default	---

--man

Description	Displays full documentation.

--quiet

Format	--quiet
Default	---
Description	Suppresses headers and success messages.

--site

Format	--site <site_name>
Default	---
Description	Obtains a response from specified site.

--verbose

Format	--verbose
Default	---
Description	Display modified object details.

--version

Format	--version
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 14.6 Making a Usage Lien - page 147
- Appendix A: Commands Reference - page 457

A.55 mam-server

A.55.1 Synopsis

mam-server [`-s`, --start] [`-k`, --stop] [`-r`, --restart] [`-c`, --reconfig] [`-l`, --status] [`--primary`] [`--backup`] [`-d`, `--debug` [<debug_level>]] [`--help`] [`--man`] [`--version`] [`--about`]

A.55.2 Overview

mam-server is a forking server that listens for and services Moab Accounting Manager client requests. It handles the startup and daemonization, shutdown and restart of the application.

A.55.3 Options

--backup	
Format	`--backup`
Default	---
Description	Causes the server to start up in the backup server role. When running under the backup server role, events are disabled.

-c, --reconfig	
Format	`-c` or `--reconfig`
Default	---
Description	Causes the server to reread the configuration files. This can also be accomplished by sending the HUP signal to the main server process.

-k, --stop	
Format	`-k` or `--stop`
Default	---

-k, --stop	
Description	Shuts down (kill) the server. This can also be accomplished by sending the TERM signal to the main server process.

-l, --status	
Format	`-l` or `--status`
Default	---
Description	Displays the status of the server, indicating whether it is running or has stopped.

--primary	
Format	`--primary`
Default	---
Description	Causes the server to start up in the primary server role. When running under the primary server role, events are enabled.

-r, --restart	
Format	`-r` or `--restart`
Default	---
Description	Restarts the server. ⓘ If MAM has been started under systemd, use `systemctl restart mam.service` instead of using this option.

-s, --start	
Format	`-s` or `--start`
Default	---

-s, --start

Description	Starts the server. This option is assumed in the absence of a stop or restart flag and may be omitted in a start request.

-d, --debug

Format	`-d` or `--debug [<debug_level>]`
Default	`DEBUG`
Description	Logs debug information to the screen. You can supply an optional debug level parameter to indicate the logging threshold. It can be one of `TRACE`, `DEBUG`, `INFO`, `WARN`, `ERROR` and `FATAL`.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

-

A.56 mam-set-password

A.56.1 Synopsis

mam-set-password [[-u] <user_name>] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.56.2 Overview

`mam-set-password` sets a user password. If the user name is not specified via an option or as the unique argument, then the invoking user will be taken as the user whose password will be set. The invoker will be prompted for the new password.

A.56.3 Options

-u	
Format	`[-u] <user_name>`
Default	---
Description	Specifies the name of user whose password is to be set. If no user is specified, the invoking user will be taken as the user whose password will be set.

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--help	
Format	`--help`
Default	---

--help

Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.57 mam-shell

A.57.1 Synopsis

mam-shell [--format csv|raw|standard] [--debug] [--site <site_name>]
[--help] [--man] [--quiet] [--verbose] [--version] [--about]
[<command>]

A.57.2 Overview

mam-shell is an interactive control program that can access all functionality available in MAM. You can invoke commands directly from the command line, or an interpreter can parse commands from stdin.

Commands follow the form:

```
<Object>[,<Object>...] <Action> [<Predicate>]...
```

 <Predicate> follows the form:

```
[<Conjunction>] [<OpenParentheses>] [<Object>.]<Name Operator>
[<Subject>.]<Value> [<CloseParentheses>]
```

Where

- <Conjunction> defaults to "And" and includes:
    - && – and
    - || – or
    - &! – and not
    - |! – or not
- <OpenParentheses> can be any number of literal open parentheses '('.
- <Name> is the name of the condition, assignment or option.
- <Operator> is one of:
    - == – equals
    - < – less than
    - > – greater than
    - <= – less than or equal to
    - >= – greater than or equal to
    - != – not equal to
    - ~ – matches

- ○ ! ~ – does not match

- ○ = – assignment

- ○ += – increment

- ○ −= – decrement

- ○ := – option

- ○ :! – negated option

- `<Value>` is the value of the condition, assignment, or option and may be enclosed in double quotes to enclose spaces or special characters.

- `<CloseParentheses>` can be any number of literal close parentheses ')'.

You can specify the desired selections (columns to be displayed) in a query via a pseudo Show option with a value of comma-separated attribute names. It can optionally include an object, operator and alias. It will follow the form:

```
Show:="[operator(][object.]name[=alias][)][,[operator(][object.]name
[=alias][)]]...".
```

See Chapter 22: Using the MAM Shell (mam-shell) - page 203 for more information on constructing requests.

A.57.3 Options

--debug	
Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--format	
Format	`--format <standard|csv>`
Default	standard
Description	Specifies the data output format. Valid values are standard, raw, and csv.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version	
Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

A.58 mam-statement

A.58.1 Synopsis

mam-statement [[-f] <fund_id>] [-n <fund_name>] [-u <user_name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] [-s <start_time>] [-e <end_time>] [--summarize] [--hours] [--debug] [--site <site_man>] [--help] [--man] [--version] [--about]

A.58.2 Overview

mam-statement displays a fund statement. For a specified time frame it displays the beginning and ending balances as well as the total credits and debits to the fund over that period. This is followed by an itemized report of the debits and credits. You can use filters to select the funds you would like to review.

A.58.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Specifies that the statement will represent a combination of information for all the funds available for this account. Note that the statement may include information from other accounts if multiple accounts share the included funds.

-c	
Format	-c <class_name>
Default	---
Description	Specifies that the statement will represent a combination of information for all the funds available for this class. Note that the statement may include information from other classes if multiple classes share the included funds.

-e	
Format	`-e <end_time>`
Default	`Now`
Description	Specifies the end of the reporting period in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-f	
Format	`[-f] <fund_id>`
Default	`Infinity`
Description	Specifies that MAM should make the fund statement for the specified fund.

-g	
Format	`-g <group_name>`
Default	`---`
Description	Specifies that the statement will represent a combination of information for all the funds available for this group. Note that the statement may include information from other groups if multiple groups share the included funds.

-m	
Format	`-m <machine_name>`
Default	`---`
Description	Specifies that the statement will represent a combination of information for all the funds available for this machine. Note that the statement may include information from other machines if multiple machines share the included funds.

-n	
Format	`[-n] <fund_name>`
Default	---
Description	Specifies that MAM will display the fund statement for funds with the given name.

-o	
Format	`-o <organization_name>`
Default	---
Description	Specifies that the statement will represent a combination of information for all the funds available for this organization. Note that the statement may include information from other organizations if multiple organizations share the included funds.

-s	
Format	`-s <start_time>`
Default	`-Infinity`
Description	Specifies the beginning of the reporting period in the format `YYYY-MM-DD[ hh:mm:ss]|-Infinity|Infinity|Now`.

-u	
Format	`-u <user_name>`
Default	---
Description	Specifies that the statement will represent a combination of information for all the funds available for this user. Note that the statement may include information from other users if multiple machines share the included users.

---hours

Format	`--hours`
Default	---
Description	Specifies that MAM should display time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), the currency is divided by 3600 to display resource-hours.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter

Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Restricts the fund to one where constraints do not conflict with the specified filters. For example, `mam-statement --filter User=amy` will restrict the fund to one usable by the user `amy`. You may specify multiple filter options that are logically ANDed together.

--filter-type

Format	`--filter-type ExactMatch|Exlusive|NonExclusive`
Default	`NonExclusive`

--filter-type

Description	Specifies the filtering type.
	• If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints.
	• If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied).
	• If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`

--site

Default	---
Description	Obtains a response from specified site.

--summarize

Format	--summarize
Default	---
Description	Displays transaction summaries only. Deposits, Refunds, Charges, and other properties will be shown as total as opposed to being itemized.

--version

Format	--version
Default	---
Description	Displays the product version.

--about

Format	--about
Default	---
Description	Displays product information.

Related Topics

- 10.10 Obtaining a Fund Statement - page 107
- Appendix A: Commands Reference - page 457

A.59 mam-transfer

A.59.1 Synopsis

mam-transfer {--from-fund <source_fund_id> &| --from-
allocation <source_allocation_id> &| --from-filter <filter_
name>=<filter_value>...} {--to-fund <destination_fund> &| --to-
allocation <destination_allocation_id> &| --to-filter <filter_
name>=<filter_value>...} [--filter-type
ExactMatch|Exclusive|NonExclusive] {[-z] <transfer_amount>} [-
d <description>] [--hours] [--debug] [--site <site_name>] [--help] [-
-man] [--quiet] [--verbose] [--version] [--about]

A.59.2 Overview

mam-transfer issues a transfer between funds.

A.59.3 Options

-d	
Format	-d <description>
Default	---
Description	Specifies the reason for the transfer. The annotation applies to the transaction description (seen via mam-list-transactions), not the allocation description.

-z	
Format	[-z] <transfer_amount>
Default	---
Description	Specifies the amount to transfer.

--hours

Format	`--hours`
Default	---
Description	Treats currency as specified in hours. In systems where the currency is measured in resource-seconds (like processor-seconds), this option allows the amount to be specified in resource-hours.

--debug

Format	`--debug`
Default	---
Description	Logs debug information to the screen.

--filter-type

Format	`--filter-type ExactMatch|Exlusive|NonExclusive`
Default	`NonExclusive`
Description	Specifies the filtering type. • If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. • If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). • If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--from-allocation

Format	`--from-allocation <source_allocation_id>`
Default	---
Description	Transfers credits from the specified allocation ID only. If you omit the allocation, only credits from active allocations will transfer in the order of earliest expiring first.

--from-filter

Format	`--from-fund <filter_name=<filter_value>`
Default	---
Description	If you specify one or more source filters and there is exactly one matching fund, MAM makes the transfer from that fund. Otherwise, it displays a list of funds for the specified filters and you will be prompted to respecify the transfer against one of the enumerated funds. You can specify multiple `--from-filter` options by logically ANDing them together.

--from-fund

Format	`--from-fund <source_fund_id>`
Default	---
Description	Specifies the fund to be debited.

--help

Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`

--quiet

Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--to-allocation

Format	`--to-allocation <destination_allocation_id>`
Default	---
Description	Transfers credits to the specified allocation ID only. If you omit the allocation, MAM transfers the credits to the allocation having the same start and end time as the source allocation the funds are taken from, or, if such an allocation is non-existent, MAM will create a new allocation in the target fund having the same start and end time.

--to-filter

Format	`--to-filter <filter_name>-<filter_value>`
Default	---
Description	If you specify one or more destination filters and there is exactly one matching fund, a transfer will be made to that fund. Otherwise, MAM displays a list of funds for the specified filters and you will be prompted to respecify the transfer against one of the enumerated funds. You may specify multiple `--to-filter` options by logically ANDing them together.

--to-fund

Format	`--to-fund <destination_fund_id>`

--to-fund	
Default	---
Description	Specifies the fund to be credited.

--verbose	
Format	--verbose
Default	---
Description	Displays modified object details.

--version	
Format	--version
Default	---
Description	Displays the product version.

--about	
Format	--about
Default	---
Description	Displays product information.

Related Topics

-
-

A.60 mam-withdraw

A.60.1 Synopsis

mam-withdraw [-f <fund_id>] [-i <allocation_id>] [-u <user-name>] [-g <group_name>] [-a <account_name>] [-o <organization_name>] [-c <class_name>] [-m <machine_name>] [--filter <filter_name>=<filter_value>]... [--filter-type ExactMatch|Exclusive|NonExclusive] {[-z] <withdrawal_amount>} [-d <description>] [--hours] [--debug] [--site <site_name>] [--help] [--man] [--quiet] [--verbose] [--version] [--about]

A.60.2 Overview

mam-withdraw makes a withdrawal from the specified fund.

A.60.3 Options

-a	
Format	-a <account_name>
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified account.

-c	
Format	-c <class_name>
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified class.

-d	
Format	-d <description>

-d	
Default	---
Description	Specifies the reason for the withdrawal. The annotation applies to the transaction description (seen via `mam-list-transactions`), not the allocation description.

-f	
Format	`-f <fund_id>`
Default	---
Description	Specifies the ID of the fund from whichMAM will make the withdrawal.

-g	
Format	`-g <group_name>`
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified group.

-i	
Format	`-i <allocation_id>`
Default	---
Description	Withdraws credits from the specified allocation ID only. If you omit the allocation, MAM only withdraws credits from active allocations in the order of earliest expiring first.

-m	
Format	`-m <machine_name>`
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified machine

-o	
Format	`-o <organization_name>`
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified organization.

-u	
Format	`-u <user_name>`
Default	---
Description	Specifies that the fund for the withdrawal should be usable by the specified user.

-z	
Format	`[-z] <withdrawal_amount>`
Default	---
Description	Specifies the amount to withdraw. You may also specify the amount as the sole argument.

--hours	
Format	`--hours`
Default	---
Description	Specifies that MAM should display time-based credits in hours. In cases where the currency is measured in resource-seconds (like processor-seconds), this option allows the amount to be specified in resource-hours.

--debug	
Format	`--debug`

--debug	
Default	---
Description	Logs debug information to the screen.

--filter	
Format	`--filter <filter_name>=<filter_value>`
Default	---
Description	Restricts the fund to one where constraints do not conflict with the specified filters. For example, `mam-withdraw --filter User=amy` will restrict the fund to one usable by the user `amy`. You may specify multiple filter options that are logically ANDed together.

--filter-type	
Format	`--filter-type ExactMatch|Exlusive|NonExclusive`
Default	`NonExclusive`
Description	Specifies the filtering type. • If the exact-match filter type is used, a fund will only be matched if the specified filters exactly match the fund constraints. • If the exclusive filter type is used, a fund will only be matched if the specified filters meet all constraints (not only must the filters be a non-conflicting superset of the fund constraints, but all constraint association dependencies must also be satisfied). • If the non-exclusive filter type is used, a fund will be matched as long as the specified filters do not conflict with the constraints.

--help	
Format	`--help`
Default	---
Description	Displays a brief help message.

--man

Format	`--man`
Default	---
Description	Displays full documentation.

--quiet

Format	`--quiet`
Default	---
Description	Suppresses headers and success messages.

--site

Format	`--site <site_name>`
Default	---
Description	Obtains a response from specified site.

--verbose

Format	`--verbose`
Default	---
Description	Displays modified object details.

--version

Format	`--version`
Default	---
Description	Displays the product version.

--about	
Format	`--about`
Default	---
Description	Displays product information.

Related Topics

- 10.8 Making Withdrawals - page 105
- Appendix A: Commands Reference - page 457

A.61 mybalance

A.61.1 Synopsis

mybalance [--hours] [--help] [--man]

A.61.2 Overview

mybalance displays balance information for the invoking user.

A.61.3 Options

--hours	
Format	--hours
Default	---
Description	Displays balance in processor-hours (instead of processor-seconds).

--help	
Format	--help
Default	---
Description	Displays a brief help message.

--man	
Format	--man
Default	---
Description	Displays full documentation.

Related Topics

-
-